A-LOGIC

RICHARD BRADSHAW ANGELL

University Press of America, Inc.
Lanham • New York • Oxford

copyright page ii (designed by UPA)

Contents

ANALYTIC LOGIC

CHAPTER 0 Introduction

PART I. ANALYTIC LOGIC

SECTION A. Synonymy and Containment

CHAPTER 1 "And" and "Or"

CHAPTER 2 Predicates

CHAPTER 4 "Not"

SECTION B. Mathematical Logic

CHAPTER 5 Inconsistency and Tautology

SECTION C. Analytic Logic

CHAPTER 6 "If . . . then" and Validity

PART II. TRUTH-LOGIC

CHAPTER 7 "Truth" and Mathematial Logic

CHAPTER 8 Analytic Truth-logic with C-conditionals

CHAPTER 9 Inductive Logic—C-conditionals and Factual Truth

Preface

The standard logic today is the logic of Frege's *Begriffschrift* (1879) and subsequently of Russell and Whitehead's great book, *Principia Mathematica* (1913) of Quine's *Mathematical Logic* (1940) and *Methods of Logic* (4th ed.,1982) and of hundreds of other textbooks and treatises which have the same set of theorems, the same semantical foundations, and use the same concepts of validity and logical truth though they differ in notation, choices of primitives and axioms, diagramatic devices, modes of introduction and explication, etc. This standard logic is an enormous advance over any preceding system of logic and is taught in all leading institutions of higher learning. The author taught it for forty years.

But this standard logic has problems. The investigations which led to this book were initially directed at solving these problems. The problems are well known to logicians and philosophers, though they tend not to be considered serious because of compensating positive achievements. In contrast to many colleagues, I considered these problems serious defects in standard logic and set out to solve them. The anomalies called "paradoxes of material and strict implication", were the first problems raised. The paradox of the liar and related paradoxes were raised later. Other problems emerged as proponents tried to apply standard logic to the empirical sciences. These included the problems of contrary-to-fact conditionals, of dispositional predicates, of confirmation theory and of probabilities for conditional statements, among others.

I

What started as an effort to deal with particular problems ended in a fundamentally different logic which I have called A-logic. This logic is closer to ordinary and traditional concepts of logic and more useful for doing the jobs we expect logic to do than standard logic. I believe that it, or something like it, will replace today's standard logic, though the secure place of the latter in our universities makes it questionable that this will happen in the near future.

To solve or eliminate the problems confronting standard logic, fundamental changes in basic concepts are required. To be sure, most *traditional* forms of valid *arguments* (vs. theorems), are logically valid according to both standard and A-logic. Further, A-logic includes the exact set of "theorems" which belong to standard logic. This set of theorems appears as a distinct subset of tautologous formulae that can not be false under standard interpretations of 'and', 'or', 'not', 'all' and 'some'.[1] Nevertheless, the two systems are independent and very different.

In the first place, A-logic does not accept all rules of inference that standard logic sanctions. There are infinitely many little-used argument forms which are "valid" by the criterion of standard logic but are not valid in traditional logic. These 'valid' arguments would be called *non sequiturs* by intelligent users of natural languages, and are treated as invalid in A-logic. They include the *argument forms* related to the

1. 'tautology' is defined in this book in a way that covers all theorems of standard logic.

so-called "paradoxes of material and strict implication"—that any argument is valid if it has an inconsistent premiss or has a tautology as its conclusion. Thus, the two systems differ on the argument forms they call valid, though both aspire to capture all ordinary arguments that traditional logic has correctly recognized as valid.

Secondly, because A-logic interprets "if…then" differently, many *theorems* of standard logic, interpreted according to standard logic's version of 'if..then', have no corresponding theorems with 'if…then' in A-logic. Prominent among these are the 'if…then' statements called "paradoxes of material implication".

Thirdly, the introduction of A-logic's "if…then" (which I call a C-conditional) brings with it a kind of inconsistency and tautology that standard logic does not recognize. It would ordinarily be said that "if P is true then P is not true" is inconsistent; thus its denial is a tautology. A-logic includes inconsistencies and tautologies of this type among its tautology- and inconsistency-theorems in addition to all of the tautologous 'theorems' of standard logic. But while such inconsistencies and tautologies hold for C-conditionals, they do not hold for the truth-functional conditional of standard logic where "If P then not-P" is truth-functionally equivalent to 'not-P'. Thus A-logic has additional 'theorems' (in standard logic's sense) which standard logic does not have.

But the heart of the difference between the two systems does not lie in their different conditionals. Standard logic need not interpret its symbol '$(P \supset Q)$' as 'if…then'. For in standard logic "if P then Q" is logically equivalent to and everywhere replacable by "either not P or Q" or "not both P and not Q". If 'if…then's are replaced by logically equivalent "either not…or…" statements, the argument form that standard logic calls *Modus Ponens* becomes an alternative syllogism and "Paradoxes of material implication" turn into tautologous disjunctions. For example, $\vdash (\sim P \supset (P \supset Q))$ becomes $\vdash (P \vee \sim P \vee Q))$ while the related 'valid'argument, $\vdash (\sim P$, therefore $(P \supset Q))$ becomes the Principle of Addition $\vdash (\sim P$ therefore $(\sim P \vee Q))$. Thus all its theorems, valid arguments, and rules of inference are expressible in logically equivalent expressions using only 'or', 'not', 'and', 'all' and 'some', without any "if … then"s. Interpreting its formulae as 'if … then' statements may be necessary to persuade people that this is really a system of *logic*; but no essential theorem or 'valid' argument form is eliminated if we forego that interpretation.

But even if no theorems of standard logic are interpreted as conditionals the critical differences remain. This would not remove the 'valid' *non-sequiturs* of strict implication. And even the Principle of Addition raises questions. Given any statement P, does every disjunction which has P as a disjunct *follow logically from* P regardless of what the other disjuncts may be? If the logic is based on a truth-functional semantics (like standard logic) the answer may be yes; for if P is *true*, then surely (P or Q) is *true*. But do all of the infinite number of disjunctions (each with a different possible statement in place of 'Q) *follow logically* from P? Is "Hitler died or Hitler is alive in the United States" *logically contained* in "Hitler died"? Does the former follow necessarily from the latter? These questions can be raised seriously. A-logic raises them and provides an answer. In doing so it must challenge the ultimacy of the truth-functional foundation on which the principle of Addition and all of standard logic rests. The question has to do with what we mean by "follows logically" and "logically contains" independently of the concepts of truth and falsehood. Thus, even if all 'if…then's are removed from standard logic, very basic differences remain.

The fundamental difference between the two systems is found in their concepts of logical validity. It is generally conceded that the the primary concern of logic is the development of logically valid principles of inference. A-logic offers a different, rigorous analysis of "logically follows from" and "is logically valid" which frees it from the paradoxes, including the Liar paradox, of standard logic.

In the past it has often been said that a valid inference must be based on a *connection* between premisses and conclusion, and between the antecedent and consequent of the related conditional. But, as in Hume's search for the connection between cause and effect, the connection between premiss and conclusion in valid inferences has been elusive.

In standard logic a connection of sorts is provided by the concepts of truth and falsity. According to standard logic an inference is valid if and only if it can never be the case that the premisses are true and conclusion is false. But this definition is too broad. It is from this definition that it follows that if a premiss is inconsistent and thus cannot be true, then the argument is valid no matter what the conclusion may be; and also that if the conclusion is tautologous or cannot be false, the inference is valid no matter what the premisses may be. In both cases we can not have both the premisses true and the conclusion false, so standard logic's criterion of validity is satisfied. In these two cases the premisses and conclusion may talk about utterly different things. The result is that infinitely many possible arguments that are ordinarily (and with justice) called *non sequiturs*[2] must be called logically valid arguments if we use the definition of validity given by standard logic.

Logical validity is defined differently in A-logic. It is defined in terms of a relation between meanings rather than a relation between truth-values. To be logically valid, the meaning of the premisses must logically *contain* the meaning of the conclusion. This is the connection it requires for validity; and it adds that the premisses and conclusion must be mutually consistent.

II

The connection of logical containment is not elusive at all. It is defined in terms of rigorous syntactical criteria for 'is logically synonymous with'. In A-logic 'P is logically contained in Q' is defined as meaning that P is logically synonymous with (P&Q). Thus if a statement of the form, "P therefore Q" is valid it is synonymous with "(P&Q) therefore Q", and if "If P then Q" is valid, it is synonymous with "If (P&Q) then Q". In standard logic this is called 'Simplification'.

How, then, is '*logical synonymy*' defined according to A-logic? To be part of a rigorous formal logic, it must be syntactically definable, and the result must be plausibly related to ordinary concepts of synonymy.

In ordinary language the word 'synonymy' has different meanings for different purposes. Roughly, if two words or sentences are synonymous, they have the same meanings. But this leaves room for many interpretations. Sameness of meaning for a dictionary is based on empirical facts of linguistic usage and is relative to a particular language or languages. Sometimes sameness of meaning is asserted by fiat as a convention to satisfy some purpose—e.g., abbreviations for brevity of speech, or definitions of legal terms given in the first paragraphs of a legal document to clarify the application of the law. In other cases it is claimed to be a result of analysis of the meaning of a whole expression into a structure of component meanings. A plausible requirement for the ordinary concept of synonymity is that two expressions are synonymous if and only if (i) they talk about all and only the same individual entities and (ii) they say all and only the same things about those individual entities. For purposes of logic we need a definition close to ordinary usage which uses only the words used in formal logic.

In purely formal A-logic the concept of logical synonymity is based solely on the meanings of syncategorematic expressions ('and', 'or', 'not', 'if ... then' and 'all'), and is determined solely by logical words and forms. Two formulae are defined as synonymous if and only if (i) their instantiations can have all and only the same set of individual constants and predicate letters (and/or sentence letters), (ii) whatever predicate is applied to any individual or ordered set of individuals in one expression will be predicated of the same individual or set of incdividuals in the other expression, and (iii) whatever is logically entailed by one expression must be logically entailed by the other. These requirements together

2. I.e., the conclusion does not *follow from* the premisses.

with definitions for ‘all’ and ‘some’, six synonymy-axioms for ‘and’, ‘or’, ‘not’, ‘if … then’, and appropriate rules of substitution yield the logical synonymity of ‘(P & (QvR))’ with ‘((Q&P)v ~ (~Rv ~ P))’ and of ‘(∃x)(∀y)Rxy’ with ‘((∀y)(∃x)Rxy & (∃x)Rxx) & (∃z)(∀y)Rzy’ for example, as well as the logical synonymy of infinitely many other pairs of formulae.

In A-logic all *principles of valid inference* are expressed in *conditional* or *biconditional* statement-schemata such that the meaning of the consequent is contained in the meaning of the antecedent. In standard logic ‘theorems’ are all equivalent to tautologous conjunctions or disjunctions *without any conditionals* in the sense of A-logic. These ‘theorems’ are all truth-functionally equivalent to, and analytically synonymous with, denials of inconsistent statement schemata. In A-logic ‘inconsistency’ and ‘tautology’ are defined by containment, conjunction and negation. Tautologous statements are not ‘logically valid’ in A-logic, for the property of logical validity is confined to arguments, inferences, and conditional statements in A-logic. No formula called *logically valid* in standard logic is a “validity-theorem” or called a *logically valid* formula in A-logic (although as we said, many *logically valid argument-forms* are found in standard logic). This result follows upon confining *logical validity* to cases in which the conclusion “follows logically” from the premisses in the sense of A-logic. Principles of logical inference are included among the validity-theorems, and validity-theorems are theorems of logic, not of a metalogic.

In standard logic valid statements are universally true. In A-logic a valid statement can not be false but need not be true. An argument can be valid even if the premisses and conclusion, though jointly consistent, are never true. A conditional can be logically valid, though its antecedent (and thus the conditional as a whole) is never true. Acknowledging the distinction between the truth of a conditional from the validity of a conditional helps solve the problem of counterfactual and subjunctive conditionals.

To recognize tautology and validity as distinct, the very concept of a theorem of logic has to be changed. Instead of just one kind of theorem (as in standard logic), A-logic has Synonymy-theorems, Containment-theorems, Inconsistency-theorems, Tautology-theorems, Validity-theorems, etc.. Each category is distinguished by whether its special logical predicate—‘is logically synonymous with’, ‘logically contains”, ‘is inconsistent’, ‘is tautologous’ or ‘is valid’—truthfully applies to its subject terms or formulae.

Validity-theorems (principles of valid inference) are the primary objective of formal logic. Inconsistency-theorems (signifying expressions to be avoided) are of secondary importance. The denials of inconsistency, i.e., tautology-theorems (e.g., theorems of standard logic) have no significant role as such in A-logic.

The concept of synonymy is the fundamental concept of A-logic from beginning to end. It is present in all definitions, axioms and theorems of A-logic. We are not talking about the synonymy that Wittgenstein and Quine attacked so effectively—real objective univocal universal synonymy. Rather we are talking about a concept which individuals can find very useful when they deliberately accept definitions which have clear logical structures and then stick with them.

Taking synonymy as the central semantic concept of logic takes us ever farther from the foundations of standard logic. To base logic on relationships of meanings is a fundamental departure from a logic based on truth-values. Since only indicative sentences can be true or false the primary objects of study for standard logic are propositions. Synonymy begins with terms rather than propositions; it covers the relation of sameness of meaning between any two expressions. Two predicates can be synonymous or not and the meaning of one can contain the meaning of another. One predicate may or may not logically entail another. Compound predicates can be inconsistent or tautologous—as can commands and value-judgments—though none of these are true or false. A-logic provides sufficient conditions for determining precisely whether the meaning of a predicate schema is inconsistent and whether it is or is not *logically* contained in the meaning of another. Thus the focus is shifted from propositions to predicates in A-logic.

The concept of synonymy in A-logic can not be reduced to sameness of truth-values. There are unlimited differences of meaning among sentences which have the same truth-value or express the same truth-function. Such differences are precisely distinguishable in A-logic among schemata with only the "logical" words. This allows distinctions which the blunt instruments of truth-values and truth-functions pass over.

II

As the Table of Contents indicates, we presuppose a distinction between pure formal logic (Part I) and truth-logic (Part II). Since standard logic presupposes that it is ultimately about statements which are true or false, A-logic must deal with the logic of statements claimed to be true or not true if it is to solve all of standard logic's problems. Such statements deal with conditional statements about matters of fact, particular or universal: empirical generalizations, causal connections, probabilities, and laws in common sense and science. Standard logic's problems are how to account for confirmation, dispositional statements, contrary-to-fact conditionals, causal statements and conditional probability among others.

Standard logic is tethered to the concept of truth. A-logic is not; Tarski's Convention T is not accepted. It is not assumed that all indicative sentences are equivalent to assertions of their truth. They may simply be used to paint a picture without any reference to an independent reality. Frege's argument that 'is true' is not a predicate is also rejected. By "Truth-logic" we mean the logic of the operator 'T' (for "It is true that…") and the predicate "…is true". The base of Analytic Truth-logic consists of two definitions, seven synonymy-axioms, and three rules of inference (see Ch. 8). Almost all of these are compatible with, even implicit in, the semantics of standard logic; the difference lies in the omission of some presuppositions of standard logic's semantics.

In contrast to problems in pure formal logic, standard logic's problems with respect to the logic of empirical sciences are all due to its truth-functional conditional. Mathematical logic greatly improved upon Aristotelian logic by treating universal statements as quantified conditionals; i.e., in replacing "All S is P" with "(∀x)(If Sx then Px)". But the interpretation of "if P then Q" as equivalent to "not both P and not-Q" and to "not both not-Q and not-not P" results in the "paradox of confirmation", the problem of counterfactual conditionals, the problem of dispositional predicates, and the problem of conditional probability, and problem of causal statements among others; all of which hamper the application of standard logic to the search for truth in empirical sciences and common sense.

Separating the meaning of a sentence from the assertion that it is true helps to solve or eliminate the major problems in applying standard logic to valid logical reasoning in science and common sense. Since predicate terms as such are neither true nor false, A-logic deals with expressions that are neither true nor false as well as with ones that are. Most importantly, it holds that in empirical generalizations the indicative C-conditionals are true or false only when the antecedent is true; they are neither true nor false if the antecedent does not obtain. It also makes it possible to interpret scientific laws as empirical principles of inference, as distinct fromuniversal truths. The separation of meaning from truth assertion, coupled with the C-conditional, allows solutions to the problems mentioned.

It follows that standard logic's presupposition that every indicative sentence is either true or false exclusively can not hold in A-logic. The Principle of Bivalence (that every indicative sentence is either true or false exclusively) gives way to the Principle of Trivalence, (that every expression is either true, false, or neither-true-nor-false). The resulting trivalent truth-logic yields three Laws of Non-Contradiction and three Laws of Excluded Middle. It also yields proofs of every truth-table principle for assigning truth-values to a compound when the truth-values of its components are given. It makes possible consistent and plausible accounts of the contrary-to-fact conditionals, dispositional predicates, principles of confirmation, the probability of conditionals, and the analysis of causal statements that standard logic has been unable to provide. It also allows the extension of A-logic to other kinds of sentences which are neither true nor false, such as questions and commands.

In separating truth-logic (Part II) from purely formal logic (Part I), A-logic rejects Quine's rejection of the distinction between analytic and synthetic statements. Clearly if we have a conditional statement in which the meaning of the antecedent *logically contains* the meaning of the consequent, we can tell by *analysis of the statement* that if the antecedent is true, the consequent will be true also, e.g., in "For all x, if x is black and x is a cat, then x is a cat". This contrasts sharply with conditionals like "All rabbits have appendices" (i.e., "For all x, if x is a rabbit then x has an appendix"). In this case the evidence that the consequent will be true if the antecedent is true, can not be determined by analysis of the logical structure. We have to open up cats and look inside to determine whether the statement is even possibly true. Claims about factual truth are synthetic. Truth-logic is primarily about logical relations between synthetic truth-claims. In A-logic the distinction between analytic and synthetic statements is clear.

The array of differences reported above between standard logic and A-logic—and there are more—may be baffling and confusing. Despite important areas of agreement the radical divergence of A-logic from the accepted logic of our day should not be under-estimated. Although A-logic is rigorous and coherent as a whole, many questions can be raised. In this book I have tried to anticipate and answer many of them by comparing A-logic step by step with standard, or "mathematical", logic. Usually I turn to W.V.O. Quine for clear expositions of the positions of standard logic. But admittedly, major conceptual shifts and increased theoretical complexity are demanded. This will be justified, I believe, by both a greater scope of application and greater simplicity in use of this logic vis-a-vis standard logic.

IV

Some things in this book require more defense than I have provided. Questions may be raised, for example, about the application of A-logic to domains with infinite or transfinite numbers of members. To be valid, quantificational statements must be valid regardless of the size of any domain in which they apply. In this book theorems about universally quantified formulae are proved using the principle of mathematical induction. This shows they hold in all finite domains. Whether they hold in infinite domains depends on a definition and theory of 'infinity' not developed here. Axioms of infinity are not included in pure formal logic; they are added to satisfy the needs of certain extensions of logic into set-theory and mathematics that use polyadic predicates for irreflexive and transitive relations. The logic of 'is a member of' and the logic of mathematical terms will deveop differently under A-logic than developments based on standard logic, and the treatment of infinity may be affected. So the answer to that question will depend on further developments.

It may be objected that the use of the principle of mathematical induction in Chapter 3, and the use of elementary set theory in Chapter 7, are illegitimate because the principles involved should be proved valid before using them to justify logical principles. I do not share the foundationalist view that logic starts from absolutely primitive notions and establishes set theory and then mathematics. Set theory, mathematics and the principle of mathematical induction were established long before contemporary logic. Systems of logic are judged by how well they can fill out and clarify proofs of principles previously established; a logic of mathematics would be defective if it did not include principles like Mathematical Induction. There is no harm in using a principle which will be proved later to establish its theorems. Our program is to establish a formal logic, not to deduce mathematics from it. There will be time enough later to try to clarify ordinary set-theory and basic mathematics using A-logic.

There are other things in this book that might be done better, particularly in choices of terminology. For example, I have used the predicate 'is a tautology' as Wittgenstein used it in the *Tractatus*, and have held that that usage is equivalent to 'is a denial of an inconsistency'. The Oxford English dictionary equates a tautology with repetition of the same statement or phrase. This is closer to ordinary usage than my usage here. Had I used this meaning, I would have had good reason to say that assertions of synonymy and containment are both essentially tautologous assertions, instead of holding that tautologies (in Wittgenstein's sense) are not valid statements. Again, thoughout the book I use "M-logic" (for "mathematical logic") to name what I call standard logic in this preface. I might better have used "PM-logic" (for that of Russell and Whitehead's *Principia Mathematica*) to represent standard logic. For surely the best logic of mathematics need not be that of *Principia*, which was based on what is now the standard logic of our day.

Finally, it may be asked why this book is titled 'A-logic'. The original title of this book was "Analytic Logic" and throughout the book I have used that term. But too much emphasis should not be placed on that name. The primary need is to find a term to distinguish the logic of this book from standard logic. Both logics are general systems of logic, with certain objectives in common; they need different names so we can tell which system we are talking about when comparing them.

My original reason for choosing the name "Analytic Logic" was that the semantical theory of A-logic holds that the over-all structure of the meanings of complex well-formed expressions can be expressed in their syntactical structure, and thus what follows is revealed by *analysis* of those structures.

A-logic, from beginning to end, is based on a relation of synonymy which can be determined by analyzing the syntactical structures of two expressions (often together with definitions of component terms). Further, syntactical criteria for applying the logical predicates 'is inconsistent', 'is tautologous', 'is logically synonymous with', 'logically contains' and 'is valid' are provided by inspection and *analysis of the structure and components of the linguistic symbols* of which they are predicated. For example, whether one expression P is *logically contained* in another expression, depends on whether *by analysis* using synonyms one can literally find an occurrence of P as a conjunct of the other expression or some synonym of it. In contrast, in standard logic semantics is grounded in a theory of reference. It relies on truth-values, truth-tables and truth-functions which presuppose domains and possible states of affairs external to the expression.

Philosophically it could also be argued that the name "Analytic Logic" is related to the view that logic is based on the *analysis of meanings of linguistic terms*. As such it has an affinity with the movement begun by Russell and Moore early in the twentieth century called "philosophical analysis", though without its metaphysical thrust and its search for 'real definitions'.

But on reflection I decided against this title because as time went on analytic philosophy became more and more associated with standard logic. I have many debts to those who carried out this development, but their position is what A-logic proposes should be replaced. For example, Quine is an analytic philosopher who denied the analytic/synthetic distinction, favored referents over meanings and espoused a policy of "extensionalism" as opposed to intensionalism, while describing theorems of standard logic as analytic statements. Thus for many philosophers and logicians the term "Analytic Logic" could mean the kind of standard logic which is opposed to A-logic here.

Hence I have entitled this book simply 'A-logic', even though I have treated this as an abbreviation of 'analytic logic' throughout the book. Those who find the phrase 'analytic logic' misleading, presumably will not deny that A-logic is at least <u>a</u> logic, and may remind themselves of the narrow meaning of 'analytic' originally intended. But whatever it is called, this logic is different than the current standard logic, and it is believed that some day it, or some related system, will be viewed as everyone's logic rather than an idiosyncratic theory

R.B.Angell, March, 2001

Chapter 0

Introduction

This book presents a system of logic which eliminates or solves problems of omission or commission which confront contemporary Mathematical Logic. It is not an extension of Mathematical Logic, adding new primitives and axioms to its base. Nor is it reducible to or derivable from Mathematical Logic. It rejects some of the *rules of inference* in Mathematical Logic and includes some theorems which are not expressible in Mathematical Logic. In Analytic Logic the very concepts of 'theorem' 'implication' and 'valid' differ from those of Mathematical Logic. Nevertheless all of the *theorems* of Mathematical Logic are established in one of its parts. Although it includes most principles of valid inference in both traditional and mathematical logic, it is a different logic.

0.1 Preliminary Statement

Like Mathematical Logic, Analytic Logic has, as its base, a system of **purely formal** logic which deals with the logical properties and relationships of expressions which contain only the "logical" words 'and', 'or', 'not', 'some', 'all', and 'if…then', or words definable in terms of these.[1] Except for 'if…then' these words have the same meaning in Analytic Logic as in Mathematical Logic. Both logics are consistent and complete in certain senses.

The purely formal part of Analytic Logic is based on a syntactical relation, SYN, which exists among the wffs (well-formed formulae) of mathematical logic but has not been recognized or utilized in that logic. SYN and a derivative relation, CONT, are connected to concepts of logical synonymy and logical containment among wffs of mathematical logic. These are two logical relationships which mathematical logic cannot express or define; although the paradigm case of logical containment, [(P&Q) CONT Q],

1. Some logicians have held that the logic of identity and the logic of classes are part of formal, or universal, logic—hence that '=' (for 'is identical with') or 'e' (for "is a member of") are also "logical words". But here we consider these logics as special logics of the extra-logical predicates, '…is identical with___' and '…is a member of___'. The principles of pure formal logic hold within all special logics, but principles of special logics are not all principles of universal, formal logic.

is akin to Simplification; and logical synonymy is equivalent to mutual containment. Logical synonymy is abbreviated 'SYN' when it holds between purely formal wffs, and 'Syn' when it holds between expressions with extra-logical terms in them; similarly for 'CONT'.

The first three chapters explore the synonymy and containment relations which exist between negation-free expressions using only the logical words 'and' and 'or', 'some' and 'all'. In the fourth chapter 'not' is introduced but with appropriate distinctions this does not change any basic principles of SYN and CONT. In the fifth chapter the concept of logical inconsistency is defined in terms of containment and negation, and precisely the wffs which are theorem-schemata in mathematical logic are proven to be tautologies (denials of inconsistencies) in Analytic Logic. Though these theorems are the same, the rules by which they are derived in Analytic Logic are not. Mathematical logic has rules of inference which declare that certain arguments which would normally be called non sequiturs, are *valid* arguments. The rules of inference used in Analytic Logic are different.

In Chapter 1 through 5 the use of 'if…then' in mathematical logic is not discussed. We will call the conditional used in mathematical logic the "truth-functional conditional" (abbr."TF-conditional"). In Analytic logic a different conditional, called the "C-conditional", is introduced to solve the problems which confront the TF-conditional. The difference in the concept of 'if…then' is of basic importance, but the problems of mathematical logic cannot be solved merely by changing the concept of a conditional. The roots of its problems lie much deeper and the most important differences between analytic and mathematical logic are not in the meaning of "if…then".

Nevertheless, the meaning given to 'if…then' in Analytic Logic, introduced in Chapter 6, makes possible the solution of many problems which arise from the truth-functional conditional, and it expands the scope of logic. The C-conditional is free of the anomalies which have been called (mistakenly) "paradoxes of material and strict implication"; and using the C-conditional makes it possible to explain the confirmation of inductive generalizations, to explicate contrary-to-fact conditionals, dispositional predicates, lawlike statements, the probability of conditionals, and the grounds of causal statements—concepts which mathematical logic has not handled satisfactorily.

For each valid C-conditional statement in Analytic Logic there is an analogous truth-functional "conditional" which is a valid theorem in mathematical logic; but the converse does not hold. Many TF-conditional statements called valid in mathematical logic, are not valid with C-conditionals of Analytic Logic. For example, the so-called "paradoxes of strict implication" are not valid.

On the other hand, Analytic Logic recognizes certain forms of C-conditionals as inconsistent where Mathematical Logic with a TF-conditional sees no inconsistency. For example 'If not-P then P' is inconsistent according to Analytic Logic, and its denial is a tautology. The same statement and its denial are contingent and can be true or false if the "if…then" is that of Mathematical Logic. Thus Analytic Logic adds tautology-theorems which mathematical logic lacks and cannot express, while statements which are inconsistent or tautologous with the conditionals in Analytic Logic are merely contingent if the truth-functional conditionals of mathematical logic are used.

Analytic Logic is broader in scope than mathematical logic, although it is more constrained in some areas. It is broader in that it includes all theorems of mathematical logic as tautology-theorems and adds other tautology-theorems not expressible in Mathematical Logic. It is more constrained in that it deliberately excludes some rules of inference found in Mathematic Logic and restricts others. But also, as a formal logic it is broader because it is a logic of predicates, not merely of sentences. It is not based on truth-values and truth-functions, but on logical synonymy and derivatively on logical containment, and, through the latter, on inconsistency of predicates. It is capable of explicit extension to cover not only the logic of truth-assertions and propositions, but kinds of sentences which are neither true nor false—questions, directives, and prescriptive value judgments. Further, it is self-applicable; it applies to the logic of its own proofs, and principles. (Mathematical Logic is inconsistent if it tries apply its semantic principles to itself; hence its separation of logic and 'metalogic'). Finally, with the C-conditional Analytic Logic provides principles for reasoning about indeterminate or unknown or contrary-to-fact situations which Mathematical Logic can not provide.

It is not very helpful to ask which logical system is the "true" logic. Whether the concepts of the conditional, conjunction, negation and generalization correspond to features of absolute or platonic reality is a metaphysical question, and the correspondence of a logic to ordinary usage is a question for empirical linguistics. One can hold with some justice that Analytic Logic is closer to ordinary usage than Mathematical Logic is at certain points, but ordinary usage should not be the final test. The amazing success of Mathematical Logic, despite some non-ordinary usage, has outstripped by far what "ordinary language" could accomplish.

Analytic Logic is presented here as a system which will do jobs that logic is expected to do better than Mathematical Logic can do them. It will do jobs which many enthusiastic supporters of Mathematical Logic wanted that logic to do, but found it incapable of doing. It is offered as an improvement over Mathematical Logic; without forfeiting any of the successful accomplishments of Mathematical Logic it avoids defects of commission, and fills in gaps of omission.

Mathematical Logic distinguishes tautologies and inconsistencies within a limited class of indicative sentences extremely well. Every tautology is equivalent to a denial of an inconsistency. This is important, but there is an objective of formal logic which is more basic than avoiding inconsistency. It is the job of keeping reasoning on track; of guiding the moves from whatever premisses are accepted, to conclusions which follow logically from their meaning. To avoid inconsistency is to avoid being derailed; but more is needed. We need rigorous criteria for determining the presence or absence of logical connections between premisses and conclusion, and between antecedent and consequent in any conditional if they are to be "logically valid". The concepts of logical Synonymy and Containment in Analytic Logic provide such criteria. These concepts and criteria are lacking in Mathematical Logic.

Whether or not Analytic Logic is "true", or in accord with common usage, are interesting questions, worthy of discussion. But regardless of the answer, Analytic Logic is proposed as a demonstrably more useful and efficient set of general rules to guide human reasoning.

0.2 Mathematical Logic and Its Achievements

What, more precisely, do we mean by "Mathematical Logic"? In this book, "Mathematical Logic" means the propositional logic and first-order predicate logic that was developed by the German mathematician Gottlob Frege in 1879, was expounded and expanded in Russell and Whitehead's *Principia Mathematica*, Vol. 1, 1910, and is currently taught in most universities from a wide variety of textbooks. We use the textbooks and writngs of Quine for a clear exposition of it.

Mathematical logic is one of the great intellectual advances of the 19th and 20th centuries. It has given rise to great new sub-disciplines of mathematics and logic, and it has rightly been claimed that the development of computers—and the vast present-day culture based on computer technology—could not have occurred had it not been for Mathematical Logic.

Before Mathematical Logic, the traditional logic based on the work of Aristotle and Stoic philosophers was considered by many philosophers to embody the most universal, fundamental laws of the universe and to be the final word on the rules of rational thought. But among other things, traditional logic was incapable of explaining the processes by which mathematicians—the paradigmatic employers of logic—moved from premisses to their conclusions. Traditional logic provided systematic treatments of only a few isolated kinds of argument. The logic of "some" and "all", was dominated by Aristotle's theory of the syllogism, which dealt with monadic predicates only, and four basic statement forms "All S are P", "No S are P", "Some S are P" and "Some S are not P". The logic of propositions from the

Stoics consisted of a rather unsystematic collection of valid or invalid argument forms involving 'and', 'or', 'not' and above all 'if…then' (Modus Ponens, Modus Tollens, Hypothetical Syllogism, Alternative Syllogism, Complex Constructive Dilemma, Fallacy of Affirming the Consequent, etc.). There was no general account of arguments using relational statements, no analysis of mathematical functions.

Mathematical Logic incorporated traditional logic within itself, albeit with some modifications; but it went infinitely farther. It developed a new symbolism, starting with a specification of all primitive symbols from which complex expressions could be constructed.

It introduced rules of formation which in effect distinguished grammatical from ungrammatical forms of sentences. The forms of grammatical sentences constructible under these rules are infinitely varied. Elementary sentences need not have just one subject; they may have a predicate expressing a relation between any finite number of subjects. Any sentence can be negated or not. Any number of distinct elementary sentences could be conjoined, or disjoined. Conjunctions could have disjunctions as conjuncts; disjunctions could have conjunctions as disjuncts. Expressions for 'all' or 'some' can be prefixed to infinite varieties of propositional forms with individual variables, including forms with many different occurrences of 'all' or 'some' within them. The infinite variety and potential complexity of well-formed formulae or sentence-schemata was and is mind-boggling.

Definitions are added in which certain formal expressions are defined in terms of a more complex logical structure containing only occurrences of primitive symbols.

Formulated as an axiomatic system, a small set of axioms and a few rules of inference are then laid down and from these one can derive from among all of the infinite variety of well-formed formulae, all and only those with a form such that no sentence with that form could be false because of its logical structure. These are called 'theorems' in Mathematical Logic, and sentences with those forms are said to be 'logically true' or 'valid'. All rules of inference are strictly correlated with precise rules for manipulating the symbols, so the determinations of theoremhood and logical truth are mechanical procedures, capable of being done by machines independently of subjective judgments.

Because of the rigor and vast scope of this logic, and especially its capacity to show the forms and prove the validity of very complex arguments and chains of arguments in mathematics, it has been widely accepted. About 65 years after Frege put it together and about 35 years after Russell and Whitehead introduced it to the English speaking world it became "standard logic" in the universities of the western world.

Mathematical Logic has been called "symbolic logic", "modern logic", and today it is often called "standard logic" and even "classical logic". Throughout this book it is referred to as "Mathematical Logic", or "M-logic" for short. This name seems most appropriate because 1) the use of symbols is not new (both Aristotle and the Stoics used symbols), 2) "modern" changes with time, 3) the adjectives "classical" and "standard" depend on societal judgments which may change. On the other hand 4) this particular logic originated with mathematicians (Frege, Whitehead, and before them, Leibniz, De Morgan, Boole, etc), 5) it has proven particularly successful in dealing with the forms of mathematical proofs, 6) it is often called "Mathematical Logic" by its foremost proponents whether philosophers or mathematicians, and 7) its limitations are due in part to the fact that the limited requirements of mathematical reasoning are too narrow to encompass all of the kinds of problems to which logic should be applied.[2]

---

2. E.g., Quine wrote, "Mathematics makes no use of the subjunctive conditional**;** the indicative form suffices. It is hence useful to shelve the former and treat the 'if-then' of the indicative conditional as a connective distinct from 'if-then' subjunctively used." Quine,W.V., *Mathematical Logic*, 1983, p.16. Efforts to apply Mathematical Logic during the 1940s and 50s to methods and laws of empirical science, disclosed that its failure to cover subjunctive or contrary-to-fact conditionals was a problem for its claim to universality.

Among many excellent expositions of Mathematical Logic, there are differences in notation, choices of axioms and rules, and alternative modes of presentation, as well as some differences in philosophical approach. In this book Analytic Logic is developed and compared with Mathematical Logic as it goes along. Rather than trying to deal with all of its different versions, when comparison will help the exposition I usually turn to the work of the American philosopher and logician, W.V.O.Quine, a preeminent proponent of Mathematical Logic and the major philosophical presuppositions behind it. The differences selected between Analytic Logic and Quine's formulation usually represent fundamental differences with all other formulations of Mathematical Logic.

A Formal Axiomatization of Mathematical Logic

One version of the "logistic base" of Mathematical Logic is presented in TABLE 0-1.[3] The formal axiomatic system consists of two parts whose sub-parts add up to five. These are called its "logistic base".

The first part defines the subject-matter (the object language, the entities to be judged as having or not having logical properties, or of standing in logical relations). In this part we first list the primitive symbols which will be used. Secondly we give rules for forming wffs ("well-formed formulae"), i.e., meaningful compound symbols; these are called Formation Rules. Thirdly, we provide abbreviations or definitions of additional new symbols in terms of well-formed symbols, or definitions of additional new symbols in terms of well-formed symbols containing only primitive symbols in the definiens.

The second part of the system consists of Axioms—primitive statements which are asserted to have the logical property, or stand in the logic relation, we are interested in—, and Rules of Transformation, or Rules of inference, which tell how to get additional sentences from the axioms, or from theorems derived from the axioms which will also have the logical property, or stand in the logical relations, we are interested in. This formal system can be viewed simply as a set of symbols without meaning, with rules for manipulating them. That it can be viewed in this way is what makes it a rigorous system But what makes such a system meaningful, and justifies calling it a logic, is provided by the theory of semantics which explicitly or informally is used to assign meanings to the symbols and thus to explain and justify calling it a system of logic.

The notion of *truth* is central to the semantic theory of Mathematical Logic. The well-formed formulas of Mathematical Logic are viewed as the forms of *propositions*—sentences which must be either *true* or false. The meanings of the logical constants are explained in the semantic theory by *truth*-tables, which lay out the conditions under which proposition of specific forms are true, or are false. Concepts of validity and logical implication are defined in terms of the *truth* of all possible interpretations of the schematic letters employed. Logic itself, as a discipline, is viewed has having, as its central aim, the identification of forms of statements which are universally and logically "*true*". Paradoxically, the word 'truth' is banned from the "object language" which Mathematical Logic talks about, because the concept of validity in the semantic theory would make the system self-contradictory if 'is *true*' were admitted as a predicate in that language. (Analytic Logic, with a different interpretation of "validity", not defined in terms of *truth*, does not have this problem).

(Statements attributing truth to a sentence are not included in purely formal Analytic Logic. This is not because the inclusion of 'is true' would lead to a paradox (as in Mathematical Logic), but because the word 'true' is not a 'logical word', i.e., it is not a word which makes clear the form but has no meaning

3. TABLE 0-1 presents a somewhat simplified version of Barkley Rosser's axiomatation of the first order predicate calculus in *Logic for Mathematicians*, 1953. Except for Axiom 6, this system is similar to the system of Quantification Theory in Quine's *Mathematical Logic*, 1940, but is simpler notationwise.

TABLE 0-1
Mathematical Logic

I. SCHEMATA (OR WFFS)

1. Primitives:

Grouping devices), (,
Predicate letters (PL): $P_1, P_2, \ldots, P_n$. [Abbr.‘P’, ‘Q’, ‘R’]
Individual constants (IC): $a_1, a_2, \ldots a_n$ [Abbr.‘a’, ‘b’, ‘c’]
Individual variables (IV): $x_1, x_2, \ldots x_n$, [Abbr.’x’, ‘y’, ‘z’]
Sentential operators: &, ~
Quantifier: $(\forall x_i)$

2. Formation Rules

FR1. $[P_i]$ is a wff
FR2. If P and Q are wffs, [(P&Q)] is a wff.
FR3. If P is a wff, [~P] is a wff.
FR5. If P_i is a wff and t_j ($1 \underline{<} j \underline{<} k$) is an IV or a IC,
then $P_i<t_1,\ldots,t_k>$ is a wff
FR6. If $P_i<1>$ is a wff, then $[(\forall x_j)P_ix_j]$ is a wff.

3. Abbreviations, Definitions

Df5-1. [(P & Q & R) Syn_{df} (P & (Q & R))]
Df5-2. $[(\forall_k x)Px\ \text{Syn}_{df}\ (Pa_1 \,\&\, Pa_2 \,\&\, \ldots \,\&\, Pa_k)]$ [Df ‘∀’]
Df5-3. [(P v Q) Syn_{df} ~(~P & ~Q)] [Df ‘v’]
Df5-4. $[(P \supset Q)\ \text{Syn}_{df}\ {\sim}(P \,\&\, {\sim}Q)]$ [Df ‘⊃’]
Df5-5. $[(P \equiv Q)\ \text{Syn}_{df}\ ((P \supset Q)\&(Q \supset P))]$ [Df ‘≡’]
Df5-6. $[(\exists x)Px\ \text{Syn}_{df}\ {\sim}(\forall x){\sim}Px]$ [Df ‘(∃x)’]

II. AXIOMS AND TRANSFORMATION RULES

4. Axiom Schemata

Ax.1. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)[P \supset (P\&P)]$
Ax.2. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)[(P\&Q) \supset P)]$
Ax.3. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)[(P \supset Q) \supset ({\sim}(Q\&R) \supset {\sim}(R\&P))]$
Ax.4. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)[(\forall x)(P \supset Q) \supset (\forall x)P \supset (\forall x)Q)]$
Ax.5. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)[P \supset (\forall x)P]$, if x does not occur free in P.
Ax 6. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)[(\forall x)R<x,y> \supset R<y,y>]$

5. Transformation Rules (Principles of Inference)

R1. If $\vdash$ P and $\vdash$ (P ⊃ Q) then $\vdash$ Q. [TF-Detachment]
R2. If $\vdash$ P and Q is a sentential variable and R is a wff,
then $\vdash$ P(Q/R). [Universal-SUB]

except as a syntactical operator. The logic of the predicate 'is true' is added as a special extension of Analytic Logic; this "analytic truth-logic" is presented in Chapters 7, 8 and 9.)

The semantical theory of Mathematical Logic contains many principles and assumptions which are used to support the claim that its formal system can represent all the major principles which properly belong to logic. Some of these principles are accepted in the branch of Analytic Logic which deals with truth; others are not. E.g., in the following list of principles all of the odd-numbered principles are logically valid with the C-conditional, but none of the even-numbered ones are:

1)	[P] is true if and only if [not-P] is false	T[P] iff F[∼P]
2)	[P] is true if and only if [P] is not-false	T[P] iff ∼F[P]
3)	[P] is false if and only if [not-P] is true	F[P] iff T[∼P]
4)	[P] is false if and only if [P] is not-true	F[P] iff ∼T[P]
5)	[P] is true, if and only if it is true that [P] is true	T[P] iff TT[P]
6)	Tarski's Convention T: [P] is true if and only if [P]	T[P] iff [P]
7)	[P] is either not-true or not-false	(∼T[P] v ∼F[P])
8)	The Law of Bivalence (often mis-called the "Law of Excluded Middle"): [P] is either true or false (exclusively)	(T[P] v F[P])
9)	[P and Q] is true iff [P] is true and [Q] is true.	T(P & Q) iff (TP & TQ)
10)	[P and Q] is true iff [P] is not-false and [Q] is not-false	T(P & Q) iff (∼FP & ∼FQ)
11)	[P or Q] is true iff [P] is true or [Q] is true.	T(P v Q) iff (TP v TQ)
12)	[P or Q] is true iff [P] is not-false or [Q] is not-false.	T(P v Q) iff (∼FP v ∼FQ)
13)	If [P] is true, and [P ⊃ Q] is true, then Q is true	If T[P] & T[P ⊃ Q] then T[Q]
14)	If [P] is true, and [P ⊃ Q] is not-false, then Q is true	If T[P] & ∼F[P ⊃ Q] then T[Q]

This shows that Analytic Logic differs from Mathematical Logic in its semantic assumptions. In these cases, M-logic treats 'not-false' as inter-changeable with 'true', and 'not-true' as interchangeable with 'false'. Analytic Logic does not. Analytic truth-logic includes all of the theorems of Mathematical Logic itself (not its semantics) construed as propositions which are either true or false; but its semantical theory differs from the semantical theory used to justify Mathematical Logic.

0.3 Problems of Mathematical Logic

Most criticisms of Mathematical Logic have been directed at its account of conditional statements. However, the most fundamental problem is prior to and independent of this question. It has to do with the concepts of "following logically from" and "validity" in Mathematical Logic. Validity is defined differently in A-logic than in M-logic. Thus there are two distinct basic problems to consider: 1) the problem of "logical validity" and 2) problems of truth-functional conditionals. The two are inter-related by the VC\VI principle: a conditional statement is valid if the inference from its antecedent to its consequent is valid. Despite this, we will try to disentangle the problem of validity from the problem of the truth-functional conditional. We begin with the problem of validity.

0.31 The Problem of Logical Validity

The problem of what we mean by logical validity is fundamental. It is the problem of the criteria by which one decides whether one idea "follows logically from", or is a "logical consequence" of, another. Consider the following statement:

"Linguistic expressions are visible or auditory signs with meanings. Their meanings convey concepts and ideas. One expression is said to *follow logically* from other expressions if the meanings of the other expressions are the same as, or contain, the meaning of the first expression."

So we might think. It is the way we do think in Analytic Logic. But this is not the way '*follow logically'* is used in Mathematical Logic.

0.311 The Problem of "Valid" *Non Sequitur*s

Consider: in Mathematical Logic, any statement of the form '((P&~P) ⊃ Q)' is a theorem of logic. These symbols, according to Mathematical Logic, represent ordinary statements of the form "If both P is true and P is false, then Q is true". Statements of this form are said to be a *logically valid* conditionals. Since the conditional statement is *logically valid*, by the VC\VI principle the inference from a premiss which is the antecedent to a conclusion which is the consequent is a valid inference. Therefore, since any indicative statement can be put for P and any other indicative statement for Q, the following arguments or inferences, according to M-logic, are both valid ones in which the conclusion "follows logically" from the premisses:

Tito died and Tito did not die.	Tito died and Tito did not die.
Therefore, Some dinosaurs had wings.	Therefore, No dinosaurs had wings.

These arguments, and many other kinds which are called valid in Mathematical Logic, would ordinarily be called *non sequiturs* by common sense and science. Siding with common sense, we say that to call such arguments valid, as Mathematical Logic must, is an oxymoron. According to Mathematical Logic then, there are *non sequiturs* that are valid.

For each actual statement which is both a valid conditional according to Mathematical Logic and is such that the consequent *logically follows* from the antecedent in any ordinary sense, there is an infinite set of statements which are valid conditionals according to Mathematical Logic but are clear *non sequiturs* in the ordinary sense. One has merely to put the denial of the initial "valid" statement as antecedent and then choose any one of an infinite number of statements which have no logical connection with the antecedent as the consequent. This consequence of Mathematical Logic is seldom mentioned and never stressed in its textbooks.

At first it might appear that the problem for Mathematical Logic lies in treating '((P&~P) ⊃ Q)' as a conditional statement. But the problem remains the same whether or not we treat this as a conditional statement. For '((P&~P) ⊃ Q)' is logically equivalent and analytically synonymous with '(~(P&~P) v Q)' and '~((P&~P) & ~Q)'. Given these equivalences, it is clear that rejecting the interpretation of '⊃' as "if…then", does not the solve the issue. For if '((P&~P) ⊃ Q)' is replaced by either of its logical equivalents, then ascribing validity to the inference from '(P&~P)' to Q does not depend on interpreting '⊃' as "if…then". We could have interpreted '⊃' as simply abbreviating "Either (P and not P) is false or Q is true", or "It is false that (P & ~P) is true and Q is false". The inference from (P&~P) to Q is still valid in Mathematical Logic; but in this case we can not invoke the VC\VI principle.

This leaves two questions: 1) why is '((P&~P) ⊃ Q)' a theorem of mathematical logic? and 2) how is the "validity" of '((P&~P) ⊃ Q)' connected to the validity of the inference from (P&~P) to Q?

The answer to the first question is clearer than it was when '⊃' was first read as 'if…then'. For '~((P&~P) & ~Q)' is clearly a denial of an inconsistent conjunction, and surely any denial of an inconsistency should be a theorem of logic. Since '((P&~P) ⊃ Q)' means the same thing as '~((P&~P) & ~Q)', the former should be a theorem of logic too regardless of how it is interpreted.

The answer to the second is harder to get at. It comes out in the semantic theory of Mathematical Logic, which includes an account of validity. There, an argument is said to be valid if there can be no case in which the premisses are true and the conclusion false.[4] Ha! Now we understand. For if the premiss is (P& ~ P), we know that it can never be true; hence there can be no case in which both (P& ~ P) is true and Q is false. Thus on this definition of "valid" it is valid.

But this account of validity—which is the underlying concept in Mathematical Logic—needs challenging since, as we have just seem, it forces us to call arguments which are *non sequiturs* 'valid' arguments. This does not rest on construing any formula of Mathematical Logic as a conditional. Negation, conjunction, alternation and quantification are quite sufficient to express all theorems; we need not interpret any statement as "if…then". Analytic Logic eliminates "valid" *non sequiturs* by defining 'validity' differently.

A major lack in Mathematical Logic's concept of validity is that it does not require any connection between the meaning of the premisses and meanings of expressions in the conclusion. Several kinds of *non sequiturs* qualify as "valid" in Mathematical Logic because of this. In Analytic Logic the relations of SYN and CONT require connections between premisses and conclusion through value-sharing and other syntactical properties which are related to containment of meanings. If the validity relation is restricted to ordered pairs with containment relations, many *non sequiturs* including the examples above, are avoided. The rest are eliminated by requiring that premisses and conclusion be jointly consistent if the inference is to be valid.

0.312 Irrelevance

Another problem of mathematical logic is that irrelevant components can be injected into the conclusion of their 'valid' arguments.

Roughly speaking if a part of a conclusion is not connected in any way to its premisses, it is irrelevant to the argument. A particular kind of irrelevance is connected with a principle which often appears as an axiom of Mathematical Logic—the "Principle of Addition". This has various forms, including "'(P or Q)' follows validly from 'P'", and "'not both (P and Q)' follows validly from 'not-P'". In Analytic Truth-logic it is clear that if P is *true*, then by rules for determining truth or falsity of (P or Q), (P or Q) *must be true*. This is due to the meaning of 'true' and of the statement that "P or Q" *is true*. But in purely formal Analytic Logic, prior to introducing the concept of truth, what is put for Q may have no connection with what was put for P; literally Q is not contained in P at all. In such cases the move from P to (P v Q) can introduce a Q which is completely irrelevant and 'If P then (P or Q)' is not valid in Analytic Logic. How are these two reflections on the "Principle of Addition" reconciled?

In Analytic Logic, this problem is recognized and a rigorous distinction is made in Chapters 7 and 8 between containments and inferences which do not depend on presuppositions about truth and falsity, and those that do. In some valid inferences, the truth of (P or Q) follows from the truth of P *together with* certain implicit presuppositions which underlie every truth-investigation. The latter are called '*analytic-implications*' of truth-logic and play an important role in valid *de dicto* reasoning, but can lead to error in *de re* argumentation.[5] Valid *analytic implications* should not be confused with valid *entailments*.[6]

---

4. See Quine, W.V.O. *Methods of Logic*, 4th Ed, Ch. 7 "Implication", especially pp 45-6. This is discussed in detail in Section 10.12 of this book.

5. See Section 8.232, especially Section 8.23231.

6. E.g., this distinction has a bearing on Goodman's formulation of the 'new riddle of induction'. See Section 10.22 p 569.

0.313 Paradoxes in the Foundations

The problem of 'validity' in Mathematical Logic is not only whether certain arguments called 'valid', should be called invalid. It is rooted in a problem concerning the truth-functional foundations of Mathematical Logic.

The foundations of Mathematical Logic—the assumptions expressed in the "semantic theory" that is used to justify it—presupppose that validity and all important logical concepts can be defined or developed in terms of the truth-values and truth-functions of propositions (sentences which are either true or false exclusively). The account of "logical consequence" or "validity" in Mathematical Logic is based on the extra-logical concepts of truth, falsehood and truth-functions. It is assumed that logic deals primarily with sentences which are either true or false, and that the sentences it deals with, including conditionals, must be either true or false.

These foundations lead to paradoxes in Mathematical Logic. Because the concept of validity is defined in terms of truth-functions, the "Paradox of the liar", is a genuine logical paradox for Mathematical Logic; a contradiction is derived from premisses like "Whatever I say is not true", with the assumption that it must be either true or false, by the rules of Mathematical Logic which make the conclusion "valid". If we wish to solve its problems of validity, we must re-consider the desirability of basing logic on truth-values and truth-functions. The "liar" is not a logical paradox for Analytic Logic, although it clearly involves a contradiction. For the rules by which the contradiction is derived do not follow from its rules for *valid* inference or from the definition of 'valid'in Analytic Logic.[7] Other paradoxes of Mathematical Logic have no analogous paradoxes in Analytic Logic for the same reason. In Analytic Logic conditionals and arguments can not be valid if the conjunction of both premiss and conclusion is inconsistent or if certain containment relations are lacking. These requirements, not present in Mathematical Logic, render the contradiction in "the liar" not a paradox for Analytic Logic.

0.314 Narrowness in Scope

The truth-functional foundations of Mathematical Logic not only lead to paradox and oxymoronic valid *non sequitur*s; they tie validity-theorems too narrowly to assertions of truth. What questions can be validly inferred from which other questions? What commands can be validly inferred from other commands? Questions and commands are neither true nor false, but they have logical properties and relations.

Various efforts have been made to develop logics of questions, of imperatives, and of value judgments, by adding appropriate primitives to Mathematical Logic. None of them have enjoyed a reception comparable to the acceptance of Mathematical Logic. Anomalies, *non sequitur*s, inadequacies and paradoxes dog these logics. Some of their problems are directly traceable to problems of the Mathematical Logic with which they begin.

The restricted focus on truth-values and truth-functions stands in the way of a broader conception of validity as a relation based on sameness and difference of meanings. Many expressions have the same truth-value and many sets of expressions express the same truth-function but have utterly different meanings. Making truth-values and truth-functions central in defining validity ignores vast differences between the meanings of expressions which share the same truth-value or the same truth-function. *Salve veritate* is not *salve sens*; preservation of truth-value does not preserve meaning.

Quine has been a leading proponent of rejecting "meanings" as the bearers of properties or relationships central to logic. He early rejected semantic theories featuring synonymy, analyticity and entailment, in favor of theories of reference featuring naming, truth, and extension[8] and has persistently advocated

7. See Section 10.11 for a more careful presentation and analysis of the Liar Paradox .

what he has called the "policy of extensionality", and accounts of logic compatible with a physicalistic and behavioristic world view.

Quine's objections to basing logic on properties or relations of meanings is not that meanings are "abstract shadowy entities" (though he holds that), but that "the appropriate equivalence relation [i.e. synonymy] makes no objective sense at the level of sentences". In criticizing the view that propositions are the *meanings* of sentences ("exalted as abstract entities in their own right") he writes,

> My objections to recognizing propositions does not arise primarily from philosophical parsimony—from a desire to dream of no more things in heaven and earth than need be. Nor does it arise, more specially from particularism—from a disapproval of intangible or abstract entities. My objection is more urgent. If there were propositions, they would induce a certain relation of synonymy or equivalence between sentences themselves: those sentences would be equivalent which expressed the same proposition. Now my objection is going to be that the appropriate equivalence relation makes no objective sense at the level of sentences. This, if I succeed in making it plain, should spike the hypothesis of propositions.[9]

The account of logical synonymy in Analytic Logic is intended to spike Quine's objections. In Analytic Logic Quine's separation of theories of reference and semantical theories of meaning is rejected. The meaning of a sentence involves both reference (for subject terms) and intension (for predicate terms), and there is no ground for holding that synonymy, as defined in Chapter 1, is any less rigorous and "objective" than the truth-functional analyses of logical properties and relationships offered in Mathematical Logic. Perhaps the resulting system of Analytic Logic can be interpreted in terms of a purely behavioristic and physicalistic view of the world—as "extensional" in some sense. But from a pragmatic point of view I believe there is no less rigor, and logic is easier to understand, if it is treated—as has logic has often been treated before—as having to do with sameness and containment of ideas or meanings. The rigor is provided by the correlation of these relations and properties with kinds of syntactical structures and with rules of transformation in the symbolic system.

'Validity' defined in terms of truth is less discriminating than 'validity' defined in terms of containment of meanings.[10] Many more pairs of expressions are differentiated by having different meanings, than by having different truth-values or expressing different truth-functions.

Purely formal Analytic Logic focuses on the logical properties of predicates rather than on those of sentences exclusively (see Chapter 2). They serve better than sentences as the subject matter of formal logic. Predicates are neither true nor false as such. Compound predicates can be constructed from simpler ones using the usual connectives and operators. They may be inconsistent or tautologous or neither, and the meaning of one may be contained in, or synonymous with, the meaning of another. The basic logical properties and relationships apply to predicates primarily and sentences derivatively. The logic of sentences follows from the logical relationships of their predicates. Sentences may be viewed as "saturated predicates"—predicates in which all the argument positions are filled with constant singular terms.

Predicates occur in sentences which are neither true nor false, such as questions, directives, and prescriptive value judgments; from this fact the logic of these modes of discourse will follow. Logical

---

8. Cf. "Notes on the Theory of Reference", CH VII in Quine's *From a Logical Point of View*, 1953, among other writings.

9. W.V.Quine, *Philosophy of Logic*, Prentice-Hall, 1970, pp 2 and 3.

10. Quine defines validity of schemata, including quantificational schemata, as "truth under all interpretations in all non-empty universes". W.V.Quine, *Methods of Logic*, 4th Ed. pp 172-3.

concepts and principles can be applied to fictional or imaginative prose as well as to truth-claims; their logical properties and relationships can be studied without passing through a truth-logic first. Thus Analytic Logic covers a broader range of expressions and yields a greater variety of valid inferences than Mathematical Logic.

The rejection of truth-values and truth-functions as essential features of validity and universality in formal logic does not entail disparagement or rejection of the logic of truth-claims or propositions. It only says that such a logic is not a purely formal logic and is not universal. Awareness of the logical properties and relations of sentences which are asserted to be true or not true is absolutely indispensable for most sound practical reasoning. But the concept of truth is not a purely formal, syncategorematic concept like 'and', 'or', 'not' and "if…then". The word 'true' has at least one very familiar substantive meaning which is spelled out in Chapter 7.

In Analytic Logic, truth-logic is treated as the special logic of the extra-logical predicate "…is true", and/or of the truth-operator, 'T(P)', for "It is true that P". Consequently, a sentence S is not equivalent in meaning to a sentence of the form ''S' is true' or 'It is true that S' as Tarski held, nor need the attribution of truth be limited to higher levels of semantic ascent, as Quine has suggested.[11]

The last four chapters of this book develop truth-logic as an extension of Analytic Logic which centers on valid principles of inference logic for inferring truths from other truths. It applies to all expressions to which "it is true that…" is prefixed, implicitly or explicitly. Differences and similarities between the semantics of A-logic and M-logic are made explicit in the theorems of Analytic truth-logic. One central difference is that the principle of trivalence, '(TP v FP v (~TP&~FP))', i.e., "every expression P is true, false, or neither", is a tautology and a logical truth of analytic truth-logic, while M-logic's Law of Bivalence (often miscalled the "Law of Excluded Middle"), '(TP v FP)', i.e., "P is either true or false", is neither a tautology nor theorem. Most of the statements treated by M-logic as semantic theorems in its metalogic, are simply theorems in Analytic truth-logic.

0.32 Problems of the Truth-functional Conditional

The problems of interpreting '(P ⊃ Q)' as "If P then Q" can now be dealt with separately.

At bottom, the question about truth-functional conditionals is the desireability of taking "If P then Q" to mean the same thing as "Either P is false or Q is true" and "It is false that both P is true and Q is false". According to Mathematical Logic all three expressions are logically equivalent. Mathematical Logicians never deny this. Most of them recognize that this usage differs in certain ways from ordinary usage, but argue that it is the best interpretation for purposes of logic.[12] Granted that logic deals with conditionals, the question is whether the C-conditional is a better explication of conditional statements for the purposes of logic.

In Analytic Logic, "valid" is, by definition, a property only of inferences or of C-conditional statements. Disjunctions and conjunctions without conditionals can be tautologous but never valid. In Mathematical Logic every tautologous conjunction or disjunction can be converted into an equivalent TF-conditional or TF-biconditional which is "valid" in M-logic. Thus, both the conditional and the term 'valid' must be given different meanings in Analytic Logic and Mathematical Logic . The problem is to see what the differences are and decide which serve better the objectives of logic.

---

11. See W.V.Quine, *Philosophy of Logic*, 1970, Prentice Hall, p.11.

12. See Quine, *Methods of Logic* (4th ed), Ch 3,"The Conditional", p 21, in which Quine claims that the material (i.e., truth-funcitonal) conditional is the "most convenient" way to regard conditionals. From the point of view of Analytic Logic, this is not the best way to view conditionals for purposes of logic.

0.321 Problem of Reconciling the TF-conditional and the VC\VI Principle

The VC\VI principle—that a conditional statement "If P then Q" is valid if and only if an argument from the premiss P to a conclusion Q, is valid—is a principle of traditional logic which is accepted in Mathematical Logic as well as Analytic Logic though it is seldom mentioned. If the *non sequitur*s mentioned above are recognized as such, there is a problem of reconciling the TF-conditional with the VC\VI principle.

Suppose one accepts the concept of valid theorems advanced in Mathematical Logic, but agrees that it is false that *every* statement *follows logically* from any contradiction; e.g., one agrees that '(P&∼P), therefore Q' is not a valid form of argument, although '((P & ∼P) ⊃ Q)' is a valid theorem. If one does this, one must either 1) reject the VC\VI principle, or, 2) reject interpreting '⊃' as "if…then". For '((P&∼P) ⊃ Q)' is equivalent to '∼((P&∼P) & ∼Q)', and the latter is clearly a tautology, and thus a 'valid' theorem of Mathematical Logic.

It is not so easy to reject the VC\VI principle. The semantic theory of M-logic reconciles both VC\VI and the interpretation of '⊃' as "if…then", by its definition of valid argument and the truth-functional interpretation of (P ⊃ Q). For, [P ⊃ Q] is called valid in Mathematical Logic if and only if there is no case such that P is true and Q is false—and this is exactly what is required in Mathematical Logic for the 'validity' of any argument from premiss P to a conclusion Q. But this doesn't make '(P&∼P) therefore Q' any less a *non sequitur*. In textbooks of Mathematical Logic the tendency is to ignore the problem, or doggedly insist that these purported *non sequitur*s but are really valid since Mathematical Logic has defined 'valid' to make them so. However, if we agree that the inference-forms are *non sequitur*s, the traditional VC\VI principles can be preserved only if we reject interpreting (P ⊃ Q) as a conditional.

In Analytic Logic this problem does not arise. The VC\VI principle is preserved and there are no *non sequitur*s analogous to those in Mathematical Logic. This is because 1) the meaning of 'if…then' and its truth-conditions are different and 2) a different meaning is given to 'valid' which yields different rules of valid inference in Analytic Logic. There is also a different concept of 'a theorem of logic'. Meanings that are assigned to these terms in Mathematical Logic don't accomplish the goals logicians seek. The appropriate questions are, what are these goals? and do the meanings assigned in Analytic Logic do the desired jobs.

0.322 The Anomalies of "Material Implication"

Second, at a more superficial level, there are what used to be called "paradoxes of material implication". This was a misleading name because they are not true logical paradoxes and they are not really problems of implication. Rather they are problems of interpreting '(P ⊃ Q)' as "if…then". The symbol '(P ⊃ Q)' means the same in mathematical (and analytic) logic as 'Either not-P or Q' and 'Not both P and not-Q)', although the latter are not ordinarily thought to mean the same as 'if P then Q'. The problems may be called anomalies, or deviations with respect to ordinary usage. A better collective name for them would be "anomalies of the truth-functional conditional".[13]

The problem is to reconcile ordinary uses of 'if…then' with certain kinds of if…then statements which are said to be universally true by virtue of their logical form in Mathematical Logic. There are

13. Anomaly: "irregularity, deviation from the common or natural order…" Anomalous—"unconformable to the common order; irregular; abnormal." *Oxford English Dictionary*.

many problems of this sort.[14] Among the simpler ones commonly mentioned are the assertions that, no matter what P and Q may be, the following forms of statement are always true:

$\vdash [\sim P \supset (P \supset Q)]$ — "If P is false, then (if P *is true* then Q *is true*)"
$\vdash [\sim P \supset (P \supset \sim Q)]$ — "If P is false, then (if P *is true* then Q is false)"
$\vdash [Q \supset (P \supset Q)]$ — "If Q is true, then (if P *is true* then Q *is true*)"
$\vdash [Q \supset (\sim P \supset Q)]$ — "If Q is true, then (if P is false then Q *is true*)"
or,
$\vdash [\sim P \supset (P \supset Q)]$ — "If P is false, then (if P then Q) *is true*)"
$\vdash [\sim P \supset (P \supset \sim Q)]$ — "If P is false, then (if P then Q is false) *is true*"
$\vdash [Q \supset (P \supset Q)]$ — "If Q is true, then (if P then Q) *is true*"
$\vdash [Q \supset (\sim P \supset Q)]$ — "If Q is true, then (if P is false then Q) *is true*"

These are all "valid" theorems according to Mathematical Logic, since it can not be the case that the antecedent is true and the consequent false. If '⊃' is not read as "if...then" and the whole expression is replaced by logically equivalent (or synonymous) expressions, it is easy to see why these are said to be laws of logic. For they are each equivalent, or synonymous, respectively to,

$\vdash \sim(\sim P \,\&\, P \,\&\, \sim Q)$, $\vdash \sim(\sim P \,\&\, P \,\&\, \sim Q)$, $\vdash \sim(Q \,\&\, P \,\&\, \sim Q)$, and $\vdash \sim(Q \,\&\, P \,\&\, \sim Q)$,
or $\vdash (P \vee \sim P \vee Q)$, $\vdash (P \vee \sim P \vee Q)$, $\vdash (\sim Q \vee \sim P \vee Q)$, and $\vdash (\sim Q \vee \sim P \vee Q)$,
which are either denials of inconsistent conjunctions, or tautologous disjunctions. But it still does not seem right to call them valid when '$((\sim P \supset (P \supset Q))$' is read "If P is false, then (If P then Q) is true".

In Analytic Logic the removal of these anomalous uses of "if...then" is accomplished by defining 'if...then' differently. The symbol '⊃' is retained as an abbreviation of '$\sim(P\&\sim Q)$', but it is not interpreted as "if...then" in A-logic. Rather, "if...then" is associated with a new symbol '$\Rightarrow$', and $[P \Rightarrow Q]$ is true under different conditions than $[P \supset Q]$. In line with an idea advanced by Jean Nicod a conditional is true if and only if both the antecedent and the consequent are true, and is false if and only if (as in Mathematical Logic) the antecedent is true and the consequent false. When the antecedent is not true, the conditional is neither true nor false.[15]

This solution requires recognition that some indicative sentences may be neither true nor false. In particular an indicative conditional sentence in which the antecedent is not true, or the consequent is neither true nor false, is neither true nor false. This means that in analytic truth-logic, its truth tables must allow three possibilities—an indicative sentence may be true, false, or neither. This rejects the presupposition of Mathematical Logic that every meaningful indicative sentence is either true or false.

The trivalent truth-tables for 'not', 'and', 'if...then' and for the sentential operator, 'T', which are presented in Chapters 7 and 8 are formally the same as truth-tables presented by DeFinetti in 1935 in "The Logic of Probability".[16]

---

14. Under this rubric, C.I. Lewis, listed 28 in "Interesting theorems in symbolic logic", *Journal of Philosophy Psychology and Scientific Methods,* v. X,(1913), pp 239-42.

15. Nicod, J. *Le probleme logique de l'induction*, Paris, 1924 (trans. by P.P.Weiner) as *Foundations of Geometry and Induction,* London, 1930, p. 219

16. Bruno De Finetti—"La Logique de Probabilite", Actualites Scientifiques et Industrialles, 391, *Actes du Congres Internationale de Philosophie Scientifique*, Sorbonne Paris, 1935, IV. Induction and Probabilit, Hermann et CLe, diteurs. Paris. 1936, pp 31-39. Translated into English by R.B. Angell, in *Philosophical Studies*, v. 77, No.1, Jan 1995, pp 181-190.

0.323 Problems Due to the Principle of the False Antecedent

The most interesting problems of the TF-conditional were discovered by prominent proponents of Mathematical Logic in the process of trying to apply Mathematical Logic to the methods of empirical science. These problems include, among others, 1) Hempel's "paradoxes of Confirmation", 2) Carnap's problem of "dispositional predicates", 3) Goodman's problems of "counterfactual conditionals", 4) the problem of "Lawlike statements", 5) the problem of using TF-conditionals in explicating causal statements, and 6) Ernest Adam's problem of reconciling the probability of conditionals with the concept of conditional probability in probability theory. These and other problems are solved in Analytic Logic.

Many of these problems have a common root: according to Mathematical Logic if the antecedent of $(P \supset Q)$ is false, then the conditional as a whole is true no matter what Q may be. This is the first, or the fifth, of the "Paradoxes of material implication" listed above—based on the theorem of Mathematical Logic, $\vdash [\sim P \supset (P \supset Q)]$. Let us call it "the principle of the false antecedent".

This principle makes some sense if '$(P \supset Q)$' is replaced by its equivalent '$(\sim P \vee Q)$' ("either P is false or Q is true"). For this is in line with some ordinary uses. One might indeed say, if "P is false" is true, then "either P is false or Q is true" is true, no matter what Q may be. In analytic *truth*- logic this is a valid implication. However, it is <u>not</u> in keeping with ordinary usage to say that if "P is false" is true, then "If P then Q" must be true. But by the principle of the false antecedent, this is what the Mathematical Logician must always say if '$(P \supset Q)$' is interpreted as "If P then Q)".

This anomaly raises special problems with respect to general statements in common sense and science, some of which are called laws of nature. In both Analytic Logic and Mathematical Logic, the first step in putting a general statement into logical symbolism, is to translate "Every S is P" into '(∀x)(If Sx then Px)'. This may be read as "For every x,…" or "For all x,…" or "No matter what x may be…, if x is S then x is P". This method of treating general statements is one of the great advances of Mathematical Logic over traditional logic. Rules of formation makes it possible to display the precise logical form of complicated statements in both mathematics and empirical science. For example, in mathematics the principle "equals of equals are equal" can be translated into logic symbolism as '(∀x)(∀y)(∀z)(If (Rxy & Rzy) then Rxz)', with 'R' for "…is equal to___".

However, by the principle of false antecedent, If $[(\forall x) \sim Px]$ is true, then '$(\forall x)(Px \supset Qx)$' is true no matter what is put for 'Qx'. For example, in a domain of 3 individuals, {a,b,c} $[(\forall x) \sim Px]$ is

$$(\forall x) \sim Px = (\underline{\sim Pa\ \ } \ \& \ \underline{\sim Pb\ \ } \ \& \ \underline{\ \sim Pc\ \ }),$$

and from this follows: $((Pa \supset Qa)\ \&\ (Pb \supset Qb)\ \&\ (Pc \supset Qc))$ which is $(\forall x)(Px \supset Qx)$ for 3 individuals.

This result creates major problems in trying to apply Mathematical Logic to the empirical sciences.

0.3231 Hempel's "Paradoxes of Confirmation"

Logicians would like to explain the logical connection between observations and inductive generalizations in science. For example, what kinds of observed facts would support, and what facts would refute, the conclusion that all ravens are black? Since "Are all ravens black?" is a question of *empirical* fact, it is assumed that being black is not part of the definition of 'a raven'.

All logicians agree "All ravens are black" would be refuted if one raven was found that was not black. What, then, would confirm it? Carl Hempel, a staunch proponent of Mathematical Logic, makes the amazing assertion,

> . . . we have to recognize as confirming for ['All ravens are black'] any object which is neither black nor a raven. Consequently, any red pencil, any green leaf, any yellow cow, etc., becomes confirming evidence for the hypothesis that all ravens are black.[17]

Hempel labels this a "paradox of confirmation" but does not reject it. He is driven to it by his commitment to M-logic. For if "All ravens are black is true" then for any objects, a,b,c,d,..., which are not ravens, "if a is a raven, then a is black" must be true. As we just saw, [$\sim$Pa $\supset$ (Pa $\supset$ Qa)] is always true according to Mathematical Logic. Thus if a is a red pencil, or a green leaf or a yellow cow, a is certainly not a raven. Hence, in those cases it follows that "if a is a raven, then a is black" is true. Therefore, the generalization is confirmed. The absurdity is obvious. Imagine an application for a research grant to confirm that "All ravens are black" by observing all objects in some region that has no ravens in it. Hempel rejected Nicod's way of solving the problem primarily because it would violate the Transposition Principles of M-logic. (See Section 10.331)

0.3232 Carnap's Problem of "Dispositional Predicates"

Science relies heavily on "dispositional predicates", which are best expressed using conditional predicates. For example, every element in the periodic table is distinguishable by the different conditions of temperature and pressure under which it becomes a solid, a liquid, or a gaseous state.

"($\forall$x)(If x is a time and a is a bit of hydrogen
and a is cooled to -259° C under 760mm mercury pressure at x,
then a changes from gas to a liquid state at x."

"($\forall$x)(If x is a time and a is a bit of iron
and a is heated to 1530°C under 760mm mercury pressure at x,
then a changes from solid to a liquid state at x."

In science these two statements, one about hydrogen and one about iron, are accepted as sound. The first describes the *disposition* of hydrogen to change from liquid to gas at -259°C, and the second the *disposition* of iron to change from solid to liquid at 1530°C. Suppose the object before me, a, is bit of liquid mercury. Then it is neither a bit of hydrogen, nor a piece of iron, and therefore the antecedents are false in both sentences, and by the principle of the false antecedent, the conditionals are true in both cases—at any time if this object, a, (which happens to be a bit of mercury) is cooled to -259°C it changes from gas to liquid, and also if heated to 1530°C, it changes from solid to liquid. Patently false and absurd! Yet, this follows from interpreting "If P then Q" as (P $\supset$ Q) in Mathematical Logic.

Around 1936, the problem of dispositional predicates was presented by Rudolf Carnap, another brilliant proponent of Mathematical Logic, who tried valiantly to explicate scientific methodology using Mathematical Logic. He explained this problem using the concept of "solubility".[18] Can we not define the predicate "...is soluble in water" as "whenever x is put in water, x dissolves"? To translate this statement into logical symbolism, the proposed definition suggests that we restate it as: "for any x, x is

---

17. Carl G. Hempel, "Studies in the Logic of Confirmation, MIND (1945). Reprinted in Brody, Baruch A. and Grandy, Richard E., *Readings in the Philosophy of Science*, Prentice Hall, NJ, 2nd Ed., 1989, see p 265.

18. Rudolf Carnap, "Testability and Meaning", *Philosophy of Science,* v. 3 & v.4, 1936,1937. Reprinted in Feigl, Herbert and Brodbeck, May *Readings in the Philosophy of Science*, Appleton-Century-Crofts, Inc, 1953. pp 47-92 see especially pp 52-3, Section 4. Definitions.

soluble in water if and only if at any time, t, if x is put in water at t, then x dissolves at t." In the symbolism of Mathematical Logic with "if…then" interpreted as '⊃', this is symbolized as

$$(\forall x)(Sx \equiv (\forall t)(Pxt \supset Dxt))$$

with 'Sx' standing for 'x is soluble in water', 'Pxt' standing for 'x is placed in water at the time t' and 'Dxt' standing for 'x dissolves at time t'. Suppose now, said Carnap, we have an object c, which is a match that was completely burned yesterday. "As the match was made of wood, I can rightly assert that it was not soluble in water; hence the sentence … which asserts that the match is soluble in water, is false". But by Mathematical Logic (and Analytic Logic) if the generalization above is true, 'c is soluble in water must be true. For by the definition of soluble,

$$\text{`}(S\underline{c} \equiv (\forall t)(P\underline{c}t \supset D\underline{c}t))\text{' is true.}$$

and since the match c was never put in water, i.e., '$(\forall t)\sim Pct$' is true, it follows that the antecedent of '$(\forall t)(P\underline{c}t \supset D\underline{c}t)$' is false at all times, hence by the principle of the false antecedent, every instantiation of the conditional as a whole is true,—i.e., the definiendum, '$(\forall t)(P\underline{c}t \supset D\underline{c}t)$' is true, and thus that '$S\underline{c}$' is true, i.e., **the match, c, is soluble in water** "in contradiction to our intended meaning".

Carnap concluded that dispositional predicates cannot be defined; i.e., "predicates which enunciate the disposition of…a body for reacting in such and such a way to such and such conditions,. e.g., 'visible', 'smellable', 'fragile', 'tearable', 'soluble', 'insoluble', etc.,…cannot be defined by means of the terms by which these reactions and conditions are described…"[19]

This argument has been widely cited as a ground for rejecting "operational definitions". What it really shows however, is the inadequacy of the truth-functional conditional of Mathematical Logic for this purpose. In Analytic Logic, dispositional predicates of this sort can be defined in synonymity statements, with a C-conditional in the definiendum. Since Analytic Logic does not accept the principle of the false antecedent, there are no contradictions with the intended meaning.

The principle that Carnap used in his argument may be called the *generalized principle of the false antecedent*: "If '$(\forall x)\sim Px$' is true, then '$(\forall x)(Px \supset Q)$' is true no matter what Q may be." I.e., if every instantiation of the antecedent is false, then every instantiation of the quantified conditional is true,—i.e., '$(\forall x)(Px \supset Q)$' is true. This, like the particular form of that principle, is a fallacy from the point of view of ordinary usage and is recognized as invalid for C-conditionals in Analytic Logic.

0.3233 Goodman's Problems of "Counterfactual Conditionals"

In 1946 Nelson Goodman, another strong defender of Mathematical Logic, implicitly invoked this same principle to introduce "The Problem of Counterfactual Conditionals":

> What, then, is the *problem* about counterfactual conditionals? Let us confine ourselves to those in which antecedent and consequent are inalterably false—as, for example, when I say of a piece of butter that was eaten yesterday, and that had never been heated,
>
> If that butter had been heated to 150°F., it would have melted.
>
> Considered as a truth-functional compounds, all counterfactuals are true, since their antecedents are false. Hence,
>
> If that butter had been heated to 150° F., it would not have melted.

19. Opus cit., p 52.

> would also hold. Obviously something different is intended, and the *problem is to define the circumstances under which a given counterfactual holds while the opposing conditional with the contradictory consequent fails to hold.*[20]

The generalized Principle of the False Antecedent enters with the statement that the butter had never been heated—hence the '$(\forall t) \sim H\underline{c}t$' is assumed true. This makes both $(\forall t)(H\underline{c}t \supset M\underline{c}t)$ and $(\forall t)(H\underline{c}t \supset \sim M\underline{c}t)$ true, and by universal instantiation the two conflicting conditionals follow.

In Analytic Truth-logic, since the generalised principle of the false antecedent does not follow from a C-conditional, this problem does not exist. Further, among its theorems is one which solves the problem as Goodman just stated it above. This principle is:

T8-818. Valid[T(P => Q) <=> F(P => ~Q)]

i.e., "If 'If P then Q' is true, then 'If P then not-Q' is false" is a valid conditional. Thus "the circumstances under which a given counterfactual holds while the opposing conditional with the contradictory consequent fails to hold" is provided by Analytic Logic. It makes them contraries. They can not be true at the same time. This theorem is derived in a clear fashion in Chapter 8 from the meaning assigned to 'if...then' following Nicod's suggestion, and is supported by trivalent truth-tables.

However, this does not solve all of Goodman's problems. For the problem is not just to explain how a particular conditional can be true and "the opposing" conditional false; we have to show on what positive grounds lawlike statements (generalized inferential conditionals of the form '(∀x)(TPx => TQx)') can be supported or confirmed. This is the the problem of what confirms or supports the "validity" of a generalized inferential conditional.

0.3234 The Problem of Explicating Causal Statements

The unfortunate principle of the false antecedent is at the root of several other problems which will be discussed, including the problem of inferences to causal connections, and the disparity between conditional probability and the probability of a conditional.

The simplest cases of establishing a causal connection seem to rest upon establishing two sorts of conditionals: 1) if the cause occurs at any time t, then the effect occurs at t (or shortly thereafter), and 2) if the cause does not occur then the effect does not occur. If conditionals are construed as TF-conditionals, then any effort to base a causal conclusion on observed evidence dissolves into chaos. For by the principle of the false antecedent, every <u>occurrence of C</u> implies "If C does not occur then E occurs" which is the opposite of 2), and every <u>non-occurrence of C at time t</u> implies "If C occurs at t, then E does not occur at t (or shortly after)", which is the opposite of 1). To be sure, $(C \& E)$ also implies $(C \supset E)$ and $(\sim C \& \sim E)$ also implies $(\sim C \supset \sim E)$, but due to the principle of the false antecedent, too many TF-conditionals follow from the facts and the result is chaos.

The analysis of causal statements and the grounds for making them is notoriously difficult. But even such plausible, though incomplete, attempts as Mill's Methods can not be accounted for with the TF-conditional.

The C-conditional of Analytic Logic, being free of the Principle of the False Antecedent, does not have these multiple consequences and makes possible a more plausible approach to the analysis of causal statements.[21]

---

20. Goodman, Nelson, "The Problem of Counterfactual Conditionals", (Italics mine) *Journal of Philosophy,* v. 44 (1947). Reprinted in Goodman, Nelson, *Fact, Fiction and Forecast*, 1955, 1965. Quote is from the latter, p. 4.

21. See Sections 9.34 and 10.336

0.3235 Ernest Adam's Problem of Conditional Probability

In 1973 Ernest Adams showed that what is called a conditional probability in probability theory is different from the probability of a conditional if the conditional is the TF-conditional of Mathematical Logic.[22] In probability theory, the probability that [(P& ~P) ⊃ Q] is 1.0, (since this is the probability of [~(P & ~P & ~Q]), but according to probability theory, there is no probability at all that Qis true, if (P & ~P) is true, because there can be no cases in which (P& ~P) is true. Surely probability theory is correct.

Again, if we have a basket with ten pieces of fruit—six apples, only one of which brown, and four pears, two of which are brown—the probability that a piece of *fruit* in the pot is brown if it is an apple is 1/6 (or one out of 6) according to probability theory. But with the "if...then" of Mathematical Logic, the probability of (If x is an apple then x is brown) is the same as the probablity of (either x is not an apple or x is brown), and the probability of that is ½; since five of the ten pieces of fruit are either not apples (4) or are brown though an apple (1). Thus conditional probability in probability theory is wildly different from the probability of a TF-conditional. Again the principle of the false antecedent screws up the works.

In Analytic Logic, the probability of a C-conditional being true is the probability of the antecedent and the consequent being both true over the probability of the antecedent's being true; this is derived from the meaning of the C-conditional and is precisely the same as the definition of conditional probability in standard probability theory. An argument against the possibility of such a definition is answered.

0.33 The Problem of Counterfactual, Subjunctive and Generalized Lawlike Conditionals

Each of the problems in the five preceding sections, 0.3231 through 0.3235, was due at bottom to the principle of the false antecedent. Analytic Logic avoids those problems with the introduction of Nicod's idea of a C-conditional. It denies that the falsehood of the antecedent makes a conditional true, or that the truth of the consequent, by itself, can makes a conditional true.

However, this is not the end of it.

The problems of counterfactual conditionals, subjunctive conditionals and lawlike statements are not solved merely by eliminating the principle of the false antecedent. The C-conditional is true if antecedent and consequent are both true, false if antecedent is true and consequent is false, and neither true nor false if the antecedent is not true or the consequent is neither true nor false. But giving the conditions of truth and falsity of the C-conditional can not explain how or why we accept or reject conditionals when we don't know whether the antecedent is true or not, or even when we know that the antecedent is not true. On what grounds, then, or in what way, do we accept or reject a conditional when either a) we know the antecedent is not true, or b) we don't know whether or not the antecedent is true or false?

What is required is a radical step; one that requires a different mind-set in thinking about how and why conditionals, as such, are accepted. One in which validity does not depend on truth-values as in Mathematical Logic. One breaks off asking if they are true or false and asks, independently, if they are valid. To grasp that truth and validity are separate and independent, requires relinquishing habits of mind deeply embedded in ordinary discourse and presupposed in Mathematical Logic.

A sharp distinction must be drawn between truth-claims about conditionals and validity claims. C-conditionals, by themselves, are basically inference vehicles—devices which sanction the passage from accepting one thought to accepting another. As such they are are symbolized by the forms [P ⇒ Q],

---

22. Adams, Ernest, *The Logic of Conditionals*, Reidel, 1975

[~Q => P], [TP => TQ], [TP => ~FQ], etc. They must not be confused with truth claims about them, symbolized by T[P => Q], T[~Q => P] or F[TP => TQ]. To ask if a conditional is true or false, is to ignore part of its essential nature *qua* a conditional. To talk about its truth or falsity is to talk only about the truth-values of its components, rather than the validity of passing from one component to the other. The assertion that a conditional is *true* is not a conditional statement—it is a truth-assertion; just as the *denia*l of a conjunction is not a conjunction, but a *denial* of conjunction. Expressions which assert a C-conditional is true or—'T(P => Q)'—are never claims of validity, and validity claims about a conditional—'Valid[P => Q]'—never entail that [P => Q] is true (though valid statements can not be false).

Validity is a possible property of an inferential conditional whether or not it is true. Inferential conditionals, qua conditionals, may be valid or invalid. In Analytic Logic, *logically valid* inferences or C-conditionals are ones in which the meanings of the conclusion or consequent are the same or contained in the meaning of the premisses or antecedent. In Analytic *Truth*-logic they are vehicles for passing from one truth-assertion to another because the meaning of the consequent is contained in the meaning of the antecedent. That an argument, or a conditional is *valid* does not entail that either premisses (antecedent), conclusion(consequent), or the inference (conditional) as a whole is *true*. Determinations of the validity or invalidity of conditionals and determinations of the truth or falsity of conditionals are processes which differ in kind.

To determine whether a conditional, [P => Q] or [TP => TQ], is *true, false, or neither*, is to determine which one of nine possible states of affairs exist by determining whether P is true, false, or neither, and whether Q is true, false, or neither. Given these determinations, the related row in the truth-table represents the one case, of the nine possible cases, which exists in the domain of reference, and tells whether the conditional as a whole is true, false or neither, in that particular case.

To determine *logical validity* one must determine whether the meaning of the conclusion (or consequent) is logically contained in premisses (or antecedent) of the inference (or conditional), and whether among the whole set of possibilities (the truth-table as a whole) there is at least one possible case in which (P => Q) is true, and none in which it is false.

Valid conditionals need not be true at all, but in truth-logic to claim they are logically valid, is to claim they might, logically, be true, and can not, logically, be false. Lawlike statements are generalizations of inferential conditionals , e.g., [(∀x)(TPx => TQx)]. Logically valid conditionals are incapable of having false instances by virtue of their meanings. But also, a C-conditional may consistently be believed to be "empirically valid" (or "E-valid") if it is it has some true instantiations and no known false instantiations.

In empirical sciences, particular observations logically entail the truth, falsity or neither (but not the validity) of particular C-conditionals. If [P & Q] is true, then [P => Q] is true. If a certain conditional predicate has no false instantiations, and many true ones, one is free to conceive and believe consistently in a natural connection, expressible in a generalized inferential conditional; though the nature of the connection may be little more than a constant conjunction or coexistence of the consequent with the antecedent. Such "empirically valid" conditionals, when quantified and generalized on the basis of careful observations, have proved very useful in human endeavors.

In probability determinations, the number of all true instantiations of a conditional predicate in a certain domain divided by the number of true-instantiations-plus-false-instantiations, gives the ratios or frequencies on which probability judgment may be made.

It does not matter, for logical validity in Analytic Logic, whether or not a conditional is always true—i.e., true for all interpretations of its variables. A conditional predicate does not have to be true for all values of its variables in order to be valid either logically or empirically. If the antecedent is not true, or the consequent is neither true nor false, the conditional is neither true nor false, hence not-true. For every signifcant and useful conditional there are many situations in which it is neither true nor false, hence not-true. No finite number of true instantiations can prove a conditional logically valid. At most the logical validity of a conditional entails that it can not be false and it may be true.

This feature of C-conditionals, the separation of truth-determinations from validity determinations, makes possible rigorous methods of deciding the logical or empirical validity or invalidity, of contrary-to-fact C-conditionals, subjunctive C-conditionals, and lawlike generalized C-conditionals. All of which is beyond the reach of the TF-conditional.

0.34 Miscellaneous Other Problems

In addition to the problems mentioned there are problems of the relation of Mathematical Logic to traditional logic, including the Squares of Opposition and theory of the Syllogism in Aristotelian logic. Here though both Mathematical Logic and Analytic Logic have differences to explain, Analytic Logic is closer to traditional logic and can explain why it differs better than standard answers from Mathematical Logic that try to explain why *it* differs.

There are also questions Analytic Logic must face where it does not fully accept certain principles which are widely thought to be universal—notably the Principle of Addition and principles of Transposition. Both are accepted, but only in limited forms, in Analytic Logic. For, example, in Analytic Logic "[If P then Q] is true" does not entail "[If not-Q then not-P] is true"; for the very case which would make the first true (P's being true and Q's being true) is contrary to what would make the second true (Q's being false and P's being false). But some versions of transposition are valid, including "[If (If P then Q) is true, then (if not-Q then not-P) is not false]". None of these restrictions affect the tautologies called "principles of transposition" in Mathematical Logic; e.g., $[(P \supset Q) \equiv (\sim Q \supset \sim P)]$ is still a tautology.

0.4 Analytic Logic

Analytic Logic is intended to solve or eliminate the problems which we have just enumerated without forfeiting those concepts, theorems, and methods which have produced the positive achievements of Mathematical Logic.

0.41 The Over-View

Chapters 1 through 8 constitute a progressive development from the introduction of SYN and CONT into a negation-free fragment of M-logic to a full version of Analytic Truth-logic which deals with the problems of Mathematical Logic discussed in this book. In this sequence, the purely formal theory of Analytic Logic is presented in Chapter 6, and Analytic Truth-logic and its applications, is developed as an extension of formal logic in Chapters 7, 8, 9 and 10.

0.42 The Logistic Base of Analytic Logic

The logical base of formal A-logic in Chapter 6 is displayed in TABLE 0-2, Analytic Logic.

The first part of this table, SCHEMATA (OR WFFS), specifies the primitive symbols and the rules for combining primitive symbols to form meaningful complex symbols that Analytic Logic will use to designate the abstracted logical forms of predicates that it talks about. The complex abstract forms are displayed in schemata or wffs (well-formed formulae). In the second part, AXIOMS AND RULES OF INFERENCE, six categorical statements (Axioms) are presented that assert that certain pairs of wffs stand in the relationship of logical synonymy or 'Syn'. The rules are given by which one can produce new theorems from the axioms, either by substituting a synonym for a component expression, or by uniformly substituting certain other well-formed expressions for all occurrences of a variable component, or by instantiating a predicate. There are no quantified expressions among the axioms since quantified formulae and their rules are derivable from definitions and acts of repeatedly instantiating some n-adic predicate on all n-tuples of individuals in the field of reference.

In general, theorems of analytic logic consist in assertions that certain predicates, here called Logical Predicates, apply to certain kinds of expressions. In TABLE 0-2 only the logical predicate 'Syn' occurs, for this is the fundamental logical predicate which occurs directly or indirectly in the definitions of all other logical predicates.

But Analytic Logic also makes use of defined terms, which do not occur in TABLE 0-2. These defined terms are shown in TABLE 0-3. They are of two sorts: those that abbreviate complex wffs making them more easy to grasp, and those that define other Logical Properties or Relations. The first kind of defined expressions are predicate operators; they are used to abbreviate or stand for more complex forms of predicate schemata. The expressions on the right-hand side (definientia) are all well-formed formulas according to TABLE 0-2, while the expression on the left hand-side (definienda) introduce symbols which do not occur in TABLE 0-2, but which stand for the expressions on the right. All definitions of predicate operators are familiar from Mathematical Logic. The second kind of defined expressions are Logical Predicates which are used to signify the presence of properties in the wffs (being Inconsistent or Tautologous or Valid) or relations between wffs (one containing the other, or entailing or implying the other). These are the properties and relations logic seeks to investigate. All of them are built up ultimately from the relation Syn and one or more predicate operators such as conjunction, or negation, in the subject expressions.

0.43 The Progressive Development of Concepts in Successive Chapters

Analytic Logic is developed in successive steps beginning with its central concepts of logical synonymy and logical containment, in order to show how these concepts are related to the concepts of negation, inconsistency, tautology, truth, and the truth-functional accounts of "logical consequence", "implication", or "validity" which are central to Mathematical Logic.

In each Chapter from 1 to 6 a new predicate operator or a new Logical Predicate is added, until in Chapter 6 we have the purely formal system of Analytic Logic. The extra-logical truth- operator, 'T', is added in Chapter 7 to give a truth-logic which can be related to the semantics of M-logic (without the C-conditional), and in Chapter 8 the C-conditional is re-introduced to produce Analytic Truth-logic. Chapter 9 deals with the validity and probability of C-conditionals with respect to empirical facts. Chapter 10 summarizes the way A-logic solves the problems of M-logic.

In each of the Chapters 1 through 8 theorems are limited to those involving the primitives notions and logical predicates that have been introduced up to that point. By inspection of the name of a theorem and the symbols in it, one can tell the chapter in which it is introduced and identify the axioms or rules peculiar to that chapter which are required to establish the theorem.

In Chapters 1 through 4 only the logical predicates 'SYN' or 'CONT' occur in theorems. In Chapter 1 these predicates are applied to pairs of sentential schemata formed from sentence letters 'P', 'Q', 'R' etc. with only the operators for "and" and "or", namely '&' and 'v'. In Chapter 2 they apply in addition to predicate schemata with individual constants or argument position-holders, such as 'P<1>', 'R<1,2>', 'P<a>', 'R(<1,b>', etc. In Chapter 3 they are applied to expressions with individual variables bound to quantifiers, as in '(∀x)(Px v Qx)' and '(∃y)((∀x)(Rxy v Ryx)'. It is remarkable how much of quantification theory can be gotten from this base withou negation.

In Chapter 4 the negation sign, '∼', is added for the first time with a new axiom for Double Negation. The logical operators 'v', '⊃' and '≡' are defined in terms of '&' and '∼', reducing the axioms of Chapter 1, to five, and '(∃x)' is defined in terms of '(∀x)' and '∼'. Syn and Cont are applied

TABLE 0-2
Analytic Logic—The Formal System

I. SCHEMATA (OR WFFS)

1. Primitives:

Grouping devices	), (,], [, $>$, $<$
Predicate letters (PL):	$P_1, P_2, \ldots, P_n$.[Abbr. ‘P’, ‘Q’, ‘R’]
Argument-position-holders (APH):	1, 2, 3,.
Individual Constants (IC):	$a_1, a_2, \ldots a_n$, [Abbr. ‘a’, ‘b’, ‘c’]
Individual variables (IV):	$x_1, x_2, \ldots x_n$, [Abbr. ‘x’, ‘y’, ‘z’]
Predicate operators:	&, $\sim$, $\Rightarrow$
Quantifier:	$(\forall x_i)$
Primitive (2nd level) Predicate of Logic:	Syn

2. Formation Rules

FR1. $[P_i]$ is a wff
FR2. If P and Q are wffs, [(P&Q)] is a wff.
FR3. If P is a wff, $[\sim P]$ is a wff.
FR4. If P and Q are wffs, $[(P \Rightarrow Q)]$ is a wff.
FR5. If P_i is a wff and each t_j $(1 \leq j \leq k)$ is an APH or a IC, then $P_i\langle t_1, \ldots, t_k\rangle$ is a wff
FR6. If $P_i\langle 1\rangle$ is a wff, then $[(\forall x_j)P_i x_j]$ is a wff.

II. AXIOMS AND TRANSFORMATION RULES

Axioms

Ax.6-1.	$\models$ [P Syn (P&P)]	[&-IDEM1]
Ax.6-2.	$\models$ [(P&Q) Syn (Q&P)]	[&-COMM]
Ax.6-3.	$\models$ [(P&(Q&R)) Syn ((P&Q)&R)]	[&-ASSOC1]
Ax.6-4.	$\models$ [(P&(QvR)) Syn ((P&Q)v(P&R))]	[&v-DIST1]
Ax.6-5.	$\models$ [P Syn $\sim\sim$P]	[DN]
Ax 6-6.	$\models$ [(P & (P$\Rightarrow$Q)) Syn (P & (P$\Rightarrow$Q) & Q)]	[MP]

Principles of Inference

R6-1. If $\models$ P, and Q is a component of P, and $\models$ [Q Syn R]
then $\models$ P(Q//R) [SynSUB]

R6-2. If $\models$ R and $P_i\langle t_1, \ldots, t_n\rangle$ occurs in R, [U-SUB]
and Q is a suitable h-adic wff, where $h \geq n$,
and Q has an occurrence of each numeral 1 to n,
and no individual variable in Q occurs in R,
then $\models [R(P_i\langle t_1, \ldots, t_n\rangle/Q)]$

R6-3. If $\models P\langle t_1, \ldots, t_n\rangle$ then $\models P\langle t_1, \ldots, t_n\rangle(t_i/a_j)$ [INST]

TABLE 0-3

Abbreviations, Definitions

Predicate Operators:

Df6-1. [(P & Q & R) Syn$_{df}$ (P & (Q & R))]
Df6-2. [(∀x)Px Syn$_{df}$ (Pa$_1$ & Pa$_2$ & … & Pa$_n$)] [Df '∀']
Df6-3. [(P v Q) Syn$_{df}$ ~(~P & ~Q)] [Df 'v',DeM1]
Df6-4. [(P ⊃ Q) Syn$_{df}$ ~(P & ~Q)] [Df '⊃']
Df6-5. [(P ≡ Q) Syn$_{df}$ ((P ⊃ Q)&(Q ⊃ P))] [Df '≡']
Df6-6..[(∃x)Px Syn$_{df}$ ~(∀x)~Px] [Df '(∃x)']
Df6-7. [(P <=> Q) Syn$_{df}$ ((P => Q)&(Q => P))] [Df '<=>']

Logical Predicates

Df 'Cont'. ['(P Cont Q)' Syn$_{df}$ '(P Syn (P&Q))']
Df 'Inc'. ['Inc(P)' Syn$_{df}$ '(((P Syn (Q&~R)) & (Q Cont R))
v ((P Syn (Q&R)) & Inc(R))
v ((P Syn (Qv R)) & Inc(Q) & Inc(R))
v ((P Syn (Q => R)) & Inc(Q&R))']
Df 'Taut'. ['Taut(P)' Syn$_{df}$ 'Inc(~P)']
Df 'Valid'. ['Valid(P => Q)' Syn$_{df}$ '((P Cont Q) & not-Inc(P&Q))']

to pairs of expressions in which the negation sign, '~', occurs, or to expressions which are synonymous, by the new axiom and definitions, with expressions in which negations signs occur (e.g., expressions like '(P ≡ (P v P))', and '((∃x)(∀y)Rxy ⊃ (∃x)Rxa)'). Negation was introduced *after* '&', 'v' and quantifiers were introduced in Chapters 1 through 3, in order to make clear that the relations of Syn and Cont are not essentially affected by the introduction of either negation or '⊃', the TF-conditional. The definitions introduced up to this point are the same as those in Mathematical Logic. The cumulative result in Chapter 4 is a theory of logical synonymy and containment for the wffs of Mathematical Logic.

In Chapter 5, the logical predicates 'Inc' and 'Taut' are defined in terms of logical containment, negation and conjunction, and the set of theorems and theorem schemata of Mathematical Logic are derived as INC-and TAUT-theorems from axioms and rules of Analytic Logic. In a sense this proves A-logic is complete with respect to M-logic. The concept of a "valid inference" in A-logic is introduced. 'Logical validity' is defined in term of containment and not being inconsistent. Using this definition, a large sub-set of inferences from one wff of M-logic to another, are proven valid in A-logic. The validity (for common sense and A-logic) of *these* inferences accounts for the wide acceptability of M-logic. In the same section, the invalidity (according to A-logic) of M-logic's "valid *non sequiturs*" is established.

In Chapter 6 the new primitive, '=>' for the C-conditional is introduced, and with it the VC\VI principle which correlates valid conditionals with valid inferences. With an additional axiom from which *modus ponens* follows, and a new clause in the definition of 'inconsistent' which defines the inconsistency of a conditional, the basic system of formal Analytic Logic is completed. VALIDITY-theorems consisting of valid C-conditionals are derived, as well as many INC- and TAUT-theorems which are not found in M-logic. No anomalies of irrelevance, or "valid *non sequiturs"* appear in the conditional logic of Chapter 6.

The problems of Mathematical Logic's logic in science and common sense are provided with solutions in chapters, 7, 8, 9, and 10. For that purpose we supplement the formal A-logic of Chapter 6 with a special logic of the predicate 'is true' and the operator 'It is true that…' which we call Analytic Truth-logic. The supplementary Axioms, Definitions, and Principles of Inference for analytic truth-logic are shown in TABLE 0-4.

In Chapter 7 the *extralogical* predicate operator, 'T', is introduced as an operator which can operate on wffs, sentences and predicates to form new well-formed expressions. The symbol '=>' is excluded from Chapter 7 in order to yield a system of analytic truth-logic which can be compared to the standard semantics of M-logic and tell how the latter should be adjusted to accord with Analytic Logic while leaving the theorems of M-logic unchanged. A distinction is made between *logical containments*, *entailments*, and *analytic implications*; the last two are based on the meanings of extra-logical presuppositions and terms. *Analytic implications* are only *de dicto* valid while entailments can be *de re* valid. For example, the principle of Addition, as the theorem |= Valid_I [TP, **.:**T(PvQ)], is only *de dicto* valid, whereas |= Valid [T(P & Q), **.:** TQ] can be valid *de re*. The idea is that in the latter case, the truth of the antecedent, insures that the consequent and all its compnents will be true of reality, but in the case of Addition, though it follows from extra-logical linquistic meanings, the truth of the antecedent does not insure that all of its components give true information about reality. This distinction helps to keep *de re* reasoning on track.

TABLE 0-4
Analytic Truth-logic
Adds the following:

1. Primitive (2nd Order) Predicate: T
2. Formation Rule: FR7-1. If P_i is a wff, then $T[P_i]$ is a wff.
3. Definitions (Abbreviations) Predicate Operators:
   Df 'F'. [F(P) Syn_{df} T(~P)]
   Df '0'. [0(P) Syn_{df} (~TP & ~FP)]
4. Axioms: T-Ax.7-01. TP Syn (TP & ~FP)
   T-Ax.7-02. FTP Syn ~TP
   T-Ax.7-03. T(P & Q) Syn (TP & TQ)
   T-Ax.7-04. T(P v Q) Syn (TP v TQ)
   T-Ax.7-05. T(~P & (PvQ)) Cont TQ
   T-Ax.8-01. T(P => Q) Syn T(P&Q)
   T-Ax.8-02. F(P => Q) Syn T(P&~Q)
5. Principle of Inference:
   R.7-2. If Inc(P) then |= ~T(P)

Chapter 8 builds on all of the primitives and wffs of preceding chapters, including ‘T’ and ‘=>’. The result is Analytic Truth-logic, the extension of the formal Analytic Logic of Chapter 6 indicated in TABLE 0-4. It emphasizes the distinction between C-conditionals with a ‘T’ prefixed to the whole, and C-conditionals with T-operators prefixed to components but not to the whole. The latter are inferential conditionals, and only they can be *valid* in analytic truth-logic. All Validity-theorems are *de dicto* valid but some are *De Re* Valid Conditionals and some are only *De Dicto* Valid. The latter are based on analytic-implication theorems, and include principles of truth-tables among other things, but there are rules for limiting their use when reasoning about what is true or false *de re*. Theorems asserting Logical Truth and Logical Falsehood are purely *De dicto*, and have no relevance to non-linguistic reality.

Chapter 9 explores the logical connections between empirical data and C-conditionals in arriving at empirical generalizations, lawlike statements, credible subjunctive or contrary-to-fact statements, dispositional predicates, causal statements and probability statements. Solutions to M-logic’s problems in these areas are presented in this chapter.

Chapter 10 summarizes how analytic truth-logic avoids the logical paradoxes and solves the problems which arose as proponents tried to incorporate the logical thinking of empirical scientists and common sense reasoners under the rubric of Mathematical Logic.

0.44 Other Extensions of Analytic Logic

Analytic Logic in the basic *narrow* sense, is just the formal logic developed in Chapter Six, which applies such predicates as “…logically contains…”, “…is inconsistent”, “…is logically valid” to forms of expression which contain only the syncategorematic “logical” words, ‘and’, ‘not’, ‘if…then’ and expression defined from them.

Analytic Logic in the *broad* sense includes all of its extensions into other disciplines and fields of knowledge. For each extension of logic there is a substantive predicate or group of predicates, which in effect define a field of investigation. From definitions with ‘Syn_{df}’ of these basic terms, if their basic logical form is clear, the postulates and axioms of the discipline will emerge and theorems will be derived, constituting the special logic of that discipline. In this book we can not pursue logic into these other fields, except for the logic of “…is true”. But we can point out that certain changes would result from using Analytic Logic instead of Mathematical Logic in these special logics. These extensions include, besides truth-logic, the logics of questions, and commands, deontic or axiological logics of value judgments, and many other logics with greater or lesser degrees of general application.

A logic of the predicate ‘<1>is different from<2>with respect to<3>’, may cover the area that has been called the logic of identity, and solve some of the problems in identity theory.[23] The logic of sets or classes, is the logic of the predicate ‘<1>is a member of<2>’. Paradoxes like Russell’s paradox which have beset this field may be avoided with the help of distinctions drawn in Analytic Logic. That certain modes of derivation produce contradictions shows only that those modes do not preserve validity.

The concept of generative definitions—including the concept of recursive functions—may be handled using dispositional predicates of Analytic Logic in ways that remove problems due to extensionalism. Some conjectures: Cantor’s theorem can perhaps be reformulated with the help of A-logic, based on the

---

23. The predicate “<1> is different than <2> with respect to <3>” is a POS predicate. As defined in Mathematical Logic, “<1>is identical to<2>” is a NEG predicate. In this Analytic Logic, “<1>is identical with<2>” Syn “(∀x) ~(<1>is different from<2>with respect to x)”. Synonymy is not the same as identity and some problems in identity theory are solved by preserving that difference.

analysis of the concept of a generative definition, asserting the impossiblity of producing a generative definition of entities in a non-finite domain, which will cover all entities definable by other generative definitions in that domain. Cantor's Paradox, like Russell's paradox, will fail to be a paradox in A-logic, because no inconsistent conditional can be a valid conditional. The logic of mathematics—the foundations of mathematics—can be rid of some of its problems without loss by starting with a generative definition of positive integers within the apparatus of Analytic Logic.

The logics of different branches of science are the logics of their basic predicates. In physics one investigates properties and relations of purely physical entities; in biology, properties and relationships of biological organisms. The branches of knowledge may be organized on the basis of a hierarchical structure of concepts, like the tree of Porphyry, in which the complexity of the meanings of the concepts is in inverse proportion to their extensions in the actual world.

In addition to the logics of special branches or sub-branches of knowledge—of truth-investigations—it is most important that rigorous treatment be provided for the logic of questions, directives and value-judgments. In a world where many individuals are increasingly making decisions about the ends towards which human energies should be expended, these areas may well be more important for the survival of human society than further advances in our knowledge about the world.

0.45 New Concepts and Principles in Analytic Logic

The solutions offered here in an effort to provide a broader, stronger theory of logic require the revision of several older concepts and the introduction of some new ones. The success of Analytic Logic as a whole is based on the interrelations of these concepts and principles with each other. Working acceptance of this set of inter-related concepts is necessary if Analytic Logic is to develop successfully.

Among concepts to be revised, is the concept of a *theorem of logic*.

In M-logic it is often said that '(∼P v P)' is a theorem, that '(∃x)((∀y)Rxy ⊃ (∃z)Rzz)' is a theorem, etc. More carefully, Quine says '(∼P v P)' and '(∃x)((∀y)Rxy ⊃ (∃z)Rzz)' are theorem schemata, not theorems. Theorems, he says, are the indicative sentences which have these forms and are thereby logically true. He says this because (i) wffs with mere sentence letters or predicate letters 'P' or 'R' are neither true nor false and (ii) he presupposes that theorems are true statements.

Analytic Logic, in contrast, distinguishes several kinds of theorems depending on the Logical Property or Relation which is asserted to apply. The job of logic is conceived as involving several jobs, each investigating the applicability of some Logical Predicate to linguistic expressions.

First, there are Syn-theorems and Cont-theorems which assert the logical synonymy of two expressions, or the logical containment of one in the other. Examples include:

T1-05. |= [(P v (Q & R)) Syn ((P v Q) & (P v R)]
T3-34. |= [(∃x)(∀y)Rxy Cont (∃x)Rxx]
T4-37. |= [(∀x)(Px ⊃ Qx) Cont ((∀x)Px ⊃ (∀x)Qx))]
T6-13. |= [((P ⇒ Q) & P) Cont Q]

These kinds of theorems are found in all chapters from Chapter 1 on, but the basic principles of logical synoymy and containement are laid down in Chapters 1 to 4.

Next there are TAUT-theorems and INC-theorems. In Analytic Logic both the wffs (schemata) and the statements that Quine calls theorems of logic are identified as *tautologous* expressions, i.e., negations of *inconsistent* expressions. Wffs are abstract expressions but they can be inconsistent or tautologous. These occur in Chapter 5 and subsequent chapters. Examples include Rosser's axioms for M-logic:

T5-501b′. TAUT[P ⊃ (P & P)]	T5-501b. INC[(P & Q) & ~P]
T5-136a′. TAUT[(P&Q) ⊃ P)]	T5-136a. INC[P & ~(P v Q)]
T5-05′. TAUT[(P ⊃ Q) ⊃ (~(Q & R) ⊃ ~(R & P))]	T5-05. INC[~(P & ~Q) & (~(Q & R) & (R & P))]
T5-10′. TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)]	T5-10. INC[(∀x)(~Px v Qx) & (∀x) Px & ~(∀x)Qx)]
T5-11′. TAUT[(∀x)Px ⊃ Pa]	T5-11. INC[(∀x)Px & ~Pa]
T5-12′. TAUT[(∀y)((∀x)R<x,y> ⊃ R<y,y>]	T5-12. INC[(∃y)((∀x)R<x,y> & ~R<y,y>]

In Chapter 6 A-logic adds other tautologies, including many with no TF-analogues, e.g.

T6-31.TAUT[~(P => ~P)]	T6-30. INC[P => ~P]
T6-33.TAUT[~((P&Q) => ~P)]	T6-32. INC[(P&Q) => ~P]

But the most important kind of theorems are Validity-theorems, the valid inferences, or conditionals or biconditionals established in Chapters 5, 6, 7 and 8. For example:

T5-101b. VALID[P, .: (P&P)]	T6-613. VALID[((P => Q) & P) => Q]
T6-620. VALID[((∀x)(Px => Qx) & (∀x)Px) => (∀x)Qx]	Ti7-780. Valid_I [~TP , .: ~T(P & Q)]
T7-730. Valid [~T(∃x)Px , .: (∀x)~TPx]	T8-818. Valid[T(P=>Q) => F(P=> ~Q)]

Validity theorems are the basic principles of logic; the grounds of those rules of inference which are reliable guides to reasoning. Other kinds of theorems serve secondary, supportive roles.

Analytic Logic covers other logical properties and relations such as entailments, analytic implications, logical unfalsifiability, logical unverifiability, logical contingency (of wffs), logical contrariety or contradictoryness, etc. But *valid inferential conditionals* of formal logic remain the primary concern of logic, for these are the guides to rational reasoning and discovery in all fields.

Principles which we call Derived Rules of inference in early chapters, are logically valid conditionals in the broad sense of A-logic, provable from definitions of their terms. These principles are theorems of A-logic in the broad sense. Valid[T(P CONT Q) => T(TAUT(P ⊃ Q))] is a theorem, not of purely formal logic, but of logic in the broad sense. Analytic Logic applies to itself.

Other concepts (besides ‘Theorem’), including “logical validity”, “logical entailment” and “logical implication”, are revised to enable logic to avoid anomalies and paradoxes. Revisions in these concepts are explicated at length in Sections 6.35, 7.16, 7.423, 8.222 and elsewhere.

In Analytic Logic as in M-logic, all theorems are implicitly truth-claims. But in A-logic what each theorem implicitly asserts is the truth of a specific sentence in which a) the predicate is a “predicate of logic”, and b) the subject-terms are linguistic expression or wffs which have the property or relation the predicate stands for.[24] In each Validity-theorem, A-logic asserts that it is true that some particular form of inference or conditional statement is *Valid*; this does not entail that every instantiation of that form is true, though it entails that every instantiation is valid and none are false. Similarly for TAUT-theorems; not every instantiation is true, but none are false.

Four essentially new concepts are introduced in Analytic Logic. They are: Referential Synonymy, the POS/NEG distinction, C-conditionals and T-operators. These are introduced and connected with more familiar concepts in sequential order.

---

24. The task of rigorous formal logic is to correlate physical properties of linguistic signs with logical properties, relations and structures of ideas. Logical properties or relations may be thought of as belonging, or not, to the meanings, or ideas, conveyed by the expressions that are the subject terms.

1)	**Synonymy & Containment** Chs 1-3		
	a) Among Negation-free '&' and 'v' statements		Ch 1
	b) Between Predicates		Ch 2
	c) Between quantified expressions		Ch 3
2)	The **POS/NEG** distinction among predicates	Ch 4	
	Inconsistency & Tautology defined by Cont		Ch 5
3)	**C-conditionals**	Ch 6	
4)	**T-operator** & Trivalence	Chs 7-9	
	a) with Math Logic		Ch 7
	b) in Analytic Truth-logic with C-conditionals		Ch 8
	c) C-conditionals and T-operator on matters of fact		Ch 9

The truth-tables in Chapter 8 for C-conditionals are not new; de Finetti's 1935 three-valued truth-tables are formally the same as those in A-logic (although the truth-table for (P => Q) was assigned to '(Q|P)' for "the conditional probability of Q, given P", and the T-operator 'T' was associated with "thesis" rather than with "it is true that ..."). But the explication of conditionals, incorporating a sharp distinction between truth-claims about conditionals, and conditionals as "inference tickets" (to use Ryle's phrase) represents at least a new concept based on earlier ideas.

With regard to Synonymy and Containment, or the POS/NEG distinction as here defined, or of truth-operators, I am not aware of any precedents. However, this may well be due to an insufficient search of the literature.

These revised or new concepts conflict at points with principles or presuppositions of Mathematical Logic. Among the principles of A-logic which are contrary to those of Mathematical Logic are the following:

1) Theorems of logic are assertions that certain logical properties or relations hold of the meanings of certain kinds of linguistic entities.
2) Logic is not basically only about propositions (sentences which are either true or false).
3) Logical properties and relations are properties and relations of the meanings of predicates, and only derivatively of sentences.
4) Every predicate is POS or NEG.
5) In truth-logic, some indicative statements are neither true nor false.
6) Conditionals are true or false only if the antecedent is true.
7) Validity applies only to inferences and C-conditionals.
8) In all *logically valid* conditionals the antecedent logically contains the consequent.
9) A conditional is not valid if antecedent and consequent are jointly incompatible.
10) Valid statements are not necessarily true.

It is unlikely that all of these principles will seem plausible to the reader initially. Perhaps some which are initially questioned will be seen as intuitively acceptable after examining the arguments for them. But if they are to be justified as a whole, it must be by their inter-related roles in solving the problems which have confronted Mathematical Logic, and in the successful performance of jobs we expect and want logic to perform.

A Note on Notational Devices and Rules of Convenience

In this book we use several notational devices for convenience and clarification. Most of them are, or are similar to, devices used by authors in Mathematical Logic:

Quasi-Quotes: '[P_i]' means 'a result of putting an actual predicate for 'P_i' in '(P_i)'
Assertion sign: '|=' for 'By formal Analytic Logic it is true that…'
Uniform Substitution: [P (e_1/e_2)] for 'the result of replacing all occurrences of expression_1 in P, by expression_2.'
Partial Substitution: [P (e_1//e_2)] for 'the result of replacing one or more occurrences of expression_1 in P, by expression_2.'
Alphabetic variants: Two quantified wffs, P and Q, have the purely notational relation of being alphabetic variants if and only if

(i) P and Q are each wffs having one or more quantifiers
(ii) P and Q differ from each other only with respect to which letters for individual variables occupy which positions in the two wffs, and
(iii) P and Q have exactly the same patterns of same-variable occurrences.

The purely notational Rule of Alphabetic Variance is:
"If R is a Q-wff and S is an alphabetic variant of R, then |= [R Syn S]"

Assertions of Formal vs Applied Logic: To distinguish theorems based only on the meanings of syncategorematic expressions ("logical words") from those based on the substantive meanings of non-syncategorematic expressions, we capitalize 'SYN' in the former theorems, and use 'Syn' for both kinds of logical synonymy. '[P syn Q]' all by itself ascribes synonymy without adding the imprimatur of logic. Letting '|=…' mean 'It is true by purely formal A-logic that …',

'[P SYN Q]' for '|= [P Syn Q]'	'INC[P]' for '|= Inc[P]'	'Inc[P]' for '|= inc[P]'
'[P CONT Q]' for '|= [P Cont Q]'	'TAUT[P]' for '|= Taut[P]'	'[P Syn Q]' for '|=[P syn Q]'
'VALID[P]' for '|= Valid[P]', etc.		'Valid[P]' for '|= valid[P]'

Assignment of Names to Axioms, Rules, Theorems, etc. We use a system of naming theorems, rules, etc, which follows the sequence in which concepts and their symbols are introduced. Axioms, theorems, metatheorems, rules, derived rules, and definitions are named by prefixing 'Ax.', 'T', 'MT', 'R', 'DR' or 'Df.' to the number of the chapter in which they are introduced, followed by a hypen, then the number assigned to the axiom, theorem, etc. in that chapter. Thus, 'Ax.5-02' means Axiom 2 in Chapter 5, and 'R4-1' means inference Rule number 1 in Chapter 4. The Axioms and Rules have the same numbers in suceeding chapters but the different chapter number often signifies a new refinement. If the theorem-number has two digits, as in 'T8-27', it is derived from axioms and rules, etc. found in the Chapter with the number following 'T'. A theorem-number with three digits, as in 'T6-136', means the theorem is derived by rules in the Chapter with the number following 'T' from a theorem in a previous chapter whose number is the first of the three digits. E.g., T6-136. VALID[(P&Q) => P] is derived in Chapter 6, from T1-36. [(P&Q) CONT P] of Chapter 1, using Derived Rule DR6-6a of Chapter 6, which says that if [A Cont B is a theorem, and (A & B) is not inconsistent, then (A=>B) is valid. Since the logical predicates 'Inc', 'Taut', and 'Valid' are all defined in terms of Logical Containment, many theorems are derived by rules in Chapters 5 to 8 from Cont-theorems in the earlier Chapters, 1 to 4.

These devices are developed in a sequence of steps from Chapter 1 to Chapter 6 (where the purely formal base of A-Logic is presented) and are continued in A-logic's extension into truth-logic in Chapters 7 and 8.Thus axioms, theorems and rules are listed consecutively. The use of brackets for quasi-quotation is rigorous for theorems but less so - more suggestive - in rules and metatheorems. The terms 'metatheorem' and 'derived rules' are somewhat interchangeable but should probably have been better distinguished.

Part I

Analytic Logic

Section A. SYNONYMY AND CONTAINMENT

Section B. MATHEMATICAL LOGIC

Section C. ANALYTIC LOGIC

Section A

Synonymy and Containment

Chapter 1

"And" and "or"

The point of departure for this chapter is the fragment of standard propositional logic in which the well-formed formulae have only propositional variables, '&', and 'v'. This will help the transition from familiar features of mathematical logic to the two basic concepts of analytic logic.

1.1 Logical Synonymy among Conjunctions and Disjunctions

The concepts of logical synonymy and logical containment are applied to a negation-free sub-set of the well-formed formulae of the sentential calculus in mathematical logic, i.e., wffs using only the conjunction-operator '&' and the disjunction-operator 'v'. This excludes wffs with occurrences of '∼', '⊃' or '≡'. A formal axiomatization of this fragment is presented with eight axioms, theorems, normal forms and a completeness proof. Theorems are either SYN-theorems or CONT-theorems, e.g., [(P&Q) SYN (Q&P)] or [(P & Q) CONT P]. These assert, respectively, that a pair of wffs are logically synonymous, or that the first wff logically contains the second. Logical containment is like Simplification writ large.

The formal syntactical relationships denoted by 'SYN' (logical synonymy) and 'CONT' (logical containment) exist and have always existed among the wffs of mathematical logic, but they have not been recognized, associated with a concept of synonymy, or proposed as foundations of formal logical theory.

The fundamental presupposition of analytic logic is that substituting the synonym of an expression P for any occurrence of P in a theorem of logic will not affect logical properties or relations asserted to hold by that theorem.

All of the essential criteria for logical synonymy and logical containment and the basic rules of inference are presented in this chapter. Subsequent chapters add classes of well-formed expressions which may stand in these relations, but do not add to or alter the fundamental criteria for these logical relations, the meaning of the two-term predicates 'SYN' (for logical synonymy) and 'CONT' (for logical containment), or the basic principles of inference employing these concepts.

1.11 The SYN-Relation and Referential Synonymy

Axioms and theorems of the form '[P SYN Q]' may be treated as mere symbols with syntactical properties and subject to transformations only in accordance with syntactical rules. They may also be interpreted as asserting that a certain kind of synonymy relation, called referential synonymy, holds between the meanings of expressions replacing P and Q. It is this interpretation which warrants calling some statements of the form 'P SYN Q' axioms or theorems of logic. Referential synonymy should be distinquished from synonymies introduced by definitions within the system. The latter introduce new symbols and are useful notationally and conceptually, but must be formulated so as to preserve referential synonymy.

1.111 Referential Synonymy

There are semantical theories of logic which separate theories of reference from theories of meaning and opt for one rather than the other as the foundation of logic.[1] In traditional philosophy, and in dictionaries, linguistic "meaning" has been treated as having different modes: denotation and connotation, extension and intension, reference and meaning. For A-logic the meaning of an indicative sentence involves both reference and intensional criteria. The meanings of subject terms are their referents—what is being talked about. The referents are conceived as fixed entities. They may be imagined or real; the concept that they exist "objectively" is an add-on, inessential to being the referent of a term, as is the concept of their being "subjective". The meanings of predicates are criteria of application, i.e., intensional meanings. Criterial meanings are something carried around with us in our heads;[2] they stand for properties or relations which may or may not belong to the referents of the subject terms. Contingent indicative sentences can convey information precisely because, from a logical point of view, their predicates may or may not apply to the fixed entities denoted by the subject terms.

For rigor, the rules for deriving theorems from axioms must be interpretable strictly as rules of syntactical transformations. But to make it plausible that the axioms and theorem belong to logic, the initial SYN-axioms and all SYN-theorems should be interpretable as conforming to the concept of referentially synonymous expressions.

The negation-free fragment of propositional logic contains only sentence letters for indicative sentences, parentheses for grouping, conjunction signs and alternation signs. Here we discuss only the logical synonymy of indicative sentences with logical structures using 'and' and 'or'. By the rules of formation we can construct unlimited numbers of negation-free wffs in this class.

The concept of referential synonymy is based on three intuitively plausible principles: If two expressions are referentially synonymous, then 1) both expressions must refer to or talk about all and only the same entities, 2) both expressions must say (or predicate) the same things about each of those entities, and 3) all and only those things contained in (or entailed or implied) by one expression must be contained in (or entailed or implied) by the other.

All essential features of the logic of referential synonmy are found in the logical structures of negation-free expressions built up only with conjunction and disjunction as connectives. The subsequent introduction of quantifiers, negation and a conditional operator[3] adds to the variety of logically synonymous pairs but does not affect the essential nature of that relationship.

As applied to the sentential logic in M-logic, referential synonymy is based on the principle that if two sentences <u>mean</u> the same thing, they must be about the same entities and must say, entail and imply the same things about those entities. Putting it negatively, if two sentences do not refer to (talk about) the

---

1. Cf. Quine, W.V.O., "Notes on the Theory of Reference", pp 130-8, in *From a Logical Point of View*, Harper, 1963.
2. Whether they also exist in some objective realm—a Platonic heaven—is a metaphysical question, not discussed here.
3. Which are introduced in Chapters 3, 4 and 6 respectively.

same individuals, or do not say the same things about each of those individuals, or do not imply or entail the same things, they can not have the same meanings.

The essential features of referential synonymy are fully displayed in the negation-free fragment of sentential logic with only "&" and 'v' as sentential operators. If two wffs of sentential logic are to represent only pairs of synonymous expressions, then each instantiation of the two wffs must have all and only the same subject terms and all and only the same predicate terms. Thus a necessary condition for the referential synonymy of all pairs of sentences gotten by instantiating the sentence-letters in pair of wffs, is that the two wffs contain all and only the same sentence letters. (I.e., complete variable sharing) and that all occurrences of a sentence letter are instantiated by the same expression. Otherwise instantiated sentences could refer to different entities and/or say different things about the same or different entities. (The formal expression of this is SYN-metatheorem 1.)

But obviously, sameness of sentence letters is not sufficient for referential synonymy. Two sentences can have all and only the same sentence letters, yet one can be true and the other false. For example '(P or Q)' can be true when '(P and Q)' is false. They can't have the same meaning. A second requirement is suggested by the negative statement,

> If two sentence schemata are such that, upon substituting the same sentences for the same sentences letters in both, one implies or entails something the other does not, then they can not be synoymous.

In terms of the sentential logic in M-logic, this means that if one resulting sentence can be true when the other is not, then the two sentence schemata **can not** have the same meaning. Thus if two sentence have the same meaning, they must be true under the same conditions and false under the same conditions. They must have the same final column in their truth tables; i.e. they must be truth- functionally equivalent, expressing the same truth-functions. In other words, truth-functional equivalence is a necessary condition of referential synonymy or sameness of meaning among wffs. (The formal expression of this is a corollary of SYN-metatheorem 2.)

The converse does not hold. Truth-functional equivalence is not sufficient for referential synonymy, since two expression can be truth-functionally equivalent though they do not have the same sentence letters, violating our first rule. For example, P and (P&(PvQ)) have the same truth- tables in M-logic, but not the same set of variables.[4]

So now we have two necessary conditions for referential synonymy which can be correlated with the purely syntactical features, 1) the two wffs must have all and only the same sentence letters, and 2) they must have the same truth-tables. If two formulae fail to meet either one they can not be synonymous. But are these two conditions sufficient?

It turns out that these two are sufficient as long as we are dealing only with negation-free wffs in which only the sentence connectives 'and' and 'or' occur. There is an easy decision procedure for this class of wffs. To be logically synonymous the two wffs, P and Q, must satisfy both (1) that P and Q are truth-functionally equivalent and (2) P and Q have all and only the same sentence letters. Thus (in this

---

4. Mathematical Logic is viewed by some mathematicians as grounded in a theory of distributed lattices; such lattices include an *absorption principle*, by which P is equivalent to (P & (PvQ)). On this view Analytic Logic may be said to be grounded in a theory of *non-absorptive* distributive lattices: it rejects the *absorption principle*.

negation-free logical language) if the answer is affirmative on both counts, P and Q are referentially synonymous, if negative on either, they are not referentially synonymous.[5]

For a more basic, general solution we turn in Section 1.12 to an axiomatic system which yields an alternative to the truth-table test—reducibility of the two wffs to a unique, synonymous "normal form".

This system starts with eight SYN-axioms which meet the two necessary conditions laid down above. They are analogues of eight widely recognized equivalences: the principles of idempotence, commutation, association and distribution for 'and' and for 'or'. These axioms, with a rule for uniform substitution of wffs for sentence letters, and the rule of substitutivity of synonyms, contain the basic core from which all SYN-theorems and CONT-theorems in analytic logic flow. Later extensions of analytic logic to cover predicates, quantifiers, negation, conditionals or other syncategorematic operators or relations, do not add or subtract any basic criteria for logical synonymy or logical containment.

After the axiomatic development in Section 1.12, principle and metatheorems in Sections 1.13 and 1.14 will be derived to help establish the completeness and soundness of the SYN-relation.

The question of the completeness and soundness of the axiomatic system for the 'SYN-relation' (in this chapter) may be phrased as whether the system will pronounce SYN all and only those pairs of negation-free sentence-schemata which satisfy the truth-table test described above. The answer to this question is provided by metatheorems relating Basic Normal Form wffs to members of the class of all wffs in the present system. In Section 1.13, we define four Basic Normal Forms of wffs. In Section 1.141 we prove SYN-metatheorems 3 to 7 which establish that every wff is SYN to just one wff in each class of Basic Normal Forms. In Section 1.142, SYN-metatheorems 8-10 show that the set of all wffs is partitioned by each class of Basic Normal forms into disjoint equivalence classes of wffs SYN to each individual Normal Form wff in that class. These results make possible the soundness and completeness proof in Section 1.142; with simple adjustments they will establish completeness and soundness proofs for the enlarged systems which follow.

1.112 Definitional Synonymy

A second kind of synonymy which is used in logic, may be called definitional synonymy. This is a useful notational device. A new symbol, not derivable from the primitive symbols and rules of formation, is introduced and an expression using this symbol is said to have the same meaning as some complex symbol expressed entirely in primitive notation. Sometimes this is merely an abbreviatory device which saves ink and time, but in other cases it reflects the result of analytical reflection on the meaning of a familiar term, defining it in terms of a complex of other terms taken as primitive.

The act of introducing a new symbol by definition into a language with fixed primitives is a selective discretionary act. There are infinities of complex expressions and any complex expression can be abbreviated. Theoretically no definitions are necessary, but clearly some are useful. The human mind often reasons more easily with abbreviations of complex expressions than with the unabbreviated definiens. However, constrains are needed. There is a utility in keeping the number of distinct symbols low; the

---

5. The use of truth-tables is an extra-logical device, and later, with the introduction of negation, these two conditions turn out to be insufficient; some non-synonymous wffs satisfy both of them. E.g., ((P&~P) & (Qv~Q)) and ((Q & ~Q) & (Pv~P)). These have all and only the same elementary wffs, but they are not synonymous. One makes a contradictory statement about P, the other does not. According to the concept of entailment in A-logic, the first logically contains P but the second does not. Although in M-logic they both "imply" each other (because both are inconsistent and inconsistencies imply everything), the concept of implication in A-logic differs on this and is such that neither A-implies the other. Nevertheless, as a temporary device the decision procedure mentioned above holds in negation-free sentential logic.

human mind works best with a lean, efficient vocabulary. More important, the predicates logic uses must not be themselves ambiguous or inconsistent (though they must be able to identify and talk about inconsistent expressions). Ambiguity and inconsistency can be introduced into a language by undisciplined introduction of new symbols. Definitions should be introduced into a logical system only in accordance with rules which keep ambiguity and inconsistency under control.

In this chapter there are only two definitions: 1) one which permits the elimination of the inner sets of parentheses in expressions of the forms '((P & Q) & R)' and '((P v Q) v R)' and 2) the definition of "CONT" in terms of SYN and conjunction: [(P Cont Q) Syn_{df} (P Syn (P&Q))]. The first serves several useful purposes, including the reduction of expressions to simply written normal forms. The second provides a conceptual clarification of central importance in analytic logic.

In later chapters other definitions are introduced and appropriate rules of definition will be discussed more fully. If appropriate rules are followed, substitutions of definitional synonyms will be preserve referential synonymity. The new symbols which are the definienda, will stand for the primitive expressions they abbreviate. They will be synonymous with other expressions if and only if the primitive expressions which are abbreviated are referentially synonymous. Thus the concept of referential synonymy is the fundamental concept for logic; definitional synonymy is a useful device which, properly used, will not conflict with or add to the concept of referential synonymy.

1.12 An Axiomatization of the SYN-Relation for 'Or' and 'And'

1.120 The Formal System

The formal system for this chapter is as follows:

I. Primitive symbols:

Grouping devices: ( )

Connectives: & v

Sentence (Proposition) Letters: P_1, P_2, P_3,…, P_i,… {PL}

(To save ink and eye-strain we often use 'P', 'Q', 'R' rather than 'P_1', 'P_2', and 'P_3'.

II. Rules of Formation:

FR1-1. If P_i e {PL}, then P_i is a wff

FR1-2. If P_i and P_j are wffs, then (P_i & P_j) and (P_i v P_j) are wffs.

Also, schemata can be made easier to read by abbreviating association to the right. For this purpose we introduce two abbreviational definitions:

III. Definitions:

Df1-1. (P & Q & R) SYN_{df} (P & (Q & R))

Df1-2. (P v Q v R) SYN_{df} (P v (Q v R))

These three sections gives us the object language we deal with in this chapter. Theorems of this chapter consist of assertions that the relationship of logical synonymy (Abbr: SYN) or the relationship of logical containment (abbr: CONT) hold of some ordered pairs of wffs. The theorems are developed from the base of eight axioms:

IV. Axioms

Ax.1-01. [P SYN (P&P)] [&-IDEM]
Ax.1-02. [P SYN (PvP)] [v-IDEM]
Ax.1-03. [(P&Q) SYN (Q&P)] [&-COMM]
Ax.1-04. [(PvQ) SYN (QvP)] [v-COMM]
Ax.1-05. [(P&(Q&R)) SYN ((P&Q)& R) [&-ASSOC]
Ax.1-06. [(Pv (QvR)) SYN ((PvQ) v R)] [v-ASSOC]
Ax.1-07. [(Pv (Q&R)) SYN ((PvQ) & (PvR))] [v&-DIST]
Ax.1-08. [(P& (QvR)) SYN ((P&Q) v (P&R))] [&v-DIST]

V. Rules of Inference

To prove theorems in this chapter, two rules of transformation, or inference, will be used. The first rule derives new theorems by substituting synonyms for synonyms within a SYN-theorem:

R1-1. If [P SYN Q] and [R SYN S], then you may infer [P SYN Q(S//R)]. "SynSUB"
(where 'Q(S//R)' abbreviates
"the result of replacing one or more occurrences of S in Q by R")

As stated this rule requires that the substitution be made in the right-hand component of the initial SYN-theorem. The following rules would be equally warranted instead of R1-1, but as we shall see, they are derivable from R1-1 with axioms and R1-2 so they need not be added.They may all be called 'SynSUB'.

R1-1b. [From (P SYN Q) and (S SYN R), you may infer (P SYN Q(S//R))]
R1-1c. [From (P SYN Q) and (R SYN S), you may infer (P(S//R) SYN Q)]
R1-1d. [From (P SYN Q) and (S SYN R), you may infer (P(S//R) SYN Q)]

The second rule, U-SUB, can be used to derive new SYN-theorems by permitting the uniform substitution of any wff at *all* occurrences of a sentence letter in the components of a SYN-theorem:

R1-2. If [R SYN S] and Pe{PL} and Q is any wff, then you may infer [(R SYN S)(P/Q)]
(where '(R SYN S)(P/Q)' abbreviates
'the result of replacing all occurrences of P in '(R SYN S)' by Q')

As written above these rules are conditioned directives. The consequent can be put in the imperative mood, although it is not a requirement or command. R1-1 and R1-2 suggest that when certain conditions are met one *may* as a next step, write down another statement of a certain form. *Rules* of inference are different from *principles* of inference. Principles are indicative conditionals. The reliability of a rule depends on the *validity* (to be established later in our logic) of the related indicative conditionals. For example, the *principle* of logic which underlies the *rule* R1-1 says, that **if** it is true that P is logically synonymous with Q, and R is logically synonymous with S, **then** it will be true that P is synonymous with the result of replacing one or more occurrences of S in Q by R. It can be expressed symbolically (and a bit elliptically) as:

R1. [If (P SYN Q) and (R SYN S), then (P SYN Q(S//R))] SynSUB

and the principle underlying R1-2 can be expressed in symbols as

R2. [If (R SYN S) and Pe{PL} and Q is any wff, then (R SYN S)(P/Q) U-SUB

The reliability of a *rule* of inference—the assurance that using the rule will not lead from a truth to a non-truth—depends on the validity of *principles* like R1. These principles of inference are what justify using the rules. In each step of a proof I use a rule, but I justify each step by referring to a *principle* of inference, such as R1.

The rule R1-1 can be formulated as a strictly mechanical sequence of steps to be performed. Initially we lay it out as just one strictly mechanical procedure to emphasize its capacity for complete rigor: Where successive steps in a derivation are shown on numbered lines, j), k), k+i), the step gotten by use of R1 is justified by the annotation '[j),k),R1]' as in:

Instance of [P SYN Q]:	Step j) [P SYN (T&**(S&U)**)]	
	......	
Instance of [R SYN S]:	Step k) [**R** SYN **(S&U)**]	
	.....	
Instance of [P SYN Q(S//R)]:	Step k+i) [P SYN (T&**R**)]	[j),k),R1]

Regardless of whether j) precedes k) or not in the derivation, in the justification, '[j),k),R1]', for the step k+i the '**j)**' will name the statement within which a substitution is to be made, and '**k)**' will name the intermediate premiss which contains an occurrence of the expression to be substituted on its left and an occurrence of the expression which is replaced on its right. Schematically,

[Step j)	,	Step k)	,	Step k + 1)	]
[P SYN	Q] ,	[R SYN	S] ,	**.:** [P SYN	Q(S//R)]
(to stay the same)	(to be changed)	(to be substituted)	(to be replaced)	(stays the same)	(Q with R substituted for S)

Usually in this and following chapters we use R1 to justify steps in a derivation. But to eliminate needless steps in some theorems and proofs, we introduce an alternative to R1 which we will call 'R1b'. It permits us to replace R by S rather than S by R. I.e.,

instead of R1. [If (P SYN Q) and (R SYN S), then (P SYN Q(S//R))],
we have, R1b. [If (P SYN Q) and (R SYN S), then (P SYN Q(R//**S**))].

Strictly we can derive R1b from R1, as follows:

R1b. [If (P SYN Q) and (R SYN S), then (P SYN Q(R//S))]

Proof:	Step 1)	P SYN Q	[Assumption]
	Step 2)	R SYN S	[Assumption]
	Step 3)	**S** SYN (S&S)	[Ax.1-01, U-SUB]
	Step 4)	S SYN **(S&S)**	[Ax.1-01, U-SUB]
	Step 5)	S SYN **S**	[4),3),R1]
	Step 6)	S SYN R	[5),2),R1]
	Step 7)	P SYN Q(R//S)	[1),6),R1]
	Step 8)	[If (P SYN Q) and (R SYN S), then (P SYN Q(R//**S**))]	[1) to 7),Cond.Pr.]

R1b allows us to replace the left-hand member S of a synonymy by the right-hand member, rather than the right-hand member R by the left-hand member, as in R1. When we use 'R1b' we will call it 'R1b' to distinquish what we are doing from steps in accordance with R1.

The two principles which justify R1-1 and R1-2 respectively, namely,

R1. [If (P SYN Q) and (R SYN S), then (P SYN Q(S//R))]
and R2. [If (R SYN S) and Pe{PL} and Q is any wff, then (R SYN S)(P/Q)

are special cases of the general rules SynSUB (the Substitutivity of Synonyms), and U-SUB or (Uniform Substitution of a wff for a variable).

The rule of SynSUB is based on a very general principle of semantics. It says "If A is any expression, and B is a well-formed component of A, and B is *referentially synonymous* with C, then the meaning of A is *referentially synonymous* with the meaning of the result of substituting C for one or more occurrences of B in A"; If A and B Syn C, then A Syn A(C//B).

In Rule R1-1 this principle is limited to cases in which A is a statement of the form '(P SYN Q)' and 'B is referentially synonymous to C' has the special form '(R SYN S)'. If R occurs as a component of Q, S may be substituted for R in Q, and the result will still be logically synonymous (SYN) to P.

The word 'synonymy' has many meanings and uses in ordinary language. In A-logic the word 'Syn' is used to abbreviate 'is referentially synonymy with' as we have just defined it for purposes of this logical system. We use 'SYN' (capitalized) for those cases in which the referential synonymy is based solely on the meanings and syntax of expressions containing only the syncategorematic words ("logical words"), 'and', 'or', 'not', 'if…then', 'all', 'some' and words defined by them, as in the axiom [(P & Q) SYN (Q&P)]. Letting '|=' mean 'It is a theorem of analytic logic that …', '(R SYN S)' abbreviates '|= (Q Syn R)'. Extra-logical expressions may be Syn, i.e., referentially synonymous, depending on definitions. For example, given certain definitions in geometry, "'a is a square' Syn 'a is an equilateral plane figure & a is a rectangle'" is proven true. Provided that a definition conforms to appropriate rules for introducing definitions in a logical system, synonyms by definition (by Syn_{df}) may be substituted by this rule to get logically valid inferences in the broad sense of 'logically'.

Thus SynSUB is doubly limited in R1. It applies only to theorems of analytic logic in which [P SYN Q] has been established, and it substitutes only expressions R and S which are proven logically synonymous (i.e., such that [R SYN S] is true). The capitalized 'SYN' in 'P SYN Q(S//R)' abbreviates 'P is *logically synonymous* with the result of replacing one or more occurrences of S in Q by R'. Thus only new *SYN*-theorems or *CONT*-theorems can be derived, by the Rule R1-1.

The principle of SynSUB is unique, central and essential to analytic logic. It is similar in certain respects to mathematical logic's rule of the "substitutivity of the biconditional", but the latter is not generally valid in analytic logic since it does not always preserve sameness of meaning. Further, referential synonymy is a stricter relation than logically valid biconditionality, even with the C-conditionals of Chapter 6. SynSUB holds whether the intitial expression P is a question, fictional statement, proposition, directive or value judgment. For logic in the broad sense, SynSUB preserves whatever logical properties or relations belong to expressions, because it perserves their referential meanings.

The second principle of inference, R2 is a special case of U-SUB or Uniform Substitution. A general, unrestricted rule of U-SUB allows substitution, at ***all*** occurrences of any sentence letter within a given *theorem of Logic*. While SynSUB preserves sameness of meaning in any sentence; U- SUB makes substitutions only in sentences which are theorems of logic.

U-SUB does not preserve referential meaning as SynSUB does. It preserves only the logical property or relation which is asserted in the theorem. If a theorem asserts the *logical synonymy* of P and

(PvP), U-SUB allows the assertion of the *logical synonymy* of the result of putting '(R&S) for all occurrences of 'P' in that theorem. The resulting expression, ''(R&S)' SYN '((R&S)v(R&S))'' is not referentially synonymous with ''P' SYN '(PvP)'' for it talks about different expressions that can have different instantiations without violating rules of uniform instantiation. But the conclusion by U-SUB that ''(R&S)' Syn '((R&S)v(R&S))'' is a *theorem of logic* because ''P' Syn '(PvP)'' is a *theorem of logic*, is based on the principle that logical synonymy (SYN) in this case holds by virtue of the meanings of '&' and 'v' and the sameness of the over-all logical stucture they have in common.[6] In the present chapter R2 limits the expressions which may be substituted to negation-free wffs with only '&' and 'v' as sentential operators, and it limits the statements within which the substitutions may be made to SYN- and CONT-theorems. As the definition of 'well-formed formula' is broadened in subsequent chapters, the applicability of U-SUB expands to allow the substitution of wffs with quantifiers, negation signs, conditional-signs and various other linguistic operators to get new theorems. In Chapter 5 and later chapters U-SUB will also preserve the propertues of logical inconsistency and tautology in INC- and TAUT-theorems, and also it can preserve synonymy and inconsistency based on extra-logical definitions. However, there are other important logical properties and relations in Chapter 6 and later chapters which are preserved by U-SUB only with certain restrictions.[7]

The Axioms and Theorems of logical synonymy ("SYN-theorems") are categorical assertions that the meanings of the two component expressions, and of any pairs of expressions with the same over-all logical forms as theirs, are referentially synonymous. Having been proved, SYN-theorems can be used with R1-1 or R1-2 to derive proofs of other SYN- or CONT-theorems. SYN-theorems can also be used with SynSUB in proofs of derived principles of inference—conditional statements validating the passage from certain expressions with some logical property or relationship to other expressions with the same or a different logical property or relationship.

When used in arriving at a proof or in deductive reasoning both SynSUB and U-SUB are treated as permissive rules—i.e., as rules which say that under certain conditions (in the antecedent) a certain other step is permissible. They are not intended to force the user to proceed from one step to another, but to guide the reader, when it serves the user's purpose to use them. Had they been presented as imperatives, commanding the reader (in the case of U-SUB) to substitute wffs for all occurrences of each sentence letter in a given axiom, the user would be off on an unending, pointless endeavor, producing theorem after theorem by U-SUB to no purpose. As permissives, these rules are not true or false (just as conditional imperatives are not true or false). But they must be valid. They are valid if they are based on valid indicative conditionals. Indicative conditionals, if valid, are also principles of Logic, e.g., "[If (P & Q) is true, then Q is true]" is a valid principle of Analytic Truth Logic; it appears in Section 8.2142 as validity-theorem T8-16. Valid[T(P & Q) => TQ].

The central objective of Formal Logic is to prove valid conditional principles which will support permissive rules of logical inference. From each such principle a valid rule may be derived and used. In constructing proofs, it is the rules, not the principles, which are used. The principles provide the grounds for considering the results of using the rule reliable. Decisions on which rules to use to get a particular proof is made by the user on the basis of the objective of his activity.

---

6. E.g., [P SYN (P v P)] has the same over-all structure
as: [(R&S) SYN ((R&S)v(R&S))]

7. Certain other logical properties, like logical satisfiability and logical validity (as defined in A-logic) are not preserved by unrestricted U-SUB. In particular, derivations of new Validity-theorems from prior Validity-theorems by U-SUB can be gotten only with restrictions on U-SUB to avoid inconsistency.

1.121 Notes on Notation

The following notes specify the meaning of various symbols in the notation which is used in the formal system. The purpose of using these symbols, rather than the long and awkward ordinary language expression which convey their meanings, is to simplify the process both visually and conceptually.

Sentence letters, Predicate letters, variable terms, etc..

(i) The meanings of sentence letters. Noun phrases of the form 'a man', 'a dog with no hair', 'a constitutional crisis', have been called indefinite descriptions. One can make differentiations with a given indefinite description. Seeing several men, one may differentiate one from another by speaking of 'a man' and 'a second man' and 'a third man' etc., or, more briefly, as 'man_1', 'man_2', 'man_3' etc.. In logic, if we are concerned with propositions, we may similarly speak of 'a $proposition_1$', 'a $proposition_2$', etc. Proposition letters, 'P_1', 'P_2', etc., may be said to have as their meanings, the ideas of propositions which may (but need not) be different, which are indefinitely descriptions:

(1) 'P_1' Syn_{df} 'a $proposition_1$' (2) 'P_2' Syn_{df} 'a $proposition_2$' etc.,

P_2 need not be different from P_1, though it may be and we must assume their instantiations are different unless there is reason to think otherwise. As a further simplifying step, we will often use 'P', 'Q', 'R' instead of 'P_1', 'P_2', 'P_3' etc to avoid the bother of subscripted numbers.

If we are concerned with sentences, we may similarly speak of 'a sentence1', 'a sentence2', etc. Sentence letters, 'S_1', 'S_2', etc., may be said to have as their meanings, the ideas of sentences which may (but need not) be different, indefinitely described:

(1) 'S_1' Syn_{df} 'a $sentence_1$'
(2) 'S_2' Syn_{df} 'a $sentence_2$' etc.,

In this chapter we follow Mathematical Logic in treating 'P_1', 'P_2', etc., as propositional variables—indefinite descriptions standing as place-holders for propositions.

(ii) Quotation Marks

Single quote-marks, put around an expression, signify that one is talking about the linguistic symbol. Expressions enclosed in single quotes, are always subject terms, and as Quine has stressed, using a linguistic expression to talk about non-linguistic entities must be clearly distinguished from mentioning (or referring to) a linguistic term in order to talk about the linguistic sign itself. 'Boston is populous' is true, as is ''Boston' has just six letters', but ''Boston' is populous' is false, as is 'Boston has just six letters'.[8]

One may say several kinds of things about linguistic expressions.

Specific predicates can be used to describe their syntactical features—the numbers of simple signs (letters) in a complex sign, the physical sounds or shapes of their elementary parts, the sequence or order in which these occur, the repetition of particular shapes in that sequence, and how two signs are differ or are the same with respect to these features, etc.

In Analytic Logic, it is the relationships and properties of the meanings of signs that are the intended object of study. All theorems of analytic logic may be viewed as assertions about the meanings of

8. See Sections 4 and 5 in Quine's *Mathematical Logic*, 1981.

linguistic expressions. The most fundamental relationship of interest is that of synonymy or sameness of meaning. The next is containment of one meaning in another. The properties of being inconsistent or tautologous are a third. The objective of formal logic is to provide rigorous ways to determine when the relationships of entailment and implication and the property of being a **valid** conditional are present in particular linguistic expressions.

Thus the job of analytic logic is to give rules for determining when specific linguistic expression in quotes have the same meaning, or are inconsistent, or or are such that one logically entails the other, etc.

(ii) Quasi-quotation; the meaning of brackets (corners).

Quine introduced a symbolic device called 'quasi-quotation', which we will use throughout this book.[9] In contrast to quotation (single quotes before and after a wff e.g., '(P_1 v ~P_1)') Quine uses "corners" (here, brackets), '['and']' as in '[P_1 v ~P_1]'. The meaning of a quasi-quotation may be presented as an indefinite description as follows:

(3) "[P_1 v ~P_1]" Syn_{df} "a result of replacing all occurrences of 'P_1' by $\mathbf{P_1}$ in '(P_1 v ~P_1)'"

which means the same as,

(3) "[P_1 v ~P_1]" Syn_{df} "a result of replacing all occurrences of 'P_1' by **a proposition$_1$** in '(P_1 v ~P_1)'"

Thus defined, the meaning of any expressions in brackets (corners) is conveyed by a noun-phrase. It may be used as a subject term or, with the copula as a predicate of the form, e.g., '… is [P_1 v ~P_1]'.

Given the meaning of corners (brackets), the following is a self-evidently true sentence:

(4) '(Jo died v ~(Jo died)' is [P_1 v ~P_1].

For 'Jo died' is a well-formed sentence, and given the meaning of the corners (brackets) in (3) it is a simple truth that

(4') '(Jo died v ~(Jo died)' is a result of replacing all occurrences of
'P_1' by P_1 in '(P_1 v ~P_1)'.

Given the meanings to be assigned in logic to the symbols 'v' and '~', (4) means the same as,

(4'') 'Jo died or it is not the case that Jo died' is a result of replacing all occurrences of
'P_1' by a proposition$_1$ in '(P_1 v ~P_1)'.

or, synonymously, re-reading (4),

(4''') 'Jo died or it is not the case that Jo died' is [P_1 v ~P_1].

This use of quasi-quotations is useful in moving from specific ordinary language expressions to their logical forms. But it does not provide answers to the questions logic is concerned with—whether the

9. See Section 6, Quine, Opus. Cit.

linguistic expressions are inconsistent, whether one expression entail or implies another, and whether an argument of inference is valid.

Quasi-quotations may also be used as subjects. It is as subject terms that they appear in analytic logic. The theorems of A-logic are statements which assert that certain logical properties or relationships belong to expressions of various forms. Thus the following is a theorem of analytic logic:

(5) $\models$ Taut[P_1 v $\sim P_1$]

The symbol '$\models$' is read "It is true, by logic that…", and 'Taut[P_1 v $\sim P_1$]' is read "[P_1 v $\sim P_1$] is tautologous". Thus, drawing on the meanings just assigned to sentence letters and to 'v', '$\sim$', '[' and ']', the sentence (5) is synonymous with, but mercifully shorter than, the English sentences, (5) means,

(5') It is true, by logic, that [P_1 v $\sim P_1$] is tautologous.

or, with more English,

(5'') It is true, by logic, that a result of replacing all occurrences of 'P_1' by P_1 in '(P_1 v $\sim P_1$)' is tautologous.

or with still more,

(5'') It is true, by logic, that a result of replacing all occurrences of 'P_1' by a proposition$_1$ in '(Either P_1 or it is not the case that P_1)' is tautologous.

In the present chapter we deal only with logical synonymy and logical containment.[10] So in this chapter these remarks apply to the expressions found in the theorems of this chapter. Strictly speaking, if we follow the rules just explained, the first SYN-theorem in this chapter, the axiom,

Ax.1-01. [P SYN (P & P)]

should be written: Ax.1-01. $\models$ ['P_1' Syn '(P_1 & P_1)'], where the brackets indicate that what we are talking about are results of replacing occurrences of 'P_1' with P_1. The same may be said of all theorems of analytic logic throughout the rest of this book But for ease of reading, we use '[…SYN—]' in place of '$\models$ […Syn—]' and we use 'P','Q' and 'R' within formulae rather than 'P_1', 'P_2' and 'P_3' simply to avoid the complexity of single quotes. This may be justified as conventions of understanding. The capitalized version, 'SYN', means 'Syn by logic alone', so that '[…SYN—]' is an abbreviation of '$\models$ […Syn—]', where 'Syn' has to do with sameness of meaning among extra-logical as well as logical expressions.

We drop the single quote marks—replacing ['P' SYN '(P & P)'] by [P SYN (P & P)]—simply to avoid undue proliferation of quote-marks. For all theorems of analytic logic are statements about the meanings of linguistic expressions, and since the predicates 'SYN', 'CONT', 'TAUT', 'INC', 'VALID' are specificly defined as predicates of logic, it must be presupposed that what they are talking about are the meanings of linquistic expressions or linguistic expressions of specified forms.

10. 'Tautologous' is defined in Chapter 5.

Further, when any well-formed formula is named as a theorem or axiom of analytic logic, we may drop the '$\models$' , i.e., the prefix that says the expression is true by logic; for to present it as a theorem or an axiom is to assert implicitly that it is true by logic.

The rules of transformation—SynSUB and U-SUB—work very well within these conventions, so the net affect is simplicity in proofs. But it must not be forgotten that the absence of single quotes is a convention; all theorems of analytic logic are about the semantic (meaning) properties and relationships of linguistic expressions which are implicitly presented in quotes as subject terms in the theorem.

The concept of a theorem or axiom in analytic logic differs from the concept in mathematical logic. For Quine theorems are actual statements. Wffs, like '(P_1 v $\sim P_1$)' are schemata which are neither true nor false and thus can't be theorems. His quasi-quotes, e.g., '[P_1 v $\sim P_1$]', mean essentially what they means above in 3): "a result of replacing all occurrences of 'P_1' by a proposition$_1$ in '(P_1 v $\sim P_1$)'".[11] But it is not the quasi-quotations but what they designate that he considers theorems. Schemata are "valid" if all sentences with that form are logically true. '$\vdash$ [P_1 v $\sim P_1$]' means "any result of replacing all occurrences of 'P_1' by a proposition$_1$ in '(P_1 v $\sim P_1$)' is logically true." Thus theorems are the actual sentences which are instances of the schema. '(P_1 v $\sim P_1$)' is not a theorem, but a theorem schema;

'(P v Q)' is a schema but not a theorem schema.[12]

Thus with the respect to the wff '(P v $\sim$P)' the relevant theorem of analytic logic says

'$\models$ Taut[P v $\sim$P]' or, just 'TAUT[P v $\sim$P]'
which means
"It is true by pure logic that [P v $\sim$P] is tautologous",

i.e., it attributes the property of being tautologous to any expression of the form '(P v $\sim$P)', including '(P v $\sim$P)' (which has a meaning) itself.

On the other hand to Quine, '$\vdash$ [P v $\sim$P]' means "Every result of putting a sentence, P, for 'P' in '(P v $\sim$P)' is a theorem of logic" His theorems are the results of these replacements, not '$\sim$[P v $\sim$P]' itself, which belongs to metalogic.

In analytic logic, the property of being inconsistent belongs to the abstract expression 'P and not-P' by virtue of its meaning. '(P and not-P)' is not a statement, and is not true or false, as Quine would agree.

---

11. In *Mathematical Logic*, Quine uses Greek letters instead of 'P', 'Q', 'R'. He limits the use of these symbols to cases "where the expression designated is intended to be a statement" and says that they are replaceable by statements in single quotes or by other names of statements. Thus the quasi-Quotation '[P $\equiv$ Q]' would mean for Quine "the result of putting P for 'P' and Q for 'Q' in '(P $\equiv$ Q)'.
Cf. Quine, W.V., *Mathematical Logic*, 1981, pp 35-36

12. "A schema is valid if satisfied by all its models. A logical truth, finally, is as before: any sentence obtainable by substitution in a valid schema." W.V.Quine *Philosophy of Logic*, Prentice-Hall, 1970. p 52.
"Logical truths . . . are statements of such forms as '*p* or not *p*', 'If *p* then *p*', 'If *p* and *q* then *q*', 'If everything is thus and so then something is thus and so', and others more complex and less quickly recognizable. Their characteristic is that they not only are true but stay true even when we make substitutions upon their component words and phrases as we please, provided merely that the so called "logical" words '=', 'or', 'not', 'if-then', 'everything', 'something', etc., stay undisturbed. We may write any statements in the '*p*'and '*q*' positions and any terms in the 'thus and so' positions, in the forms cited above, without fear of falsity. All that counts, when a statement is logically true, is its structure in terms of logical words. Thus it is that logical truths are commonly said to be true by virtue merely of the meanings of the logical words." Quine W.V., *Methods of Logic* (4th Ed.), 1982. Harvard University Press, p. 4

But it has a meaning, its meaning is inconsistent, and the statement that its meaning is inconsistent is true by logic. To say it is inconsistent is a theorem, a true statement of analytic logic. Similarly, the property of being tautologous belongs to the abstract expression '(P or not-P)' by virtue of its meaning. '(P or not-P)' is not a statement, and is not true or false. But its meaning is has the property of being tautologous; and the statement that its meaning is tautologous is true by logic. Thus to say it is tautologous is a theorem of analytic logic. Similarly, 'P_1' and '(P_1 & P_1)' have meanings; they mean or stand for the concept of a proposition$_1$ and the concept of a result of conjoining that same proposition$_1$ with itself. These are concepts of referentially synonymous expressions. No matter what P_1 may be it will talk about the same entities, says the same thing about them, and be true or not true under exactly the same conditions as [P_1 & P_1]. Thus the statement that they are referentially synonymous is true by logic alone. As abbreviated in '[P SYN (P&P)]' it is an axiom of analytic logic, Ax.1-01.

Mathematical logic seems to have only one kind of theorem (are the other kinds all in the meta-logic?). These all reduce to tautologies or "denials of inconsistencies" (See Chapter 5). Analytic logic has a variety of different kinds of theorem, depending on which predicate of logic is being used. These include TAUT-theorems, which attribute tautology to selected expressions, INC-Theorems, which attribute inconsistent to expressions, SYN-theorems, CONT-theorems, and most important, VALIDITY-theorems. All of this is developed in chapters which follow.

1.122 A Distinguishing Property of SYN

It is clear by examination of the axiom schemata Ax.1-01 to Ax.1-08 and R1 that all pairs of wffs that are SYN will also be TFeq (truth-functionally equivalent). But SYN does not hold of some truth-functionally equivalent pairs of wffs, including primarily the following:

Absorption: i.e., [P TFeq (P&(PvQ))], or [P TFeq (Pv(P&Q))]

That P is not SYN to (P&(PvQ)) or to (Pv(P&Q)) is clear, for the components do not have the same variables. Inspection of Axiom Schemata Ax.1-01 to Ax.1-08 shows that:

SYN-metatheorem 1. Each pair of synonymous wffs will have all and only the same elementary expressions in each of its members.

U-SUB and SynSUB preserve this property. Thus SYN is stronger (more restricted) than truth- functional equivalence (which is called "logical equivalence" in M-logic—mistakenly according to A-logic). This is a central distinguishing feature of SYN; unlike TF-equivalence, one can quick decide that two expressions are not SYN simply by finding that one contains an elementary expression which does not occur in the other.

Another feature of referential synonymy is that if two wffs, P and Q, are synonymous and R is a conjunctive component of one, then R must be a conjunctive component of the other. A conjunctive component is any wff which can be a conjunct not in the scope of a disjunction sign, by transformations using only &v-DIST or v&-DIST. This may be expressed as the following:

SYN-metatheorem 2. If [P SYN Q] and [P SYN (P&R) then [Q SYN (Q&R)]

Nevertheless, the pairs of negation-free wffs which are SYN constitute a proper sub-class of the class of pairs of wffs which are TF-equivalent. Thus we have, for this chapter the corollary for negation-free sentential M-logic:

SYN-metatheorem 2a: If P and Q are negation-free sentential wffs and [P SYN Q] then P and Q are truth-functionally equivalent.

Proof: Use standard truth-tables on Ax.1-01 to Ax.1-08; Rule R1 and U-SUB preserve TF-equivalence.

1.123 SYN is an Equivalence Relation

Next we prove that SYN is reflexive, symmetrical and transitive, i.e., an equivalence relation (as ordinarily defined).

T1-11. [P SYN P]

Proof:	1) [**P** SYN (P&P)]	[Axiom Ax.1-01]
	2) [P SYN **(P&P)**]	[Axiom Ax.1-01]
	3) [P SYN **P**]	[1),2),R1]

Hence, 'P SYN P' is a theorem, i.e., by logic, SYN is reflexive.

To prove that a relation is symmetrical, is to prove that the conditional statement "if R<P,Q> is true, then R<Q,P> is true" is valid. If a conditional is valid, it can not be false, but to assert that a conditional is valid does not entail that its antecedent is true, or that its consequent is true. In the proof we assume that some expression with the form in the antecedent is true; but we don't assert that any particular expression is true categorically. The validity of a conditional statement is proved by what is called a conditional proof; it is shown that if any premiss of the form in the antecedent were assumed to be true, then the conclusion would be true, because one can get from the premisses to the conclusion step by step using only theorems and rules of inference from logic as intermediaries.

The result is a derived principle of inference.[13] It will allow us to do in one step what would otherwise take four steps.

DR1-01.If [P SYN Q] then [Q SYN P]

Proof:	Step 1) [**P** SYN Q]	[Assumption]
	Step 2) [Q SYN **Q**]	[T1-11,U-SUB]
	Step 3) [Q SYN **P**]	[2),1),R1]
	Step 4) [If [P SYN Q] then [Q SYN P]	[1) to 3),Cond. Pr]

Thus 'If [P SYN Q], then [Q SYN P]' is valid. I.e., SYN is symmetrical. The transitivity of SYN is provable, by the validity of a conditional statement of the form "If aRb and bRc then aRc":

DR1-02. [If [P SYN Q] and [Q SYN R], then [P SYN R]

Proof:	Step 1) [P SYN **Q**]	[Assumption]
	Step 2) [Q SYN R]	[Assumption]
	Step 3) [**R** SYN Q]	[2),DR1-01]
	Step 4) [P SYN **R**]	[1),3),R1]
	Step 5) If [P SYN Q] and [Q SYN R], then [P SYN R]	[1) to 4), Cond. Pr.]

13. In order to eliminate needless theorems and proofs, we introduced an alternative version of R1. Strictly speaking the version of SynSUB we called R1b was a derived rule, but since it is used very frequently we simply call it 'R1b'.

Therefore, SYN is transitive.

SYN-metatheorem 3. SYN is reflexive, symmetrical and transitive, i.e., an equivalence relation.
[Proof: T1-11,DR1-01,DR1-02,usual definition of "equivalence relation"]

1.124 Derived Rules and Theorems

The theorems to be proved in this chapter include those needed to establish that the axioms Ax.1-01 to Ax.1-08 with R1, are complete with respect to the logical synonymy of any pairs of wffs in standard logic through quantification theory.

Thus the first theorem is T1-11, [P SYN P], proved above. In each proof, the justification of each step is given in square brackets on the right. A previous theorem or axiom in which a substitution is to be made is cited first, followed by the theorem or axiom containing the expression to be substituted if SynSUB is used, followed by the rule of inference used. It is left to the reader to figure out how the rule is applied in a given instance. Sometimes two steps using the same intermediate premiss will be telescoped into one step.

Two more liberal derived rules, &-ORD and v-ORD, will save many steps. They are based on Commutation, Association and Idempotence.

DR1-03. "**&-ORD**". Any complex wff containing a string of components connected only by conjunction signs is SYN to any other grouping or ordering of just those conjunctive components with or without conjuncts occurring more than once.

Proof: The derived rule &-ORD is based on R1 and R1b with
Ax.1-01. [P SYN (P&P)] [&-IDEM],
Ax.1-03. [(P&Q) SYN (B&P)] [&-COMM], and
Ax.1-05. [(P&(Q&R)) SYN ((P&Q)&R)] [&-ASSOC1].

DR1-04. "**v-ORD**". Any complex wff containing a string of components connected only by disjunction signs is SYN to any other grouping or ordering of just those disjunctive components with or without disjuncts occurring more than once.

Proof: The derived rule v-ORD is based on R1 and R1b, with
Ax.1-02. [P SYN (PvP)] [v-IDEM],
Ax.1-04. [(PvQ) SYN (BvP)] [v-COMM], and
Ax.1-06. [(Pv(QvR)) SYN ((PvQ)vR)] [v-ASSOC1].

The following theorems are useful later on. They are justified by the general rules of &-ORD or v-ORD. The first two use only Commutation and Association (Ax.1-03 and Ax.1-05, or Ax.1-04 and Ax.1-06). The next two also use Idempotence (Ax.1-01 or Ax.1-02) as well. These theorems help prove quantification theorems T3-13 to T3-18.

T1-12. [((P & Q) & (R & S)) SYN ((P & R) & (Q & S))]
T1-13. [((P v Q) v (R v S)) SYN ((P v R) v (Q v S))]
T1-14. [(P & (Q & R)) SYN ((P & Q) & (P & R))]
T1-15. [(P v (Q v R)) SYN ((P v Q) v (P v R))]

To show how &-ORD saves steps, we derive T1-12, using only T1-11, Axiom 1-03 and Axiom 1-05, Rule U-SUB and R1(i.e., SynSUB). For T1-14 we make U-SUB implicit and also use DR1-01:

T1-12. [((P & Q) & (R & S)) SYN ((P & R) & (Q & S))]

Proof: 1)	[P SYN P]	**[T1-11]**
2)	[((P&Q) & (R&S)) SYN ((P&Q) & (R&S))]	[1),U-SUB('((P&Q) & (R&S))'for 'P')]
3)	[(P & (Q & R)) SYN ((P & Q) & R)]	**[Ax.1-05]**
4)	[(P&(Q & (R&S))) SYN ((P&Q) & (R&S))]	[3),U-SUB('(R&S)'for'R']
5)	[((P&Q) & (R&S)) SYN (P&(Q & (R&S))]	[2), 4), R1]
6)	[(Q & (R & S)) SYN ((Q & R) & S)]	[3),U-SUB('Q'for'P', 'R'for'Q' 'S'for'R']
7)	[((Q & R) & S) SYN ((Q & R) & S)]	[1),U-SUB('((Q&R)&S)'for'P']
8)	[((Q & R) & S) SYN (Q & (R & S))]	[7), 6), R1]
9)	[((P&Q) & (R&S)) SYN (P&((Q&R) &S))]	[5), 8), R1]
10)	[(P & Q) SYN (Q & P)]	**[Ax.1-03]**
11)	[(Q&R) SYN (R&Q)]	[10),U-SUB('Q'for'P', 'R'for'Q']
12)	[((P&Q) & (R&S)) SYN (P&((R&Q)&S))]	[9), 11), R1]
13)	[(R&(Q&S)) SYN ((R&Q)&S))]	[6),U-SUB('R'for'Q', 'Q'for'R')]
14)	[((P&Q) & (R&S)) SYN (P&(R&(Q&S)))]	[12), 13), R1]
15)	[((P&R) & (Q&S)) SYN (P&(R&(Q&S)))]	[8),U-SUB('P'for'Q', '(Q&S)'for 'R']
16)	[((P&Q) & (R&S)) SYN ((P&R) & (Q&S))]	[14), 15), R1]

T1-14. [(P & (Q & R)) SYN ((P & Q) & (P & R))]

Proof: 1)	[(P & (Q & R)) SYN (P & (Q & R))]	[T1-11]
2)	[(P & (Q & R)) SYN ((P & P) & (Q & R))]	[1),Ax.1-01(DR1-01),R1]
3)	[(P & (Q & R)) SYN (P & (P & (Q & R)))]	[2),Ax.1-05(DR1-01),R1]
4)	[(P & (Q & R)) SYN (P & ((P & Q) & R))]	[3),Ax.1-05,R1]
5)	[(P & (Q & R)) SYN (P & ((Q & P) & R))]	[4),Ax.1-03,R1]
6)	[(P & (Q & R)) SYN (P & (Q & (P & R)))]	[5),Ax.1-05(DR1-01),R1]
7)	[(P & (Q & R)) SYN ((P & Q) & (P & R))]	[6),Ax.1-05,R1]

The derived rules DR1-05 and DR1-06 are Generalized Distribution rules, based on Ax.1-08 and Ax.1-07, with R1, R1b, DR1-01,DR1-03 and DR1-04. They, like DR1-03 and DR1-04, will be centrally important for establishing principles of logical synonymy in Quantification Theory.

DR1-05. **Generalized &v-DIST**. If P is a conjunction with n conjuncts, and one or more of the conjuncts are disjunctions, P SYN Q if Q is a disjunction formed by disjoining all of the distinct conjunctions which are formed by taking just one disjunct from each conjunct of P.

E.g.**:** [((PvQ) & R & (SvTvU)) SYN
((P&R&S) v (P&R&T) v (P&R&U) v (Q&R&S) v (Q&R&T) v (Q&R&U))]

The proof in any use of DR1-05 repeatedly uses &v-DIST (Ax.1-08) with R1b, or some variant on Ax.1-08 by means of v-ORD or &-ORD on parts of Ax.1-08. In the proof below of our example we treat all variations on Ax.1-08 as '&v-DIST', and omit mention of R1b or R1. (The reverse proof would have similar repeated uses of '&v-DIST'(Ax.1-08) or some variant by v-Ord or &-ORD on parts of Ax.1-08).

|= [((PvQ) & R & (SvTvU)) SYN
((P&R&S) v (P&R&T) v (P&R&U) v (Q&R&S) v (Q&R&T) v (Q&R&U))]

Proof: 1) [((PvQ)&R&(SvTvU)) SYN ((PvQ)&R&(SvTvU))] [T1-11,U-SUB]
2) [(") SYN (((PvQ)&R)&(SvTvU))] [1),&-ORD]
3) [(") SYN (((P&R)v(Q&R))&(SvTvU))] [2),&v-DIST]
4) [(") SYN (((P&R)&(SvTvU)) v ((Q&R)&(SvTvU)))] [3),&v-DIST]
5) [(") SYN (((P&R)&(Sv(TvU))) v ((Q&R)&(Sv(TvU))))] [4),v-ORD]
6) [(") SYN (((P&R&S)v((P&R)&(TvU))) v ((Q&R&S)v((Q&R)&(TvU))))] [5),&v-DIST(Twice)]
7) [(") SYN (((P&R&S)v((P&R&T)v(P&R&U))) v((Q&R&S)v((Q&R&T)v(Q&R&U))))] [6),&v-DIST(Twice)]
8) [((PvQ)&R&(SvTvU)) SYN((P&R&S) v (P&R&T) v (P&R&U) v(Q&R&S) v (Q&R&T) v (Q&R&U))] [7),v-ORD]

DR1-06. Generalized v&-DIST. If P is a disjunction with n disjuncts and one or more of the disjuncts are conjunctions and Q is a conjunction formed by conjoining all of the distinct disjunctions which are formed by taking just one conjunct from each disjunct of P, then [P SYN Q].

E.g., ((P&Q) v R v (S&T&U)) SYN
((PvRvS) & (PvRvT) & (PvRvU) & (QvRvS) & (QvRvT) & (QvRvU))]

The proof in any use of DR1-05 repeatedly uses v&-DIST (Ax.1-07), or some variant on Ax.1-07 by means of &-ORD or v-ORD on parts of Ax.1-07. In the proof below of our example we treat all variations on Ax.1-07 as 'v&-DIST', and omit mention of R1 or R1b. (The reverse proof would have similar repeated uses of 'v&-DIST'(Ax.1-07) or some variant by means of &-ORD or v- ORD on parts of Ax.1-07).

Proof: 1) [((P&Q)vRv(S&T&U)) SYN ((P&Q)vRv(S&T&U))] [T1-11,U-SUB]
2) [(") SYN (((P&Q)vR)v(S&T&U))] [1),v-ORD]
3) [(") SYN (((PvR)&(QvR))v(S&T&U))] [2),v&-DIST]
4) [(") SYN (((PvR)v(S&T&U))&((QvR)v(S&T&U)))] [3),v&-DIST]
5) [(") SYN (((PvR)v(S&(T&U)))&((QvR)v(S&(T&U))))] [4),&-ORD]
6) [(") SYN (((PvRvS)&((PvR)v(T&U))) & ((QvRvS)&((QvR)v(T&U))))] [5),v&-DIST(Twice)]
7) [(") SYN (((PvRvS)&((PvRvT)&(PvRvU))) & & ((QvRvS)&((QvRvT)&(QvRvU))))] [6),v&-DIST(Twice)]
8) [((P&Q)vRv(S&T&U)) SYN ((PvRvS)&(PvRvT)&(PvRvU)& &(QvRvS)&(QvRvT)&(QvRvU))] [7),&-ORD]

Among the theorems numbered T1-12 to T1-36, the even-numbered theorems will all be theorems in which the major connective in the right-hand wff is '&'. The odd-numbered theorems will be their duals; i.e. the result of interchanging '&' and 'v' throughout the even-numbered theorems.

The reason for numbering SYN-theorems T1-12 to T1-36 in this manner is that it provides an systematic procedure for constructing a "dual proof" for the dual of any given theorem. The reason for having wffs with conjunctions as right-hand components be even-numbered theorems is that all significant containment theorems will be derivable from these theorems using R1 and the definition of logical containment, whereas the odd-numbered theorems will not produce significant (non-convertable) containments.[14]

T1-16. [(Pv(P&Q)) SYN (P&(PvQ))]
Proof: 1) [(Pv(P&Q)) SYN ((PvP)&(PvQ))] [Ax.1-07,U-SUB]
2) [(Pv(P&Q)) SYN (P&(PvQ))] [1),Ax.1-02,R1]

T1-17. [(P&(PvQ)) SYN (Pv(P&Q))]
Proof: 1) [(P&(PvQ)) SYN ((P&P)v(P&Q))] [Ax.1-08,U-SUB]
2) [(P&(PvQ)) SYN (Pv(P&Q))] [1),Ax.1-01,R1]

T1-18. [(P&(Q&(PvQ))) SYN (P&Q)] [C-Max]
Proof: 1) [(P&(Q&(PvQ))) SYN (P&(Q&(PvQ)))] [T1-11,U-SUB]
2) [" SYN (P&((Q&P)v(Q&Q))] [1),Ax.1-08(DR1-01),R1]
3) [" SYN ((P&(Q&P))v(P&(Q&Q)))] [2),Ax.1-08(DR1-01),R1]
4) [" SYN ((P&Q)v(P&Q))] [3),&-ORD]
5) [(P&(Q&(PvQ))) SYN (P&Q)] [4),Ax.1-02,R1]

T1-19. [(Pv(Qv(P&Q))) SYN (PvQ)] [D-Max]
Proof: 1) [(Pv(Qv(P&Q))) SYN (Pv(Qv(P&Q)))] [T1-11,U-SUB]
2) [" SYN (Pv((QvP)&(QvQ))] [1),Ax.1-07(DR1-01),R1]
3) [" SYN ((Pv(QvP))&(Pv(QvQ)))] [2),Ax.1-07(DR1-01),R1]
4) [" SYN ((PvQ)&(PvQ))] [3),v-ORD]
5) [(Pv(Qv(P&Q))) SYN (PvQ)] [4),Ax.1-01,R1]

T1-20. [(P&(Q&R)) SYN (P&(Q&(R&(Pv(QvR)))))]
Proof: 1) [(P&(Q&R)) SYN (P&(Q&R))] [T1-11,U-SUB]
2) [" SYN (P&(Q&(R&(QvR))))] [1),8,R1]
3) [" SYN ((Q&R)&(P&(QvR)))] [2),&-ORD]
4) [" SYN ((Q&R)&(P&((QvR)&(PvQvR))))] [3),8,R1]
5) [" SYN (P&((Q&(R&(QvR)))&(PvQvR)))] [4),&-ORD]
6) [" SYN (P&((Q&R)&(Pv(QvR)))))] [5),8(DR1-01),R1]
7) [(P&(Q&R)) SYN (P&(Q&(R&(Pv(QvR))))))] [6),&-ORD]

14. A Note on Duality. If the logical structures of two wffs are exactly the same except that '&' and 'v' are interchanged, the two wffs are said to be duals of each other.

Axioms and theorems are numbered and arranged in the first three chapters so that each even-numbered SYN-theorem has a one member which is a conjunction equal to or larger than the other member. Every odd-numbered SYN-theorem will be the dual of the preceding even-numbered theorem; that is, it will be identical except for the thorough-going interchange of '&'s and 'v's. Further, the proofs of every theorem in this range will be the dual of the proof of its dual; every step in one proof will be the dual of the same step in the other.

This makes the proof of every SYN-theorem from T1-12 to T1-35 the dual of the proof of its dual SYN-theorem. Theorems cited in brackets on the right to justify each step are unchanged except that even-numbered theorems replace the successor odd-numbered theorems, and odd-numbered theorems replace the preceding even-numbered theorems. This duality of SYN proofs has an important consequence for completeness in Quantification Theory.

Proofs of CONT-theorems do not obey these rules since every CONT-theorem is derived from a SYN-theorem with a conjunction as the right-hand component.

T1-21. [(Pv(QvR)) SYN (Pv(Qv(Rv(P&(Q&R)))))]

<u>Proof</u>: 1)[(Pv(QvR)) SYN (Pv(QvR))] [T1-11,U-SUB]
2) [“ SYN (Pv(Qv(Rv(Q&R)))] [1),9,R1]
3) [“ SYN ((QvR)v(Pv(Q&R)))] [2),v-ORD]
4) [“ SYN ((QvR)v(Pv(Q&R)v(P&(Q&R))))] [3),9,R1]
5) [“ SYN (Pv(Qv(Rv(Q&R)))v(P&(Q&R)))] [4),v-ORD]
6) [“ SYN (Pv((QvR)v(P&(Q&R))))] [5),9(DR1-01),R1]
7) [(Pv(QvR)) SYN (Pv(Qv(Rv(P&(Q&R)))))] [6),v-ORD]

T1-22. [(Pv(P&(Q&R))) SYN (P&((PvQ)&((PvR)&(Pv(QvR)))))]

<u>Proof</u>: 1) [(Pv(P&(Q&R))) SYN (Pv(P&(Q&R)))] [T1-11,U-SUB]
2) [“ SYN ((PvP)&(Pv(Q&R)))] [1),Ax.1-07(DR1-01),R1]
3) [“ SYN (P&(Pv(Q&R)))] [2),Ax.1-02,R1]
4) [“ SYN (P&((PvQ)&(PvR)))] [3),Ax.1-07(DR1-01),R1]
5) [“ SYN (P&((PvQ)&((PvR)&((PvQ)v(PvR)))))] [4),8,R1]
6) [“ SYN (P&((PvQ)&((PvR)&((PvP)v(QvR)))))] [5),v-ORD]
7) [(Pv(P&(Q&R))) SYN (P&((PvQ)&((PvR)&(Pv(QvR)))))] [6),Ax.1-02,R1]

T1-23. [(P&(Pv(QvR))) SYN (Pv((P&Q)v((P&R)v(P&(Q&R)))))]

<u>Proof</u>: 1) [(P&(Pv(QvR))) SYN (P&(Pv(QvR)))] [T1-11,U-SUB]
2) [“ SYN ((P&P)v(P&(QvR)))] [1),Ax.1-08(DR1-01),R1]
3) [“ SYN (Pv(P&(QvR)))] [2),Ax.1-01(DR1-01),R1]
4) [“ SYN (Pv((P&Q)v(P&R)))] [3),Ax.1-08(DR1-01),R1]
5) [“ SYN (Pv((P&Q)v((P&R)v((P&Q)&(P&R)))))] [4),9,R1]
6) [“ SYN (Pv((P&Q)v((P&R)v((P&P)&(Q&R)))))] [5),v-ORD]
7) [(P&(Pv(QvR))) SYN (Pv((P&Q)v((P&R)v(P&(Q&R)))))]
[6),Ax.1-01(DR1-01),R1]

T1-24. [(Pv(P&(Q&R))) SYN (P&(Pv(QvR)))]

<u>Proof</u>: 1) [(Pv(P&(Q&R))) SYN (Pv(P&(Q&R)))] [T1-11,U-SUB]
2) [“ SYN (P&(PvQ)&(PvR)&(Pv(QvR)))] [1),T1-22(DR1-01),R1]
3) [“ SYN ((P&P)&(PvQ)&(PvR)&((Pv(QvR))&(Pv(QvR))))]
[2),Ax.1-01(DR1-01),R1(twice)]
4) [“ SYN ((P&(Pv(QvR)))&(P&(PvQ)&(PvR)&(Pv(QvR))))] [3),&-ORD]
5) [“ SYN ((P&(Pv(QvR)))&(Pv(P&(Q&R))))] [4),T1-22,R1]
6) [“ SYN (P&((Pv(QvR))&(Pv(P&(Q&R)))))] [5),Ax.1-05,R1]
7) [“ SYN (P&(((Pv(QvR))&P)v((Pv(QvR))&(P&(Q&R)))))]
[6),Ax.1-08(DR1-01),R1]
8) [“ SYN ((P&((P&(Pv(QvR)))v(P&(Q&(R&(Pv(QvR))))))] [7),&-ORD]
9) [“ SYN (P&((P&(Pv(QvR)))v(P&(Q&R)))))] [8),T1-20,R1]
10) [“ SYN (P&(((P&P)v(P&(QvR)))v(P&Q&R)))] [9),Ax.1-08(DR1-01),R1]
11) [“ SYN (P&((Pv(P&(QvR)))v(P&Q&R)))] [10),Ax.1-01,R1]
12) [“ SYN (P&((Pv((P&Q)v(P&R)))v(P&Q&R))) [11),Ax.1-08(DR1-01),R1]
13) [“ SYN (P&(Pv((P&Q)v((P&R)v(P&Q&R)))))] [12),v-ORD]
14) [“ SYN (P&(P&(Pv(QvR))))] [13),T1-23,R1]
15) [“ SYN ((P&P)&(Pv(QvR)))] [14),Ax.1-05(DR1-01),R1]
16) [(Pv(P&(Q&R))) SYN (P&(Pv(QvR)))] [15),Ax.1-01,R1]

T1-25. [(P&(Pv(QvR))) SYN (Pv(P&(Q&R)))]
<u>Proof</u>: [T1-24,DR1-01]

T1-26. [(P&(PvQ)&(PvR)&(Pv(QvR))) SYN (P&(Pv(QvR)))] [C-Expansion]
<u>Proof</u>: 1) [(P&(PvQ)&(PvR)&(Pv(QvR)))) SYN (Pv(P&Q&R)))] [T1-22,DR1-01]
2) [(P&(PvQ)&(PvR)&(Pv(QvR)))) SYN (P&(PvQvR)))] [1),T1-25,R1]

T1-27. [(Pv(P&Q)v(P&R)v(P&(Q&R))) SYN (Pv(P&(Q&R)))] [D-Expansion]
<u>Proof</u>: 1) [(Pv(P&Q)v(P&R)v(P&(Q&R)))) SYN (P&(PvQvR)))] [T1-23,DR1-01]
2) [(Pv(P&Q)v(P&R)v(P&(Q&R)))) SYN (Pv(P&Q&R)))] [1),T1-24,R1]

The theorems above have been selected because they are all required for metatheorems which follow about properties of the SYN-relation, including the completeness and soundness of the formal system of SYN relative to its intended semantic interpretation.

The last eight theorems are introduced for their use in connection with quantification theory. They establish basic synonymies between pairs of logical structures which provide a basis from which all theorems in quantification theory can be derived.

T1-28. [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & (PvR))]
<u>Proof</u>: 1) [((P&Q)v(R&S)) SYN (((P&Q)vR)&((P&Q)vS))] [Ax.1-07,U-SUB]
2) [((P&Q)v(R&S)) SYN ((Rv(P&Q))&(Sv(P&Q))))] [1),Ax.1-02(DR1-01),R1(twice)]
3) [((P&Q)v(R&S)) SYN ((RvP)&(RvQ)&(SvP)&(Sv Q))] [2),Ax.1-07(DR1-01),R1(twice)]
4) [((P&Q)v(R&S)) SYN (((RvP)&(RvP))&(RvQ)&(SvP)&(SvQ))] [3),Ax.1-01(DR1-01),R1]
5) [((P&Q)v(R&S)) SYN ((RvP)&(RvQ)&(SvP)&(SvQ))&(RvP))] [4),&-ORD]
6) [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & (RvP))] [5),4),R1]
7) [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & (PvR))] [6),Ax.1-02),R1]

T1-29. [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v (P&R))]
<u>Proof</u>: 1) [((PvQ)&(RvS)) SYN (((PvQ)&R)v((PvQ)&S))] [Ax.1-08,U-SUB]
2) [((PvQ)&(RvS)) SYN ((R&(PvQ))v(S&(PvQ)))] [1),Ax.1-01(DR1-01),R1(twice)]
3) [((PvQ)&(RvS)) SYN ((R&P)v(R&Q)v(S&P)v(S&Q))] [2),Ax.1-08(DR1-01),R1(twice)]
4) [((PvQ)&(RvS)) SYN (**((R&P)v(R&P))**v(R&Q)v(S&P)v(S&Q))] [3),Ax.1-02(DR1-01),R1]
5) [((PvQ)&(RvS)) SYN (((R&P)v(R&Q)v(S&P)v(S&Q))v(R&P))] [4),v-ORD]
6) [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v (R&P))] [5),3),R1]
7) [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v (P&R))] [6),Ax.1-01),R1]

T1-30. [((P&Q)&(RvS)) SYN ((P&Q) & ((P&R)v(Q&S)))]
<u>Proof</u>: 1) [((P&Q)&(RvS)) SYN (((P&Q)&R)v((P&Q)&S))] [Ax.1-08,U-SUB]
2) [((P&Q)&(RvS)) SYN (((P&Q)&(P&R))v((P&Q)&(Q&S)))] [1),&-ORD]
3) [((P&Q)&(RvS)) SYM ((P&Q) & ((P&R)v(Q&S)))] [2),Ax.1-08(DR1-01),R1]

The dual SYN-theorem of T1-30, which is T1-31, will be of little use to us; we list it with proof (which is dual to the proof of T1-30).

T1-31. [((PvQ)v(R&S)) SYN ((PvQ) v ((PvR)&(QvS)))]
Proof: 1) [((PvQ)v(R&S)) SYN (((PvQ)vR)&((PvQ)vS))] [Ax.1-07,U-SUB]
2) [((PvQ)v(R&S)) SYN ((PvQ)v(PvR))&((PvQ)v(QvS))] [1),v-ORD]
3) [((PvQ)v(R&S)) SYM ((PvQ) v ((PvR)&(QvS)))] [2),Ax.1-07(DR1-01),R1]

The next theorem, T1-32, is the wff from which eventually we derive (in Chapter 4) Russell's version of Leibniz' "praeclarum theorema".[15] It also expresses the basic semantic relationship upon which Quine's meta-axiom, *102. $\vdash [(\forall x)(P \supset Q) \supset ((\forall x)P \supset (\forall x)Q))]$, rests, as will be shown in Chapter 4.

T1-32. [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) & (PvRv(Q&S))]
Proof: 1) [((PvQ)&(RvS)) SYN ((RvS)&(PvQ))] [Ax.1-02,U-SUB]
2) [((PvQ)&(RvS)) SYN (((RvS)&P)v((RvS)&Q))] [1),Ax.1-08(DR1-01),R1]
3) [((PvQ)&(RvS)) SYN (((P&(RvS))v(Q&(R vS)))] [2),Ax.1-01,R1(twice)]
4) [((PvQ)&(RvS)) SYN (((P&R)v(P&S))v((Q&R)v(Q&S)))
[3),Ax.1-08(DR1-01),R1(twice)]
5) [((PvQ)&(RvS)) SYN ((PvPvQv(Q&S))&(PvPvRv(Q&S))
&(PvSvQv(Q&S))&(PvSvRv(Q&S))
&(RvPvQv(Q&S))&(RvPvRv(Q&S))
&(RvSvQv(Q&S))&(RvSvRv(Q&S))] [4),Gen v&-DIST]
6) [((PvQ)&(RvS)) SYN ((PvPvQv(Q&S))&((PvPvRv(Q&S))&(PvPvRv(Q&S)))
&(PvSvQv(Q&S))&(PvSvRv(Q&S))
&(RvPvQv(Q&S))&(RvPvRv(Q&S))
&(RvSvQv(Q&S))&(RvSvRv(Q&S))] [5),Ax.1-01(DR1-01),R1]
7) [((PvQ)&(RvS)) SYN ((PvPvQv(Q&S))&(PvPvRv(Q&S))
&(PvSvQv(Q&S))&(PvSvRv(Q&S))
&(RvPvQv(Q&S))&(RvPvRv(Q&S))
&(RvSvQv(Q&S))&(RvSvRv(Q&S)))
&(PvPvRv(Q&S)))] [6),&-ORD]
8) [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) & (PvPvRv(Q&S)))] [7),5),R1]
9) [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) & (PvRv(Q&S)))] [7),Ax.1-02,R1]

The dual of T1-32, which is T1-33, will be of little use to us; we list it without its proof (which is dual to the proof of T1-32).

T1-33. [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) v (P&R&(QvS)))]

Theorems 8 to T1-35, with Ax.1-01 to Ax.1-08 are sufficient to yield proofs of the SYN-Metatheorems 1-12 which establish completeness and soundness of this portion of analytic logic relative to the concept of logical referential synonymy.

---

15. Cf. Russell, B. & Whitehead A.N., *Principia Mathematica*, Vol 1, 1925, page 113, theorem *3.47. Russell cites Gerhardt's edition of Leibniz' *Philosophical Works*, Vol. vii, page 223.

The final two theorems are useful in proofs of quantification theory which deal with scope changes in prenex quantifier sequences.

T1-34. [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & ((PvR)&(QvS)))]
Proof: 1) [((P&Q)v(R&S)) SYN ((P&Q)v(R&S))] [T1-11,U-SUB]
2) [((P&Q)v(R&S)) SYN ((Q&P)v(S&R))] [1),Ax.1-03,R1(twice)]
3) [((Q&P)v(S&R)) SYN (((Q&P)v(S&R)) & (QvS))] [T1-28,U-SUB]
4) [((P&Q)v(R&S)) SYN (((Q&P)v(S&R)) & (QvS))] [2),3),DR1-02]
5) [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & (QvS))] [4),2),R1]
6) [(((P&Q)v(R&S)) & (PvR)) SYN ((P&Q)v(R&S))] [T1-28,DR1-01]
7) [((P&Q)v(R&S)) SYN ((((P&Q)v(R&S)) & (PvR)) & (QvS))] [5),6),R1]
8) [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & ((PvR) & (QvS)))]
[7),Ax.1-05(DR1-01),R1]

T1-35. [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v ((P&R)v(Q&S)))]
Proof: 1) [((PvQ)&(RvS)) SYN ((PvQ)&(RvS))] [T1-11,U-SUB]
2) [((PvQ)&(RvS)) SYN ((QvP)&(SvR))] [1),Ax.1-04,R1(twice)]
3) [((QvP)&(SvR)) SYN (((QvP)&(SvR)) v (Q&S))] [T1-29,U-SUB]
4) [((PvQ)&(RvS)) SYN (((QvP)&(SvR)) v (Q&S))] [2),3),DR1-02]
5) [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v (Q&S))] [4),2),R1]
6) [(((PvQ)&(RvS)) v (P&R)) SYN ((PvQ)&(RvS))] [T1-29,DR1-01]
7) [((PvQ)&(RvS)) SYN ((((PvQ)&(RvS)) v (P&R)) v (Q&S))] [1),6),R1]
8) [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v ((P&R)v(Q&S)))]
[7),Ax.1-06(DR1-01),R1]

1.13 Basic Normal Forms

With the SYN-relation a unique logical function is established which has no analogue based on truth-functional relations alone.

To show this we define syntactically four kinds of normal forms; we call them 'Basic Normal Forms'. Then we show that for every SYN-equivalence class, X, (i.e., every class X of all wffs which are SYN to each other) there will be one and only one member in each class of Basic Normal Forms which is a member of X. This means there is a many-one functional relation between the members a SYN-equivalence class and one wff of each kind of Basic Normal Form. In effect each of the four classes of Basic Normal Form wffs partitions the whole class of wffs into disjoint classes of mutually SYN wffs.

In this section we define four Basic Normal Forms of wffs syntactically. In Section 1.14 which follows we prove SYN-metatheorems 3 to 7 which establish that every wff is SYN to just one wff in each class of Basic Normal Forms.[16] Metatheorems 3 to 7 are used again in Chapter 4 to provide a decision procedure and a completeness proof for our axiomatization of SYN with respect to any pair of wffs in standard logic.

The sub-class of normal form wffs I call "Basic Normal Form Wffs" divides into the following four distinct sub-sub-classes:

---

16. In this chapter the term 'wffs' refers only to negation-free wffs composed of letters with conjunction and disjunction, as defined for the present axiomatic system. In following chapters the 'wff' is defined more broadly with the successive introduction of predicates and individual constants, variables and quantifiers,

1) MOCNFs—Maximal Ordered Conjunctive Normal forms
2) MODNFs—Maximal Ordered Disjunctive Normal Forms
3) MinOCNFs—Minimal Ordered Conjunctive Normal Forms
4) MinODNFs—Minimal Ordered Disjunctive Normal Forms

The logical SYN-relationships betweem members of these classes and other wffs have no analogues in the logical truth-functional relationships of any normal form wffs to other wffs. The four different sub-classes serve various different purposes, but any two of them can be put in one-to-one correspondence under the SYN-relation.

To define these four classes it suffices to define the MOCNFs first, then define the other sub-classes by reference to them. A composite definition of "basic normal forms" may be constructed from the definitions of the four sub-classes.

1.131 Maximal Ordered Conjunctive Normal Forms (MOCNFs)

Three types of normal forms are defined before we get to MOCNFs, as follows:

Df 'NF'. A wff, A, is a **normal form** wff (abbr.'NF')
$\underline{Syn}_{df}$ (i) A has no sentential operators other than '&','v' and '∼',
(ii) and negation signs,if any, occur only in elementary wffs.

Df 'CNF'. A wff A is a **conjunctive normal form** wff (abbr.'CNF')
$\underline{Syn}_{df}$ (i) A is a normal form wff
(ii) and no '&' lies in the scope of any 'v' in A.

Df 'MCNF'. A wff, A, is a **maximal conjunctive normal form** wff, (abbr. 'MCNF')
$\underline{Syn}_{df}$ (i) A is a CNF, and
(ii) A has a conjunct (the maximal conjunct) which contains as disjuncts every elementary wff which occurs in A, and
(iii) for any conjunct, C_i, of A which lacks any elementary wff in the maximal conjunct of A, there is another conjunct, C_j, of A which contains as disjuncts all elementary wffs in C_i plus the one C_i lacks, and
(iv) no conjunct of A contains two occurrences of the same elementary wff as disjuncts and
(v) no two conjuncts of A contain the same set of elementary wffs.

Df 'MOCNF'. A, is a **maximal ordered conjunctive normal form** wff(abbr. 'MOCNF')
Syn_{df} (i) A is an MCNF, and
(ii) conjuncts with fewer elementary wffs occur to the left of conjuncts with more elementary wffs, and
(iii) within each conjunct elementary wffs occur left-to-right in alphabetical order, and
(iv) among conjuncts having the same number of elementary wffs each conjunct, Ci, occurs to the left of any conjunct whose left-most elementary wff is alphabetically later than Ci's left-most different elementary wff, and
(v) complex conjunctions and complex disjunctions are all associated to the right.

EXAMPLES: (All are TF-equivalent, as well as SYN, to each other):
Given: 1) ((BvA) v ((DvC) & A))
CNF: 2) ((BvAvCvD) & (BvAvA))
MCNF: 3) ((BvAvC) & (BvAvCvD) & (BvA) & (AvBvD))
MOCNF: 4) ((AvB) & (AvBvC) & (AvBvD) & (AvBvCvD))

1.132 Maximal Ordered Disjunctive Normal Forms (MODNFs)

Corresponding to CNFs, MCNFs and MOCNFs are disjunctive normal forms (DNFs), maximal disjunctive normal forms (MDNFs), and maximal ordered disjunctive normal forms (MODNFs). These may be defined by interchanging 'v' and '**.**', 'conjunct' and 'disjunct', 'conjunctive' and 'disjunctive', 'conjunction' and 'disjunction' in the definiens of Df 'CNF', Df 'MCNF', and Df 'MOCNF'.

EXAMPLES: (All are TF-equivalent,as well as SYN, to each other).
Given: 1) ((BvA) v ((DvC) & A)) [Same as above]
DNF: 2) (B v A v (A&D) v (A&C))
MDNF: 3) (B v A v (B&A) v (A&C) v (A&D) v (B&C) v(B&D) v
v (B&A&C) v (B&A&D) v (A&C&D) v (B&D&C) v (B&A&C&D))
MODNF: 4) (A v B v (A&B) v (A&C) v (A&D) v (B&C) v (B&D) v
v (A&B&C) v (A&B&D) v (A&C&D) v (B&C&D) v (A&B&C&D))

1.133 Minimal Ordered Conjunctive and Disjunctive Normal Forms

Minimal Ordered CNFs and DNFs are like maximal ordered forms, except that "intermediate conjuncts" of MOCNFs and "intermediate disjuncts" of MODNFs are deleted. "Intermediate" conjuncts (disjuncts) are ones that are neither maximal nor elementary and contain all elementary components of some other conjunct (disjunct). MOCNFs and MODNF are left with only Maximal Conjunct (Disjunct) and the Minimal Conjuncts (Disjuncts) which are defined as follows,

Df 'MinCONJ' 'z is a minimal conjunct of x'
Syn_{df} (i) 'x is a CNF wff, and (ii) z is a conjunct of x, and
(iii) every other conjunct of x has some disjunct which is not a disjunct of z'.

Df 'MinDISJ' 'z is a minimal disjunct of x'
Syn_{df} (i) 'x is a DNF wff, and (ii) z is a disjunct of x ,and
(iii) every other disjunct of x has some conjunct which is not a conjunct of z'.

Intermediate conjuncts (disjuncts) of basic normal forms are those that are neither maximal nor minimal. By definition, MinOCNF wffs (MinODNF wffs) have no intermediate conjuncts (disjuncts).

Examples of MinOCNF and MinODNF: (All are SYN to examples in Sections 1.131 and 1.132)
Given: ((BvA) v ((DvC) & A))
MinOCNF: ((AvB) & (AvBvCvD))
MinODNF: (A v B v (A&B&C&D))

These may be compared to MOCNF and MOCDF forms of the same wff:

MOCNF: ((AvB) & (AvBvC) & (AvBvD) & (AvBvCvD))
MODNF: (A v B v (A&C) v (A&D) v (B&C) v (B&D) v
v (A&B&C) v (A&B&D) v (A&C&D) v (B&C&D) v (A&B&C&D))
(the eliminated intermediate disjuncts are underlined.

1.134 "Basic Normal Forms" in General

The composite general concept of a basic normal form may be expressed as follows: 'A wff, A, is **a basic normal form wff**' Syn_{df} 'A is a uniquely ordered conjunctive (disjunctive) normal form wff, with all of its elementary wffs disjoined (conjoined) in one conjunct (disjunct), the Maximal Conjunct (Disjunct); with no elementary wffs occurring twice in any conjunct (disjunct); with no two conjuncts (disjuncts) containing the same set of elementary wffs; and with either all, or no, intermediate conjuncts (disjuncts)'.

Obviously, the particular method of ordering spelled out in clauses (ii) to (v) of Df 'MOCNF' (and derivatively Df 'MODNF', Df 'MinOCNF' and Df 'MinODNF') is not the only way to provide a unique order for the symbols in a CNF or DNF wff. Thus this particular rule is not essential; but some such rule of ordering is necessary to get the Uniqueness Theorem (SYN- Metatheorem 7).

Each of the different sub-classes of Basic Normal Forms have distinct advantages for some syntactical or semantic problems over the others. The MOCNFs are particularly useful in defining the concept of analytic containment or entailment, and in defining the concept of a necessary condition (the truth of each conjunct is a necessary condition of the truth of the whole. The MODNFs are useful in defining the concept of a sufficient condition and of an INUS conditions (the truth of any disjunct is a sufficient condition for the truth of the whole). Minimal forms of both sorts are particularly useful in simplifying derivations and proofs, especially in quantification theory.

1.14 SYN-metatheorems on Basic Normal Forms

1.141 Metatheorems about SYN-equivalent Basic Normal Forms

Basic Normal Form Metatheorems are those which relate wffs in basic normal forms by the SYN-relation to other wffs, or classes of wffs, of standard sentential logic. In this section I prove four metatheorems, culminating in the Uniqueness Theorem, SYN-metatheorem 7 which says that every wff is SYN to one and only one basic normal form wff of any one of the four types.

This uniqueness is significant. It is essential for the completeness proof of our axiomatization of the SYN relation among wffs of standard M-logic (see Section 5.34). In 1938 Hilbert and Ackerman pointed out that the normal forms used in standard logic are not unique. They stated that,

"the following equivalences [are] of importance for representung the relation of having the same truth value: (29) X ≡ Y eq ((~XvY) & (~YvX))
(30) X ≡ Y eq ((X&Y) v (~X&~Y))",

and they wrote two pages later,

> "We note, incidently, that the normal form belonging to a combination of sentences is not unique. For example, by (29) there belongs to (X≡Y) on the one hand the normal form ((~X v Y) &

(~YvX)). On the other hand, by applying the distributive law to the right side of (30), we have (Xv~X & (~XvY) & (Yv~X) & (Yv~Y))."[17]

The latter two normal forms are truth-functionally equivalent but not SYN, though they contain all and only the same elementary wffs. By defining SYN and normal forms independently of negation we establish equivalence classes under the SYN relation that have no analogues under the relation of truth-functional equivalence.

We first prove three metatheorems for MOCNFs:

SYN-metatheorem 4. Every wff is SYN to at least one MOCNF wff.
SYN-metatheorem 5. No wff is SYN to more than one MOCNF-formula.
SYN-metatheorem 6. Every wff is SYN to one and only one MOCNF wff.

Since MODNF wffs are closely related by duality principles to MOCNF wffs, analogous proofs can be given for SYN-metatheorems like SYN-metatheorem 6 for MODNFs. And since MinOCNF and MinODNF wffs are uniquely defined from MOCNF and MODNF wffs, similar SYN-metatheorems can be proved for MinOCNF and MinODNF wffs. These consequences are summarized in the Uniqueness Metatheorem,

SYN-metatheorem 7. Every wff is SYN to one and only one member from each class of MOCNF, MinOCNF, MODNF, and MinODNF wffs.

These SYN-metatheorems have no analogues with TF-equivalence. Due to Absorption, there are an infinite number of MOCNFs, etc., which are TF-equivalent to any given wff, and thus an infinite number of MOCNFs, MODNFs, MinOCNFs and MinODNFs, which are TF-equivalent to a given wff. [Proof: given any MOCNF wff, A, simply add (Cv-C) as the right-most disjunct in every conjunct of A, where C is a letter alphabetically later than any sentence letter contained in A; the result will be a new MOCNF which is not SYN, but is TF-equivalent to A].

Since the particular mode of ordering in Df 'MOCNF' is not the only way to get unique orderings of DNF and CNF wffs, we can't say that each wff is SYN-equivalent to just one of each of only four **basic normal forms**; innumerable other conventions might be adopted to get unique orders for wffs. But this is immaterial.

1.142 Proofs of SYN-metatheorems 4-7

To prove SYN-metatheorem 7 we need SYN Metatheorems 4, 5 and 6. The task is to prove there is a derivation which starts with U-SUB in the theorem, [A SYN A], and ends with a theorem [A SYN B], where B is a MOCNF wff.

SYN-metatheorem 4. Every wff is SYN to at least one MOCNF wff.
Proof:
1) The necessary and sufficient conditions of being an MOCNF-wff are the following [by definitions Df 'NF', Df 'CNF', Df 'MCNF', and Df 'MOCNF']:

---

17. From D.Hilbert and W.Ackerman, *Principles of Mathematical Logic*, Chelsea Publishing Company, New York 1950, pp 11 and 13.(Translation of *Grundzuge der Theoretischen Logik*, 2nd Ed.,1938). The logical notation used is the notation of this book, rather than Hilbert and Ackerman's.

(i) A is a normal form wff, i.e., no operators other than '&'s, 'v's and '∼' prefixed only to sentence letters.
(ii) no '&' lies in the scope of any 'v' in A;
(iii) A has a conjunct (called "the maximal conjunct") containing as disjuncts every elementary wff which occurs any where in A;
(iv) for any conjunct, C_i, of A which lacks any one of the elementary disjuncts in the maximal conjunct of A, there is another conjunct, C_j, of A which contains all of the elementary disjuncts of C_i plus just one elementary disjunct which C_i lacks;
(v) no conjuncts of A contain two occurrences of the same elementary wff;
(vi) no two conjuncts of A contain the same set of elementary wffs;
(vii) conjuncts with less elementary wffs precede conjuncts with more elementary wffs;
(viii) within each conjunct, elementary wffs occur left-to-right in alphabetical order;
(ix) among conjuncts having the same number of distinct elementary wffs, each conjunct C_i occurs to the left of any conjunct whose left-most different component is alphabetically later than C_i's left-most different elementary component;
(x) conjunctions and disjunctions are all grouped to the right.

Now we must show that: given any wff A, it is possible to get a wff B which both satisfies all of the conditions (i) to (x) for being an MONCF, and is such that [A SYN B]:

For any wff, A, the first step is [A SYN A] [by 1,U-SUB].
Condition (i) is already met since no wff, A, will not contain any operators other than '&'s and 'v's.
(When negation and quantifiers are introduced in later chapters this condition will be appropriately modified.)
Condition (ii) is satisfied by using Ax.1-07 (v&-Distribution), assisted by Ax.1-04 (v-Commutation) and R1, to yield [A SYN A_1], where A_1 has every occurrence of 'v' in any component (Av(B&C)) or ((B&C)vA) moved into the scope of '&', as in '((AvB)&(AvC))'.
Condition (iii) is reached with [A SYN A_2], where A_1 has a maximal conjunct, by using [(A&(B&(AvB))) SYN (A&B)] [T1-18], as needed assisted by Ax.1-03 (&-Commutation), Ax.1-05 (&-Association), R1 and DR1-01.
Conditions (iv) thru (vi) are satisfied in [A SYN A_3], where A_3 is a <u>maximal</u> conjunctive normal form, gotten by using the Principle of C-Expansion, i.e., T1-26 [(A&((AvB)&((AvC)&(Av(BvC))))) SYN (A&(Av(BvC)))], assisted by &-ORD (DR1-03), v-ORD (DR1-04), and R1.
Condition (iv) requires that if any conjunct, C_i, of A_1 lacks any elementary wff in the maximal conjunct of A_1, another conjunct, C_j, which contains all elementary wffs in C_i plus the lacking elementary wff of C_i, be inserted in A_3.
Conditions (v) and (vi) will be met, using &-Idempotence and v-Idempotence to eliminate unnecessary repetitions.
Conditions (vii) thru (x) are satisfied at [A SYN B], where B is a Maximal <u>Ordered</u> conjunctive normal form, by using DR1-03 (&-ORD) and DR1-04 (v-ORD) with R1 on A_3 in A SYN A_3.

Hence by the steps above it is always possible, given any wff, A, to derive [A SYN B], such that B is an MOCNF wff. I.e., Every wff is SYN to at least one MOCNF wff.

To prove SYN-metatheorem 5 —that no wff is SYN to more than one MOCNF formula—we need the following Lemma which says that two CNF wffs which are SYN must have the same set of minimal conjuncts.

Lemma. If A and B are CNF wffs and [A SYN B]
then A and B have the same set of minimal conjuncts.

The proof for this Lemma is as follows: given the definition of a minimal conjunct, if A and B are MOCNFs and have a different set of minimal conjuncts, they would not be TF-equivalent (since A could be false when B was true). Hence, by the contrapositive of SYN-metatheorem 2a, they could not be SYN. Consequently, if A and B are both CNF and are SYN, they must have the same set of minimal conjuncts.

The crucial part of this proof is the truth-table test.[18] To see how this works, note that according to the definition of 'MinCONJ', z is a minimal conjunct of x if and only if

(i) x is a CNF wff and (ii) z is a conjunct of x and
(iii) every other conjunct of x has at least one disjunct which is not a disjunct of z'.

Therefore, 1) Let x = A such that A is a CNF wff and z = C a minmal conjunct of A,

[e.g., A = '((QvPvR) & (QvPvRvS) & (QvP) & (PvQvS))', and C = '(QvP)',
so that (QvP) is a minimal conjunct of A]

2) Assign F to all elementary wffs in C and T to all elementary wffs not in C.
[e.g. ((QvPvR) & (QvPvRvS) & (QvP) & R & (PvQvS))
f f **t** f f **t** t f**F**f **F** **t** f f **t**

On this assignment every conjunct other than C is either an elementary wff not in C which is assigned t, or is a disjunction of elementary wffs at least one of which is not in C and which is assigned t, This makes each conjunct other than C have the value t, while C itself is f. Thus C alone makes A take the value f on that assignment.

3) Let B be any other CNF wff which has the same elementary wffs as A in the maximal conjunct, but does not have C as a minimal conjunct. Since A and B are both CNF and have the same set of n elementary wffs, the truth-tables of A and B will have the same set of 2^n rows and the values of components of A and B are determined by the values assigned to the elementary wffs.

4) but there will be an assignment of truth values to the elementary wffs in A and B such that A is f and B is t. This is due solely to the lack of the minimal conjunct C in B. For example

[E.g.**:** A = ((QvPvR) & (QvPvRvS) & (QvP) & R & (PvQvS))
f f f **t** f f **t** t <u>f f f</u> **t** f f **t**
B = ((QvPvR) & (QvPvRvS) & R & (PvQvS))
t f f **t** f f **t** t **t** f f **t**]

18. This test does not involve the concepts of truth and falsehood essentially; essentially any two values, e.g., 0 and 1, would provide an appropriate model for this type of equivalent testing. We call them "truth-tables" only because this the most familiar version.

5) Hence A is not TF-equivalent to B, by truth-tables.
6) Hence A is not SYN to B [Contrapositive of SYN-metatheorem 2a].
7) Hence, if A and B are both in conjunctive normal form and do not have the same set of minimal conjuncts, they can't be SYN.
   [By conditional proof from premisses in 1) and 3) to the conclusion 7)].
8) Hence, Lemma 1. If A and B are CNF wffs and [A SYN B] then A and B have the same set of minimal conjuncts.

Next we use this Lemma to prove that if A is a wff and B is an MOCNF wff which is SYN to A, then there is no C, not identical with B, which is an MOCNF and is SYN to A. In other words,

SYN-metatheorem 5. No wff is SYN to more than one MOCNF-formula.

Proof: Suppose [A SYN B] has been proven, where A is any wff and B is an MONCF. Suppose also that we a proof for any other wff, C which is an MOCNF wff, that [C SYN A]. Then [B SYN C] [by DR1-02, the transitivity of SYN]. By Lemma 1, B and C have the same set of minimal conjuncts. And since there are SYN, by SYN-metatheorem 1, they have the same set of elementary wffs, hence identical Maximal Conjuncts. Further, by the definition of 'MOCNF', if they have the same minimal and maximal conjuncts they will have identical intermediate conjuncts. Hence all conjuncts of B and C will be the same, and since both are ordered by the same ordering rules, the order of conjuncts will be the same, and B will be identical to C. Since C represents any arbitrarily chosen wff which is MONCF and SYN to A, it follows that every MOCNF wff which is SYN to A is identical, hence no wff, A, is SYN to more than one MOCNF formula, i.e., SYN-metatheorem 5.

SYN-metatheorem 6, that every wff is SYN to one and only one MOCNF wff, follows from SYN-metatheorem 4 and SYN-metatheorem 5.

SYN-metatheorem 6. Every wff is SYN to one and only one MOCNF wff.

Proof: Every wff is SYN to at least one MOCNF wff.[SYN-metatheorem 4] and no wff is SYN to more than one MOCNF-formula. [SYN-metatheorem 5].

Hence, Every wff is SYN to one and only one MOCNF.

We now define the functional relation, '... **is the MOCNF of ---**'. By SYN-metatheorem 6, we now know this will be a one-many (i.e.a functional) relation in terms of which the functional expres sion, 'the MOCNF of B' (Abbr.,'MOCNF(B)') can be defined.

Df 'A=MOCNF(B)' [For 'A is the MOCNF of B']
   Syn_{df} '(i) B is a wff of sentential logic, and
   (ii) A is an MOCNF wff, and
   (iii) [A SYN B']
   I.e, [A=MOCNF(B) Syn_{df} (BeWff & AeMOCNF & (A SYN B))]

Since MODNF wffs are related by duality principles to MOCNF wffs, parallel proofs can be given for metatheorems, like SYN-metatheorem 6, for MODNFs. And since MinOCNF and MinODNF wffs are uniquely defined from MOCNF and MODNF wffs, similar SYN-metatheorems can be proved for MinOCNF and MinODNF wffs. These consequences are summarized without proof, in

SYN-metatheorem 7. Every wff is SYN to one and only one member from each class of MOCNF, MinOCNF, MODNF, and MinODNF wffs.

On the basis of these analogues to SYN-metatheorem 6, the further functional relations can be defined,

Df 'A	= ModNF(B)'	(for 'A is the MODNF of B),
Df 'A	= MinOCNF(B)'	(for 'A is the MinOCNF of B)'
and Df 'A	= MinODNF(B)'	(for 'A is the MinODNF of B)'

in the same way as Df 'A=MOCNF(B)', changing only the kind of basic normal form.

These uniqueness metatheorems will enable us to develop later a completeness proof for our final axiomatization of the SYN relation for all wffs of standard logic in Chapter 4.

1.2 Logical Containment Among Conjunctions and Disjunctions

The paradigmatic form of logical containment yields the principle of Simplification in mathematical logic. "[P & Q] logically contains P" is its simplest form. It yields "If (P&Q) is true, then P is true"; this is T8-15 in Chapter 8. Logical containment is positive—it can be formalized completely in a negation-free language. To say P logically contains Q is to say that the meaning of Q is logically contained in the meaning of the logical structure of P.

Like, logical synonymy, logical containment is independent of inconsistency and negation. The concept of inconsistency, and denials of inconsistency, require the introduction of negation.

1.21 Definition of 'CONT'

The logical predicate '...CONT—' may be thought of as meaning 'the meaning of...is logically contained in the meaning of—'. But we define it syntactically in terms of logical synonymy and conjunction:

Df 'CONT': [A CONT B] $\underline{\text{Syn}}_{\text{df}}$ [A SYN (A&B)]

Logical containment of B in A obtains only if the synonymy of A and (A&B) is a logical synonymy—i.e., only if [A SYN (A&B)] is a SYN-theorem. Thus to say that a wff of the form [A CONT B] is a theorem is to say that regardless of content, the meaning of A contains the meaning of B by virtue of the meanings of the logical constants and logical structures in A and B.

The paradigm case of logical containment is simplification: (A&B) contains B. In general whatever can be a conjunct in any conjunctive normal form of a wff is logically contained in that wff. In contrast, although (AvB) is true if A is true, (AvB) is not logically contained in A or in B; the meaning of (AvB) is not part of the meaning of A. This coincides with the fact that A is not SYN to (A & (AvB)).

Containment is non-symmetrical:

[A SYN (A&B)] Syn [(A&B) SYN A] Syn [A CONT B]
[B SYN (A&B)] Syn [(A&B) SYN B] Syn [B CONT A]

But [A SYN (A&B)] does not mean the same as [B SYN (A&B)]
and [A CONT B] does not mean the same as [B CONT A]

From Df 'Cont', with &-ORD, we get the general rule,

DR1-10. If [P Syn (Q_1 & Q_2 &…Q_n)] then [(P Cont Q_i) ($1 \leq i \leq n$)]

<u>Proof</u>: 1) [P Syn (Q_1 & Q_2 &…Q_n)] [Assumption]
2) Q_i is any one of _Q_1 ,Q_2 ,..Q_n_ [Assumption]
3) |= [P Syn ((Q_1 & Q_2 &…Q_n) & Q_i)] [1),2), &-IDEM and &-ORD]
4) |= [P Syn (P & Q_i)] [3),1),R1]
5) [P CONT Q_i] [4),Df 'CONT']

As a short-cut we will frequently pick out a Q_i from the conjuncts on the right-hand side of SYN-theorem in some, Step k, [P SYN (..& Q_i ..&..)] and conclude in Step k+1, [P CONT Q_i], citing "Df''CONT'" to justify this step.

1.22 Containment Theorems

Analytic logic begins with logical synonymy. Logical containment is defined in terms of logical synonymy and conjunction. Synonymy is more fundamental than containment. The latter presupposes the more fundamental concept, synonymy, or sameness of meaning. Containment varies with linguistic operators whereas synonymy does not. P and Q mutually contain each other if and only if they are synonymous. This is not to deny that the most interesting cases for logic are the asymmetrical cases where synonymy fails, i.e., where P contains Q but Q does not contain P. However, analytic logic begins with logical synonymy: axioms of the form 'P SYN Q' which mean "P is logically synonymous with Q".

There are many more theorems of the form '[A CONT B]' than there are SYN-theorems. In the first place, every case of logical synonymy is a case of logical containment. Thus, trivially, every SYN-theorem with its converse gives rise to two CONT-theorems. But in addition, for every wff, A, there are as many one-way, non-synonymous CONT-theorems as there are distinct conjuncts in the maximal ordered conjunctive normal form of A (in the MOCNF[A]). Finally, additional infinities of CONT-theorems between wffs which are not SYN, can be derived by replacing either side of the CONT-Theorem by some logical synonym.

In what follows we shall present only CONT-theorems which are derived initially from even-numbered SYN-Theorems. These SYN-theorems, it will be remembered, have (by our convention) the form A SYN (B_1 &…& B_m). Given any such even-numbered theorem it is a simple matter to derive, for any B_i where $1 \geq i \leq m$, a theorem [A CONT B_i] by the following steps:

1) [A SYN (B_1 &…& B_m)] [Assumption]
2) [A SYN ((B_1 &…& B_m) & B_i)] [1),&-ORD]
3) [A SYN (A & B_i)] [2),1),R1]
4) [A CONT B_i] [3),Df 'CONT']

To avoid excessive new theorem numbers, and to display the relation of CONT-theorems to the SYN-theorems from which they are derived, we may use the following system in numbering containment theorems.

Given any SYN-theorem named [Tx-n], where x is the number of the chapter in which the theorem first occurs, and n is numeral expressed in two digits. Suppose A is a wff and (B_1 &…& B_k) is its MOCNF wff so that the following is a theorem: Tx-n. [A SYN (B_1 &…& B_k)].
The Containment theorems derived from A, will be numbered as Tx-nc(1), Tx-nc(2), …Tx- nc(1,2),…,Tx-nc(2,4,5) where any number i following the dash represents the ith conjunct B_i on the right-hand side of

T-xn. For example, T1-22 [(Pv(P&(Q&R))) SYN (P & (PvQ) & (PvR) & (Pv(QvR)))],
has the form, T1-n. [(A) SYN (B_1 & B_2 & B_3 & B_4)]

The right-hand side is the MOCNF of the left-hand side, and therefore, the containment theorem named T1-22c(1,3) is [(Pv(P&(Q&R))) CONT (P&(PvR))]. In general, 'Tx-nc(i,j,k). [A CONT B]' signifies that [A SYN C], where C is the MOCNF of A, and B is the conjunction of the ith, jth and kth conjuncts of C.

CONT-theorems do not have to have a 'c' in their names. Theorems of the form [P CONT Q] may be given an independent number [T1-k]. Also they may appear as steps in a proof without any number. For example, '|= [P CONT P]' is directly derivable by Df 'CONT' from Ax.1-01, which is '[P SYN (P & P)]', and often occurs that way in proofs. Also by Df 'Cont', one can derive [P SYN (P&Q)] as a step in a proof if '[P CONT Q]' has been proven.

To save writing down all steps explicitly, we will often justify a step '(i+1) [P CONT Q]' by [i),Df 'Cont'], if the previous step (i) has the form 'P SYN R', and Q is a conjunct of a Conjunctive Normal Form of R. (The MOCNF of R is simply the CNF of R which contains all conjuncts that occur in any normal form of R.)

As mentioned above, logical synonymies are special cases of logical containment. The whole meaning of A Contains the whole meaning of A. Thus if [A SYN B] then [B SYN A] and both [A CONT B] and [B CONT A]. Formally this is expressed in the form of the two derived rules of inference for containment, DR1-11. If [A SYN B] then [A CONT B]
and DR1-12. If [A SYN B] then [B CONT A]. Their proofs follow:

DR1-11. If [A SYN B] then [A CONT B]

Proof:	1) [A SYN B]	[Assumption]
	2) [B SYN (B&B)]	[Ax.1-01,U-SUB]
	3) [A SYN (B&B)]	[1),2),R1]
	4) [A SYN (A&B)]	[3),1),R1]
	5) [A CONT B]	[4),Df 'CONT',R1]

DR1-12. If [A SYN B] then [B CONT A]

Proof:	1) [A SYN B]	[Assumption]
	2) [B SYN (B&B)]	[Ax.1-01,U-SUB]
	3) [B SYN (B&A)]	[2),1),R1]
	4) [B CONT A]	[3),Df 'CONT',R1]

It follows that all SYN-theorems and their converses transform to CONT-theorems.

What distinguishes CONT from SYN, however, are asymmetrical cases where [A CONT B] is a theorem but its converse is not. These are CONT-theorems whose converses are not theorems. The sinplest and most basic CONT-theorems are: T1-36, T1-37 and T1-38. They can be derived in many different ways, including the following:

T1-36 [(P&Q) CONT P)] [SIMP1]

Proof:	1) [(P&Q) SYN (Q&P)]	[Ax.1-02]
	2) [(P&Q) SYN (Q&(P&P))]	[1),Ax.1-01,SynSUB]
	3) [(P&Q) SYN ((P&Q)&P)]	[2),&-ORD]
	4) [(P&Q) CONT P]	[3) Df 'CONT']

T1-37 [(P&Q) CONT Q)] [SIMP2]
Proof: 1) [(P&Q) SYN (Q&P)] [Ax.1-02]
2) [(P&Q) SYN ((Q&Q)&P))] [1),Ax.1-01,SynSUB]
3) [(P&Q) SYN ((P&Q)&Q)] [2),&-ORD]
4) [(P&Q) CONT Q] [3) Df 'CONT']

T1-38 [(P&Q) CONT (PvQ)]
Proof: 1) [(P&Q) SYN (P&(Q&(PvQ)))] [T1-18,DR1-01]
2) [(P&Q) SYN ((P&Q)&(PvQ)))] [1),&-ORD]
3) [(P&Q) CONT (PvQ)] [2),Df 'CONT']

Another interesting CONT-theorem which will be useful later is,

T1-39 [(P& (QvR)) CONT ((P&Q) v R)]
Proof: 1) (P&(QvR)) SYN ((P&Q)v(P&R)) [T1-08(&v-DIST),U-SUB]
2) (P&(QvR)) SYN ((Pv(P&R)) & (Qv(P&R)) [1),T1-08,SynSUB]
3) (P&(QvR)) SYN (((PvP) & **(PvR))** & ((QvP) & **(QvR)**)) [2),T1-04,(twice)]
4) (P&(QvR)) CONT ((PvR) & (QvR)) [3)Df 'CONT']
5) (P&(QvR)) CONT ((P&Q)vR) [4),T1-04,v&-Dist]

The enormity of the set of new theorems added by non-symmetrical CONT-theorems is suggested by the following considerations. By SYN-metatheorem 6, every wff is SYN to one and only one MOCNF wff. From the non-synonymous conjuncts of the MOCNF form of any wff, A, theorems of the form [A CONT B] can be derived in which B is not SYN with A so its converse fails.

The simplest paradigm case is the following. The MOCNF of (P&Q) is (P & (Q & (PvQ))). That the latter is SYN with (P&Q) is established in T1-18,. [(P & (Q & (PvQ))) SYN (P&Q)]. Thus we will label the three simple CONT-theorems for (P&Q),8c(1), 8c(2) and 8c(3), where the number in parentheses after the 'c' represents the ith conjunct in the MOCNF form of the wff on the left. These can be derived from 8 as the first step but they can also be proved more easily in other ways.

Thus, T1-36 can also be presented as T18c(1) [(P&Q) CONT P)] and proved as follows:

T1-18c(1) [(P&Q) CONT P)]
Proof: 1) [(P&Q) SYN (P&Q)] [T1-11,U-SUB]
2) [(P&(Q&(PvQ))) SYN (P&Q)] [T1-18]
3) [(P&Q) SYN (P&(Q&(PvQ)))] [1),2),SynSUB]
4) [(P&Q) SYN ((P&(Q&(PvQ)))&P)] [3)&-ORD]
5) [(P&Q) SYN ((P&Q)&P)] [3),4),SynSUB]
6) [(P&Q) CONT P] [5),Df 'CONT']

Similarly T1-37 may be established as T1-18c(2) [(P&Q) CONT Q)] and T1-38 might also have been identified as T1-18c(3) [(P&Q) CONT (PvQ)]. Consider also the unique MOCNF of (Pv(P&(Q&R))) which is given on the left hand side of T1-22.

T1-22. [(P v (P & (Q&R))) SYN (P & ((PvQ) & ((PvR) & (Pv(QvR)))))]
has the form, [(A) SYN (B_1& B_2 & B_3 & B_4)]

Using the steps mentioned above we can derive containment theorems for (Pv(P&(Q&R))) which state that each conjunct, or conjunction of conjuncts on the right-hand side of T1-22 (Pv(P&(Q&R))) itself, is logically contained in (Pv(P&(Q&R))). In this case we immediately get nine CONT-theorems which are not synonyms. Each establishes the logical containment of expressions of a certain form within the left-hand expression, with which they are not synonymous (either because they don't have all the same sentence letters, or because they are not truth-functionally equivalent):

T1-22. [(P v (P & (Q&R))) SYN (P & ((PvQ) & ((PvR) & (Pv(QvR)))))]
[(A) SYN (B_1& B_2 & B_3 & B_4)]
T1-22c(1) [(Pv(P&(Q&R))) CONT P]
T1-22c(2) [(Pv(P&(Q&R))) CONT (PvQ)]
T1-22c(3) [(Pv(P&(Q&R))) CONT (PvR)]
T1-22c(4) [(Pv(P&(Q&R))) CONT (Pv(QvR))]
T1-22c(1,2) [(Pv(P&(Q&R))) CONT (P&(PvQ))]
T1-22c(1,3) [(Pv(P&(Q&R))) CONT (P&(PvR))]
T1-22c(2,3) [(Pv(P&(Q&R))) CONT ((PvQ) & (PvR))]
T1-22c(2,4) [(Pv(P&(Q&R))) CONT ((PvQ) & (Pv(QvR)))]
T1-22c(3,4) [(Pv(P&(Q&R))) CONT ((PvR) & (Pv(QvR)))]

Further, (Pv(P&(Q&R))) can be replaced by any logical synonym of it (of which there are infinitely many) on the left side; the result will still be an assertion of a non-synonymous logical containment. Similarly the right hand sides can be replaced by any logical synonyms (of which there will be infinitely many) preserving the assertion of a non-synoymous logical containment.

In general given any wf, A, we prove theorems of this sort as follows:

(1) start with U-SUB in T1-11, i.e., [A SYN A] where A will be the containing wff,
then (2) by methods described in Section 1.142 we derive [A SYN B] where B is the MOCNF of A, and
then (3) by &-Idempotence and DR1-03 we get [A SYN (B&C)] where B is the MOCNF of A, and C is a conjunct, or conjunction of conjuncts of B;
then (4) from [A SYN B] and [A SYN (B&C)] by R1 we get [A SYN (A&C)];
then (5) by Df 'CONT', we get [A CONT C], where C is a conjunct or conjunction of conjuncts of the MOCNF of A.
then (6) Further contaiment theorems can then be gotten by replacing C, the contained wff, with any of its infinite set of synonyms.

Thus in addition to all theorems of the form [A CONT B] where A and B are SYN, we have a vast sub-set of ordered pairs of wffs, <A,B>, where B is not SYN to A but is logically contained in A.

A will logically contain B only if every elementary wff in B occurs in A. Also, A will contain B if and only if B is SYN to a conjunct or conjunction of conjuncts of some conjunctive normal form of A. After negation is introduced we will see that more constraints appear, and the gap between containment and "logical implication" in standard logic is increased. But the two constraints just mentioned are the basic ones which make containment a stricter relation than the "logical implication" of standard logic.

Among ordered pairs of wffs which are truth-functional "implications" but not logical containments are Principles of Addition. These have no analogue in containment. A does not, in general, logically contain (AvB), since B does not occur in A.

So far as they relate to standard logic (including quantification theory) all fundamental principles of logical containment, like all basic principles of logical synonymy, are rooted in the negation-free logic of 'and' and 'or'. Later introductions of negation, quantifiers and individual variables yield new classes of wffs and thus many new theorems. But what makes pairs of these new wffs logically synonymous remains relationships between logical structures established by '&' and 'v'. The new theorems add expressions in which the meanings of 'it is not the case that', 'all' and 'some' are included, but they do not add anything new to the criterion of whether A logically contains B or not. Thus the fundamental concepts of logical synonymy and containment as they relate to standard logic, will not undergo any further basic changes. Logical containment and logical synonymy of canonical expressions in the language of standard logic is completely defined in the logic of 'and' and 'or'.

1.23 Derived Containment Rules

The definition Df 'CONT'is basically an abbreviation in the metalanguage; '[A CONT B]' abbreviates '[A SYN (A&B)]'. Using this we can establish many derived rules which render the formal system of 'CONT' and 'SYN' more germane. Containment theorems are important because of the intimate connection between logical containment and the concepts of "following logically from", and of logically valid conditionals. In this section several derived rules of inference for reaching containment theorems are established. Later (Chapter 6) valid logical rules of inference, expressed as conditionls, will be derivable from them. The first two rules are about the derivability of logical containment from logical synoymy:

DR1-11. If [A SYN B], then [A CONT B]

Proof:	Step 1) [A SYN B]	[Assumption]
	Step 2) [B SYN (B&B)]	[Ax.1-01,U-SUB]
	Step 3) [(B&B) SYN B]	[2),DR1-01]
	Step 4) [A SYN (B&B)]	[3),1),R1]
	Step 5) [A SYN (A&B)]	[4),1),R1]
	Step 6) [A CONT B]	[5),Df 'CONT']

DR1-12. If [A SYN B] then [B CONT A]

Proof:	Step 1) [A SYN B]	[Assumption]
	Step 2) [B SYN A]	[1),DR1-01]
	Step 3) [B SYN (A&A)]	[1), Ax.1-01,SynSUB]
	Step 4) [B SYN (B&A)]	[3),2),SynSUB]
	Step 5) [B CONT A]	[5),Df 'CONT']

Obviously, If [A CONT B] then [A SYN (A&B)]

Proof:	Step 1) [A CONT B]	[Assumption]
	Step 2) [A CONT B] Syndf [A SYN (A & B)]	[Df 'CONT']
	Step 3) [A SYN (A&B)]	[1),2),SynSUB]

The following are helpful:

DR1-13. If [A CONT (B&C) then (A CONT B)]

Proof: Step 1) A CONT (B&C) [Assumption]
Step 2) A SYN (A&(B&C)) [1),Df 'CONT']
Step 3) A SYN ((A&B)&C) [2), Ax.1-05]
Step 4) (A&B) SYN (A&B) [T1-11,U-SUB]
Step 5) (A&B) SYN ((A&B&C)&B) [3), DR1-21,R1]
Step 6) (A&B) SYN ((A&(B&B)&C) [5),DR1-03(&-ORD)]
Step 7) (A&B) SYN (A&B&C) [6),Ax.1-01,(&-IDEM),R1]
Step 8) A SYN (A&B) [2),7),R1]
Step 9) A CONT B [8),Df 'CONT']

DR1-14. If [(A SYN B) then ((A CONT B) and (B CONT A))]
Proof: Step 1) A SYN B [Assumption]
Step 2) (A CONT B) [1),DR1-11,MP]
Step 3) (B CONT A) [1),DR1-12,MP]
Step 4) ((A CONT B) & (B CONT A)) [2),3),ADJ]

The following rule is obvious:

DR1-15. If [A SYN (B&C)], then [A CONT B]

and the following rules express the special case of SynSUB applied to CONT-theorems:

DR1-16. If [A SYN B] and [C CONT D], then [C CONT D(A//B)]
or, If [A SYN B] and [C CONT D], then [C(A//B) CONT D]

The converses of these derived containment rules do not hold; it does not follow from the fact that A CONT B, that A SYN B and/or that B CONT A.

DR1-17. If [A CONT B], then [(CvA) CONT (CvB)]
Proof: Step 1) [A CONT B] [Assumption]
Step 2) [A SYN (A&B)] [1),Df 'CONT']
Step 3) [(A&B) SYN B] [2),DR1-01]
Step 4) [(CvA) SYN (CvA)] [T1-11,U-SUB]
Step 5) [(CvA) SYN (Cv(A&B))] [4),3),R1]
Step 6) [((CvA)&(CvB)) SYN (Cv(A&B))] [Ax.1-07,DR1-01]
Step 7) [(CvA) SYN ((CvA)&(CvB))] [5),6),R1]
Step 8) [(CvA) CONT (CvB)] [7),Df 'CONT']

On the other hand synonymy is derivable from mutual containment and conversely:

DR1-18. If [A CONT B] and [B CONT A] then [A SYN B]
Proof: Step 1) [A CONT B] [Assumption]
Step 2) [B CONT A] [Assumption]
Step 3) [A SYN (A&B)] [1),Df 'CONT',R1]
Step 4) [B SYN (B&A)] [2),Df 'CONT',R1]
Step 5) [(B&A) SYN (A&B)] [Ax.1-03]
Step 6) [A SYN (B&A)] [5),3),R1]
Step 7) [A SYN B] [4),6),R1]

But though non-symmetrical, logical containment is transitive: DR1-19

DR1-19. If [A CONT B] and [B CONT C], then [A CONT C]

<u>Proof</u>: Step 1) [A CONT B] [Assumption]
Step 2) [B CONT C] [Assumption]
Step 3) [A SYN (A & B)] [1),Df 'CONT',R1]
Step 4) [B SYN (B & C)] [2),Df 'CONT',R1]
Step 5) [A SYN (A & (B & C))] [3),4),R1b]
Step 6) [A SYN ((A & B) & C)] [5),Ax.1-05,R1b]
Step 7) [A SYN (A & C)] [6),3),R1]
Step 8) [A CONT C] [7),Df 'CONT'),R1]

This rule is a special case of the following broad rule which covers sorites of any length:

DR1-20. If [A_1 CONT A_2] and [A_2 CONT A_3] and…and [A_{n-1} CONT A_n],then [A_1 CONT A_n].

<u>Proof</u>: Step 1) [A_1 CONT A_2]. [Assumption 1]
Step 2) [A_2 CONT A_3] [Assumption 2]
Step 3) [A_1 SYN (A_1 & A_2) [1),Df 'CONT']
Step 4) [A_2 SYN (A_2 & A_3)] [2),Df 'CONT']
Step 5) [(A_1 & A_3) SYN A_1] [4),DR1-01]
Step 6) [A_1 SYN (A_1 & (A_2 & A_3))] [3),4),R1b]
Step 7) [A_1 SYN ((A_1 & A_2) & A_3)] [5),Ax.1-05,R1b]
Step 8) [A_1 SYN (A_1 & A_3)] [7),3),R1]
Step 1+8(1)-0) = 9) [A_1 CONT A_3] [8),Df 'CONT]
Step 1+8(2)-7) = 10) [A_3 CONT A_4] [Assumption 3]
Step 1+8(2)-6) = 11) [A_1 SYN (A_1 & A_3)] [9),Df 'CONT']
Step 1+8(2)-5) = 12) [A_3 SYN (A_3 & A_4)] [10),Df 'CONT']
Step 1+8(2)-4) = 13) [(A_3 & A_4) SYN A_3] [12),DR1-01]
Step 1+8(2)-3) = 14) [A_1 SYN (A_1 & (A_3 & A_4))] [11),13),R1]
Step 1+8(2)-2) = 15) [A_1 SYN ((A_1 & A_3) & A_4)] [14),Ax.1-05,R1]
Step 1+8(2)-1) = 16) [A_1 SYN (A_1 & A_4)] [15),11),R1]
Step 1+8(2)-0) = 17) [A_1 CONT A_4] [16),Df 'CONT']
. . . .
For any given n in 'A_n', j=(n-2), and
Step 1+8(j-1)-0) [A_1 CONT A_{n-1}] [1+8(j-1)-1),Df 'CONT]
Step 1+8j-7) [A_{n-1} CONT A_n] [Assumption n+1]
Step 1+8j-6) [A_1 SYN (A_1 & A_{n-1})] [1+8(j-1)-0),Df 'CONT']
Step 1+8j-5) [A_{n-1} SYN (A_{n-1} & A_n] [1+8j-7),Df 'CONT']
Step 1+8j-4) [(A_{n-1} & A_n) SYN A_{n-1}] [1+8j-5),DR1-01]
Step 1+8j-3) [A_1 SYN (A_1 & (A_{n-1} A_n))] [1+8j-6),1+8j-4),R1]
Step 1+8j-2) [A_1 SYN ((A_1 & A_{n-1}) A_n)] [1+8j-3),Ax.1-05,R1]
Step 1+8j-1) [A_1 SYN (A_1 & A_n)] [1+8j-2),1+8j-1),R1]
Step 1+8j-0) [A_1 CONT A_n] [1+8j-1),Df 'CONT]

The following three rules are helpful:

DR1-21. If [A CONT B] then [(C&A) CONT (C&B)]

Proof: Step 1) [A CONT B]	[Assumption]
Step 2) [A SYN (A&B)]	[1),Df 'CONT']
Step 3) [(A&B) SYN B]	[2),DR1-01]
Step 4) [= [(C&A) SYN (C&A)]	[T1-11,U-SUB]
Step 5) [(C&A) SYN (C&(A&B))]	[4),3),R1]
Step 6) [((C&A)&(C&B)) SYN (C&(A&B))]	[&-ORD]
Step 7) [(C&A) SYN ((C&A)&(C&B))]	[5),6),R1]
Step 8) [(C&A) CONT (C&B)]	[7),Df 'CONT']

DR1-22. If [A CONT B] then [(C&A) CONT (CvB)]

Proof: Step 1) [A CONT B]	[Assumption]
Step 2) [(C&A) CONT (C&B)]	[1),DR1-21]
Step 3) [(C&A) SYN ((C&A)&(C&B))]	[2),Df 'CONT']
Step 4) [(C&B) SYN (C&(B&(CvB)))]	[T1-18]
Step 5) [(C&B) SYN ((C&B)&(CvB))]	[4),Ax.1-05,R1b]
Step 6) [((C&B)&(CvB)) SYN (C&B)]	[5),DR1-01]
Step 7) [(C&A) SYN ((C&A)&((C&B)&(CvB)))]	[3),6),R1]
Step 8) [(C&A) SYN (((C&A)&(C&B))&(CvB)))]	[7),Ax.1-05),R1b]
Step 9) [(C&A) SYN ((C&A)&(CvB))]	[8),3),R1]
Step 10) [(C&A) CONT (CvB)]	[9),Df 'CONT']

DR1-23. If [A CONT C] and [B CONT C], then [(AvB) CONT C]

Proof: Step 1) [A CONT C]	[Assumption]
Step 2) [B CONT C]	[Assumption]
Step 3) [A SYN (A & C)]	[1),Df 'CONT',R1]
Step 4) [B SYN (B & C)]	[2),Df 'CONT',R1]
Step 5) [(AvB) SYN (AvB)]	[T1-11,U-SUB]
Step 6) [(AvB) SYN ((A & C)v B)]	[3),5),R1b]
Step 7) [(AvB) SYN ((A & C)v(B & C))]	[4),6),R1b]
Step 8) [(AvB) SYN ((A v B) & C)]	[T1-17,&v-DIST]
Step 9) [(AvB) CONT C]	[6),Df 'CONT']

Other useful derived rules, easily proved, are:

DR1-24. If [A CONT C] and [B CONT C], then [(A&B) CONT C]
DR1-25. If [A CONT B] and [A CONT C], then [A CONT (B & C)]
DR1-26. If [A CONT B] and [A CONT C], then [A CONT (B v C)]
DR1-27. If [A CONT C] and [B CONT D], then [(AvB) CONT (C v D)]
DR1-28. If [A CONT C] and [B CONT D], then [(A&B) CONT (C & D)]

All of rules of containment depend upon the general principles of logical synonymy and the property of a conjunction that if the conjunction is proposed or is true, then any of its conjuncts is proposed or is true. This is the basic concept of logical containment—i.e., containment based on the meaning of 'and' and logical synonymy.

All of the rules above govern movements from logical containment to logical containment, preserving the logical relationship of containment through transformations. Similar, more basic rules, hold for semantic containments (based on definitions of extra-logical terms). The basic rules for applying given logical or semantic containments differ, depending on the sentential operators being employed. (I.e., depending on the mode of discourse—whether it is about truths or necessity or probability or right and wrong or beauty.) Basic rules for applying what logic teaches about containments, include the following:

Valid: [If (A is true & (A Cont B)) then B is true]
[If (A is necessary & (A Cont B)) then B is necessary]
[If (A is tautologous & (A Cont B)) then B is tautologous]

But not valid: [If (A is false & (A Cont B)) then B is false]
[If (A is Impossible & (A Cont B)) then B is Impossible]
[If (A is inconsistent & (A Cont B)) then B is incosistent]

1.3 Equivalence Classes and Decision Procedures

1.31 SYN-Equivalence Metatheorems 8-10

An equivalence class is a class of all elements in a domain which stand in a certain equivalence relation to a given member of that domain. We have seen (SYN-metatheorem 3) that SYN is an equivalence relation. Thus for the present domain of wffs:

Df 'SYNeq'. X is an SYN-equivalence class of wffs
Syn_{df}
X is the class of all and only those wffs which are SYN with some given sentential wff.
or, ['x ∈ SYNeq' Syn_{df} '(∃y)(y∈Wff & (z)(z∈X iff (z∈Wff & (z SYN y)))']

Among the properties of SYN-equivalence classes in the sentential calculus is the following:

SYN-metatheorem 8. Every member of a SYN-equivalence class will contain occurrences of all and only the same atomic wffs which occur positively and the same atomic wffs which occur negatively as are contained in each other member of the class.

[Proof: Df 'SYNeq', SYN-metatheorem 1 and SYN-metatheorem 3]

This metatheorem has no analogue for truth-functional equivalence classes due to the principle of absorption.

The uniqueness theorem, SYN-metatheorem 9, follows from SYN-metatheorem 6 and the definition of SYN-equivalence classes, Df 'SYNeqCL':

SYN-metatheorem 9. Every SYN-equivalence Class contains one and only one MOCNF wff.[19]

Similar SYN-metatheorems for MODNF, MinOCNF and MinODNF wffs can be established. These consequences are summarized, analogously to SYN-metatheorem 7, in the Uniqueness Metatheorem,

SYN-metatheorem 10. Every SYN-equivalence class has one and only one wff from each of the classes of MOCNF, MinOCNF, MODNF, and MinODNF wffs.

SYN-metatheorem 10 has no analogue for TF-equivalence classes. There are an infinite number of MOCNFs, etc., which are TF-equivalent to any given wff, and thus an infinite number of MOCNF, MODNF, MinOCNF and MinODNF wffs in every TF-equivalence class. By the "paradoxes of strict implication", all inconsistent wffs are in one TF-equivalence class, and all tautologies (theorems) are in another. By contrast, SYN-equivalence classes separate inconsistencies and tautologies by potential differences in content—i.e., in what is being talked about or what is said about what is talked about. But also, among contingent statements, TF-equivalence classes include infinite numbers of MOCNFs, MODNFs, MinOCNFs and MinODNFs which are truth-functionally equivalent because of Absorption and thus have wide differences in meaning and content.

1.32 Characteristics of SYN-Equivalence Classes

To envisage more clearly what a SYN-equivalence class would look like, consider the following. If x is a sentential wff, its SYN-equivalence class will consist of all and only those wffs in which have the same set of elementary wffs as x, and whose instantiations are true if and only if similar instantiations for A are true. Thus,

a) The eleven smallest wffs (in number of symbols) of the infinite number in the SYN-equivalence class of the wff, P, are:

P,

(P&P), (PvP), and

(P&(P&P)), ((P&P)&P), ((P&P)vP), (P&(PvP)), ((PvP)&P), (Pv(P&P)), (Pv(PvP)), ((PvP)vP).

In the absence of negation every wff containing only occurrences of P is SYN-equivalent to P.

b) the fourteen smallest members of the SYN-equivalence class of '(((P&Q)vP)&Q)' are:

(P&Q), (Q&P),

((P&P)&Q), (P&(P&Q)), ((P&Q)&P), (P&(Q&P)), ((Q&P)&P), (Q&(P&P)), ((P&Q)&Q), and (P&(Q&Q)), ((Q&Q)&P), (Q&(Q&P)), ((Q&P)&Q), (Q&(P&Q)), ((Q&P)&Q), (Q&(P&P)).

The MOCNF wff determining this class is (P&Q&(PvQ)), which is also the MinOCNF of this class. The MODNF wff of this class is (P&Q), i.e., the maximal disjunct standing alone, which is also the MinDNF of this class.

19. The first version of this metatheorem was first presented in a paper entitled "A Unique Form for Synonyms om the Propositional Calculus", at the Assoc. for Symbolic Logic Meeting, in New York, 12/27/69. An abstract of the paper, entitled, R.B.Angell, "A Unique Normal Form for Synonyms in the Propositional Calculus", was published in the *Journal of Symbolic Logic*, v.38 (June 1973), p. 350.

TABLE 1-1
Basic Normal Forms of SYN-Classes with 3 Variables

With just three elementary wffs, P,Q,R, there will be just eighteen SYN-equivalence classes, each having all wffs SYN to just one of the 18 MOCNFs and to one of the 18 MinOCNFs (which drop intermediate conjuncts (underlined) from MOCNFs):

MOCNFs	MinOCNFs
(P&Q&R&(PvQ)&(PvR)&(QvR)&(PvQvR))	(P&Q&R&(PvQvR))
(P&Q&(PvQ)&(PvR)&(QvR)&(PvQvR))	(P&Q&(PvQvR))
(P&R&(PvQ)&(PvR)&(QvR)&(PvQvR))	(P&R&(PvQvR))
(Q&R&(PvQ)&(PvR)&(QvR)&(PvQvR))	(Q&R&(PvQvR))
(P&(PvQ)&(PvR)&(QvR)&(PvQvR))	(P&(QvR)&(PvQvR))
(Q&(PvQ)&(PvR)&(QvR)&(PvQvR))	(Q&(PvR)&(PvQvR))
(R&(PvQ)&(PvR)&(QvR)&(PvQvR))	(R&(PvQ)&(PvQvR))
(P&(PvQ)&(PvR)&(PvQvR))	(P&(PvQvR))
(Q&(PvQ)&(QvR)&(PvQvR))	(Q&(PvQvR))
(R&(PvR)&(QvR)&(PvQvR))	(R&(PvQvR))
((PvQ)&(PvR)&(QvR)&(PvQvR))	((PvQ)&(PvR)&(QvR)&(PvQvR))
((PvQ)&(PvR)&(PvQvR))	((PvQ)&(PvR)&(PvQvR))
((PvQ)&(QvR)&(PvQvR))	((PvQ)&(QvR)&(PvQvR))
((PvR)&(QvR)&(PvQvR))	((PvR)&(QvR)&(PvQvR))
((PvQ)&(PvQvR))	((PvQ)&(PvQvR))
((PvR)&(PvQvR))	((PvR)&(PvQvR))
((QvR)&(PvQvR))	((QvR)&(PvQvR))
(PvQvR)	(PvQvR)

Each of the same eighteen SYN-equivalence classes, listed in the same order, have one of the 18 MODNFs and, dropping the intermediate (underlined) disjuncts, the 18 MinODNFs below:

MODNFs	MinODNFs
(P&Q&R)	(P&Q&R)
((P&Q)v(P&Q&R))	((P&Q)v(P&Q&R))
((P&R)v(P&Q&R))	((P&R)v(P&Q&R))
((Q&R)v(P&Q&R))	((Q&R)v(P&Q&R))
((P&Q)v(P&R)v(P&Q&R))	((P&Q)v(P&R)v(P&Q&R))
((P&Q)v(Q&R)v(P&Q&R))	((P&Q)v(Q&R)v(P&Q&R))
((P&R)v(Q&R)v(P&Q&R))	((P&R)v(Q&R)v(P&Q&R))
(Pv(P&Q)v(P&R)v(P&Q&R))	(Pv(P&Q&R))
(Qv(P&Q)v(Q&R)v(P&Q&R))	(Qv(P&Q&R))
(Rv(P&R)v(Q&R)v(P&Q&R))	(Rv(P&Q&R))
((P&Q)v(P&R)v(Q&R)v(P&Q&R))	((P&Q)v(P&R)v(Q&R)v(P&Q&R))
(Pv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(Pv(Q&R)v(P&Q&R))
(Qv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(Qv(P&R)v(P&Q&R))
(Rv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(Rv(P&Q)v(P&Q&R))
(PvQv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(PvQv(P&Q&R))
(PvRv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(PvRv(P&Q&R))
(QvRv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(QvRv(P&Q&R))
(PvQvRv(P&Q)v(P&R)v(Q&R)v(P&Q&R))	(PvQvRv(P&Q&R))

With two sentence letters and no negation, there are just four equivalence classes. For example, for P and Q there is the class above, SYN-equivalent to the MOCNF, (P&Q&(PvQ)), and the classes of wffs SYN-equivalent to the wffs, (P&(PvQ)), (Q&(PvQ) and (PvQ)—each has a different truth-function than its MinODNF wff.

With three sentence letters there will be just eighteen SYN-equivalence classes, each consisting of wffs SYN to just one of 18 MOCNFs, just one of 18 MinOCNFs, just one of 18 MODNFs and just one of 18 three-lettered MinODNFs. (see TABLE 1-1). Each of these classes has an infinite number of members due to idempotences and re-orderings, but just one member from each of the four classes of MOCNF, MODNF MinOCNF and MinODNF wffs.

c) The 24 smallest wffs in the SYN-equivalence class of '(Pv(P&(QvR))v((QvP)&(P&R)))' which has the MOCNF '(P&(PvQ)&(PvR)&(PvQvR))', and the MinOCNF '(P & (PvQvR))' are:

{(P&(Pv(QvR))),	(P&((PvQ)vR)),	(P&(Pv(RvQ))),	(P&((PvR)vQ)),
(P&(Qv(PvR))),	(P&((QvP)vR)),	(P&(Qv(RvP))),	(P&((QvR)vP)),
(P&(Rv(QvP))),	(P&((RvQ)vP)),	(P&(Rv(PvQ))),	(P&((RvP)vQ)),
((Pv(QvR))&P),	(((PvQ)vR)&P),	((Pv(RvQ))&P),	(((PvR)vQ)&P),
((Qv(PvR))&P),	(((QvP)vR)&P),	((Qv(RvP))&P),	(((QvR)vP)&P),
((Rv(QvP))&P),	(((RvQ)vP)&P),	((Rv(PvQ))&P),	(((RvP)vQ)&P)}

These are the minimal conjunctive normal forms (MinCNFs) of which the first is the unique MinOCNF wff of this class. The four unique basic normal form wffs in this class (listed fifth in TABLE I) are:

MOCNF = (P&(PvQ)&(PvR)&(PvQvR)); MinOCNF = (P&(PvQvR));
MODNF = (Pv(P&Q)v(P&R)v(Q&R)v(P&Q&R)); MinODNF = (Pv(Q&R)v(P&Q&R)).

Our objective has been to prove that each of the four Basic Normal Form classes have the following property: each member of each Basic Normal Form class determines a SYN-equivalence class which is disjoint from all SYN-equivalence classes determined by other members of that class. In other words, the members of each basic normal form class partition all wffs into sets of classes of wffs which are SYN with one and only one basic normal form wff of that kind. It follows that since every wff is reducible to a unique normal form, one can decide whether two negation-free wffs are SYN or not by seeing if the same unique normal form is deducible from each. From this completeness and soundness proofs for the axiomatic system of SYN with respect to pairs of wffs in this chapter are forthcoming.

1.33 Syntactical Decidability and Completeness

1.331 A Syntactical Decision Procedure for SYN

Given any two negation-free sentential wffs, A and B, there are several effective decision procedures for determining whether they are SYN-equivalent or not. The one we are about to establish is purely syntactical.

The syntactical decision procedure is simple: First, reduce both wffs to one of the four kinds of Basic Normal Forms (e.g., to an MOCNF-wff as laid down in SYN-metatheorem 4). According to SYN-metatheorem 10 each will reduce to one and only one such wff. Then see if the two Basic Normal Forms reached are identical. The statement [A SYN B] will be a theorem, by SYN-Metatheorem 6, if and only if they both reduce to the same MOCNF, or (by SYN-metatheorem 7) to the same MODNF or MinOCNF or MinODNF wff. Thus, SYN-metatheorem 11. Given any two wffs whether they are SYN or not is effectively decidable by determining whether they both are SYN to the same basic normal form(s).

From this decision procedure it follows that the axiomatic system of SYN-equivalence presented above is complete with respect to all pairs of wffs in logic which are TF-equivalent and can not have instantiations which refer to different entities or say different things about the entities they refer to—i.e., it is complete and sound with respect to referential synonymy.

Another method is called the Method of Analytic Truth-tables (a proof of this has been presented elsewhere).[20] For negation-free wffs, this method is similar to a standard truth-table test for TF-equivalence, except that if **n** is the number of elementary wffs in any wff, its truth-table can not have more than 2^n rows; A and B are SYN-equivalent by this test if and only if they have the same final columns in their truth-tables.The constraint preserves the SYN-properties mentioned in SYN-Metatheorem 1 as well as other SYN-properties. This method will be developed in Chapter 4 as a semantical decision procedure.

1.332 Decisions on the Number of SYN-eq-Classes

Each SYN-equivalence class has an infinite number of members due to Idempotence (using Ax.1-01 or Ax.1-02 with R1 or R1b on any component of a SYN-equivalence class) Distribution and other SYN axioms. But it is always decidable how many SYN-equivalence classes are determined by any given finite set of elementary wffs.

SYN-metatheorem 12. Given any set of n distinct elementary wffs there is a finite (theoretically) decidable number of A-equivalence classes uniquely determined by that set.

For example, for a single elementary wff, there is just one SYN-equivalence class; for any two elementary wffs there will be just four SYN-equivalence classes; and for any set of three elementary wffs, there are just eighteen SYN-equivalence classes determined by this set; the 18 classes for the set {P,Q,R} are represented by the following 18 distinct non-SYN MinOCNFs:

1) P&(PvQvR)
2) Q&(PvQvR)
3) R&(PvQvR)
4) P&Q&(PvQvR)
5) P&R&(PvQvR)
6) Q&R&(PvQvR)
7) P&Q&R&(PvQvR)
8) (PvQ)&(PvQvR)
9) (PvR)&(PvQvR)
10) (QvR)&(PvQvR)
11) (PvQ)&(PvR)&(PvQvR)
12) (PvQ)&(QvR)&(PvQvR)
13) (PvR)&(QvR)&(PvQvR)
14) (PvQ)&(PvR)&(QvR)&(PvQvR)
15) P&(QvR)&(PvQvR)
16) Q&(PvR)&(PvQvR)
17) R&(PvQ)&(PvQvR)
18) (PvQvR)

---

20. See Abstract in JSL, Vol. 46, No 3, Sept 1981, p 677, of "Analytic Truth-tables", by R. B. Angell, presented at the Annual Meeting of the Association for Symbolic Logic, Washington DC, 1977. A more detailed proof is in an unpublished paper. The heart of the proof is that for each row which comes out T in the analytic truth-table of a wff, A, one of the disjuncts of the **MODNF of** A is constructible by conjoining the elementary wffs assigned T in that row, and the set of such disjuncts constitutes all and only the disjuncts belonging to the unique **MODNF of** A.

If negation is introduced (see Chapters 4), negated and unnegated occurrences of the same atomic wff in normal form wffs must be assigned T's and F's as if they represented different sentences since negated and unnegated occurrences of a sentence will obviously say different things about their subjects.

No two of these 18 MinOCNFs are TF-equivalent,[21] although all 18 have occurrences of the same set of elementary wffs.

The property described in SYN-metatheorem 12 does not hold of TF-equivalence classes; for any given set of elementary wffs, there will be an infinity of TF-equivalence classes, due to the Principle of Absorption.

I believe no single formula is known that defines the number of SYN-equivalence classes as a function of an arbitrary number, n, of elementary wffs; but clearly that number is theoretically decidable for any finite number, n. The proof of SYN-metatheorem 12 is as follows:

Proof of SYN-metatheorem 12:

1) Given a number **n**, we need only consider the classes of MOCNFs which can be constructed from a set of **n** distinct elementary wffs.

2) Given any **n** elementary wffs, there are 2^n -1 possible wffs of just one those n wffs or of several connected by 'v'—i.e., the number of non-empty sub-sets of the set of **n** elementary wffs. The number of possible distinct wffs consisting of one of *those* wffs, or some n-tuple of them conjoined by '&', will be: $2^{(2^n -1)}$ -1 This, then, is the finite upper bound on the number of conjunctions of disjunctions of non-synonymous wffs (among which are all of the MOCNFS) constructible from the **n** elementary wffs. [Note that 2^{2^n} is the number of distinct binary truth-functions for **n** distinct sentential variables].

3) In theory, then, a decision procedure is possible for every finite **n**: 1) one lists all possible distinct conjunctions of disjunctions containing one or more of the **n** elementary wffs, ordered and grouped as prescribed by Df 'MCNF' and Df 'MOCNF'; 2) one eliminates by inspection those wffs on the list which do not satisfy the definition of MOCNFs in Df 'MOCNF'; and 3) the number f(**n**) sought is the number of MOCNFs left on the list. This is the number of distinct truth-functions for the group of f(n) wffs remaining.

In practice, this decision procedure quickly gets out of hand (just as testing by truth-tables does). The following table makes this clear:

n	$2^n - 1$	$2^{(2^n -1)} - 1$	Number of SYN-equivalence Classes
1	1	1	1
2	3	7	4
3	7	127	18
4	15	32,767	166
5	31	2,147,483,647	?

Nevertheless, SYN-metatheorem 13 shows that theoretically the number of SYN-equivalence classes, MOCNFs, MODNFs, MinOCNFs and MinODNFs built with any finite set of n letters is decidable.

21. In standard sentential logic there are 2^{2^n} truth-functions for wffs with n sentential variables; e.g., for 3 variables, 2^8, or 256, distinct truth-functions. Why, then, are there only 18 analytic truth-functions for these three letters?

The negated sentence which instantiates (in any normal form) a negated atomic wff will always say something different about certain entities than the same sentence, unnegated, which instantiates the unnegated atomic wff. Therefore, for SYN, 'P' and '∼P' are assigned truth-values as if they were different letters, to preserve referential synonymy. In general, the semantics of SYN-equivalence in full sentential logic works just like the semantics of a negation-free sentential logic with just 'or' and 'and' as primitive logical constants. If 'R' is replaced by '∼P' throughout the list above, the following pairs are TF-equivalent but still not SYN-equivalent: 2) and 16); 4) and 15); 5) and 7); 6) and 17); 8) and 11); 9) and 18); 10) and 13); 12) and 14).

Some proponents of mathematical logic (C.I. Lewis, Carnap, and Goodman among others) tried to define 'synonymity' within the framewhork of mathematical logic, but the results were not satisfactory. Quine abandoned theories of meaning as the semantic basis for logic in 1950 and 1951. He committed himself to using a "theory of reference" the main concepts of which are "naming, truth, denotation (or truth-of) and extension", while eschewing the use in logic of any "theory of meaning" which deals with concepts such as "synonymy (or samenss of meaning), significance (or possession of meaning), analyticity (or truth by virtue of meaning) ...[and] entailment, or the analyticity of the conditional."[22] In A-logic the concepts of meaning, synonymy and entailment are central. This is justified because these concepts can be correlated in a plausible way with the purely syntactical relation, SYN, which has not previously been recognized, thought it is applicable to the language of mathematical logic. It can be correlated with the concept of sameness of meaning just as rigorously as the concept of 'logical equivalence' in propositional logic is correlated with pairs of truth-functions which have the same final columns in their truth-tables.

It is possible that opponents of talk about meanings could give a different interpretation to the syntactical relation we associate here with 'synonymy', but I do not know what that would be. In any case the connection with synonymy is in line with ordinary discussions as to samenesses and differences in the meanings of expressions, as well as a long tradition in logic. As presented here, it poses no threat to the very real accomplishments of mathematical logic, though it may conflict with some philosophical preferences. We do not oppose a theory of reference to a theory of meaning. The reference of the subject terms of a sentence is one aspect of its meaning. The intension or idea expressed by the predicate is another aspect of its meaning. The truth or falsity of the sentence has to do with the correspondence of the predicate's meaning with facts about the properties or relations of the referent(s) of the subject term(s).

---

22. These characterizations of theories of meaning and reference are in Willard Van Orman Quine, *From a Logical Point of View*, Harper, (2nd Ed.) 1961, essay VII, "Notes on the Theory of Reference", p 130.)

Chapter 2

Predicates

2.1 Introduction: Over-view and Rationale of This Chapter

Mathematical logic takes propositions, or sentences that are either true or false (exclusively), as its fundamental objects of study. Analytic logic takes predicates as its fundamental objects of study.

Mathematical logic is customarily divided into 1) Propositional (or Sentential) Logic and 2) 1st Order Predicate Logic (or Quantification theory). The latter employs two distinct concepts: **predicates** and **quantifiers**. Predicate letters 'P', 'Q', 'R', are used as schematic representives of actual predicates. Quantifiers, '(∀x)' and '(∃x)' are used to represent the operators, 'For all x' and 'For some x'. Predicate schemata do not require quantifiers, but quantification requires predicates. This chapter, still confined to negation-free wffs, developes a logic of predicates without quantifiers. The next chapter treats predicates with quantifiers. Chapter 4 treats predicates with quantifiers and negation.

The basic difference between this and the last chapter is one of interpretation. If logic is about properties and relationships of meanings, it is better to take predicates as its elements than propositions. The meaning of an indicative sentence is independent of its truth or falsehood. To understand what the sentence means is one thing; to know whether it is true or false is another. One can understand very well what a sentence means, without having the slightest idea whether it true or false. In this and following chapters it is assumed that the properties and relationships relevant to logic reside in the meanings of predicates, whether or not embedded in sentences. The letters 'P_1', 'P_2', 'P', 'Q', 'R', etc. hereafter are construed as placeholders for predicates, rather than exclusively for propositions or sentences.

The focus is on the logical structure of predicates, whether they occur in sentences or stand by themselves. Simple "atomic" predicates vary in several ways; chiefly, in being monadic, dyadic, or polyadic, etc., but also in being particularized vs abstract, or being quantified or not. In this chapter, compound predicates are formed from simple predicates with conjunction or disjunction. One predicate may or may not be synonymous with another by virtue of its logical form, and the meaning of one may or may not be logically contained in the meaning of another. Logic determines whether these relations hold or not. In general, all logical properties and relationship are attributable to predicates. They are attributable to sentences only because of the logical structure of their predicates.

The underlying logic of synonymy and containment for conjunction and disjunction in Chapter 1 is not altered. It simply applies to a greater variety of schemata. Axioms, theorems, metatheorems, and proofs in Chapter 1 remain as written, but are re-interpreted as statements about logical containment or synonymy of pairs of predicate structures, (whether embedded in sentences or not) rather than sentences only. However, Rules of Inference must be adjusted or added.

Thus the first rule of inference, SynSUB, remains unchanged though re-interpreted as a rule for substituting synonymous predicates for on another.

But the second rule, U-SUB, must be augmented to take into account new classes of schemata formed with two new kinds of primitive symbols associated with the predicate letters—“argument position holders” and individual constants. The simple form of U-SUB in Chapter 1 remains intact, but is supplemented by provisions for substituting more complex kinds of predicate schemata for simpler components in predicate structures. The major task of this chapter, is to provide rigorous and correct rules such that the uniform substitution of a complex predicate schema at all occurrences of a predicate letter in a theorem, will preserve theoremhood. Technical distinctions introduced below, especially in Sections 2.34 to 2.37, lead to such a procedure.

A new Rule of Inference, Instantiation (abbr. INST) is added, It is the simple rule that if predicates have logical properties or stand in logical relations with one another, then expressions which apply those predicates to individual entities also have those logical properties and stand in those same logical relations. For example, since ‘((—is P) SYN (—is P & —is P)))’ is a theorem (Ax.101), we may infer that ‘(Pa SYN (Pa & Pa))’ is theorem, where ‘a’ may be any individual constant.

In this chapter, as in Chapter 1, only relationships of SYN and CONT between negation-free predicates are considered. Logical properties or relationships which depend on negation, or on quantifiers, are not. Therefore, since predicates are neither true nor false, we no longer appeal to truth-tables to determine logical properties and relationships in formal analytic logic. Reductions to basic normal forms provide all that is needed to test for SYN and CONT.

By construing logic as a logic of predicates rather than propositions, the reach of analytic logic is increased. The logical properties and relationships of predicates are found not only in indicative sentences, but in questions and commands and directives; not only in truth-assertions but in value judgments. A way is opened to extend formal logic into these other modes of discourse including questions and directives without being restricted to the logic of expressions which are true or false.

Purely formal logic—the relations and properties which are determined by the meanings of ‘and’, ‘or’, ‘not’ ‘if-then’ and ‘all’—is not changed essentially by shifting from propositions to predicates. But logic in the broader sense—applied logic where valid inference is based on the definitions and meanings of extra-logical terms—is greatly advanced by attention to predicates rather than sentences and truth-functions. This will be made clearer below.

2.2 The Formal Base for a Logic of Unquantified Predicates

The formal base of the logic of predicates in this chapter, which includes added primitive symbols, added rules of formation, and the augmented rule of uniform substitution, is given below:

<u>I. Primitive symbols:</u>

(i) Grouping devices: ( ) **< >** — <u>**Added**</u>
(ii) Connectives: & v
(iii) {PL}—Predicate Letters: **(rather than ‘proposition letters’)**
$P_1, P_2, P_3, \ldots, P_i, \ldots$
(iv) {APH}—Argument Position Holders: 1,2,3,…,i,… — <u>**Added**</u>
(v) {IC}—Individual Constants: $\mathbf{a_1, a_2, a_3, \ldots, a_i, \ldots}$ — <u>**Added**</u>

II. Rules of Formation:

FR1. If P_i e {PL}, P_i is a wff

FR2. If P_i and P_j are wffs, then (P_i & P_j) and (P_i v P_j) are wffs.

FR3. If P_i e {PL} and $\{t_1,t_2,\ldots,t_n\} \in \{\{APH\}\cup\{IC\}\}$ then $P_i<t_1,t_2,\ldots,t_n>$ is a wff **Added**

Definitions and Axioms are unchanged:

III. Definitions:

D1-1a. (P&Q&R) SYN_{df} (P&(Q&R))

D1-1b. (PvQvR) SYN_{df} (Pv(QvR))

IV. Axioms

Ax.1-01. [P SYN (P&P)]	[&-IDEM]
Ax.1-02. [P SYN (PvP)]	[v-IDEM]
Ax.1-03. [(P&Q) SYN (Q&P)]	[&-COMM]
Ax.1-04. [(PvQ) SYN (QvP)]	[v-COMM]
Ax.1-05. [(P&(Q&R)) SYN ((P&Q)&R)]	[&-ASSOC]
Ax.1-06. [(Pv(QvR)) SYN ((PvQ)vR)]	[v-ASSOC]
Ax.1-07. [(Pv(Q&R)) SYN ((PvQ)&(PvR))]	[v&-DIST]
Ax.1-08. [(P&(QvR)) SYN ((P&Q)v(P&R))]	[&v-DIST]

The rule of Substitution of Synonyms, SynSUB, remains the same. But the Rule of Uniform Substitution, U-SUB, is augmented and supplemented by INST, a Rule of Instantiation:

V. Transformation Rules

R1.From $\models$ [P SYN Q] and $\models$ [R SYN S], infer $\models$ [P SYN Q(S//R)] SynSUB

R2-2 If $\models$ R and (i) $P_i<t_1,\ldots,t_n>$ occurs in R,
and (ii) Q is an h-adic wff, where h $\geq$ n, **Augmented, Revised**
and (iii) Q has occurrences of all numerals 1 to n,
then $\models$ [R($P_i<t_1,\ldots,t_n>$/Q) may be inferred U-SUB

R2-3. If $\models$ [P<1>] then $\models$ [Pa] may be inferred **Added** INST

For the purposes of this chapter, R2-2 says in effect, "If [R SYN S] is a theorem of logic, and P_i occurs in '[R SYN S]' then the result of introducing a suitable predicate, Q, at all occurrences of P_i in '[R SYN S]' is a theorem of logic". The problem is to define "suitable" and "introducing" so that the consequent will hold whenever the antecedent does. The rule must preserve the logical properties or relations upon substitution of finer-structured predicate schemata for predicate letters or atomic predicate schemata in a wff R, whether R is a SYN-theorem or a CONT-theorem, or some other kind of logical theorem.

In accordance with the rule of formation, FR3, predicates may henceforth be symbolized by wffs of the form '$Pi<t_1,t_2,\ldots>$' where the terms t_1, t_2, etc., are either argument position-holders represented by numerals {1, 2, 3,…etc.} or placeholders for individual constants $\{a_1,a_2,\ldots$etc.} If all singular terms are filled by individual constants the predicate is a "saturated predicate", i.e., a sentence.

The following two examples illustrate the procedure of making substitutions in accord with R2-2 and R2-3:

Example #1: Illustrates how R2-2, U-SUB, preserves theoremhood upon substitution of a finer-structured predicate schema at all occurrences of a predicate letter in a SYN-theorem.

Let [R SYN S] be 1) $\models$ [$(P_1$ & $P_2)$ Syn $(P_2$ & $P_1)$] [Ax.1-03],
then R is [$(P_1$ & $P_2)$ SYN $(P_2$ & $P_1)$]
and let P_1, which occurs in R, be the relevant '$P_i\langle t_1,\ldots,t_n\rangle$' (i.e., i=1 and n=0).

Thus Q can be '$P_3\langle 1,2\rangle$', since '$P_3\langle 1,2\rangle$' satisfies clauses (i), (ii) and (iii) of R2-2, making R$(P_1/P_3\langle 1,2\rangle)$ become ((P_1 & P_2) SYN (P_2 & P_1)) ($P_1/P_3\langle 1,2\rangle$))
(R) ($P_1/P_3\langle 1,2\rangle$))
and the result of applying U-SUB to Ax.3 is
2) $\models$ [($P_3\langle 1,2\rangle$ & P_2) Syn (P_2 & $P_3\langle 1,2\rangle$)] [1),U-SUB]
In this case U-SUB involves the simple replacement of 'P_1' by '$P_3\langle 1,2\rangle$' throughout R.

R2-3, Instantiation, may now be applied to 2). It says in effect that any theorem which has argument position holders in it, remains a theorem if all occurrences of a given argument position holder are replaced by any individual constant. Thus appying R2-3 to 2) we can get,

$\models$ [($P_3\langle a_1,2\rangle$ & P_2) Syn (P_2 & $P_3\langle a_1,2\rangle$)] [2),INST]
$\models$ [($P_3\langle a_3,a_1\rangle$ & P_2) Syn (P_2 & $P_3\langle a_3,a_1\rangle$)] [2),INST(twice)]
$\models$ [($P_3\langle a_1,a_1\rangle$ & P_2) Syn (P_2 & $P_3\langle a_1,a_1\rangle$)] [2),INST(twice)]
etc…

Example #2: The next example is more complex. It illustrates how the augmented U-SUB also preserves theoremhood by introducing at all occurrences of atomic predicate schemata with the same predicate letter and argument positions another predicate schema which has as many or more, open argument positions.

By R2-2 (U-SUB), '$P_3\langle 2,a,1\rangle$ 'can be introduced for the two symbols '$P_1\langle 1,7\rangle$' and '$P_1\langle 7,1\rangle$' in,

3) [($P_1\langle 1,7\rangle$ & $P_1\langle 7,1\rangle$) SYN ($P_1\langle 7,1\rangle$ & $P_1\langle 1,7\rangle$)] [Ax.3,U-SUB]

For '$P_1\langle 1,7\rangle$' and '$P_1\langle 7,1\rangle$' are different predicate schemata of the same predicate—a predicate represented by the predicate letter 'P_1' followed by an ordered pair of subject terms. The result gotten by the augmented U-SUB is another theorem, namely,

4) $\models$ [($P_3\langle 7,a,1\rangle$ & $P_3\langle 1,a,7\rangle$) SYN ($P_3\langle 1,a,7\rangle$ & $P_3\langle 7,a,1\rangle$)]

The rules for making this substitution are more complicated than in example #1. They will be spelled out in Section 2.42 after drawing the necessary distinctions in Section 2.3, and they will be supplemented further to allow indivdual variables of quantified wffs in Section 3.223.

The application of Rule R2-3 to 4) can yield a variety of instantiated theorems by replacing argument-position-holders represented by numerals with individual constants, including:

5) [($P_3\langle 7,a,b\rangle$ & $P_3\langle b,a,7\rangle$) SYN ($P_3\langle b,a,7\rangle$ & $P_3\langle 7,a,b\rangle$)] [4),INST]
6) [($P_3\langle c,a,b\rangle$ & $P_3\langle b,a,c\rangle$) SYN ($P_3\langle b,a,c\rangle$ & $P_3\langle c,a,b\rangle$)] [5),INST]
or 5') [($P_3\langle c,a,1\rangle$ & $P_3\langle 1,a,c\rangle$) SYN ($P_3\langle 1,a,c\rangle$ & $P_3\langle c,a,1\rangle$)] [4),INST]
6') [($P_3\langle c,a,b\rangle$ & $P_3\langle b,a,c\rangle$) SYN ($P_3\langle b,a,c\rangle$ & $P_3\langle c,a,b\rangle$)] [5'),INST]

or 5'') $[(P_3<a,a,1> \,\&\, P_3<1,a,a>)$ SYN $(P_3<1,a,a> \,\&\, P_3<a,a,1>)]$ [4),INST]
6'') $[(P_3<a,a,a> \,\&\, P_3<a,a,a>)$ SYN $(P_3<a,a,a> \,\&\, P_3<a,a,a>)]$ [5''),INST]

The application of rule R1 (SynSUB) to SYN-schemata containing predicate schemata, is the same as before. Two wffs are SYN only if they have exactly the same set of elementary wffs. In the present system an "elementary schema" is any schema $[P_i<t_1,\ldots,t_n>]$ with $i \geq 1$ and $n \geq 0$. Rule R1, SynSUB, is the only rule which can change the over-all logical structure (i.e., the linear order, groupings and repetitions of wffs) while preserving theoremhood. U-SUB leaves the over-all structure intact, but can replace elementary components with more complex ones. INST retains the logical form of the initial statement, but allows its repetition with different subject terms.

The following sections work out distinctions between kinds and modes of predicates, and detail the nature of Rule 2-2. Rule 2-3, Instantiation, is the basic principle upon which Quantification theory is developed in chapter 3.[1]

2.3 Schematization: Ordinary Language to Predicate Schemata

For practical applications of formal logic we begin with sentences of ordinary language, abstracting the logically relevant features by displaying them in logical schemata. We then analyze or process the features

1. The rules for U-SUB developed in this chapter and Chapter 3, are due in very large degree to Quine's treatment of the substitution of complex predicate schemata for all occurrences of an atomic predicate or predicate schema in *Methods of Logic,* 2nd Ed (1950), Section 25, 3rd edition (1970) Sections 26 and 27, and to a lesser extent in the 4th Ed., (1982), Sections 26 and 28, where he abandons predicates in favor abstracts, and in. his *Elementary Logic* (1965). He used circled numerals '①', '②',... where we use numerals in angle brackets. We use the term 'predicate' essentially as Quine used it in this period. We will also speak of "introducing a predicate"; e.g., introducing the predicate schema 'Q<2,a,1>' for 'R<1,2>' in the theorem [(∃x)(∀y)Rxy CONT (∃x)Rxx)] to get the theorem [(∃x)((∀y)(Qyax CONT (∃x)Qxax)]. Quine used the term 'introduction' in the 1950 and 1965 editions mentioned, but abandoned it in favor of other terms in the 1970 edition. Although he abandoned the term 'predicate' in favor of class abstracts in the 1982 edition, the process of transforming an intitial formula into the result of the substitution in it was essentially unchanged.

Quine struggled with the notion of predicates throughout his career. In the earlier 1940 edition of the Elementary Logic he used the term 'stencils' instead of 'predicates'. He contrasted his use of 'predicate' with the medieval use (e.g., 'P' is the predicate in the categorical proposition 'All S is P'.) In the several decades in which he used the term 'predicates' he referred to them in a slightly derogatory way: as 'devices for substitution in quantification Theory' (p 132, 2nd ed), as 'an intermediate diagram, or so to speak a template or stencil.' (Ibid.), 'it will be convenient to introduce the auxiliary device of predicates" (p. 99 EL 1965), 'these strange expressions called predicates' (145, 3rd ed), 'Predicate schemata are doubly artificial expressions conceived in the image of sentence schemata but containing circled numerals", (p. 150, 3rd Ed) In replacing predicates with "class abstracts" in the 4th Ed of M of L, he wrote, "It is a pleasure now to invoke the versatile abstracts and banish those intrusions [i.e., predicates]" (p. 181, 4th Ed.)

Predicates are no more artifical than sentences, and certainly are less artificial than class abstracts. Though they are neither true nor false, it is perfectly natural to ask what a predicate (e.g., "is a neutrino") means, and to such questions there are often ready answers about ordinary meanings or about technical meanings in special disciplines. And no law prevents idiosyncratic definitions to serve special purposes. Quine's earlier treatment of predicates was easier to grasp and apply, and no less rigorous, than his final use of class abstracts with its dependence on set theory. We have followed the earlier approach with minor changes, adding supplementary analyses.

so displayed using definitions or rules of formal logic, and finally return to ordinary language by instantiating the schematic results of our logical operations.

If logic deals essentially with properties and relations of predicates, then it is predicates that must be schematized, predicate schemata to which the logical definitions and rules will apply, and predicate schemata which must be instantiated prior to instantiation of subject-terms.

To eliminate content altogether and focus upon the purely logical structure of a predicate requires two steps: first abstraction of predicates from sentences; second, abstraction of the purely logical structure from the content of simple or compound predicates. The result of the second step will be a predicate schema—a predicate letter affixed to ordered n-tuples of subject-term placeholders as in $P_2<1>$, $P_2<3>$ $P_2<a_1>$, $P_3<1,2>$, $P_3<3,3>$, $P_3<1,a_2>$, $P_3<a_2,a_1>$,...etc. Numerals in angle-brackets are called argument-position-holders. Subscripted lower-case letters in angle-brackets represent individual constants. But we can also use predicate letters by themselves (‘P_1’, ‘P_2’, etc.) to stand for predicates of unspecified structure.

Ordinary sentences are composed of two sorts of components: an ordered set of subject terms, and predicate terms. The subject terms are used to refer to what one is talking about. The predicate terms express criteria which are said to apply to the objects referred to by the subject terms. Traditional grammarians have held that each simple sentence has only one subject and one predicate. The great power of modern logic comes from the recognition that a simple sentence with one predicate may have more than one subject term. This analysis is indispensable in the extremely successful use of polyadic predicates in mathematical and computer logic and for the logic of relations generally.

2.31 Simple Sentences, Predicates and Schemata

It is a simple matter to abstract the predicate completely from a simple or compound sentence. One merely replaces each distinct subject term with a distinct symbol which acts as an “argument-place-holder”. For example, the predicate which is abstracted from the sentence,

1) Steve a child of Jean,

is simply ‘…is a child of —’. However, ‘…’ and ‘—’ are not the most convenient symbols for argument place-holders since one soon runs out of distinct symbols for distinct subject terms. Consequently, we will use the numerals of positive integers enclosed in angle-brackets, as **argument position holders.**[2] Thus the predicate of sentence 1) is written,

---

2. Syntactically, I follow Quine, who used circled numerals as argument position holders in predicates in *Elementary Logic* (1965), p 99, and in early editions of *Methods of Logic*. *E*.g., he would write ‘P①②’ instead of ‘$P<1,2>$’. Quine described what he was doing as “putting circled numerals for free variables”. Since expressions with free variables are not well-formed in A-logic, argument position holders are not described this way. While indebted to Quine for syntax, I do not share his aversion to predicates. Numerals as argument position holders appear to me no more artificial than any other logical symbol, and are certainly, as Quine said, useful devices. Predicates clearly have meanings and the positioning, repetitions and dissimilarities of numerals in a predicate schema also mean something, namely that the position occupied is one where a subject term can be put and cross-referenced (or not) with subject terms elsewhere in a complex predicate.These numerals are not necessarily surrogates for free variables; in this chapter they represent positions that can be occupied by individual constants. In Chapter 3 R2-2 is adjusted to allow individual variables to replace argument position holders provided certain conditions needed to preserve logical theoremhood are met.

2) <1>is a child of<2>.

The numerals in angle-brackets indicate that a denoting subject term may be put in each position they occupy. The referential content conveyed by subject terms like ‘Steve’ and ‘Jean’ is eliminated. What remains—what has been abstracted—is the predicate that was embedded in sentence 1).

A predicate is not the same as a predicate schema. The predicate 2) is an **instance of** the predicate schema ‘P_1<1,2>’. Actual predicates, like 2), have words in them which have semantic “content”; a substantive, criterial meaning. A predicate schema is much more abstract. It eliminates all substantive content (but not all meaning) and displays only the logically relevant structure or form of predicates. The logically relevant structure of the simple predicate ‘<1>is a child of<2>’ is displayed in the predicate schema ‘P_1<1,2>’, where ‘<1,2>’ represents an ordered couple of subject terms. This feature of structure is common to all dyadic predicates.

2.32 Compound Predicates

If a sentence is compound there is a single over-all compound predicate for the whole sentence, consisting of simple predicates embedded in a logical structure. In our present negation-free, quantifier-free logic the logical structure of a compound predicate is determined by linear order, grouping, patterns of repetition and difference of numerals and the placement of connectives for ‘and’ and ‘or’ only. The compound predicate in the compound sentence,

3) Steve is a child of Jean and Dick is a brother of Jean.

is:

4) (<1>is a child of<2> & <3> is a brother of<2>)

A compound sentence has a compound predicate which, as a whole, is not simply a conjunction or disjunction of component predicates viewed separately. If we view the second conjunct of the sentence 3), namely the sentence,

5) Dick is a brother of Jean

as a sentence unrelated to any other sentence, then we could express its predicate in any of several ways:

6) <1>is a brother of<2> [an instance of ‘P_1<1,2>’ or ‘P_2<1,2>’, etc.]
or 6’) <2>is a brother of<3> [an instance of ‘P_1<2,3>’ or ‘P_5<2,3>’, etc.]
or 6’’) <3>is a brother of<4> [an instance of ‘P_2<3,4>’ or ‘P_5<3,4>’, etc.]

But to abstract the predicate structure of compound sentence 3) correctly the numerals must mirror the repetitions and differences in subject-term occurrences, permitting only 4) or some variant of it with the same pattern of repetition and difference.

To see that different predicate structures express different abstract meanings, consider the different inferences common sense would draw from 4) and the results of using 6), 7), or 8) below to represent the second conjunct of 3). The only changes in the four compound predicates which result are changes in the numerals in the 3rd and 4th argument positions—i.e. differences in predicate structure. Presupposing familiar meanings associated with ‘child’, ‘brother’, ‘uncle’ etc.,

4) (<1>is a child of<2> & <3>is a brother of<2>)
Hence, <3>is an uncle of<1>.

6) (<1>is a child of<2> & <1>is a brother of<2>)
Hence, <2>is related by incest to <1>.
7) (<1>is a child of<2> & <2>is a brother of<3>)
Hence, <2) is male and <3>is an uncle or aunt of<1>.
8) (<1>is a child of<2> & <3>is a brother of<4>)
does not suggest any relation between <1>,<2>,<3> and <4>).

These four different predicates are instances respectively of the following predicate schemata (listed downwards in the same order):

(P_1<1,2> & P_2<3,2>)	(or, using	(P_7 <5,3> & P_3<6,3>)
(P_1<1,2> & P_2<1,2>)	other numerals	(P_7 <5,3> & P_3<5,3>)
(P_1<1,2> & P_2<2,3>)	but the same	(P_7 <5,3> & P_3<3,6>)
(P_1<1,2> & P_2<3,4>)	pattern)	(P_7 <5,3> & P_3<6,2>)

The second schema, formed by simply conjoining the two predicates independently schematized in 2) and 6), does not correctly describe the predicate of 3). It has the same numeral in the 1st and 3rd positions when 3) does not have the same term in 1st and 3rd positions. The third schema would misdescribe the predicate in 3) by having the same numeral in two positions that don't have the same subject and by having different numerals in positions that do have the same subject. The fourth schema is inadequate because it fails to show the repetitions that are present in 3) at all.

Thus the logical structure of a compound predicate as a whole is not simply the result of conjoining and/or disjoining simple predicates as they might be viewed independently. All logical properties and relationships, including those essential to valid logic inference and deduction, rely absolutely upon repetitions among argument position holders. Within any given problem, whether of identifying logical properties, or of establishing by logical deduction and proofs that certain predicates entail others, it is essential to maintain accurate patterns of repetition and difference among argument position holders.

2.33 Meanings of Predicate Content and Predicate Structure

The analysis of any predicate, simple or compound, must distinguish two things: the predicate content, and the predicate's structure. Predicates and predicate schemata are identified and differentiated by reference to these two features.

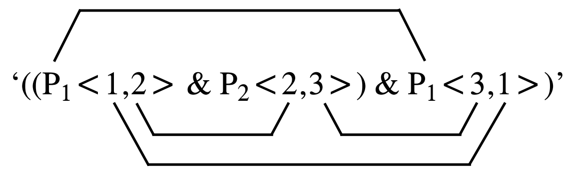
1) The predicate content is expressed through words or phrases which express the criterial meaning of the predicate. Thus 'is a child of', is the predicate phrase which expresses the content of the predicate '<1>is a child of<2>, and 'is male' is the predicate phrase which expresses the content of '<1>is male'. In compound predicates, the content is expressed by the words or phrases expressing the content of the elementary predicates as embedded in the logical structure of the predicate. The content-meaning of a predicate is combination of the meaning of the content-words and the logical form of the predicate as a whole. Ordinary language users readily grasp the content-meanings of predicates in the their language, though they may have difficulty providing synonyms or explications of these meanings. Whether two people have "have the same meaning in mind" for a predicate is tested by whether, given the same data, they agree on what the predicate would apply to it or not.

2) The predicate's logical structure. The "content" of a predicate is eliminated, leaving only the predicate's structure, when the words or phrases expressing the content of elementary predicates are

replaced by predicate letters, ‘P_1’, ‘P_2’, ‘P_3’,… etc. or ‘P’, ‘Q’, ‘R’,… A predicate letter by itself, shows no internal structure. But a compound wff of two predicate letters shows, at least, a compound predicate structure, since a) the two letter are either the same or different, and b) they are connected either by ‘and’ or by ‘or’. But elementary predicate schemata can also display the internal structure of atomic predicates in an ordered n-tuple of subject-term symbols (individual constants or argument position holders) affixed to the predicate letter, as in ‘P_1<1,a_1,2>’.

Thus two kinds of structures are displayed in a predicate schema; 1) the internal structure of atomic predicate schema expressed in symbols of the form ‘P_i<t_1,…,t_n>’ and cross-references between them, and 2) the over-all logical structure expressed by groupings, linear orders, and the connectives (here only ‘&’ and ‘v’) embedded in them.

Putting these two kinds of structure together, a compound predicate schema’s structure is displayed in the left-to-right order of groupings of conjunctive or disjunctive components and in the pattern of differences and/or repetitions and cross-references of predicate letters and placeholders for subject-terms, as in,

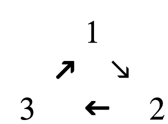


This diagram displays the logical structure of the compound predicate ‘(<1>married<2> and <2>knows<3> and <3>married<1>)’ which is the predicate of the sentence ‘(Ann married Bob and Bob knows Cal and Cal married Ann)’. (I.e., Spouse marries acquaintance of spouse.). The logical structure by itself, which is displayed in the schema ‘((P_1<1,2> & P_2<2,3>) & P_1<3,1>)’, has a meaning, however abstract, which is unchanged by &-commutation, or &-association..

The criterial meaning of such a predicate schema is determined in part by the meaning and positions of ‘&’ and ‘v’, but also by the meanings of parentheses and of linear orders. It is the idea of a certain way in which two predicates might be applies to three entities. We use this idea whenever we try to find predicates and names which, if structured in this way, would yield a meaningful or true or false expression. Putting ‘v’ in place of one or more occurrences of ‘&’ would expresses a structure with a different meaning, as would some alterations of the grouping or linear order of elementary components in such a structure.

Each different predicate schema has a meaning of its own, though it may be difficult or impossible to express directly. To show that schemata which are not SYN have distinctly different meanings, and how these differ, we keep the elementary predicate schemata constant and assigning some content to each of them, while varying the order, grouping, and occurrences of ‘&’ and ‘v’. Thus the basic components of content remain the same, while only the logical form changes. In the following example, we first list the predicate schema, then provide a complex predicate with that form, then provide a full sentence which makes some sense. Though we may not find words which express what the abstract meanings of the following six schemata are apart from all content, it is clear that these different logical structures have meanings, expressed by the schemata, which are not synonymous and are independent of any content assigned to the elementary components.

1) Initial predicate schema: ‘((P_1<1,2> & P_2<2,3>) & P_1<3,1>)’
‘(<1>married<2> and <2>knows<3> and <3>married<1>)’
Ann married Bob and Bob knows Cal and Cal married Ann



2) <u>Mixing & and v</u>: ‘$(P_1<1,2> \& (P_2<2,3> \text{ v } P_1<3,1>))$’
   [SYN ‘$((P_1<1,2> \& P_2<2,3>) \text{ v } (P_1<1,2> \& P_1<3,1>))$’]
   ‘(<1>married<2> and (<2>knows<3> or <3>married<1>))’
   Ann married Bob and either Bob knows Cal or Cal married Ann

3) <u>Interchanging & and v</u>: ‘$(P_1<1,2> \text{ v } (P_2<2,3> \& P_1<3,1>))$’
   ‘(<1>married<2> or (<2>knows<3> and <3>married<1>))’
   Either Ann married Bob or (Bob knows Cal and Cal married Ann)

4) <u>Changing the order of elementary predicate schemata</u>:
   ‘$(P_2<2,3> \text{ v } (P_1<1,2> \& P_1<3,1>))$’
   ‘(<2>knows<3> or (<1>married<2> and <3>married<1>))’
   Either Bob knows Cal or (Ann married Bob and Cal married Ann)

5) <u>Changed the grouping of elementary predicate schemata</u>:
   ‘$((P_2<2,3> \text{ v } P_1<1,2>) \& P_1<3,1>)$’
   [SYN ‘$((P_2<2,3> \& P_1<3,1>) \text{ v } (P_1<1,2>) \& P_1<3,1>))$’]
   ‘((<2>knows<3> or <1>married<2>) and <3>married<1>)’
   (Either Bob knows Cal or Ann married Bob) and Cal married Ann

6) <u>Making all connectives ‘v’</u>: ‘$((P_1<1,2> \text{ v } P_2<2,3>) \text{ v } P_1<3,1>)$’
   ‘(<1>married<2> or <2>knows<3> or <3>married<1>)’
   Either Ann married Bob or Bob knows Cal or Cal married Ann

Predicate schemata need not always represent all details of a predicate’s structure; e.g., we can use the predicate letter, ‘P_1’, to represent the saturated predicate ‘Steve married Jean’, or ‘$P_1<1>$’ to respresented ‘<1>married Jean’, even though ‘$P_1<1,2>$’ is the only full schematization of the predicate in the first sentence. A predicate schema is instantiated by an actual predicate if some appropriate predicate phrase replaces the predicate letter. Thus ‘<1>is child of<2>’ is an appropriate instantiation of ‘$P_1<1,2>$’. Also ‘<1>is a child of a brother of<2>’ would also be an appropriate instantiation of ‘$P_1<1,2>$’ even though this might be analyzed further. In general, logic is best served when all relevant details are schematized. But the monadic predicate, ‘<1>is male’ would not be an appropriate instantiation for ‘$P<1,2>$’; ‘<1>is male<2>’ is inappropriate because ‘is male’ is prima facie monadic.

The intuitively best way to understand the whole meaning of a compound predicate is to reduce its content to logically simple elements and put it into a synonymous disjunctive normal form. (This was not done in the second and fifth cases; the others are already in disjunctive normal form.) This is because understanding comes with a disjunction of sufficient conditions, and these are expressed in disjunctive normal form. Conjunctive normal forms with many disjunctive conjuncts list necessary conditions but leave obscure the sufficient conditions to apply or withold the predicate.

2.34 Numerals as Argument Position Holders

In schematizing the predicates in a segment of ordinary language for a given problem, it does not matter which numerals are used provided the same numeral is used at all occurrences of the same subject-term and different numerals are used for different subject-terms. Thus the predicate ‘Steve is a child of Jean’, if viewed as unrelated to others, can as well be expressed as ‘<5>is a child of<3>’ and taken as an instance of the predicate schema ‘$P_7<5,3>$’ or of ‘$P_2<5,3>$’ etc..

However, if other predicates or predicate schemata are involved in a given problem, the numerals placed in given positions must always be governed by the rules (i) sameness of numerals mirrors sameness of subjects, and (ii) differences among numerals must mirror possible differences in subject-terms, throughout the problem. To preserve synonymy the schemata must preserve possible differences.

Numerals within angle-brackets must not be confused with mathematical numbers, individual constants or individual variables. They do not stand for, or range over, any individual objects. As argument position holders, they have just two basic functions:

First, the presence of a numeral represents a position or location in which a subject-term can be placed—a blank space which would be left if we erased an occurrence of a subject term in an atomic sentence.

We refer to these positions as "argument positions" and any numerals placed in them as "argument position holders". Thus in the predicate schema '$P_7<5,3>$' the numeral '3' is the **argument position holder** in the **2nd argument position** and in any predicate instantiating '$P_7<5,3>$' e.g., in '<5>is a child of<3>'.

Secondly, in the ordered n-tuple of argument position holders in any simple or compound predicate, each numeral helps determine a pattern of cross-reference (repetition) and/or difference among argument terms in the predicate. If all argument positions in a schema have the same numeral, this indicates that they must all be occupied by the same subject term. If two or more different numerals occur as argument position holders, the positions they occupy may (but need not) be occupied by different subject terms.However, to be universal, logical laws must always cover the most differentiated possibilities, i.e., the possibility that every different numeral in angle- brackets is replaced by a different subject term. Thus the pattern of cross-references displays a predicate structure which is part of the meaning of any predicate which instantiates that schema.

2.35 Abstract vs. Particularized Predicates

Predicates and predicate schemata can be completely abstract or particularized. A predicate is completely abstract if all of its argument positions are open—occupied only by numerals, the argument position holders—so that no particular subject terms are left. It is particularized if the predicate is treated as if tied to some individual entity, i.e., if one or more of its argument positions is occupied an individual constant—by a term denoting a particular individual.

The sentence 1), 'Steve a child of Jean', can be said not only to have the predicate '<1>is a child of<2>'; it may also be said to have the monadic predicates '<1>is a child of Jean', or 'Steve is a child of <2>' (ordinarily taken as synonymous with '<2>is a parent of Steve'). A question 'Which is Jean's child?' may be viewed as a request to identify an individual to which '<1>is child of Jean' will apply. Such predicates are instances of monadic predicate schemata like '$P_1<1>$' or $P_2<2>$'.

However for many logical problems it is advantageous to recognize all distinctions between subject and predicate terms, so as to preserve the connection between such monadic predicates and the completely abstract predicates like '<1>is a child of<2>' to which they belong. To accomplish this we can replace singular terms like 'Jean' and 'Steve' which have specific referential content, by individual constants instead of numerals. This gives us particularized predicates, like

9) <1>is a child of$<a_1>$
10) $<a_2>$is a child of<2>

which are instances of the schemata '$P_1<1,a_1>$' and '$P_1<a_2,2>$' respectively, and can be used where finer detail is needed. Since at least one argument position is open (has a numeral) in 9) and 10) they are still only predicates—not sentences. A completely particularized predicate, with individual constants in every argument position, would represent a sentence, i.e., a saturated predicate, a predicate embedded in a sentence.

Thus a completely abstract predicate is one which has words and phrases expressing content, but has all subject-terms replaced by numerals. A particularized predicate is a predicate that has one or more of its argument positions occupied by individual constants.[3] The individual constant means that a specific subject term is to be included in the meaning of the predicate phrase at that point in the predicate structure. Positions occupied by numerals are 'open positions'; they do not preclude any individual constants being put in their place. Positions occupied by individual constants signify a fixed part of the predicate; one and only one individual can occupy such positions and this does not vary.

Predicate schemata are distinguished as abstract or particularlized on the same basis. (From now on I use 'abstract predicate' for 'completely abstract predicate'). Particularized predicate schemata like '$P_1<1,a_1>$' signify that they are schemata only for predicates whose criterial meanings are tied to as many individual entities as individual constants in their argument positions. Numerals in argument positions, on the other hand, represent predicates so far as they are free of attachment to specific individual entities.

2.36 'n-place' vs. 'n-adic' Predicates

Predicates and predicate schemata must be distinguished by both a) the number of argument places in the predicate, and b) the number of argument positions filled by argument-position-holders (numerals) as distinct from logical constants. A predicate is an n-place predicate if it has n argument positions—i.e., n places that are occupied by either numerals or individual constants. It is an **m-adic** predicate if it has **m** open positions, i.e., **m** argument positions which are occupied by numerals.

We noted above that two particularized predicates could be abstracted from sentence 1). More generally, from any singular sentence with n distinct argument places, it is possible to abstract 2^n-2 different particularized predicates. For example, the sentence 'Al is between Ben and Cal' has the completely abstract 3-place predicate,

$P<1,2,3>$ '<1>is between<2>and<3>' (3-adic,3-place predicate)

but also, we could abstract six (= 2^3-2) particularized 3-place predicates, three of which are 1-adic, and three 2-adic:

$P<1,a_1,a_2>$	'$<1>$is between$<a_1>$and$<a_2>$),	1-adic, 3-place.
$P<a_1,1,a_2>$	'$<a_1>$is between$<1>$and$<a_2>$'	
$P<a_1,a_2,1>$	'$<a_1>$is between$<a_2>$and$<2>$'	
$P<a_1,1,2>$	'$<a_1>$is between$<1>$and$<2>$'	2-adic, 3-place
$P<1,a_1,2>$	'$<1>$is between$<a_2>$and$<2>$'	
$P<1,2,a_1>$	'$<1>$is between$<2>$and$<a_2>$'	

3. Completely abstract predicates (vs. predicate schemata) are linguistic symbols which some philosophers from Plato on, could interpret as standing for universals. Whether universals in the Platonic sense exist objectively is a matter for metaphysics. But completely abstract predicates can be presented and their meanings understood though the symbols (tokens) for predicates and predicate schemata are physical objects, not universals.

Of course, we can also get six concretely particular predicates, like the instance of P<1>, '<1>is between Ben and Cal', and the instance of P_2<1,2>, 'Al is between <1>and<2>'. But we ignore these. Infinitely many additional such predicates can be formed from the same abstract predicate by using other names than 'Al', 'Ben' or 'Cal', and by interchanging positions of names.

TABLE 2-1

PREDICATES	**PREDICATE SCHEMATA**	
Abstract Predicates	**Abstract Predicate Schemata**	
Simple	Strict	Simplified
2) <1>is a child of<2>	$P_1<1,2>$	P12
6) <1> is a brother of <2>	$P_2<1,2>$	Q12
Compound	Compound	
4) (<1>is a child of<2> and <3>is a brother of<2>)	$(P_1<1,2>\&P_2<3,2>)$	(P12 & Q32)
Particularized Predicates	**Particularized Predicate Schemata**	
9) <1>is a child of<a_1>	$P_2<1,a>$	Q1a
10) a_2 is a brother of<2>	$P_2<a,1>$	Qa1
SENTENCES	**SENTENCE SCHEMATA**	
Simple	Strict	Simplified
1) Steve is a child of Jean	$P_1<a_1,a_2>$	Pab
5) Dick is a brother of Jean	$P_2<a_2,a_1>$	Qba
Compound	Compound	
3) Steve is a child of Jean and Dick is a brother of Jean	$(P_1<a_1a_2> \& P_2<a_3a_1>)$	(Pab & Qcb)

The distinctions between simple and compound predicates and predicate schemata, between abstract and particularized predicates and schemata and between simple and compound sentences and sentence schemata, for expressions 1) to 6), 9) and 10) in Sections 2.32 and 2.35 are summarized in TABLE 2-1.

[Notes on TABLE 2-1: 1) Schemata on different lines are independent of each other, although P_1 and P_2 are instantiated by 'is a child of' and 'is a brother of' throughout the left-hand column

2) The next to the right-most column shows the strictly correct method of schematizing. The right-most column shows a simplified, easier-to-read method which is often used instead. In that column 'P' replaces 'P_1', 'Q' replaces 'P_2', and so on. Likewise, 'a' replaces 'a_1', 'b' replaces 'a_2', 'c' replaces 'a_3'. Also the angles '<' and '>' used to signify ordered n-tuples, and the commas within them, are dropped.]

2.37 Modes of Predicates and Predicate Schemata

In one sense an abstract predicate and its particularizations are different predicates. Though all have the same n-place predicate phrase, no two of them are co-applicable to exactly the same set of ordered n-tuples. Each has a different predicate structure suggesting differences in referential meaning. For the referential meaning is different when a predicate is tied to different individuals or applied to different n-tuples of individuals.

On the other hand, predicates which have the same predicate phrase and the same number of argument positions, all have the same general criterial meaning. The general criterion of one thing's being between two others is the same in the purely abstract predicate, '<1>is between<2>and<3>' and in all of its particularizations.

We will call a predicate Q a mode of a predicate P, if P expresses a more general criterial meaning which belongs to Q. Thus all six particularized predicates listed in the preceding section are modes of the abstract predicate '<1>is between<2>and<3>'. But the relation is not symmetric. An abstract predicate is not a mode of its particularizations nor are the latter necessarily modes of each other. When a predicate is tied to a constant, that constant becomes an added component in the predicate's meaning. To tell whether the predicate '<1>is a child of<a>' applies or not, one must be able to identify an individual specified by a, as well as apply the criterion of the relation of 'being a child of '. The relation of being-a-mode-of, which may hold of an ordered pair of predicates, is reflexive, non-symmetrical, and transitive.

Different modes of the same predicate vary in their referential meaning. But if Q is a mode of P, then the criterial meaning of P will be part of the meaning of Q. All modes of a given predicate share its criterial meaning. When $[P_i<a,1>]$ is a mode of $[P_i<1,2>]$, the criterial meaning of $[P_i<1,2>]$ is part of the meaning of $[P_i<a,1>]$, But $[P_i<1,2>]$ does not logically Contain $[P_i<a,1>]$, nor does $[P_i<a,1>]$ logically Contain $[P_i<1,2>$. Rather the relation between them is that the logical properties and relations of $[P_i<1,2>]$ are preserved in its modes, e.g., in $[P_i<a,1>]$. This is relies on the rule of Instantiation, R2-3.

Theorems of analytic logic having the form '[A SYN B]' assert the sameness of both the criterial and referential meanings of predicates which instantiate the schemata of A and B, regardless of what is put for 'A' and 'B'. Predicate schemata do not have criterial meanings of any actual predicates because the predicate phrases which convey such meanings are replaced by predicate letters. All that is left are predicate schemata which display sameness or differences of criterial meanings and the sameness and differences of the referential meanings of subject terms (represented by occurrences of the same or different predicate letters or individual constants). However, the criterial meaning represented by a predicate schema and schemata of its modes are the same.

The purpose of formal logic, is to teach the criterial meanings of the specific logical predicates used in formal logic—SYN, CONT, INC, TAUT, LOGICALLY VALID, etc.. That is, to teach the criteria of application and construction—i.e., how to tell when one expression entails another, when an argument is valid, a sentence is tautologous, a group of statements inconsistent—i.e., how to tell when the predicates 'is tautologous', 'is valid' etc., apply and do not apply.

How do we distinguish whether one predicate, or predicate schema is a mode of another? The symbols, 'P<1,2>', 'P<2,1>', 'P<a,1>', 'P<7,b>' and '(P<1,2> & P<2,a>)' are each whole well-formed predicate schemata. Which are modes of which? What follows is an explication of criteria for the predicate 'is a mode of ':

Within any given logical context, a predicate P_2 is a mode of predicate P_1
Syn (in the discussions which follow)
(i) P_2 and P_1 have the same predicate phrase, and
(ii) P_2 and P_1 have the same number of argument positions, and
(iii) P_2 has the same individual constant in any position at which P_1 has an individual constant.
(iv) P_2 has the same or more patterns of cross-referenced positions than P_1.

This definition has the following consequences:

1) All occurrences of actual predicates which have the same predicate phrase, the same number of argument positions, the same individual constants in the same positions, the same number of numerals, and the same pattern of cross-reference are modes of each other.

For example, in any given context,

‘<1>is between<2>and<3>’ ‘<7>is between<3>and<5>’	are modes of each other
‘<a>is between<3>and<3>’ ‘<a>is between<1>and<1>’	are modes of each other

2) If an actual predicate Q is the same as predicate P except for having one or more individual constants where P has numerals, then Q is a mode of P, but P is not a mode of Q. Thus [Q mode P] is non-symmetrical.

For example, in any given context,

Q	Is a mode of	P
‘<a>is between<2>and<3>’	is a mode of	‘<1>is between<2>and<3>’
‘<a>is between<7>and<5>’	is a mode of	‘<1>is between<2>and<3>’
‘<7>is between<a>and<b>’	is a mode of	‘<1>is between<2>and<3>’
‘<a>is between<2>and<b>’	is a mode of	‘<a>is between<1>and<3>’

But the converses do not hold, the P’s on the right-hand side are not modes of the Q’s on the left-hand side because that would violate clause (iii) in the definition.

3) If Q is the same as P except for having one or more cross-references among numerals where P has none, then Q is a mode of P but P is not a mode of Q because that would violate clause (iv).

For example, in any given logical problem,

‘<**1**>is between<2>and<**1**>’	is a mode of	‘<**1**>is between<2>and<**3**>’
‘<1>is between<b>and<1>’	is a mode of	‘<1>is between<2>and<3>’
‘<a>is between<a>and<b>’	is a mode of	‘<1>is between<2>and<3>’
‘<a>is between<2>and<a>’	is a mode of	‘<a>is between<1>and<3>’

Note that these modes of the specific predicate ‘between’ are usually considered inapplicable—i.e., the usual meaning of ‘between’ is such that its criterion only applies to three entities which are different from each another; nothing is between itself and something else, or between the same thing. That these expressions are inapplicable does not mean that these are not modes of ‘between’; it means that the meaning of “between” excludes these modes—applications to less than three distinct individuals.

The examples just given were examples of an actual predicate, ‘between’. More generally, the meaning of ‘<1>is a mode of<2>’ can be extended to apply to predicate schemata. We want it to be the case that if a predicate schema B is a mode of a predicate schema A, then every predicate which instantiates the schema B, will be a mode of every predicate which instantiates A (assuming similar replacements for similar symbols).

Predicate schema, Q, <u>is a mode of</u> predicate schema P
<u>Syn</u> Q and P are predicate schemata <u>and</u>
(i) Q and P have the same predicate letter <u>and</u>
(ii) Q and P have the same number of argument positions <u>and</u>
(iii) Q has the same individual constant in any position at which P has an individual constant <u>and</u>
(iv) Q has the same or more cross-referenced positions than P.

It follows, that 1) All occurrences of a predicate letter standing alone are modes of each other. Since, for any i, all occurrences of ‘P_i’ in a given context have the same predicate letter, and, trivially, show no differences of internal structure, since no structure is shown.

2) All occurrences of predicate schemata with the same predicate letter, the same number of argument positions, the same constants in the same positions, the same number of numerals, and the same pattern of cross-reference among its numerals, are modes of each other.

<u>For example</u>, in any given logical problem,

‘$P_2<1,2>$’, ‘$P_2<2,1>$’, ‘$P_2<7,3>$’, are modes of each other,

‘$P_1<1,2,3>$’
‘$P_1<7,3,5>$’ } are modes of each other

‘$P_3<a,2,3>$’
‘$P_3<a,7,5>$’ } are modes of each other

‘$P_2<1,a,1,b>$’
‘$P_2<7,a,7,b>$’ } are modes of each other.

3) If Q is the same as P except for having one or more individual constants where P has numerals, then Q is a mode of P, but P is not a mode of Q. [(Q is a mode of P) is non-symmetrical]

<u>For example</u>, in any given logical problem,

[Q	is a mode of	P]
‘$P_1<a,2,3>$’	is a mode of	‘$P_1<1,2,3>$’
‘$P_1<a,7,5>$’	is a mode of	‘$P_1<1,2,3>$’
‘$P_1<7,a,b>$’	is a mode of	‘$P_1<1,2,3>$’
‘$P_1<a,2,b>$’	is a mode of	‘$P_1<a,1,3>$’
‘$P_2<1,a>$’	is a mode of	‘$P_2<2,1>$’ and of ‘$P_2<7,5>$’,
‘$P_2<a,a,c,b>$’	is a mode of	‘$P_2<2,a,1,b>$’

But the converses do not hold, because they violate clause (iii) in the definition.

4) If Q is the same as P except for having one or more cross-references among numerals where P has none, then Q is a mode of P but P is not a mode of Q. <u>For example</u> in any given logical problem:

‘$P_1<1,2,1>$’	is a mode of	‘$P_1<1,2,3>$’
‘$P_1<1,b,1>$’	is a mode of	‘$P_1<1,2,3>$’
‘$P_1<a,a,b>$’	is a mode of	‘$P_1<1,2,3>$’
‘$P_1<a,2,a>$’	is a mode of	‘$P_1<a,1,3>$’

But the converses do not hold, for they do not satisfy clause (iv) in the definition.

2.4 Rules of Inference

Given the distinctions of Section 2.3 how should the substitution rules be changed to provide for all substitutions which can be guaranteed to preserve logical properties and relations?

2.41 No change in R1. SynSUB, Substitutability of Synonyms

No basic changes need be made in R1, SynSUB,the rule of substitution of logical synonyms. As before, if two wffs are to be logically synonymous, they must contain all and only the same elementary wffs. Nothing has been changed except that elementary wffs now include atomic n-place predicate schemata, as well as the 0-place predicate letters. To be counted as two occurrences of the same elementary predicate place-holder, two symbols must be identical in all respects—have the same symbol(-type)s in the same positions at each position in the left-to-right sequence of primitive signs.

Starting with our original axioms, a typical use of SynSUB with predicate place holders involves an implicit use of U-SUB in the simple original way. Consider first a more explicit version of the proof of T1-17:

T1-17. (P & (P v Q)) SYN (P v (P & Q))
Proof: 1) (P & (Q v R)) SYN ((P & Q) v (P & R)) [Ax.1-08]
2) (P & (P v R)) SYN ((P & P) v (P & R)) [1),U-SUB('P' for 'R')]
3) P SYN (P & P) [Ax.1-01]
4) (P & (P v R)) SYN (P v (P & R)) [2),3),R1]
5) (P & (P v Q)) SYN (P v (P & Q)) [4),U-SUB('Q' for 'R')]

A proof of an analogous theorem in the logic of predicates, differs only because U-SUB allows new kinds of predicate schemata with occupied argument positions to replace single predicate letters uniformly. Thus using the augmented U-SUB to substitute '$P_1<1,2>$' for 'P' in step 5) and '$P_2<2,1>$' for 'R' in step 6), we prove the SYN-theorem:

$\models [(P_1<1,2>\&(P_1<1,2>vP_2<2,1>))$ SYN $(P_1<1,2>v(P_1<1,2>\&P_2<2,1>))]$
Proof: 1) [(P & (P v R)) SYN (P v (P & R))] [T1-17]
2) $[(P_1<1,2>\&(P_1<1,2>vR))$ SYN $(P_1<1,2>v(P_1<1,2>\& R))]$
[1),U-SUB('$P_1<1,2>$' for 'P')]
3) $[(P_1<1,2>\&(P_1<1,2>vP_2<2,1>))$ SYN $(P_1<1,2>v(P_1<1,2>\&P_2<2,1>))]$
[2),U-SUB('$P_2<1,2>$' for 'R')]

Step 4) and 5) in this proof has the same over-all logical form as step 5) in the preceding proof. The only thing that is new is the greater complexity of the logical structures of predicates substituted.

Thus the rule R1 (or SynSUB), which is the only rule which changes the overall logical structure of synonymous predicate schemata, is unchanged by the new additions to the system of Chapter 1. No change in over-all structure can be brought about by the augmented U-SUB—only changes in the detailed internal structure of subordinate components. The requirement that all elementary wffs on one side of 'SYN' also occur on the other side is unaffected.We have merely introduced new sets of elementary wffs, from which new compound wffs can be constructed.

2.42 The Augmented Version of R2, U-SUB

Attention to the modes of a predicate or predicate schema plays an important role in the expanded rule of U-SUB. What is wanted is a rule which will provide for uniform substitution of a complex predicate schema for all occurrences of any modes of an atomic predicate schema in all and only those cases in which such a substitution will preserve theoremhood. The rule must allow the substitution of a new predicate schema (simple or compound) for each mode of any abstract atomic predicate schema which has modes in the given logical context. In general theoremhood can be preserved by the substitution of a suitable complex predicate at all modes of an atomic predicate schema, but theoremhood is not preserved if a predicate is substituted for an expression which is not a mode of the predicate schema to be replaced (the substituend), or if the result of substitution (the substuents) are not a mode of the predicate schema to be substituted.

Initially, the U-SUB rule is quite simple and involves nothing new: If A is a theorem and B is the result of replacing all occurrences of some 0-place predicate-letter in A by some predicate schema, then B is a theorem, as in Example #1, Section 2.2. But the real power of predicate logic shows itself in a more complicated form of substitution, namely, when a theorem or set of wffs contains several different n-place modes of an abstract atomic predicate schema ($n>0$). Each distinct mode will have the same same predicate letter and the same number of argument places but different numerals and/or individual constants in different argument positions. The augmented rule of U-SUB preserves theoremhood by introducing an n-adic predicate or predicate schema at each occurrence of these different modes of the predicate schema. U-SUB is augmented and supplemented by INST, a Rule of Instantiation:

An example of the result of using R2-2, the current version of U-SUB, was presented in Example #2, Section 2.2. We now discuss it in detail, and make explicit the rules which govern its application. The rules of formal logic require consistency within a given context of rational inquiry but allow variations in different contexts. All n-place predicate schemata whose members have the same predicate letter, P_i, must be viewed as modes of the same completely abstract n-place predicate schema, $P_i<1,2,\ldots,n>$, provided they occur in the same logical context.

They "occur in the same logical context" if they all occur,

(i) within a given compound predicate schema which is being studied for logical properties or relations;[4]

or (ii) within two predicate schemata which are being examined for logical synonymy or containment;

or (iii) within sequences of SYN statements, intended as a logical deduction or proof;

or (iv) in any set of well-formed predicate schemata which is investigated for a logical property (e.g. inconsistency), or logical relationship to another wff or sets of wffs.

In any of these contexts all n-place predicate schemata which have the same predicate letter, P_i, affixed, are taken to be modes of a predicate-schema $P_i<1,2,\ldots,n>$ which represents the common criterial meaning of them all.

Uniform substitution of a new n-adic predicate schema, Q, (which may be simple or compound) for all modes of an abstract atomic predicate schema $P<1,2,\ldots,n>$ requires introducing Q at each occurrence of each mode of $P<1,2,\ldots,n>$ in the context.

4. We do not deal with any logical properties of negation-free predicate schemata standing by themselves in this chapter. But there will be some such properties (inconsistency and tautologousness) in Chapter 5 after negation is introduced. Here we deal only with the logical relations, SYN and CONT.

The crucial step of introducing Q at a single occurrence of an atomic wff is now explained.

1. Let R represent the predicate schema, or set of predicate schemata, within which the univrsal substitution is to take place.

2. Let the expressions to be replaced be the modes of $P<1,2,\ldots,n>$ in R;
   i.e., the set of atomic wffs $\{A_1,A_2,\ldots,A_m\}$ in R which meet the following conditions:
   (i) every A_i is an atomic wff with the same predicate letter, P, and just n occupied argument positions;
   (ii) any argument position in any A_i may be occupied by
   any individual constant or any numeral,
   (thus different modes will be represented by different predicate schemata)

3. A well-formed predicate schema, Q, simple or compound, is a suitable expression to be substituted at every occurrence of a mode of $P<1,2,\ldots,n>$ in R, Syn_{df}
   (i) Q is a wff, and
   (ii) Q is h-adic, where $h \geq n$. [Note that the abstract predicate for the modes $P<1,2\ldots,n>$, is by hypothesis n-place] and
   (iii) Q has at least one occurrence of each numeral 1 to n (in any order) in at least one position attached to it.

4. Rule of introduction:
   If Q is a suitable predicate and
   and A_i is a mode of $P<1,2,\ldots,n>$
   then Q is introduced at an occurrence of A_i by
   (i) replacing that occurrence of A_i by Q, and then
   (ii) replacing each occurrence of a numeral 'i' in Q with the occupant of the ith argument position in A_i.[5]

5. Q is uniformly substituted for modes of $P<1,2,\ldots,n>$ in a wff R,
   iff
   Q is suitable and Q is introduced at each occurrence
   of each mode, $P<t_1,\ldots,t_n>$, of $P<1,2,\ldots,n>$ in R.

6. Abbreviations: Let '$R([P<t_1,\ldots,t_n>]//Q)$' abbreviate
   'the result of introducing Q at an occurrence of a mode of $P<1,2,\ldots,n>$ in R'.
   Let '$R([P<t_1,\ldots,t_n>]/Q)$' abbreviate
   'the result of introducing Q at *every* occurrence of a mode of $P<1,2,\ldots,n>$ in R'.

---

5. This process of "introduction" comes directly from Quine. In his Elementary Logic, (1965), p 99 he wrote: "Introduction of a predicate or predicate schema P at a given occurrence of a predicate letter consists in supplanting that occurrence and the attached string of variables … by the expressions which we get from P by putting the initial variable of the string for '①', the next variable for '②', and so on." Quine used this only to replace individual variables; we use it to replace individual constants and argument position holders as well. We of course use '$<1>$' instead of '①' etc.

7. The (augmented) Rule of U-SUB, R2-2:

R2-2 If [R SYN S] and $P<t_1,\ldots,t_n>$ occurs in R, [U-SUB]
and Q is an h-adic wff, where $h \geq n$,
and Q has occurrences of all numerals 1 to n,
then $\models$ [R($P<t_1,\ldots,t_n>$/Q) SYN S($P<t_1,\ldots,t_n>$/Q)].
(where 'R($P<t_1,\ldots,t_n>$/Q)' means the result of replacing each occurrence of '$P<t_1,\ldots,t_n>$' by Q with the ith argument in '$P<t_1,\ldots,t_n>$' replacing an occurrence of i in Q.)

Example: Let R be = [($P_1<1,7>$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]
To introduce Q = $Q<2,a,1>$ for all occurrences of $P_1<t_1,t_2>$ in R
[I.e., $P_1<t_1,\ldots,t_n>$, here, is any 2-place wff preceded by the predicate letter 'P_1']

Step 1) Determine whether '$Q<2,a,1>$' (for Q) is a suitable predicate schema to introduce at $P_1<t_1,t_2>$.
Answer: it is suitable because (i) Q is a wff, and
(ii) '$Q<2,a,1>$' is h-adic, and $h \geq n$ where h=3, n =2 ($P_1<t_1,t_2>$ is 2-place) and
(iii) '$Q<2,a,1>$' has at least one occurrence of each numeral 1 to n (n=2) in at least one position.

Step 2) introduce Q (in this case,'$Q<2,a,1>$')
at the first occurrence of '$P_1<t_1,t_2>$' (in this case 'P_1 (1,7)')
by first replacing that occurrence, by '$Q<2,a,1>$':

(i) $\models$ [($\mathbf{P_1<1,7>}$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]
[($\mathbf{Q<2,a,1>}$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]

and then (ii) replace each occurrence of a numeral 'i' in Q (I.e., in '$Q<2,a,1>$')
with the occupant in the ith argument position in '$P_1<t_1,t_2>$':

(ii) $\models$ [($\mathbf{P_1<1,7>}$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]
[($\mathbf{Q<2,a,1>}$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]
[($\mathbf{Q<7,a,1>}$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]

This introduces '($Q<2,a,1>$' at the first occurrence of a mode of $P_1<1,7>$. But the augmented Rule U-SUB requires that it be introduced at all occurrences of any mode of $P_1<t_1,t_2>$ to preserve logical synonymy. So the final result (from which the intermediate steps are eliminated) is b) below:

a) $\models$ [($P_1<1,7>$ & $P_1<7,b>$) SYN ($P_1<7,b>$ & $P_1<1,7>$)]
[$Q<2,a,1>$ $Q<2,a,1>$) ($Q<2,a,1>$ $Q<2,a,1>$)] (Introduction Step (i))
b) $\models$ [($Q<7,a,1>$& $Q<b,a,7>$) SYN ($Q<b,a,7>$& $Q<7,a,1>$)] (Step (ii)) **.:** [a),R2-2]

and the rule is: **R2-2 If $\models$ R and (i) $P_i<t_1,\ldots,t_n>$ occurs in R, and**
(ii) Q is an h-adic wff, where $h \geq n$, and
(iii) Q has occurrences of all numerals 1 to n,
then $\models$ [R($P_i<t_1,\ldots,t_n>$/Q) may be inferred

As a special case of this rule, if all occurrences of P_i are affixed to exactly the same j-adic n-tuple of terms, the augmented rule allows uniform replacement of that atomic j-adic n-place predicate by an (n+m)-place predicate. I.e., it allows substitution simpliciter of more complicated predicate schemata for less complicated ones. The uniform substitution of any wff whatever for all occurrences of a 0-place predicate letter is a special case of this special case.

What is the value of this augmented rule of substitution? How can it be used in logic? Viewed merely as a device to get a new unquantified theorem-schemata from an unquantified theorem previously established, there are no theorems gotten by using the augmented U-SUB with its intricate rule of "introduction" that could not have been gotten with the simpler U-SUB rule, R1-2 of Chapter 1 with the introduction of m-place predicates as wffs.[6]

Further, since the synonymies and containments of unquantified logic rely only on the meanings of the connectives 'and' and 'or', the relationships between internal structures (modes) of a predicate do not affect logical containment or logical synonymy in its purely formal logic. Given any SYN-theorem or CONT-theorem with one or more occurrences of some mode of a predicate, uniform replacement of all occurrences of that mode by any other expression, whether a mode of the same predicate or not, preserves SYN-theoremhood and CONT-theoremhood.

What then, is the value of augmenting the simpler rule R1-2? Its chief advantage for formal logic lies in quantification theory. There it is the basis of a rule for deriving new theorems in quantification theory by introducing complex predicate schmata for all modes of an abstract atomic predicate with individual variables. This will be spelled out in the Chapter 3. But the preview below will illustrate how the method of introducing a predicate schema in R2-2 is related to U-SUB in Quantification theory. Consider the theorem T3-35. (∃x)(∀y)Rxy CONT (∃x)Rxx. In a domain of 2 individuals, {a,b}, '(∃x)(∀y)Rxy CONT (∃x)Rxx' means

1) (R<a,a> & R<a,b>) v (R<b,a> & R<b,b>) CONT (R<a,a> v R<b,b>)

This is a theorem of unquantified logic. Introducing Q<2,c,1> for R<1,2> in 1) by R2-2 we get.

2) (Q<a,c,a> & Q<b,c,a>) v (Q<a,c,b> & Q<b,c,b>) CONT (Q<a,c,a> v Q<b,c,b>)

This result of using R2-2 is a new theorem of unquantified logic; theoremhood has been preserved. But in the domain of the 2 individuals. {a,b} 2) means, '(∃x)(∀y)Qycx CONT (∃x)Qxcx.'

In the same way, in Chapter 3 "suitable" predicate schemata permitted by R3-2 can be introduced directly into quantified wffs regardless of the size of the domain, thus,

from 3) |= (∃x)(∀y)Rxy Cont (∃x)Rxx. [T3-35]

by introducing Q<2,c,1> for R<1,2> in **3)** we derive directly

4) |= (∃x)(∀y)Qycx Cont (∃x)Qxcx [3),R3-2]

and this derivation is valid in all domains, preserving theoremhood.

R3-2 is the version of U-SUB which is introduced in Chapter 3. It is the same as R2-2 except for the addition of a clause concerning individual variables designed to prevent quantifiers from kidnapping variables that don't belong to them. Thus theoremhood is preserved in quantification theory by introduc-

---

6. All theorems gotten by the augmented rule could be gotten by substituting the different modes for different 0-place predicate letters. For example, in the preceding section we used the new U-SUB rule to replace all occurrences of P1<1,2> by '(P3<2,a,1>' in the theorem,

a) |= [(P1<1,7> & P1<7,b>) SYN (P1<7,b> & P1<1,7>)]

to get the theorem,

b) |= [(P3<7,a,1> & P3<b,a,7>) SYN (P3<b,a,7> & P3<7,a,1>)]

But we could have gotten this result directly, using R1-2, from Axiom 1-03 [(P & Q) SYN (Q & P))], from which a) was derived, now that 'P1<1,7>', 'P1<7,b>', 'P3<7,a,1>' and 'P3<b,a,7>' are admitted as wffs.

ing predicate schemata at all occurrences of a given atomic component of a theorem by using R3-2 in the same way that R2-2 preserves theoremhood with unquantified wffs.

The Rule of U-SUB in quantification theory will be qualified further and explained more fully in the next chapter.

2.43 The Rule of Instantiation, INST

The new Rule of Inference, Instantiation (abbr. INST) is the simple rule that if predicates have logical properties or stand in logical relations with one another, then expressions which apply those predicates to individual entities also have those logical properties and stand in those same logical relations. For example, since '(P<1> SYN (P<1> & P<1>))' is a theorem by U-SUB in Ax. 1.01. We may infer that '(Pa SYN (Pa & Pa))' is theorem, where 'a' may be any individual constant by INST.

Note that we do not say that 'P<1>' logically contains 'Pa',or vice versa. Nor do we say that 'P<1>' is logically synonymous with 'Pa'. We do not even say that if 'Pa' is true, then 'P<1>' is true or vice versa, for 'P<1> being a predicate is neither true nor false.

What the rule says, in this chapter, is that if two predicates are logically synonymous then any syn-statement formed by putting an individual constant at all occurrences of some argument position holder which occurs in the former, is also logically synonymous. Derivatively, if one predicate logically contains another, the result of replacing any argument position holder at all occurrences by an individual constant, will also be a logical containment. In symbols,

If [P<1> SYN Q<1>] then [Pa_i SYN Qa_i], and If [P<1> CONT Q<1>] then [Pa_i CONT Qa_i].

More generally, if any statement correctly attributes a logical property to a predicate, or a logical relation to an n-tuple of predicates, then that logical property, or relation, will be correctly ascribed to the result of substituting an individual constant at all occurrences of any argument position holder that occurs in the intitial statement. In other words theoremhood will be preserved by Instantiation. In symbols, the more general rule of Instantiation is 'If |= [P<1>] then |= [Pa_i].

All three Rules of Inference are substitution rules: SynSUB substitutes synonynous expressions for each other at one or more occurrences, preserving theoremhood. U-SUB substitutes complex predicates or predicate schemata for all occurrences of an atomic predicate or predicate schemata in a theorem of logic preserving theoremhood in the result, and INST substitutes individual constants at all occurrences of an argument-position-holder in a theorem, preserving theoremhood in the result. The second rule, U-SUB, bears the closest resemblance to the rule that has usually been called the Rule of Uniform Substitution.

Many new theorems may be derived from the new rules of Substitution and Instantiation, but we will not list these theorems in this chapter.

DR2-2 If A is a theorem and P_i is an n-place predicate schema ($n \geq 0$) with occurrences in A
and Q is a h-adic predicate schema with $h \geq n$,
and Q has at least one occurrence of each numeral from 1 to n,
and A(P_i /Q) is the result of introducing Q at every occurrence of $P_{i,}$
then A(P_i /Q) is a theorem.

From DR2-2 one can derive an unlimited number of new theorems from each of the theorems of Chapter 1. For example, from T1-16. [(P **v** (P & Q)) SYN (P & (P **v** Q))] one can substitute for P alone, any purely abstract predicate, as in,

|= [(P<1> **v** (P<1> & Q)) SYN (P<1> & (P<1> **v Q**))]

or a predicate schema with a different predicate letter and the same argument position holder:

|= [(P<1> **v** (P<1> & R<1>)) SYN (P<1> & (P<1> **v** R<1>))]

or one or the other may be substituted with an individual constant instead of an argument position holder, making the components into particularized predicates:

|= [(Pa **v** (Pa & Q)) SYN (Pa & (Pa **v** Q))]
|= [(Pa **v** (Pa & R<2>)) SYN (Pa & (Pa **v** R<2>))]
|= [(P<1> **v** (P<1> & Ra)) SYN (P<1> & (P<1> **v** Ra))]

and these may be replaced by more complicated predicates such as introducing Q<2,b,1> for P<1>:

|= [(Q<2,b,1> **v** (Q<2,b,1> & Ra)) SYN (Q<2,b,1> & (Q<2,b,1> **v** Ra))]

or, instead we could have introduced Q<2,b,1> at all occurrences of Ra:

|= [(P<1> **v** (P<1> & Q<2,b,a>)) SYN (P<1> & (P<1> **v** Q<2,b,a>))] etc..

Thus from every theorem of Chapter 1, many new theorems are derivable using U-SUB in the form of R2-2, and INST. As a starter from T1-16 we derived,

T1-16. [(P v (P & Q)) SYN (P & (P v Q))]
1) [(P<1> v (P<1> & Q)) SYN (P<1> & (P<1> v Q))] [T1-16,R2-2]
2) [(P<1> v (P<1> & R<1>)) SYN (P<1> & (P<1> v R<1>))] [2),R2-2]
3) [(P<1> v (P<1> & Ra)) SYN (P<1> & (P<1> v Ra))] [3),INST]
4) [(Pa v (Pa & Q)) SYN (Pa & (Pa v Q))] [1),INST]
5) [(Pa v (Pa & R<2>)) SYN (Pa & (Pa v R<2>))] [4),R2-2]
6) [(Q<2,b,1> v (Q<2,b,1> & Ra)) SYN (Q<2,b,1> & (Q<2,b,1> v Ra))] [3),R2-2]
7) [(P<1> v (P<1> & Q<2,b,a>)) SYN (P<1> & (P<1> v Q<2,b,a>))] [3),R2-2]

INST, the rule of Instantiation, is like U-SUB in that something is substituted at all occurrences of a simple component. But it is different than U-SUB in other respects. It interchanges symbols for singular terms rather than symbols for predicates, and it has no analogue for the procedure of introducing a predicate schema at all occurrences of any mode of some abstract predicate schema. It simply substitutes individual constants directly for argument-position-holders.

INST allows us to preserve theoremhood while substituting any individual constant we wish at every occurrence of a given argument-position-holder in a theorem. There can be many particularized instantiations of the same predicate. This provides the basis for generalized theorems with quantificational wffs.

7) |= [(P<1> v (P<1> & Q<2,b,a>)) SYN (P<1> & (P<1> v Q<2,b,a>))] [3),R2-2]
|= [(Pa v (Pa & Q<2,b,a>)) SYN (Pa & (Pa v Q<2,b,a>))] [7),INST]
|= [(Pb v (Pb & Q<2,b,a>)) SYN (Pb & (Pb v Q<2,b,a>))] [7),INST]
|= [(Pc v (Pc & Q<2,b,a>)) SYN (Pc & (Pc v Q<2,b,a>))] [7),INST]
|= [(Pd v (Pd & Q<2,b,a>)) SYN (Pd & (Pd v Q<2,b,a>))] [7),INST]

Hence for the domain of four, {a,b,c,d}, we derive:

.: |= [(∀x)[(Px v (Px & Q<2,b,a>)) SYN (∀x)(Px & (Px v Q<2,b,a>))]

The fact that U-SUB will preserve theoremhood by introducing many different abstract predicate schemata at all occurrences of a given atomic predicate schema can not be used as a basis for generalization. We can introduce "P<1>' or 'Pa' or 'Q<2,b,1>' etc., for all occurrences of 'P' in T1-16, without changing the other components and thereby preserve theoremhood, as in

T1-16.	[(P	v (P	& Q)) SYN (P	& (P	v Q))]	
1)	|= [(P<1>	v (P<1>	& Q)) SYN (P<1>	& (P<1>	v Q))]	[T1-16,R2-2]
2)	|= [(Pa	v (Pa	& Q)) SYN (Pa	& (Pa	v Q))]	[T1-16,R2-2]
3)	|= [(Q<2,b,1>	v (Q<2,b,1>	& Q) SYN (Q<2,b,1>	& (Q<2,b,1>	v Q)]	[1),R2-2]
4)	|= [(Q<2,b,a>	v (Q<2,b,a>	& Q) SYN (Q<2,b,a>	& (Q<2,b,a>	v Q)]	[2),R2-2]

But this does not provide any basis for a general statement.

Thus, the principle that logical properties and relationships between predicates are preserved in expressions which instantiate those predicates (i.e., which put some particular singular terms for all occurrences of a given argument-position-holder in the predicate), is the basic principle on which logical generalization rests.

In the next chapter it is shown how Instantiation, with the definitions of the quantifiers, previous theorems, and rules, leads to the derivation of quantificational theorems.

2.5 Predicate Schemata and Applied Logic

By "Applied Logic" I mean the use of principles of pure formal logic in dealing with actual predicates and sentences, rather than only predicate schemata. Large portions of applied logic are still very abstract, but they are based on actual abstract predicates like "<1>is equal to<2>", "<1>is a member of <2>". In this group are included "formal sciences" such as mathematics and geometry, but also an infinite number of possible disciplines based on particular abstract predicates or clusters of such predicates. But beyond this are still more applications of logic which deal with the many predicates used in everyday affairs. We will mention the ways in which logical analysis deals with these expressions in preparation for logical deductions and reasoning.

2.51 Predicate Schemata in the Formal Sciences

It is customary to speak of mathematics and geometry, along with logic, as "formal sciences". Pure Formal Logic is the formal science *par excellence*, since it is based only on the meanings of syncategorematic words including 'and', 'or', 'all', 'some', 'not', 'if...then', and terms defined by them. The subject terms of its theorems (the wffs it talks about in its theorems) do not have any actual predicates; they have only predicate schemata, i.e., the forms of predicates. Nor do they have any actual names or singular terms. They have only placeholders for singular terms—argument- position-holders, individual constants and variables. The predicate terms in the theorems of pure formal logic 'VALID', 'INC', 'TAUT', 'SYN', 'CONT', etc.,—are the "predicates of logic" which stand for the properties and relationships that it is the business of formal logic to investigate.

The other formal sciences all presuppose the principles of pure formal logic, but their special theorems are developed from the meanings of singular terms and actual predicates which it is their particular business to investigate. Their theorems contain both categorical statements of fact based on meanings, and conditional statements which are validly deduced from those meanings.

In A-logic the predicates ‘SYN’ and ‘CONT’ stand for a sub-species of semantic synonymy and containment relations. SYN-statements are the Syn-statements in which the synonymy expressed is based solely on the syntax and meanings of the “logical words”. Syn-statements in formal sciences are based on the meanings of actual predicates and the logical structure of their meanings.

Mathematics as a formal science needs names for numbers and various kinds of predicates including ‘<1>is equal to<1>’ (which it is its primary business to investigate) and ‘<1>is odd’ and ‘<1>is the square of<2>’ and many others definable from a small base.[7] Based on the meaning of ‘<1>is odd’ one can derive categorical statements like ‘213 is odd’ and valid conditional statements like ‘If <1>is odd and <2>is the square of <1> then <2>is odd’ and many much more complicated truths of mathemtics.

Geometry needs actual predicates to distinguish points, lines, angles, planes, and different kinds of plane or solid figures, etc which are the subjects of its investigations, as well as basic predicates for ‘<1>equals<2>’, ‘<1>is congruent with<2>’, ’<1>is similar to<2>’. The theorems of these formal sciences must follow logically from the meanings of the basic terms and predicates used. Some are basic theorems of equality (of length or area or volumne), congruence or similarity (of shape). Others based on logical containments in their definitions express relationships between properties, such as “If <1>is an equilateral triangle, then <1>is an equiangular triangle”.

2.52 Formal Theories of Particular Predicates

Formal theories include many more disciplines than those customarily spoken of as formal sciences. A formal theory of identity can be developed from the meaning of the predicate ‘<1>is identical to<2>’; a formal theory of classes, from the meaning of ‘<1>is a member of<2>’, a formal theory of biological relationships among bisexual mammals can be developed from formal characteristics of the meanings of ‘<1>is a parent of<2>’ and ‘<1>is male<2>’ and ‘<1>is female’. Given any specific predicate, it is possible to develop at least a partial formal logic of that predicate. We may speak not only of the logic of mathematics, and the logic of physics, but also of the logic of any polyadic predicate, or any predicate definable with a polyadic predicate.

The difference between pure formal logic and the formal logic of specific predicates is that since pure formal logic has no actual predicates, no atomic predicate schema semantically contains any of its modes or any other atomic predicate schema. In applied formal logic—pure formal logic applied to the logic of some actual predicate or predicates—actual predicates significantly do, or do not, semantically contain one or more of their modes.

The predicate “<1>equals<2>” Contains all its abstract modes, i.e., the meaning of “equals” is such that “<1>equals<2>” is referentially Synonymous with “<1>equals<2> and <1>equals <1> and <2>equals<1> and <2>equals<2>”; thus ‘<1>equals<2>’ Cont ‘<1>equals<1>’. Expressing this formally, with ‘E’ for ‘equals’ and ‘Syn’ meaning semantic but not logical synonymy,

---

7. A typical axiomatization of elementary number theory begins with at least one singular term (for the number 1 or for zero), two dyadic predicates ‘<1>is equal to<2>’ and ‘<1>is the successor of<2>’, and two triadic predicates, ‘<1>is the product of<2>and<3>’, and ‘<1> is the sum of <2>and<3>’, although the last three are typically expressed as functions with the second and third terms taken as the independent variables.

[E<1,2> Syn (E<1,2> & E<2,1>, & E<1,1>, & E<2,2>)] (‘E’ for ‘is equal to’)
hence, by Df.’Cont’, [E<1,2> Cont E<2,1>], (i.e., E is symmetric)
and [E<1,2> Cont E<1,1>]
and [E<1,2> Cont E<2,2>] } (i.e., E is reflexive)
and [(E<1,2> & E<2,c) Cont E<1,c] (i.e., E is transitive)

Every instantiation of this synonymy statement will be true on the left side when and only when it is true on the right side. This is referential synonymy but not logical synonymy, since it is not based only on the meanings of the syncategorematic operators.

By contrast, “<1> is greater than <2>” contains none of its proper modes, though it is transitive.

By U-SUB on T1-11, [G<1,2> SYN G<1,2>] (‘G’ for ‘is greater than’)
but not [G<1,2> Cont G<2,1>] (i.e., ‘G’is not symmetric)
and not [G<1,2> Cont G<1,1>]
and not [G<1,2> Cont G<2,2>] } (i.e., ‘G’ is not reflexive)
but [(G<1,2> & G<2,a>) Cont G<1,a>] (i.e., ‘G’ is transitive)

And the usual sense of ‘<1> is a mother of <2>’, like ‘<1> loves <2>’, contains none of its modes and is not transitive. Letting ‘M’ stand for ‘is the mother of’ ‘, by T1-11 and U-SUB,

[M<1,2> SYN M<1,2>] by logic,
but not [M<a,b> Cont M<b,a>], (i.e., ‘M’is not symmetric)
and not [M<a,b> Cont M<a,a>],
and not [M<a,b> Cont M<b,b>], } (i.e., ‘M’ is not reflexive)
and not |= [M<a,b> & M<b,c>) Cont M<a,c>] (i.e., ‘M’ is not transitive)

A necessary condition for the truth of a SYN-Statement is that the predicates or predicate schemata, on both sides of ‘SYN’ must have all and only the same elementary predicates or elementary predicate schemata. With extra-logical Syn-statements it is different. It is not necessary that the two sides of the Syn-statement have the same predicates. The most useful Syn-statements have different elementary predicates on either side.The definition of an actual predicate usually has a simple predicate as definiendum and a compound predicate composed of other elementary predicates as the definiens. Such Syn-statements have different predicate letters on the two sides of ‘syn’. But though the predicates on either side may differ, the set of numerals for argument-position-holders, and individual constants, must be same on both sides since two expressions can not be synonymous if they talk about different things.[8]

Unlike SYN-theorems, the synonymies of extra-logical synonymy statements, are not preserved by substituting in accordance with U-SUB. Given the meaning of ‘G<1,2>’ as ‘<1> is greater than <2>’, substituting ‘G<3,2>’ for ‘G<2,3>’ does not preserve the special synonymy in

8. Note that definitions of the logical predicates themselves, when they occur, represent non-logical synonymies, e.g., [P CONT Q] Syn [P SYN (P&Q)]. The axioms of A-logic display properties of the meanings of the logical constants. E.g., Ax.1-03 [(P&Q) SYN (Q&P)] shows that ‘&’ is commutative; Ax.1-05 shows it is associative. These are primarily commitments on how we will use ‘&’. Whether they are acceptable or not should not be viewed as a question of absolute truth, but as one of the usefulness of such principles as parts of a total system of logic.These are useful commitments.

[G<1,2> & G<2,3>) Syn (G<1,2> & G<2,3> & G<1,3>)]

The result of such a substitution would have the following instantiations which clearly can not be valid or, in the second case, true:

If a is greater than b and c is greater than b, then a is greater than c:
If 6 is greater than 4 and 7 is greater than 4, then 6 is greater than 7

This shows that the rule of U-SUB in pure formal A-logic is designed to preserve the truth of statements about logical properties and relationships. It is not designed to preserve truth in general. By contrast, if we have a true SYN-statement, then uniform substitution of any wff for any atomic component, preserves logical synonmy in the result. For example,

[G<1,2> & (G<3,2> v G<1,3>)) SYN ((G<1,2> & G<3,2>) v (G<1,3> & G<3,2>))]

is gotten by INST and simple U-SUB(R1-2) from Ax.1-08. Logical synonymy is preserved if any wff whatever replaces ‘G<3,2>’ at all occurrences, e.g., as in the following:

[G<1,2> & (G<3,2> v G<1,3>)) SYN ((G<1,2> & G<3,2>) v (G<1,3> & G<3,2>))]
[G<1,2> & (G<2,3> v G<1,3>)) SYN ((G<1,2> & G<2,3>) v (G<1,3> & G<2,3>))]
[G<1,2> & ((P&Q) v G<1,3>)) SYN ((G<1,2> & (P&Q)) v (G<1,3> & (P&Q)))]

2.53 Formal Properties of Predicates

The meanings assigned to predicates have, or fail to have, certain formal properties. They may be reflexive, or symmetric, or transitive or connected, or have other formal properties not yet named. Abstract predicate schemata can be used to define formal properties or relations in meanings of actual predicates. These properties can be defined in terms of synonymies and/or containments of their modes. For example,

‘Reflexive[P]’ Syn$_{df}$ ‘[(P<1,2> Syn (P<1,2> & P<1,1> & P<2,2>))]’
‘Reflexive[P]’ Syn$_{df}$ ‘[P<1,2> Cont (P<1,1> & P<2,2>)]’

‘Symmetric[P]’ Syn$_{df}$ ‘[P<1,2> Syn (P<1,2> & P<2,1>)]’
‘Symmetric[P]’ Syn$_{df}$ ‘[P<1,2> Cont P<2,1>)]’

‘Transitive[P]’ Syn$_{df}$ ‘[(P<1,2> & P<2,3>) Syn (P<1,2> & P<2,3> & P<1,3>)]’
‘Transitive[P]’ Syn$_{df}$ ‘[(P<1,2> & P<2,3>) Cont P<1,3>]’

These properties and others hold not only of extra-logical predicates, as in illustrations above, but of predicates used in logic and semantics. Among the predicates used in semantics,

‘Syn’ is transitive, symmetric and reflexive
‘Cont’ is transitive, non-symmetric and non-reflexive.
Hence, [(P Syn Q) Cont (Q Syn P)]
[(P Syn Q) Cont (P Syn P)]
[(P Syn Q) Cont (P Cont Q)]
[(P Syn Q & Q Syn R) Cont (P Syn R)]
[(P Cont Q & Q Cont R) Cont (P Cont R)]

Since SYN (logical synonymy) and CONT (logical containment) are sub-species of semantic synonymy and containment, they have these same properties:

‘SYN’ is transitive, symmetric and reflexive
‘CONT’ is transitive, non-symmetric and non-reflexive.
Hence, [(P SYN Q) Cont (Q SYN P)]
[(P SYN Q) Cont (P SYN P)]
[(P SYN Q) Cont (P CONT Q)]
[(P SYN Q & Q SYN R) Cont (P SYN R)]
[(P CONT Q & Q CONT R) Cont (P CONT R)]

Definitions of other formal properties that may or may not belong to actual predicates will be added as quantifiers, negation and conditionals are introduced in subsequent chapters.

2.54 The Role of Formal Properties of Predicates in Valid Arguments

In applied logic, Syn-statements asserting substantive, non-logical synonymies serve as **a priori** premisses, whether stated explicitly or not. Consider the following argument,

1) $(4^4 > 3^5)$ “4^4 is greater than 3^5” $G<a,b>$
2) $(3^5 > 6^3)$ “3^5 is greater than 6^3” $G<b,c>$
Therefore, $(4^4 > 6^3)$ “4^4 is greater than 6^3” .: $G<a,c>$

This is immediately accepted as valid—an argument in which the conclusion follows from the reasons. We know that if the premisses are true, then the conclusion must be true. We know this even if we don’t know whether the premisses are true and the argument sound because we haven’t calculated the values of 3^5, 4^4 and 6^3. But it is not valid by formal logic. There is no theorem of logic that says $P_1<1,3>$ follows logically from $(P_1<1,2>$ and $P_1\ <2,3>$. If there were such a theorem, then the following argument would be valid:

1) Lady Elizabeth Bowes-Lyon is the mother Queen Elizabeth II
2) Queen Elizabeth II is the mother of Prince Charles
Therefore, Lady Elizabeth Bowes-Lyon is the mother of Prince Charles.

This can not be valid since the premisses are true and conclusion is false.Thus the validity of the argument to ($4^4$is greater than 6^3) does not come from pure formal logic; it comes from the meaning of “is greater than” which is an extra-logical predicate.

The binary predicate, ‘G’, for “$<1>$is greater than$<2>$” is a transitive relation. This is expressed in A-logic by the statement that the meaning of ‘ is greater than’ is such that two of modes of $G<1,2>$ contain a third mode: $|(G<1,2>\ \&\ G<2,3>)\ \mathrm{Syn}\ (G<1,2>\ \&\ G<2,3>\ \&\ G<1,3>)]$,
hence, by Df’Cont’, $[(G<1,2>\ \&\ G<2,3>)\ \mathrm{Cont}\ G<1,3>)]$[9]

9. or, $\models$ $[(a>b\ \&\ b>c)$ Cont $a>c]$.The symbolism of formal logic shows how such premisses can be accurately schematized, but from pure formal logic one can not derive the acceptability of any substantive definition or synonymy-statement involving an actual non-logical predicate.

In "[G<1,2> & G<2,3>) Syn (G<1,2> & G<2,3> & G<1,3>)]" both sides of the Syn-statement have occurrences of only the same abstract predicate, G<1,2>, and talk about all and only the same entities {1,2,3}. This makes it a statement solely about a logical structure of the meaning of '<1>is greater tha<2>'. The synonymy is not based on the meanings of the operators '&' and 'v', for, as a synonymy of instantiated conjunctions it has the form '((P&Q) Syn (P&Q&R)', which can not be a purely logical SYN-statement since one side contains a component which the other side lacks..Thus if it is to be considered valid, the premisses of this argument must include not only the factual statements,

1) $G(4^4,3^5)$	"4^4 is greater than 3^5" or "$4^4 > 3^5$"	[Premiss]
2) $G(3^5,6^3)$	"3^5 is greater than 6^3" or "$3^5 > 6^3$"	[Premiss]

but also implicit premisses based on the meaning of "is greater than"; namely that "If G<1,2> and G<2,3> then G<1,3>" is a valid conditional. In A-logic this conditional is grounded in statements of synonymy and containment with respect to being greater-than: Thus the full valid argument includes

3) [G<1,2> & G<2,3>) Syn (G<1,2> & G<2,3> & G<1,3>)]	[Premiss from logic of G]
4) [G<1,2> & G<2,3>) Cont G<1,3>)]	[3),Df 'Cont']
5) [G<4^4,3^5 > & G<3^5,6^3>) Cont G<4^4,6^3>)]	[4),INST(thrice)]
6) Valid[If G<4^4,3^5 > & G<3^5,6^3>) then G<4^4,6^3>)]	[5), DR6-6a][10]
7) $G(4^4,6^{3)}$	[1),2), MP]

Though steps 3) to 7) were not stated initially, they are in accord with what is needed to make the argument valid and sound, and may perhaps account for our initial acceptance of its validity.

In the greatest examples of applied logic—in mathematics, in natural science, in law, and in common sense—premisses must frequently include implicit or explicit analytic assertions of synonymy or containment based on meanings assigned to extra-logical expressions of the particular discipline involved. What makes a deduction logically valid is the use of logical rules of inference—SynSUB and U-SUB and Instantiation—to get from such premisses to a conclusion. In the logic of family relationships, the following Syn-statements are acceptable for use with SynSUB in many ordinary contexts:

'<1>is a brother of<2>' Syn '(<1>is a sibling of<2> & <1>is male)'
'<1>is a spouse of<2>' Syn '<2>is a spouse of <1>'
'<1>is the mother of<2>' Syn '(<1>is a parent of<2> & <1>is female)'

These Syn-statements asserting the sameness of meanings of two predicates having the logical forms, respectively,

[P_1<1,2> Syn (P_2<1,2> & P_3<1>)]
[P_4<1,2> Syn P_5<2,1>]
[P_6<1,2> Syn (P_5<1,2> & P_6<1>)]

None of these are synonymies of pure formal A-logic, for they have different elementary predicate schemata on either side of 'Syn' and they cannot be justifed by reference to meanings of syncategorematic

10. See Chapter 6, Section 6.342, page 305 for DR6-6a.

expressions alone. But they express the kind of substantive, non-logical synonymies which abound in applied logic. Arguments like

> Jean is the brother of Lauren
> therefore, Jean is a male.
>
> Lee is Jean's child and Jean is the brother of Lauren.
> therefore, Jean is Lee's father.

are shown valid only by adding non-logical synonymies and containments, like those above as explicit or implicit premisses.

2.55 Abstract Predicates and Modes of Predicates in Logical Analysis

In the 2nd and 3rd editions of *Methods of Logic* Quine introduced the notion of predicate as a supplementary device to keep uniform substitution under control when we get to polyadic quantification theory.[11] He wanted a rule which would preserve validity if all occurrences of atomic components with the same predicate letter in a quantificational theorem were replaced by some more complex component.The rule used in A-logic preserves validity by essentially the same device.

However, predicates have a much broader role in A-logic than Quine would grant them. Besides replacing sentences as the ultimate subjects of logical scrutiny, the logical analysis which separates predicates from the expressions denoting individual entities to which predicates apply, is an important preparatory step for determining what a sentence entails. For entailment is based on containment.relations between predicates, not on the referents of singular terms.

Quine speaks as if meaning belongs primarily to sentences, including "open sentences" with free variables like 'Fx', 'Gxy', '(∃y)Gxy', '(∀x)Gxy ⊃ Gyy)' etc. Then he asks what term should be associated with the 'F', or 'G', standing alone. The results of efforts to answer this question are characterized as "awkward" and "fussy". For example, taking 'Fx' to mean

> 'x used to work for the man who murdered the second husband of x's youngest sister',

he asks what meaning to give to the predicate, 'F'. The answer he offered was "former employee own youngest sister's second husband's murderer" (so that 'Fx' means 'x is a former employee of own youngest sister's second husband's murderer"). Such an expression is indeed awkward, though it seems to capture the meaning sought in an awkward way. It is of no help in determining which expressions could be put for all occurrences of 'Fx', 'Fy', and 'Fz' when they all occur in the same theorem so as to preserve theoremhood. In place of this term he proposed "predicates" of the sort we have been using. But he describes the circled numerals (argument-position-holders) and thus the predicates as "simply as a supplementary device, more convenient and systematic that those existing in ordinary language, for abstracting complex terms out of complex sentences." Thus a predicate to associate with 'F' in the present example, is

> (i) '<1>used to work for the man who murdered the second husband of <1>'s youngest sister'

---

11. See Sections 23 and 25, *Methods of Logic*, 2nd Edition and Sections 26 and 27, 3rd Edition. Predicates were abandoned in favor of abstracts in the 4th Edition, but the substitution procedure was formally the same.

And if the open sentence 'Gxy' means 'x used to work for the man who murdered y & y was the second husband of x's youngest sister', then the predicate for 'G' should be,

(ii) '<1>used to work for the man who murdered <2> & <2> was the second husband of <1>'s youngest sister'.

Quine tried to get at what meaning is common to the different occurrences of the same predicate letter in a complex quantification theorem. He wrote, "a certain correspondence must be preserved between the substitutes for 'Fx' and 'Fy'. What correspondence?". Then he points out that if 'F' stands for 'is proud of z', then '(∃x)(x is proud of z)', says the same thing about x as '(∃y)(y is proud of z)' says about y, but if 'F' stands for 'is proud of y' then '(∃x)(x is proud of y)' does not say the same thing about x that '(∃y)(y is proud of y)' says about y. How then do we make clear what the correspondence is?

In A-logic what is common is that they are both modes of the same abstract predicate. In some cases the modes are identical modes, but in others they are not. The common feature that they share is only that they are modes of the same predicate—the criterial meaning of '<1>is proud of <2>' will remains the same whether a singular term put for <2> happens to refer to the same entity as the term put for <1>, or not. Viewing abstract polyadic predicates as having criterial (intensional) meanings of their own, apart from their being applied to any individual or occurring in a sentence, is what distinguishes taking predicates seriously. This was the position that Quine, with his "extensionalist" leanings was not inclined to make.

It is interesting that Quine ascribed meaning, but not truth or falsity, to his open sentences; but had difficulty in ascribing meaning to his predicates. For A-logic this is backwards. Predicates, as Quine defined them in his earlier writings, are meaningful; open sentences are not.

Although Quine does not say so, presumably he would agree that (ii) is a synonym of (i), making 'Gxy' an expression which, with proper precautions, could be substituted for 'Fx' without affecting validity. The step from (i) to (ii) was a step by logical anlysis towards clarifying an awkward ordinary language expression.

We may grant the awkwardness of saying that the 'F' above means "former employee of own youngest sister's second husband's murderer." But this meaning is clarifed by the kind of logical analysis that Quine started in (i) and extended in (ii).The abstract predicate common to both is made clearer by a third and a fourth step of analyzing the logical structure of its meaning into a compound predicate with 4 dyadic predicates as components, namely by steps

(iii) '<1>used to work for<3> & <3>murdered<2> & <2>was the second husband of <1>'s youngest sister.'

and (iv) '<1>used to work for<3> & <3>murdered<2> & <2>was the second husband of<4> & <4>was the youngest sister of<1>.

The result in (iv), which has filtered out all relevant argument-position-holders for terms designating individual entities, is a single, complex, abstract predicate. As a whole it may be viewed as an 8-place, quadratic predicate of the form 'P_7<1,3,3,2,2,4,4,1>' which has been analysed into the logical structure composed of four conjoined dyadic predicates with cross-referenced subject terms, of the form '(P_3<1,3> & P_4<3,2> & P_5<2,4> & P_6<4,1>)'. Given this analysis, containment-theorems of A-logic yield a variety of predicates which are logically entailed from this conjunction. The account Quine gave of 'F<1>' in (i) is synonymous with his account of 'G<1,2>' in (ii), and both are synonymous with the more detailed analyses of (iii) and (iv).

We can, if we wish, define an actual 8-place, 4-adic predicate of the form $P_7<1,3,3,2,2,4,4,1>$ which is synonymous by definition to ($P_3<1,3>$ & $P_4<3,2>$ & $P_5<2,4>$ & $P_6<4,1>$). Let the initial monadic predicate which Quine said meant ' **f**ormer **e**mployee of **o**wn **y**oungest **s**ister's **s**econd **h**usband's **m**urderer) be abbreviated by the word "feoysshm", thus (i) is synonymous with "<1>is a feoysshm", which in turn is synonymous by definition with (iv). But abbreviating compound predicates by a single predicate term is not very helpful unless the complex of relationships and properties is one frequently met with, or of some special crucial interest which is frequently referred to. In this case the compound predicate is not common and would be of passing interest if it applied. Though such a definition is allowable, it is highly unlikely that it would ever gain wide enough usage to qualify it for a dictionary entry. Further, it adds nothing to understanding, for understanding of the awkward complex predicate is brought about by its logical analysis into a structure of simpler predicates. This logical analysis of the meaning of a complex non-logical predicate into a structure of simpler predicates '<1> used to work for<2>', '<1> murdered<2>', '<1>was the husband of<2>', and '<1> was the youngest sister of <2>' is what logical analysis facilitates and requires to do its work.

(i) is a mode of (iv) which centers on the relationships emanating from whatever occupies positions occupied by <1>. We can construct equally meaningful modes of <iv> by centering on whatever occupies <2>, <3> or <4>; thus we get from (iv):

<1>is a former employee of own youngest sister's second husband's murderer
<2>is the second husband of the youngest sister of a former employee of 2's own murderer
<3>is murderer of a former employee's youngest sister's second husband.
<4>is the youngest sister of a former employee of own second husband's murderer

All of these, like (i) are synonymous with and reducible to (iv).

The complex predicate schema, '($P_3<1,3>$ & $P_4<3,2>$ & P_5 $<2,4>$)' also fits "(<1>is a child of<3> & <3>is a sibling of<2> & <2>is a parent of <4>)" and this can profitably be abbreviated for the logic of blood relationships as a logical analysis and synonym of "<1>is a cousin of<4>". In this case a complex predicate has a useful abbreviation.

In mathematics, science, law, and elsewhere, many a predicate is built up as a compound combination of simpler predicates, and abbreviated with a single simple predicate as definiendum, then widely used in logical deductions and proofs based on the definiens.

In rational problem solving, it helps to begin with definitions of the terms which describe the nature of the things sought. The problem is to find facts or pre-existing principles which lead to a solution which fits the definition of the goal. In reasoning definitions are not absolute truths forced on us but conventions agreed upon because they fix the objective sought. In law, for example, bills about duties and responsibilities of members of a marriage, might begin with a definition—"for purposes of this law 'two persons a and b will be said to be married to each other iff and only if (i) one is a male and one is a female,…" or it might begin with, "for purposes of this law 'two persons a and b will be said to be married to each other iff and only if (i) they have been registered in some state or country as a married partners…". Clearly the definition of what will be meant by terms, fixes the subjects of the regulations which follow. And this is a function that the originators of bill intend to achieve. In rule books of various sports, the definition of 'the winner', and of the various events which will gain or lose points, fix certain things beyond discussion. After that who wins and who loses will depend not on changing the rules, but of which events fit the definitions of events connected to the objective of the game. The situation is not essentially different in science. The definitions of extra-logical predicates predicates fix the acceptable subject-matter and the structure of the principles and results which it is the objectives of the science to develop.

Chapter 3

Quantification

3.1 Quantification in Analytic Logic

In this chapter, the fragment of logic in Chapters 1 and 2 is expanded to include theorems of logical synonymy and logical containment between quantificational wffs. After preliminary remarks in Section 3.1, we discuss the syntactical and semantic features of quantificational schemata (Section 3.2), the axioms and rules of derivation by which quantificational theorems are derived from theorems of Chapters 1 and 2, (Section 3.3) and the the specific SYN- and CONT-theorems which are derivable in this stage of developing Analytic Logic (Section 3.4.).Finally we establish procedures for finding a synonymous prenex normal form of any Q-wff (Section 3.5). A Q-wff is any wff with a quantifier in it.

3.11 A Fundamentally Different Approach

In Analytic Logic, as in mathematical logic, expressions in which “all” and “some” occur are expressed symbolically with the help of the symbols ‘$(\forall x)$’ and ‘$(\exists x)$’. ‘$(\forall x)Px$’ is read, “For all x, x is P’ or “all things are P”, and ‘$(\exists x)Px$’ is read “For some x, x is P’ or “some things are P”.

Quantification theory is here treated as a theory of quantificational predicates; quantified wffs, abbreviated:“Q-wffs”, are treated as predicate schemata. There are four kinds of quantified predicate schemata or Q-wffs:

(i) Quantified schemata with individual variables and one or more argument-position-holders; e.g., the predicate ‘$<1>$is a father’ is used to mean “$<1>$is a parent of someone and $<1>$ is male”. In symbols this is a quantificational predicate: ‘$(\exists x)(<1>$ is a parent of x & $<1>$is male)’, which has the abstract form, ‘$(\exists x)(P_1<x,1> \;\&\; P_2<1>)$’.

(ii) Predicate schemata with bound variables, constants and argument-positions-holders, e.g., ‘$(\exists x)(P_3<1,x> \;\&\; P_1<x,a_1>)$’ for “$<1>$is the spouse of some child of a_1”.

(iii) Quantified wffs in which all argument positions are occupied either by individual constants or individual variables. These include sentence schemata which are treated as as “applied” or “saturated” predicate schemata. Thus pure predicate schemata, like ‘$P<1>$’ or ‘$(\exists x)R<x,1>$’, become applied or saturated predicate schemata when argument-position-holders are replaced by individual constants (by INST), as in ‘Pa_1’ or ‘$(\exists x)R<x,a_1>$’.

(iv) Completely quantified predicate schemata,with bound variables in every argument position e.g., '(∃x)Px' or '((∀x)(∃y)Rxy'.[1]

Sentential or propositional functions with unbound variables (e.g., 'Px' or '(∃y)Rxy') are not well-formed formulae in A-logic.[2] Operations involving "free" or "unbound" variables in some versions of M-logic, are taken over by operations with predicate schemata with argument-position-holders. Expressions with vacuous quantifiers, (e.g., '(∀x) P' or '(∀z)(∃y)Rxy') are also not well-formed formulae in A-logic. Individual variables only occur when bound to quantifers in wffs of A-logic.

The general idea behind this theory of quantification is that logical relationships belong to predicates, and are preserved when those predicates are applied to individuals. If a predicate schema, or a pair of predicate schemata, have a certain logical property or stand in a certain logical relationship, then every application of such schemata to individual entities will also have that logical property or stand in that relationship. If a logical property belongs to [P<1>], then it belongs to [Px] no matter what x may be.[3] This allows the transmission of logical properties and relations from unsaturated predicates to universal or existential quantifications of those predicates.

All SYN-theorems and CONT-theorems in this chapter are derivable by extension from the theory of synonymy and containment in Chapters 1 and 2 together with definitions of "all" and "some", as abbreviating special sorts of conjunctions and disjunctions. All theorems in this chapter are 'SYN'-for-'≡' or 'CONT'-for-'⊃' analogues of theorems of quantification theory in mathematical logic.[4]

Negation is not needed at the foundations of this quantification theory. All basic SYN-theorems and CONT-theorems are independent of negation. It is true that negation adds new logical schemata with different meanings and with its help many new definitions and SYN-theorems follow (see Chapter 4). But the basis for all SYN and CONT relations among quantified expressions are established in this chapter, prior to the introduction of negation. No additional SYN- or CONT-theorems of Quantification, except those gotten by uniform substitution of negative predicate schemata for simpler unnegated predicate schemata, emerge as the result of introducing negation. No theorem of this chapter needs modification or adjustment after negation is introduced. Thus this chapter presents a complete foundation, though negation-free, for the theory of logical synonymy and containment among quantified expressions, and the subsequent development of quantification theory rests on this foundation.

Without a negation operator, no proof in this chapter can contain truth-functional conditionals or biconditionals in premisses, rules of inference or conclusions. The use of the truth-functional conditional and the truth-functional version of *modus ponens* are not essential or required for proof in this or any subsequent chapter (though analogous rules are allowable for sub-classes of cases in Ch. 5 and thereafter). Due to the absence of negation and a truth-functional conditional some principles which have been

---

1. Nothing is lost if the reader prefers to think of the quantificational theorems in this chapter as statement schemata since the principles of quantification here are all expressed without argument-position-holders. By uniform substitution, however, any of these wffs can yield a theorem about quantificational predicates, cf. Section 3.14, Quantificational Predicates. E.g., '(∃x)(P3<1,x> & P1<x,a1>)' as an allowable U-SUB substitute for '(∃x)Px'.

2. Some logicians have treated sentential functions as predicates. Quine calls them "open sentences". But they are neither sentences nor predicates. We distinguish predicates from sentences, viewing the latter as fully applied or "saturated" predicates.

3. '(∀x)Px' is often read as "no matter what x may be, x is P".

4. By "a is a 'SYN'-for-'≡'-analogue of b" I mean a is a SYN-theorem of the form '|= [A SYN B]' and b is a theorem of Quine's of the form, '⊢ [A ≡ B]'. By "a is an 'CONT'-for-'⊃'-analogue of b" I mean that a is a CONT-theorem of the form '|= [A CONT B]' and b is a theorem of Quine's the form '⊢ [A ⊃ B]'. Adjustments are added later to handle Quine's special version of '⊢' in *Mathematical Logic*.

taken as axioms of quantification theory in M-logic do not appear in this chapter;[5] they only appear in Chapter 4 and 5, where they are instances of theorems in this chapter with negated expressions introduced by U-SUB, and truth-functional conditionals introduced as abbreviations.

Thus all theorems in this chapter are SYN-theorems or CONT-theorems. Quantified wffs represent special kinds of logical structures—structures defined only in terms of positions and cross-references among elementary formulae in components grouped by conjunction or disjunction.[6] Theorems of this chapter assert only that logical synonymy and/or containment holds between such conjunctive/disjunctive structures.

In this chapter, as in Chapter 1, each SYN-theorem or CONT-theorem purports only to be a true statement about a logical relationship between the meanings of two wffs. Nothing is here said about logical properties of wffs taken singly. Every Q-wff in this chapter, taken singly, is contingent; none by itself is necessarily true, false, always true, always false, tautologous, or inconsistent. And no inference from one theorem to the next (in this or the next chapter) represents a passage from the truth or theoremhood of a single Q-wff to the truth or theoremhood of another. We are concerned only with sameness of meaning, or containment of meaning. Here the derivation of one theorem from others is based only on sameness of meanings in two different structures regardless of their truth.

Thus the foundation of quantification theory provided in this chapter is fundamentally different, semantically and syntactically, from foundations of quantification theory in mathematical logic, though for every SYN theorem [P SYN Q] in this chapter there is an analoque [P ≡ Q] which is a theorem of mathematical logic, and for every CONT-theorem [P CONT Q] in this chapter there is an analogue [P ⊃ Q] which is a theorem of mathematical logic.[7]

3.12 To be Compared to an Axiomatization by Quine

SYN- and CONT-theorems in this chapter will be related one-by-one to analogous theorems of quantification theory presented in Chapter 2, "Quantification", of W. V. Quine's *Mathematical Logic,* 1981 ed. (abbr. 'ML'). This should help to clarify and compare precise similarities and differences between quantification theory in M-logic and in A-logic.

The system in *Mathematical Logic* is only one of several methods Quine used to present quantification theory, and many other systems might be used to stand for standard quantification theory in general. Quine's version is chosen partly because of lessons that are learned by contrasting it with A-logic, and partly because it is elegant, concise and rigorous. It covers a larger than usual array of theorems which express basic relationships between different quantified and non-quantified forms of statements. It is complete with respect to the semantic theory of mathematical logic. Every statement whose denial would be inconsistent by virtue of the logical structure of negation, conjunction, disjunction and 'all' or 'some' can be established as a theorem of logic by derivation from Quine's axioms and his rule of inference.

---

5. E.g., Quine's *101 (∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx) in *Mathematical Logic*.

6. With the introduction of a negation-free, non-truth-functional "C-conditional", in Chapter 6, the concept of "logical structure" is expanded to include structures built up by conditional connectives as well as conjunctive or disjunctive connectives. As a result, certain logical properties of non-truth-functional conditionals (based on containment, not tautology) are re-introduced, and a genuine principle of Modus Ponens is formulated, read back into the logic presented in these earlier chapters, and used to formalize its rules of inference with "if…then"s construed as C-conditionals.

7. In the Chapter 5, after having introduced negation in Chapter 4, we define a fragment of analytic logic which yields all and only the theorems of M-logic (including its Quantification Theory) construed as ascribing tautologousness to a wff, while excluding M-logic's counter-intuitive rules of valid inference.

The axiomatization of quantification theory in Analytic Logic differs in various ways from Quine's system. Some differences are superficial: the primitive symbols employed are different—Quine uses only individual variables while A-logic uses individual constants and argument-position-holders also; the formation rules which exclude sentential functions and vacuous quantifications as wffs though Quine includes them; the meaning Quine gives to the symbol '$\vdash$' differs from the meaning given to '$\models$' in A-logic. Despite reasons used to justify these differences, the set of significant quantificational theorems is the same as in all axiomatizations of mathematical logic. The only substantial differences between standard quantification theory and the fragment of Analytic Logic which yields all of its theorems in Chapter 5 lie in interpretations and Rules of Inference. The primitive rule of inference in ML is the truth-functional rule of detachment. No derivation in this chapter, or Chapter 4, uses, or can use, this rule either implicitly or explicitly, and formal Analytic Logic never requires it.[8]

In this and the next two chapters, it it is shown that none of these differences affect the thesis that all and only the standard accepted theorems of quantification theory are derivable within a circumscribed fragment of Analytic Logic. However, the rules of inference are not the same and the differences there are significant. Analytic Logic ties its rules and theorems to meaning relationships, whereas the axiomatic system in ML, like all systems of standard logic based on truth-values and truth-functions, includes as rules of valid inference, derivation rules which ignore the meaning relationships on which A-logic rests.[9]

In this chapter the foundation for such proofs is provided by proving analogues of twenty-five ML theorems. With the introduction of negation in Chapter 4, thirteen more AL-analogues are established by definitional substitution thus accounting for 38 of the 56 "theorems" listed in Chapter 2 of ML. The rest are accounted for in Chapter 5.

In Chapter 5 Quine's theorems are derived from the AL-analogues by the addition of derived rules permitting passage from $\models$ [A CONT B] to INC[A & $\sim$B], and thus to TAUT[A $\supset$ B]. Semantically, this says that if A logically contains B, then A and not-B are inconsistent and the denial of [A & $\sim$B) is tautologous. Thus containment provides the ground on which assertions of truth-functional tautologies are based. The converse relation does not hold, but the other tautologies which have no AL-analogues are all derivable using SynSUB, the substitution of logical synonyms. In Chapter 5, I will show that all theorems of first order logic in Quine can be derived independently of his semantic theory without using rules of truth-functional inference.[10]

---

8. A properly restricted version of truth-functional detachment rule is cited in Chapter 5 as a reliable device for moving from two tautologies (denials of inconsistency) to a third one. This is all that standard logic requires and all that Quine uses it for, in *Mathematical Logic*. But this makes it a special rather than a general rule of logical inference. These comments will be extended in pages that follow.

9. What I call ML's theorems, or theorem schemata, are there called "metatheorems". Arguments can be made for that, but here I shall skirt the issue, noting that a metatheorem is a kind of theorem.The theorems of ML are not actually derived in this chapter; they will be derived in Chapter 5 independently of ML *104, the truth-functional version of *modus ponens*.

10. To be independent of truth-functional rules is not to conflict with them. Truth-functional rules can be recognized as useful devices for deriving all and only standard theorems, without being accepted as rules of logical inference. Two ways of getting the same results can differ without being either inconsistent with each other, or equivalent.

Of 56 metatheorems in Quine's Ch2, 10 are rules of inference and 46 are theorem schemata. Of these 46, 16 are conditional theorems, 11 of which are conditioned on variables not being free in some component, and 5 of which are conditioned on other states of bondage or freedom of variables in the wff. All are accounted for in Ch. 5.

3.13 Relation of This Chapter to Later Chapters

This chapter provides the basis for the major results in the chapters which follow, namely:

1) the conclusion in Chapter 4 that the axiomatic system of logic in that chapter is sound and complete with respect to relations of logical synonymy and containment between pairs of quantified wffs of mathematical logic, and,
2) the conclusion in Chapter 5 that all and only theorems (vs principles of inference) of mathematical logic can be derived, using U-SUB, Double Negation and DeMorgan theorems, from the fragment of Analytic Logic in the present chapter which deals with logical synonymy and containment among quantified wffs.
3) the conclusion in Chapters 6 and 8, that SYN and CONT-theorems of this chapter and Chapter 4 provide the basic for the *validity* of C-conditional theorem, and
4) the conclusion in Chapters 7, 8 and 9, that SYN and CONT-theorems of this chapter and Chapter 4 provides the basis for valid theorems of truth-logic which help solve the problems which have confronted Mathematical logic with respect to the logic of empirical sciences.
5) Part of the argument in Chapter 10, that Analytic Logic solves certain problems which confront M-logic, is based on the demonstration in this chapter that Analytic Logic does not need the rules of inference which lead to "valid non-sequiturs" and other problems in the quantification theory of mathematical logic.

3.2 Well-formed Schemata of Negation-Free Quantifications

The following four sub-sections are intended to clarify the meanings of quantificational wffs by: (1) defining well-formed quantificational predicate schemata (Section 3.21), thus expanding the schematic language of Chapters 1 and 2, (2) explicating the semantic meaning of quantificational wffs (Section 3.22), (3) explaining how the concepts of logical synonymy and containment between quantificational schemata are grounded in the concepts in Chapters 1 and 2 (Section 3.23), and (4) more on how quantified wffs are interpretable as quantificational predicates (Section 3.24).

3.21 The Language of Negation-Free Quantificational wffs

The formal language still lacks negation, but all additions from the last chapter remain, especially argument-position-holders, individual constants, and predicate schemata with predicate letters prefixed to ordered n-tuples of subject terms like '<a,1,2,b>'. Quantifiers and individual variables are added to the primitive symbols. The expanded language is defined as follows:

<u>I. Primitive symbols:</u>

(i) Grouping devices: ( ) < >

(ii) Connectives: & v ∀ ∃ **(∀, ∃ Added)**

(iii) {PL}—Predicate Letters: P_1, P_2, P_3,..., P_i,...

(iv) {APH}—Argument-position-holders: **1,2,3,...,i,...**

(v) {IC}—Individual Constants: $\mathbf{a_1,a_2,a_3,\ldots,a_i,\ldots}$

(vi) {IV}—Individual Variables: $\mathbf{x_1,x_2,x_3,\ldots,x_i,\ldots}$ **(Added)**

[(iv)–(vi)] {ST} = Subject terms: t_1,...

II. Rules of Formation:

FR3-1. If $P_i \in \{PL\}$, P_i is a wff

FR3-2. If P_i and P_j are wffs, $[P_i \& P_j]$ and $[P_i \text{ v } P_j]$ are wffs.

FR3-3. If $P_i \in \{PL\}$ and $\{t_1,\ldots,t_n\} \in \{\{APH\} \cup \{IC\}\}$ then $P_i<t_1,t_2,\ldots,t_n>$ is a wff,

FR3-4. If $P_i<t_1,\ldots,t_n>$ is a wff and t_i {APH}, then $(\forall x)P_i<t_1,\ldots,t_n>t_i/x$ and $(\exists x)P_i<t_1,\ldots,t_n>t_i/x$ are wffs. (Added)

III. Definitions:

D1a. $[(P\&Q\&R) \text{ SYN}_{df} (P\&(Q\&R))]$

D1b. $[(PvQvR) \text{ SYN}_{df} (Pv(QvR))]$

Df '∀'. $[(\forall x)Px \text{ SYN}_{df} (Pa_1 \& Pa_2 \& \ldots \& Pa_n)]$ (Added)

Df '∃'. $[(\exists x)Px \text{ SYN}_{df} (Pa_1 \text{ v } Pa_2 \text{ v } \ldots \text{ v } Pa_n)]$ (Added)

(As before, 'P', 'Q', 'R', 'S' abbreviate 'P_1', 'P_2', 'P_3', 'P_4'; 'a', 'b', 'c', 'd' abbreviate 'a_1', 'a_2', 'a_3', 'a_4' and 'x', 'y', 'z', 'w' abbreviate 'x_1', 'x_2', 'x_3', 'x_4'.)

The Rules of Formation differ from Quine's in excluding "open sentences" and vacuous quantifiers, neither of which is essential for first order logic, though sometimes convenient.[11] These omissions are based on the principle that symbols which can not be correlated with clear and unambiguous distinctions in meaning should not be allowed in a well-formed formula. The most significant additions, for proofs of theorems, are the definitions of '(∀x)' and '(∃x)' as abbreviation symbols for conjunctions or disjunctions of instantiations of the matrix of the quantification.

The following differences between Quine and this book may be noted:

I. Primitive Symbols

1) Quine's primitive symbols include only individual variables {IV}, whereas A-logic includes individual constants {IC}, argument-position-holders {APH},
2) Quine uses Greek letters as placeholders for "sentences" ("open" and "closed"), whereas A-logic uses predicate letters as placeholders for predicates ("saturated" or not);
3) Quine's logical constants are only '|', '(x)' and '∈', whereas A-logic has only conjunction and disjunction at this point, with '(∀x)' and '(∃x)' used in representing special kinds of conjunction and disjunction.

II. Rules of Formation

1) Quine allows expressions with vacuous quantifiers as wffs; these expressions are not wffs for A-logic. Vacuous quantifiers are meaningless symbols; they don't say anything about anything.
2) Quine allows expressions with unbound variables, "open sentences", as wffs; these are not wffs of A-logic. They are ambiguous in meaning standing alone; only when bound to quantifiers do they provide definite meanings which can be true or false of objects.

---

11. The departure from Quine in formation rules, and in Quine's special use of '⊢' makes some translations from Quine to this system more difficult, but does not affect any essential properties or relationships of Quine's theorems of quantification.

III. The Assertion Sign

Quine uses '⊢ ϕ' for "the universal closure of ϕ is a theorem"; I use '⊨' in the customary sense of "It is true in this system of logic that…".

Elaborating on these comments, according to Formation Rule FR3-4, no expressions which contain a free variable, like 'P_i<y,z>' or 'P_i<x,1>', will be wffs, for variables must be introduced together with a quantifier that binds them. Such expressions are well-formed parts of a wff only if all their variables lie in the scope of some quantifier. By themselves they are neither true nor false and in any domain their meaning is ambiguous since it is not clear whether they are to be expanded conjunctively or disjunctively. Semantically they neither represent predicates with a clear criterial meaning, nor statements with a clear referential meaning. In A-logic all wffs are closed quantificationally; i.e., no sentential or predicate functions or 'open sentences' are wffs.

By the same formation rule, FR3-4, vacuous quantifiers (quantifiers which do not have any occurrences of their variable in their scope) are not well-formed parts of wffs. To introduce a quantifier that does not bind any variable is to add a meaningless element into an expression. For example, 'No matter what x may be, that John loves Mary' and 'There is some x such that John loves Mary' are referentially synonymous with 'John loves Mary', as well as with each other. The first two locutions should be eliminated in favor the last one, which is concise with no meaningless additions.

This does not mean that every thing in the scope of a quantifier must predicate something of its variable. Although '(∃x)P_1' is not a wff, '(∃x)(P_1 & P_2<x>)' is. Again, "Whoever is a brother of anybody is a male", symbolized by '(∀x)(∀y)(If Bxy then Mx)' in which 'x is a male', symbolized by 'Mx', has no occurrence of 'y' though it lies in the scope of '(∀y)'. On the other hand '(∀y)' is not vacuous in this case since 'y' occurs in 'Bxy'.[12] The "Rules of Passage" in theorems T3-17 to T3-20, which show that how a wff can occur either inside or outside the scope of a quantifier without change of referential meaning, do not need conditional clauses like "If x is not free in P…", for P will not be a wff if it has an unbound ("free") variable in it. This deviation from Quine's system make it unnecessary to add the restrictions which Quine must place on his more liberal substitution rules in order to avoid passing from theorems to non-theorems.

Additions to the symbolic language by Rule FR3-4 are supplemented in Section 3.3 by the convenient notational device, the Rule of Alphabetic Variance, "If P and Q are alphabetic variants, ⊨ [P] iff ⊨ [Q]".

3.22 Conjunctive and Disjunctive Quantifiers

We next examine in more detail the additions to the symbolic language resulting from the new primitive symbols, quantifiers and individual variables, the new formation rule FR3-4, and the new definitions Df '∀' and Df '∃'.

The primitive symbols '∀' and '∃', are symbols which occur in the Quantifiers (i) '(∀x)' for "For all x" and (ii) (∀x)' for "for some x".

The expression '(∀x)', usually called the 'universal quantifier', will be called henceforth the **conjunctive quantifier**, because it is always interpretable as abbreviating a conjunction which has the same abstract predicate schema for all of its conjuncts although each conjunct applies the predicate to a possibly different individual or n-tuple of individuals. Each such conjunct is formed by replacing every occurrence

12. Actually, the "rules of passage" in theorems T3-17 to T3-20 show that a well-formed part of a wff can occur either inside or outside the scope of a quantifier if it has no occurrence of that quantifier's variable without change of meaning.

of the quantified variable in the over-all wff which lies in the scope of the quantifier by an individual constant standing for a distinct individual member of the domain. Such quantifiers mean "all members of what ever domain is being talked about". They do not specify the number of members in the domain and the domain talked about does not necessarily have as its range all possible individuals. It can be any domain containing a finite or indefinite number and/or kind of individuals. The subscripted numeral 'n' in the definiendum may be viewed as standing indefinitely for any finite positive integer that might be chosen.

Using subscripted quantifiers, we can define certain quantificational expressions as abbreviations. Letting the subscript 'k' stand for the number of members in a domain of reference, we define quantified wffs in domains with a specified number, k, of members as follows:

Df '$\forall_k$'. [$(\forall_k x)$Px SYN_{df} (Pa_1 & Pa_2 & … & Pa_k)]
Df '$\forall_k$'. [$(\exists_k x)$Px SYN_{df} (Pa_1 v Pa_2 v … v Pa_k)]

The right-hand side is always a wff as defined in Chapter 2. The left hand side simply abbreviates the right-hand side. For example, if the domain specified is a three member domain, $\{a_1,a_2,a_3\}$, then for that domain,

'$(\forall_3 x)$(Px v Qx)' abbreviates '((Pa_1 v Qa_1) & (Pa_2 v Qa_2) & (Pa_3 v Qa_3))'

and for the 4-membered domain, $\{a_1,a_2,a_3,a_4\}$,

'$(\forall_4 x)$(Px v Qx)' abbreviates '((Pa_1 v Qa_1) & (Pa_2 v Qa_2) & (Pa_3 v Qa_3) & (Pa_4 v Qa_4))'

Each conjunct in the wff which is abbreviated is an instantiation of '$(\forall x)$(Px v Qx)'. The conjunction as a whole is called "a Boolean expansion of '$(\forall x)$(Px v Qx)' in a domain of k". In the examples above k=3 and k=4. In our metalanguage we use 'B_k[P]', as in 'B_k[$(\forall x)$(Px v Qx)]', to stand for 'the Boolean expansion of P in a domain of k'. Alternatively we can use 'B[$(\forall_k x)$Px]' for 'B_k[$(\forall x)$Px]'. In the absence of a numeral subscript to '$\forall$' as in 'B[$(\forall x)$Px]' the size of the domain is unspecified.

The defined expressions for sub-scripted quantifiers are used in proofs by mathematical induction to extablish theorems containing only un-subscripted quantifiers.

The other quantifier, '$(\exists x)$' for "For some x", usually called the "existential quantifier", b will be called the **disjunctive quantifier**. Despite its customary name this quantifier does not always entail an assertion of existence. It can be used to make statements about entities in "possible worlds" which do not exist at all in the actual or real world. Thus, without a specification of the domain or "possible world" referred to, the meaning of 'exists' is ambiguous. On the other hand, such quantifiers are always interpretable as representing a disjunction with a disjunct for every member of whatever (unspecified) domain or "possible world" might be the domain of reference.

We can define wffs with disjunctive quantifiers in domains with a specified number, k, of members, just as we did with subscripted conjunctive quantifiers. If the domain is $\{a_1,a_2,a_3\}$, then for that domain,

'$(\exists_3 x)$(Px & Qx)' abbreviates '((Pa_1 & Qa_1) v (Pa_2 & Qa_2) v (Pa_3 & Qa_3))'

and for the domain $\{a_1,a_2,a_3,a_4\}$,

'$(\exists_4 x)$(Px & Qx)' abbreviates '((Pa_1 & Qa_1) v (Pa_2 & Qa_2) v (Pa_3 & Qa_3) v (Pa_4 & Qa_4))'

Each disjunct in such an expansion may be labeled an "instantiation" of '(∃x)(Px & Qx)', and the last disjunction may be named "a Boolean expansion of '(∃x)(Px & Qx)' in a domain of 4", expressed by either, 'B_4[(∃x)(Px & Qx)]' or 'B[($∃_4$x)(Px & Qx)]'. In the absence of a numeral subscript for '∃', as in 'B[(∃x)Px]', the size of the domain is unspecified.

In certain proofs which follow, we will introduce steps like

....) ($∀_7$x)Px SYN (Pa_1&Pa_2&...&Pa_7) [$∀_7$-Exp]
or ....) ($∃_k$x)Px SYN (Pa_1v Pa_2v....v Pa_k) [$∃_k$-Exp]

The justifications on the right, signifying a Boolean expansion in a domain of 7 or k, are elliptical. In the first case it stands for two steps, such as:

a) (∀x)Px SYN_{df} (Pa_1&Pa_2&...&Pa_n) [Df '∀']
b) ($∀_7$x)Px SYN (Pa_1&Pa_2&...&Pa_7) [a),SynSUB]

The reverse process of introducing an abbreviation, as in

a) (Q & (Pa_1vPa_2v...vPa_k)) SYN (Q & (Pa_1vPa_2v...vPa_k)) [T1-11]
b) ($∃_k$x)Px SYN_{df} (Pa_1vPa_2v...vPa_k) [Df '∃']
c) (Q & (Pa_1vPa_2v...vPa_k)) SYN (Q & ($∃_k$x)Px) [a),b),SynSUB]

may be represented elliptically in one step as in,

....) (Q & (Pa_1&Pa_2&...&Pa_k)) SYN (Q & ($∀_k$x)Px) [$∀_k$-Abbr]

For an adequate logical theory of relations—the most important improvement by quantification theory over traditional logic—quantificational wffs must include wffs in which some quantifiers lie in the scope of other quantifiers. How are these to be handled?

The Boolean expansion of a Q-wff in a given finite domain, D, is the result of replacing each occurrence of a non-vacuous quantification, [(∀x)Px], by a conjunction of its instantiations in that domain, and each occurrence of a non-vacuous quantification [(∃x)Px] by a disjunction of its instantiations in that domain. When one quantifier occurs within the scope of another, one may expand either one first, then expand the other.

In any given problem every subscripted quantifier must have the same subscript, i.e., be expanded in the same domain. This agrees with presuppositions of standard logic on interpreting nested quantifications or metatheorms asserting logical equivalence of two wffs (analoguously to assertions of SYN of two wffs). Metatheorems asserting the logical equivalence of A and B are interpreted on the presupposition that all quantified variables in both wffs range over the same domain, no matter how big or small that domain may be.[13]

The following example shows how to get, in seven steps, a Boolean expansion in a domain of 3 of the wff '((∀x)(∃y)(Pxy & ∼Pyx) & (∀z)Rzz)':

13. A and B can be logically equivalent when both are inconsistent in domains of 2 but consistent in a domain of 4; they are not necessarily non-equivalent because A is inconsistent in a domain of 2 while B is consistent in a domain of 4.

The Boolean Expansion of ‘(($\forall_3$x)($\exists_3$y)(Pxy & ~Pyx) & ($\forall_3$z)Rzz)’ in the domain {a,b,c}:

Step 1. Expanding ‘($\forall_3$z)Rzz’ within the whole:

1) ($\forall_3$z)Rzz SYN (Raa & Rbb & Rcc) [$\forall_3$-Exp]

2) (($\forall_3$x)($\exists_3$y)(Pxy & Pyx) & ($\forall_3$z)Rzz)
SYN ($\forall_3$x)(($\exists_3$y)(Pxy & Pyx) & (Raa & Rbb & Rcc)) [2),1),R1b]

Step 2. Expanding ‘($\exists_3$y)(Pxy & Pyx)’ within the result of Step 1:

3) ($\exists_3$y)(Pxy & Pyx) SYN ((Pxa & Pax) v (Pxb & Pbx) v (Pxc & Pcx)) [$\exists_3$-Exp]

4) (($\forall_3$x)($\exists_3$y)(Pxy & Pyx)&(Raa & Rbb & Rcc))
SYN (($\forall_3$x)((Pxa & Pax) v (Pxb & Pbx) v (Pxc & Pcx)) & (Raa & Rbb & Rcc))
[2),3),R1b]

Step 3. Expanding the result of Step 2 by instantiating the ‘($\forall_3$x)’ in ‘($\forall_3$x)((Pxa & Pax) v (Pxb & Pbx) v (Pxc & Pcx)))’:

5) ($\forall_3$x)((Pxa & Pax) v (Pxb & Pbx) v (Pxc & Pcx))
SYN(((Paa & Paa) v (Pab & Pba) v (Pac & Pca))
&((Pba & Pab) v (Pbb & Pbb) v (Pbc & Pcb))
&((Pca & Pac) v (Pcb & Pbc) v (Pcc & Pcc))) [$\forall_3$-Exp]

6) (($\forall_3$x)($\exists_3$y)(Pxy & Pyx) & (Raa & Rbb & Rcc))
SYN(((Paa & Paa) v (Pab & Pba) v (Pac & Pca))
&((Pba & Pab) v (Pbb & Pbb) v (Pbc & Pcb))
&((Pca & Pac) v (Pcb & Pbc) v (Pcc & Pcc)))
&(Raa & Rbb & Rcc)) [4),5),R1b]

7) (($\forall_3$x)($\exists_3$y)(Pxy & Pyx) & ($\forall_3$z)Rzz)
SYN (((Paa & Paa) v (Pab & Pba) v (Pac & Pca))
&((Pba & Pab) v (Pbb & Pbb) v (Pbc & Pcb))
&((Pca & Pac) v (Pcb & Pbc) v (Pcc & Pcc)))
& (Raa & Rbb & Rcc)) [2),6)R1b]

Each of the seven steps here, and in other derivations based on Boolean expansions, is a SYN-theorem. But at bottom they are only theorems by virtue of **de dicto** conventions on abbreviation. They are useful, but not the ultimate targets of quantification theory. We use them primarily as steps in proving theorems without quantifier subscripts by mathematical induction.

The significant theorems of quantification theory do not have quantifiers with subscripts. They have to do with the referential synonymy of two kinds of logical structures. Quantifiers without subscripts do not specify any particular size of a domain; they represent a kind of structure for domains of all different sizes. Whether the range of the quantifiers' variables is everything that exists, or some finite set of fictional entities, or something else, is left open. What is asserted is that no matter which domain both wffs are referred to, or how big it is, their Boolean expansions will be logically synonymous.

It is always presupposed in general quantification theory that in any given context of investigation all quantified wffs are referred ultimately to the same domain.

In ordinary language the domains over which the quantified variables range will vary with the field of reference, or universe of discourse—what the speaker or speakers are talking about. Within rational discourse care should be taken to be clear when different quantificational statements have the same or different domains as the range of their quantified variables. Ordinarily the domain or reference is tacitly understood and kept constant throughout a given context of discussion. But not always; different presuppositions about the domain can be a source of misunderstandings, confusion, and irrelevance.

In logic this confusion is avoided because it is presupposed that the variables of every quantifiers occurring in any given logical investigation ranges over the same domain of individuals. This holds whether we are interpreting a single quantificational wff which contains two or more quantifiers or examining whether the logical forms of two or more quantificational wffs are logically synonymous, or presenting a chain of theorems or of wffs with various quantifiers in a given deduction or proof. The domain of reference is assumed to remain constant throughout the context. This holds for standard logic as well as the present system.

The **theorems** concerning logical synonymy and containment in this chapter and the next are thus intended to describe pairs of logical structures with quantifiers which would be referentially synonymous in any domain regardless of (i) the size of the domain, (ii) the kinds of individuals which belong to it, (iii) the properties and relations symbolized by predicate place-holders in the two wffs, or (iv) facts about whether such properties and relations obtain or not in any actual field of reference.

In ordinary discourse, defining the specific domain of reference is sometimes necessary to avoid misunderstanding. But in logic lack of specificity does not raise difficulties; because the constancy of the domain of reference is presupposed in every problem.

3.23 The Concept of Logical Synonymy Among Quantificational Wffs

What conditions must be satisfied for two quantificational wffs to be **logically** synonymous?

The answer is that given any two Q-wffs, Q_1, Q_2, in Analytic Logic, [Q_1 SYN Q_2] will be a theorem if and only if in every domain the Boolean expansions of Q_1 and Q_2 in would be provably SYN.[14] This requires that we first explain what it would be for two Q-wffs to be logically synonymous in a given single finite domain of k members.

To say that two Q-wffs are logically synonymous in a single finite domain, is to say that their Boolean Expansions in that domain are logically synonymous.

Let 'Q_1', 'Q_2',…etc, be schematic letters for distinct wffs which contain unsubscripted quantifiers; let '$B_k[Q_i]$' stand for 'the Boolean Expansion of Q_i in a domain of k'; and let '$B_k[Q_1$ SYN $Q_2]$' stand for sentences (true or false) of the form 'Q_1 is logically synonymous to Q_2 in a domain of k individuals'. Then the logically synonymy of two negation-free Q-wffs in a finite domain of k members may be explicated as follows:

***D20.** $\models [B_k(Q_1 \text{ SYN } Q_2)]$ if and only if
(i) the same assignment-function of individual constants to members of the domain is used in both $B_k[Q_1]$ and $B_k[Q_2]$;
(ii) $\models [B_k(Q_1) \text{ SYN } B_k(Q_2)]$ [Decidable by methods of Chapters 1 and 2]

The concept of a pair of Q-wffs (without subscripts) being logically synonymous is expressed as:

***D21.** For any two quantificational wffs (with no subscripted quantifiers),
$\models[Q_1 \text{ SYN } Q_2]$ if and only if **For every k**, $\models [B_k(Q_1 \text{ SYN } Q_2)]$.

*D21 goes much farther than *D20. If [Q_1 SYN Q_2] is logically true, each of the following must be true:

---

14. We proceed on the assumption that it is sufficient to deal with all finite domains, provided there is no limit on the number of members. We argue elsewhere that the concept of infinite domains is unnecessary since all propositions about them can be replaced, without loss, by statements involving generative definitions intensionally construed.

1) [Q_1 SYN Q_2] in at least one non-empty finite domain,
2) [Q_1 SYN Q_2] in all finite domains, no matter how large.
3) [Q_1 SYN Q_2] in all domains, finite or infinite.

Before proceeding, let us consider *D20 in more detail: A Boolean expansion of a Q-wff, $B_k[Q_1]$, in any single finite domain D of k members, will be a sentential wff, all of whose atomic sentence schemata are of the form Pi(t_1,…,t_n) with Pi some predicate letter and each ti an individual constant denoting just one member of the domain D. Thus the test for logically synonymy of the Boolean expansions of two negation-free Q-wffs in a given domain, is simply a case of testing for logical synonymy of negation-free wffs in standard sentential logic, with atomic sentence wffs used instead of simple letters, P_1, P_2, P_3,…etc.. This is decidable by reduction to the same MOCNF wff, or by analytic "truth-tables" per SYN-metatheorems 11 and 12 in Chapter 1.

To insure that sameness of elementary components is preserved the proviso in clause (i) of *D20 is required: Each distinct individual constant in $B_k[Q_1]$ and $B_k[Q_2]$ must be assigned to the same member of the domain. Assured of this constancy of interpretation, we can be assured that if they are logically synonymous, their atomic components will have the same argument-position-holders in the same positions, will talk about all and only the same individuals, and will say the same things about the same individuals. If this requirement is satisfied, two wffs in the expansion will be **distinct** if and only if either (i) they have different predicate letters, or (ii) one has more argument positions than the other, or (ii) they have the same predicate letter and the same number of argument positions, but different individual constants or argument-position-holders in one or more argument positions.

Thus there is a decision procedure for whether two arbitrary Q-wffs satisfy the components of *D20. In theory it is always possible to decide with respect to a single given finite domain, whether Boolean expansions of two quantificational wffs are logically synonymous in that domain.But this fact does not give us a way to prove that the two wffs would be logically synonymous in **all** finite domains. And of course no human being or group of human beings could ever test a quantificational wff in all finite domains even in theory, since the number of finite domains is infinite. Thus to establish a theorem of the form '(Q_1 SYN Q_2)' we must find a way to prove **For every k** [B_k(Q_1 SYN Q_2)],as i.e., we must prove that *D21 holds for Q_1 and Q_2.

The basic method of proving that Q_1 and Q_2 satisfiy ***D21** is by using the principle of mathematical induction. If we can prove that 1) the given pair {Q_1,Q_2} are logically synonymous in the domain of 1 member (the base step) and also (2) that if their Boolean expansions in a domain of k are logically synonymous, then their Boolean expansions in a domain of k+1 will be logically synonymous (The induction step), then it follows by the principle of mathematical induction that (3) their Boolea n expansions will be logically synonymous in all finite domains. This is the method we use to prove a basic set of theorems in Section 3.332 and APPENDIX III.

Because the number of theorems in quantification theory is unlimited, we can not prove that all are provable in our system by proving each one using mathematical induction. What is needed is an axiomatic system, with a finite number of initial theorems proved by mathematical induction, plus rules of inference sufficient to get a completeness proof—i.e., a proof that every pair of Q-wffs which is synonymous in all finite domains, will be provable from these axioms and rules. The small set first proven by mathematical induction (T3-19—T3-28) will serve as basic theorems from which many others can be derived using the rules of inference.[15]

3.24 Quantificational Predicates

As was said in Chapter 2, every wff can be understood as a predicate schema. This includes Q-wffs. Every statement, including quantificational statements, has a predicate which can be extracted. Quantificational statement-schemata are Q-wffs which have no argument-position-holders (numerals) in the scope of their quantifiers. Thus statement-schemata are in effect 'saturated' predicate-schemata. They represent predicates in which every argument position is occupied by an individual constant.

It follows that '(∀x)(∃y)R<x,y>' is both a statement-schema and a predicate schema, but '(∀x)(∃y)R<x,y,1>' is not a statement-schema since it has an argument-position-holder and is not "saturated". In both cases properties relevant to logic are just the properties of their predicate schemata.

But also every statement-schema can be interpreted as representing other unsaturated predicate schemata which will preserve its logical properties. For suppose we have a theorem in which only predicate letters occur, such as [(P_1 & P_2) CONT (P_2 v P_1)]. According to the Rule of U-SUB, logical synonymy and logical containment are preserved by substitutiing at all occurrences of the predicate letter P_2 in both sides of the CONT-theorem, any wff—such as 'P_3<1,2,a,>' or 'P_4<a,2>', or '(∀y)(∃x)R<1,x,y>' etc. Also SYN and CONT are preserved by the augmented U-SUB which allows the **introduction** at all occurrences of **modes** of an abstract h-adic elementary predicate schema, of a predicate which is (h+j)-adic, ($j \geq 0$), i.e., which has as many or more than h argument-position-holders and/or additional individual constants. Thus for example, the quantified wff '(∀x)(∃y)R<x,y>' while not synonymous with '(∀x)(∃y)R<x,y,1>' or with '(∀x)(∃y)(R<x,y,1> v Q<x,2>)' may be said to stand for or represent, by virtue of its logical form, the more complex structures of either of the latter wffs. For, by U-SUB, either of the latter could be uniformly substituted for the former in a SYN-theorem, preserving SYN in the result.

In short, all wffs, including Q-wffs, may be viewed as representing unsaturated predicates and predicate schemata which display as much, or more, logical structure than they do.

In dealing with quantificational statements and predicates, the **quantificational predicate** itself must be distinguished from the ***predicate of*** **the quantificational predicate**. The distinction is between the whole well-formed predicates which quantifiers are used to abbreviate, and the predicate which lies within the scope of the quantifiers and is repeated in each instantiation.

Thus [P<1>] is ***the predicate*** **<u>of</u> the quantificational predicate, [(∀x)P<x>]**. But **the quantificational predicate [(∀x)Px]** is a complex predicate of the form [Pa_1 & Pa_2 & ... & Pa_n]. Again, in a domain of n, ***the predicate of*** **the quantificational wff, '(∃x)R<1,x>'** is 'R<1,2>', which has no quantifier but is systematically repeated in **the quantificational wff '(∃x)R<1,x>'**, i.e., the wff '(R<1,a_1> v R<1,a_2> v...v R<1,$a_{(n-1)}$> v R<1,$_n$>)'.

---

15. The concept of a Boolean expansion in an infinite domain is more difficult than the concept of Boolean expansions in finite domains because one can never complete the expansion of an outside quantifer, e.g., one can not complete the expansion of '(∀x)' in '(∀x)(∃y)(Pxy &~Pyx)', so as to begin expanding what is embedded in it namely,' (∃y)(Pxy & ~Pyx)')—or vice versa. The present system deals with validity in all finite domains, and there are an infinite number of them. But I will not discussed validity in infinite domains—domains with an infinite number of members. I believe it can be shown that such domains need not be posited for any purposes of analytic logic; and that the problems of infinite sets can be reduced to problems of relations and properties of generative definitions—definitions in A-logic which have no finite limit on the terms they generate. (E.g, the definitions of '<1> is a wff', '<1> is a positive integer' and '<1> is an arabic numeral').

Again, the abstract ***predicate of*** **the Q-wff ‘(∀x)(∃y)(∀z)R<x,y,z>’** is ‘R<1,2,3>’, which is also the abstract ***predicate of*** the **Q-wffs ‘(∀z) R<1,2,z>’** and **‘(∃y)(∀z) R<1,x,y>’**. In domains of n members ‘R<1,2,3>’ will have n^3 instantiations. **The Q-wff ‘(∀z)R<1,2,z>’** stands for a complex 2-adic, 3n-place **predicate schema** of the form,

$$\text{‘((R<1,2,}a_1\text{> \& (R<1,2,}a_2\text{> \& … \& (R<1,2,}a_{(n\text{-}1)}\text{> \& R<1,2,}a_n\text{>))’}$$

and **the Q-wff ‘(∃y)(∀z)R<1,x,y>’** stands for a 1-adic, $3n^2$-place **predicate schema** of the form,

$$\text{‘((R<1,}a_1,a_1\text{> \& R<1,}a_1,a_2\text{> \& … \& R<1,}a_1,a_{n\text{-}1}\text{> \& R<1,}a_1,a_n\text{>)}$$
$$\text{v (R<1,}a_2,a_1\text{> \& R<1,}a_2,a_2\text{> \& … \& R<1,}a_2,a_{n\text{-}1}\text{> \& R<1,}a_2,a_n\text{>)}$$
$$\text{v….}$$
$$\text{v (R<1,}a_n,a_1\text{> \& R<1,}a_n,a_2\text{> \& … \& R<1,}a_n,a_{n\text{-}1}\text{> \& R<1,}a_n,a_n\text{>))’}$$

In some cases the synonymy of two different Q-wffs is due solely to the synonym of ***the predicates of*** the quantificationl wffs involved. For example,

1) [(P & Q) SYN (Q & P)]
2) [(P<1> & Q<1>) SYN (Q<1> & P<1>)] [1),R2-2,(twice)]
3) [(∀x)(Px & Qx) SYN (∀x)(Qx & Px)] [2),DR3-3c]

will be a valid proof in which the Synonymy of the Q-wffs, comes from the synonymy of the ***predicates of*** the **Q-wffs** ‘(∀x)(Px & Qx)’ and ‘(∀x)(Qx & Px)’, namely ‘(P<1> & Q<1>)’ and ‘(Q<1> & P<1>)’.

In other cases the synonymy of two different Q-wffs comes from the structure of **quantificational wffs themselves**; i.e., from analysis of the meanings of ‘∀’ or ‘∃’. For example, T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx] can not be gotten from any SYN- theorems which directly relate R<1,2> and R<1,1>, since there are no such theorems. Rather it is based on the relationship of containment between the expansions of ‘(∃x)(∀y)Rxy’ and ‘(∀x)Rxx’ in finite domains.

For example in a domain of 3 individuals, {a,b,c},

‘(∃x)(∀y)Rxy’ stands for ‘((Raa & Rab & Rac) v (Rba & Rbb & Rbc) v (Rca & Rcb & Rcc))’

and by derivations of the sorts allowed in Chapter 1, it is proven that

[((Raa & Rab & Rac) v (Rba & Rbb & Rbc) v (Rca & Rcb & Rcc)) CONT (Raa v Rbb v Rcc)],

and in a domain of 3, by Df ‘∃’, ‘(Raa v Rbb v Rcc)’ means ‘(∃x)Rxx’. Proof that such a result obtains in every finite domain yields the theorem T3-35.

The introduction of quantifiers and variables makes it possible to define extra-logical predicates which contain quantifiers. We often define a simple predicate in terms of another predicate which involves one or more quantifiers. For example, the simple 1-adic predicate

1) <1>is a parent P<1>

is different syntactically than the 1-adic quantificational predicate,

2) (∃x) x is a child of<1> (∃x)R<x,1>

yet it is compatible with ordinary usage to consider 1) and 2) as referentially synonymous. The statement,

3) ‘<1>is a parent’ Syn$_{df}$ ‘(∃x) x is a child of<1>’,

has the form: ‘P<1>’ Syn$_{df}$ ‘(∃x)R<x,1>’. It is an *a priori* assertion of semantic, or *definitional,* synonymy (but not an assertion of *logical* synonymy since the synonymy is not based on the meanings of syncategorematic words only). Nevertheless, logical inference in the broad sense (in applied logic) is based largely on this kind of semantic synonymy and containment. In such definitions, the definiens and definiendum must have all and only the same argument-position-holders. The inference from ‘Joe is a parent’ to ‘some one is Joe’s child’ has the form, ‘P<a> therefore (∃x)R<x,a>’ and is a logically valid deduction from the the definitional premiss, 3), ‘P<1> syn$_{df}$ (∃x)R<x,1>’.

By the definition of ‘∃’, the logical structure of the definiendum is displayed in a complex, but predictably regular, abstract predicate of indefinite length, freed of content.

[P<1> syn$_{df}$ (R<a_1,1> v R<a_2,1> v...v R<a_n,1>)]

On this definition the apparently simple predicate ‘<1>is a parent’ is synonymous with a disjunctive predicate of the form,

4) ‘<1>is a parent’
Syn$_{df}$ ‘(a_1 is the child of<1>) v (a_2 is the child of<1>) v … v (a_n is the child of<1>)’

Thus the 1-adic, 1-place simple quantificational predicate, ‘<1>is a parent’ is synonymous by this definition with a 1-adic, 2n-place predicate; for 4) has n places with n distinct individual constants, plus n places with just the numeral 1 in them.

For a quantificational predicate with more than one quantifier, consider ‘<1>is loved by someone who loves everybody’ which may be schematized as ‘(∃x)(∀y)(L<x,y> & L<x,1>)’ and has the form ‘(∃x)(∀y)(P_3<x,y> & P_3<x,1>)’. This predicate can be proved to logically contain the statement, ‘Some people love themselves ‘(∃x)(L<x,x>’ (the proof is based in part on T3-35, mentioned above). Thus some predicate schemata that are not statement schemata, may contain statement-schemata.

3.3 Axioms and Derivation Rules

No additional axioms are needed to establish logical synonymy and containment among negation-free quantificational predicate schemata. The Rule of Instantiation and the definitions Df ‘∀’ and Df ‘∃’ with the Principle of Mathematical Induction, are sufficient to establish derived rules for proving quantificational theorems that will hold in all domains.

Thus though the set of wffs have been expanded significantly, the axioms of Chapter 1 and 2 are sufficient and unchanged:

IV. Axioms: Ax1-01. ⊨ [P SYN (P&P)] [&-IDEM]
Ax1-02. ⊨ [P SYN (PvP)] [v-IDEM]
Ax1-03. ⊨ [(P&Q) SYN (Q&P)] [&-COMM]
Ax1-04. ⊨ [(PvQ) SYN (QvP)] [v-COMM]

Ax1-05. |= [(P&(Q&R)) SYN ((P&Q)&R)] [&-ASSOC]
Ax1-06. |= [(Pv(QvR)) SYN ((PvQ)vR)] [v-ASSOC]
Ax1-07. |= [(Pv(Q&R)) SYN ((PvQ)&(PvR))] [v&-DIST]
Ax1-08. |= [(P&(QvR)) SYN ((P&Q)v(P&R))] [&v-DIST]

From these axioms, and theorems derived from them in Chapter 1, with the rule of U-SUB augmented in Chapter 2 and in this chapter, and with SynSUB based on Df '∀' and Df '∃', we can derive unlimited numbers of new quantificational theorems employing the expanded vocabulary.

Derived Rules DR3-3a and DR3-3b are akin to the rules of UG ("Universal generalization") and EG ("Existential Generalization") in M-logic. They are established by mathematical induction. See Section 3.323These two rules (which will be reduced to one in Chapter 4) sanction inferences from non-quantificational SYN-theorems or CONT-theorems (in Chapters 1 or 2) to quantificational theorems.

Every quantification theorem in this chapter can be shown to hold in any finite domain by a sort of mathematical induction. However, it will be convenient to designate certain quantificational theorems as "basic theorems", then derive other theorems from the basic theorems without having to resort to mathematical induction for each one. Each basic theorems is shown to hold in all finite domains by mathematical induction.

There are differences of kind among theorems, principles of inference (or transformation), and rules of inference (or transformation), as indicated by the following outline:

- I. Transformation Principles and Rules (expressed in conditional statements)
  - Ia. Principles of Transformation
    - A. Principles of Derivation
      - 1) General Principles of Logical Inference
      - 2) Specific Derivation Principles
    - B. Translations
  - Ib. Transformation Rules
    - A. Derivation Rules
      - 1) Rules of Inference
      - 2) Derivation Rules
    - B. Alternative Notations
- II. Theorems
  - A. Categorical
  - B. Conditional

The difference between Principles of Transformation and Rules of Transformation is that the former are conditional statements in the indicative or subjunctive mood and the latter have a consequent in the imperative mood. Principles assert that if the antecedent is true, then the consequent is true. For example:

If [P SYN (Q&R)] is true, then [P CONT R] is true,
If [P SYN (Q&R)] is true, then INC[P & ~ R] is true, etc...

The corresponding rules are in the imperative mood:

If [P SYN (Q&R)] is true, then infer [P CONT R] !
If [P SYN (Q&R)] is true, then infer INC[P & ~ R]! etc...

Each principle supports the reliability of the corresponding rule, indicating that if the antecedent is fulfilled, then you may infer the consequent with confidence. Although the rules are in the imperative mood, they are not commands. They are conditioned directives or suggestives. Users of logic are not enjoined to make inferences. Rather, in trying to arrive at a proof or in reasoning out a problem, there may be many rules of inference at their disposal, and at each step they choose selectively a rule which, hopefully, will bring them closer to their objective.

Transformation *rules*, including Logical Rules of Inference, are conditional permissions based on an assertion that if given expressions of one sort have a certain logical property or relation, then expressions of a different syntactical form will have the same, or some other logical property or relationship. The general purpose served by a transformation rule is to show how a reliable assertion can be gotten by syntactical operations on a previously established but different reliable assertion. But not all transformation rules used in logic should be treated as Logical Rules of Inference.

A Rule of Inference for Logic says that if certain kinds of expressions possess a logical property or stand in a certain logical relationship, then we may correctly infer that certain other kinds of expressions will have the same, or some other, logical properties or relationships. They are rules of transformation which can be used to transform one theorem into a different theorem of logic. The basic Rules of Inference for A-logic apply to all predicates and sentences. They are

SynSUB. If $\models$ [P] and $\models$ [Q Syn R] then $\models$ [P(Q//R)]
U-SUB. If $\models$ [P] and Q is an atomic predicate schema in P
and R is a suitable predicate schema, then $\models$ [P(Q/R)]
INST. If $\models$ [P<1>] then $\models$ [Pa_i]

SynSUB changes the logical structure but retains the same components and preserves or establishes the logical property or relation asserted in the initial theorem. U-SUB retains the overall logical structure, but changes the detailed predicate schemata in a component, while preserving the logical property or relation asserted. INST applies predicates in a theorem to one or more different individuals while retaining the logical property or relation asserted in the theorem.

Derived Rules of Inference, based on definitions of additional logical properties and relations, allow the transformation of a theorem asserting certain logical properties or relations into another theorem asserting a different kind of logical property or relation.

Some transformation rules are useful, convenient, even practically indispensable, but are not Rules of Logical Inference either (i) because they are expedient devices based on artificial extra-logical features of expressions, or (ii) because there is no difference in the logical structures or meanings of the expressions before and after (as in rules of re-lettering, including "alphabetic variance").

In the first case, the rule is useful and does involve changes in logical structure or meaning, but is not a logical rule, because it goes outside of logic to get its results. Rules which allow vacuous quantifiers and expressions with "free" variables, can be useful but are artificial in the sense that they rely on expressions or devices unrelated to logical structures. Rules for testing the validity of an argument by truth-tables and other modeling systems like Venn or Euler Diagrams fall in this category.

In the second case, some rules preserve logical properties of different expressions and may be helpful in deriving theorems which are easier to express or grasp, but do not involve any changes in logical structures or relationships. The prime example is Alphabetic Variance. Others include rules for relettering a proof in sentential logic; this would present the property of 'logically following from' in two different versions (vis-a-vis lettering) of a logical proof, but nothing new is derived. This is merely a translation, though it has its own rules. Translations from the notation of one system to that of another, and from one natural language to another, are roughly in the same class.

Categorical theorems of logic simply assert that certain expressions or kinds of expression possess this or that logical property, or stand in this or that logical relation to each other. All axioms and theorems of A-logic are clearly of this sort. They are purported to be true of logical predicates applied to entities, including forms of expressions, which have meanings. Conditional expressions are said to be valid (rather than true or false) in logic. The most important theorems of logic are those which assert that some conditional statement, or form of statement, is logically valid. Eventually the principles of logical inference are proven to be valid conditional statements based on definitions of the predicates of logic. Thus they become theorems of A-logic itself. Each such theorem asserts that a certain conditional is valid, implicitly claiming that it is true that that conditional is valid. The corresponding *rule* of inference, is gotten by the tranformation of the consequent of the *principle* from a truth-assertion into a directive or suggestive clause in the imperative mood in the *rule*.

In considering Quine's axiomatization of M-logic, conditional Rules of Transformation and conditional Theorems should be distinguished from conditioned theorems and axioms. Both can be expressed in conditional sentences but there is an important difference. A conditional theorem or rule, is a conditional statement which is logically valid. A conditioned theorem asserts that wffs will have a certain logical property or relationship if they have certain internal characteristics; i.e., the logical property or relationship applies to any wffs which meet the specified conditions. It asserts that if certain features belong to components, then some logical property or some logical relationship will hold of or between first-level wffs. They are used to establish a single theorem or theorem-schema. Example: 'If x is not free in P, then [(P & (∃x)Qx) iff (∃x)(P v Qx)] is a conditioned theorem.(In A-logic, well-formed formulas are defined so as to avoid the need to condition a theorem on x is free in P. Every 'P_i' stands for a well-formed expression, and no well-formed expression has an unbound individual variable in it.)

3.31 Alpha-Var—The Rule of Alphabetic Variance

Before discussing other rules and proofs we discuss the rule of notational convenience, Alphabetic Variance, which is useful in some derivations but which is not a substantive Rule of Logical Inference.

Applied to components of SYN- and CONT- theorems, the Rule of Alphabetic Variance, which permits certain notational variations which preserve logical synonymy or containment. According to this rule the individual variables in any quantified wff can be uniformly re-lettered so that the resulting wff is logically synonymous with the original wff and has an identical logical form, structure and meaning in all of its parts. Because the the re-lettered wff are the same in all its parts as the original both in logical structure and in meaning, this rule has no logical significance. The Boolean expansions of two alphabetic variants in any given domain are identical in every numeral and individual constant; alphabetic re-lettering occurs only at the level of the quantified abbreviations for Boolean expansions.

Nevertheless the rule is useful. Under certain conditions, the same variable letter can be used in different quantifiers. This often decreases the number of letters we need for variables and makes it possible to re-letter different quantificational wffs in ways which make points of similarity and difference more obvious to the eye.

Such re-lettering of variables must be done according to certain rules however, or wffs which are non-synonymous will be wrongly presented as SYN-theorems. Hence 'P is an alphabetic variant of Q' has a more complicated meaning than first appears. Two quantificational wffs are alphabetic variants of one another if and only if they satisfy all three clauses of the following definition:

'P is an alphabetic variant of Q'

$\underline{\text{syn}}_{\text{df}}$

'(i) P and Q are wffs containing one or more quantifiers;

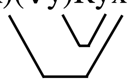
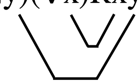
(ii) P and Q differ only with respect to which letters for individual variables occupy which positions in the respective wffs; and
(iii) P and Q have exactly the same bondage patterns.

Given this definition, the Rule of Alphabetic Variance is formulated as follows:

<u>Alph.Var</u>. If P is an alphabetic variant of Q, then [P SYN Q]

The following are examples of alphabetic variants, by the definition above:

a)'(∃z)(∀w)Rwz' is an alphabetic variant of '(∃y)(∀x)Rxy'
b)'(∃x)(∀y)Ryx' is an alphabetic variant of '(∃z)(∀w)Rwz'
c)'(∃x)(∀y)Ryx' is an alphabetic variant of '(∃y)(∀x)Rxy

In the case of a) the alphabetic variants use entirely different letters, but the bondage pattern remains the same. In the case of c) the same letters are used and the bondage pattern is still the same. Each of the three wffs in quotes is an alphabetic variant of the other two. Every quantified wff has an unlimited number of possible alphabetic variants.

The following theorems (and their converses) follow from the truth of a) and b) above, by the rule of alphabetic variance:

1) [(∃z)(∀w)Rwz SYN (∃y)(∀x)Rxy] [a),Alph.Var.]
2) [(∃x)(∀y)Ryx SYN (∃z)(∀w)Rwz] [b),Alph.Var.]
3) [(∃x)(∀y)Ryx SYN (∃y)(∀x)Rxy] [2),1) SynSUB]

But there is a mistake that must be avoided: changing the bondage pattern in the process of substituting one letter for another. The following steps make this error:

Step 1) (∃x)(∀y)Rxy [Initial wff]
Step 2) (∃y)(∀y)Ryy [Replacing 'x' at all occurrences by 'y']

The difference in bondage patterns of (∃x)(∀y)Ryx and (∃y)(∀y)Ryy affects the Boolean expansions, hence the meanings. For example in a domain of 3,

B_3[(∃x)(∀y)Ryx] SYN ((Raa&Rba&Rca) v (Rab&Rbb&Rcb) v (Rac&Rbc&Rcc))
B_3[(∃y)(∀y)Ryy] SYN (Raa&Rbb&Rcc)

Not only are the two wffs are not synonymous. In addition '(∃y)' becomes vacuous in Step 2) so that the formula is not well-formed.

A procedure for deriving a synonymous alphabetic variant of any Q-wff, must be formulated so as to avoid such results, and satisfy clause (iii) of the definition of an alphabetic variants. The following auxiliary rule, similar to previous rules of relettering, suffices:

1) To get an alphabetic variant of any Q-wff, at each step replace all occurrences of one of the variables by a variable that has no occurrences in the wff of the preceding step.

This rule guarantees that the set of wffs from all steps in the process will be alphabetic variants in accordance with clauses (i)-(iii) of the definition. For example, (∃y)(∀x)Rxy can be derived from (∃x)(∀y)Ryx using the three alphabetic variants in 1) to 3) above by the following steps:

> Step **1)** (∃x)(∀y)Ryx [Initial wff]
Step 2) (∃z)(∀y)Ryz [Replacing 'x' at all occurrences in Step 1) by 'z', which has no occurrences in Step 1)]
> Step **3)** (∃z)(∀w)Rwz [Replacing 'y' at all occurrences in Step 2) by 'w', which has no occurrences in Step 2)]
Step 4) (∃y)(∀w)Rwy [Replacing 'z' at all occurrences in Step 3) by 'y', which has no occurrences in Step 3)]
> Step **5)** (∃y)(∀x)Rxy [Replacing 'w' at all occurrences in Step 4) by 'x', which has no occurrences in Step 4)]

This rule for getting an alphabetic variant, A', of a wff, A, results in the same number of distinct variables in A and in A'. This is not incompatible with using the same variable in two separate quantifications. If desired, this result can be obtained as follows:

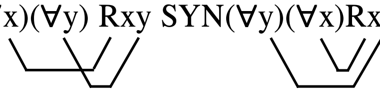
1) [(∃x)(∀y)Rxy CONT (∀y)(∃x)Rxy]	[T3-37]
2) [(∀z)(∃w)Rwz SYN (∀y)(∃x)Rxy]	[Alpha.Var.]
3) [(∃x)(∀y)Rxy CONT **(∀z)(∃w)Rwz]**	[1),2),R1]
4) [(∀x)(∃y)Rxy SYN (∀z)(∃w)Rzw]	[Alpha.Var.]
5) [(∃x)(∀y)Rxy CONT **(∀x)(∃y)Ryx]**	[3),4),R1]

If two wffs are alphabetic variants, they will have exactly the same Boolean expansion in every finite domain. For example, given the domain of three objects, {a,b,c}, the alphabetic variants, '(∃x)(∀y)Rxy', '(∃z)(∀y)Rzy', '(∃z)(∀w)Rzw', '(∃y)(∀w)Ryw', '(∃z)(∀x)Rzx', and '(∃y)(∀x)Ryx' (for instance, "Somebody loves everybody") all have the same Boolean expansion, namely,

((Raa & Rab & Rac) v (Rba & Rbb & Rbc) v (Rca & Rcb & Rcc)).

This should be clear because even before expansion, the logical structure of the two variants is unaltered—re-lettering the variables makes no change in any logical constants, in grouping, in linear order, or in the positions or occurrences of any predicate letters. If A and B are wffs which Alphabetic Variants, one can be substituted for the other in any Q-wff by SynSUB.

In contrast consider |=[(∀x)(∀y) Rxy SYN(∀y)(∀x)Rxy],



which are synonymous, but not alphabetic variants because their bondage patterns are different. This shows up in the different structures of the Boolean expansions of the two wffs. For example in a domain of 2, the Boolean expansion of (∀x)(∀y)Rxy is ((Raa & Rab) & (Rba & Rbb) whereas the Boolean expansion of (∀y)(∀x)Rxy is ((Raa & Rba) & (Rab & Rbb). This is a significant synonymy, signifying sameness of meaning of two different logical structures, whereas the synonymy of alphabetic variants signifies no difference in logical structure or form, and merely a notational difference between to expressions which mean the same thing.

The principles of Alphabetic Variance above are in accord with Quine's metatheorems *170 and *171 regarding alphabetic variance and his discussion in Section 21, *Mathematical Logic*.

A clarification. Syntactically, Q-wffs may appear to lack the requirement of sentential logic which says that if an expression of the form [P SYN Q] is a theorem of logic, then P and Q must have all and only the same atomic wffs. This may seem violated, for example, in theorems like [(∀x)Px SYN (∀y)Py], since the variables on one side are not the variables on the other. But there is a misleading analogy here. Different individual variables are not placeholders for different linguistic entities, as predicate letters, argument-position-holders, or individual constants are. That 'x' is a different variable that 'y' does not signify that '(∀x)' and '(∀y)' cover different classes.

Ordinary language has not, so far, included quantifiers and variables, 'x', 'y', 'z', etc., in its vocabulary. Nevertheless, quantifiers and variables are simply more expeditious ways of expressing what ordinary language expresses more clumbsily, and often ambiguously, using pronouns for cross-references. They should be viewed as improvements within ordinary language, rather than as logical symbols or place-holders for expressions in ordinary language, e.g., for predicates, for descriptions or names of individual objects, or for sentences.

Further, since quantified expressions abbreviate conjunctions or disjunctions of expressions, there is no problem in having a conjunctive quantifier and variables on the left side of [(∀x)Px CONT (Pa & Pb)], and individual constants but no variables on the right side. Not only is there no problem; this provides a basis for the time-honored rule of Universal Instantiation.

3.32 Rules of Inference

Since no new axioms are added for quantification theory, all theorems must be derived from axioms and theorems in Chapter 1 and 2 with the help of inference rules and the new definitions.

Proofs that quantificational SYN-theorems hold in all finite domains may be established by the principle of mathematical induction. Except for the final step (the conclusion drawn from the base step and the induction step) they use only theorems and axioms of Chapter 1 for steps in the proof. Proofs by mathematical induction can prove only one theorem at a time, for such proofs always presuppose only the components and logical structure of the given wffs.

No human being or finite group of human beings could prove all theorems in this way, for there is no limit to the number of them. What can be proved is that the system of axioms and rules are sufficient to prove all and only those theorems which conform to our semantic concept of logical synonymy. And since every quantificational theorem must hold in all domains, this entails that no matter what pair of quantified wffs might occur in a SYN-theorem we could prove, using the principle of mathematical induction, that the Boolean expansions of that pair would be logically synonymous in every domain.

A proof by mathematical induction could be given for any theorem of analytic quantification theory, but such proofs can be avoided by deriving them from a few theorems already established by such proofs.In this chapter certain quantificational theorems have been selected as basic ones and each is proved to hold in any domain by mathematical induction. We call this set, a set of "basic-theorems". Then beginning with the basic-theorems, by using INST, U-SUB, SynSUB and definitions of quantifiers, additional theorems are derived with logical structures different from any previous ones. Proofs of this sort may be called axiomatic proofs to distinguish them from proofs by mathematical induction. Given the basic theorems and rules, we will show that all and only logically synonymous pairs of wffs would be derivable from this set with these rules. In this chapter the basic theorems consist of the sixteen theorems T3-13 to T3-28 in Sections 3.411 and 3.412. These assert the logical synonymy in all finite domains of sixteen pairs of quantificational wffs and each is established by mathematical induction. From these, we argue, all other theorems of a complete quantification theory can be derived. In Chapter 4, by using certain definitions involving negation, this basic list of theorems is reduced to eight; the others are derived from these eight by U-SUB and definitional substitution. Of these eight, three can be replaced by superficial rules requiring only inspection and superficial changes, leaving just five which are sufficient to generate the full variety of logical structures which must be accounted for.

The previous transformation rules require one added constraint. To avoid substitutions by U-SUB which would not preserve logical synonymy an added constraint on the augmented rule of U-SUB is required, namely, that no variable in a predicate schema about to be substituted should have another occurrence in the wff within which the substitution is to be made.

However, in addition to the convenient notational rule of Alphabetic Variance, new logical rules of inference are needed, the derived rules DR3-3a and DR3-3b (these will be reduced to one, DR4-3, in Chapter 4.) Thus our present basic principles of logical inference are three.

V. Transformation Rules

R1. From |= [P SYN Q] and |= [R SYN S], infer |= [P SYN Q(S//R)] [SynSUB]

R3-2. If [R SYN S] [U-SUB]
and (i) $P_i<t_1,\ldots,t_n>$ occurs in R,
and (ii) Q is an h-adic wff, where h $\geq$ n,
and (iii) Q has occurrences of all numerals 1 to n,
and (iv) no variable in Q occurs in R or S **(Added)**
then it may be inferred that |= $[R(P_i<t_1,\ldots,t_n>/Q)$ SYN $S(P_i<t_1,\ldots,t_n>/Q)]$

R2-3. If |= [P<1>] then |= $[Pa_i]$ [INST]

from R2-3 we derive the following Derived Instantiation Rules: **(Added)**

DR3-3a If |= [P<1>] then |= [(∀x)Px] 'CG' (Conjunctive Generalization)
DR3-3b If |= [P<1>] then |= [(∃x)Px] 'DG' (Disjunctive Generalization)
DR3-3c If |= [P<1> Syn Q<1>] then |= [(∀x)Px Syn (∀x)Qx]
DR3-3d If |= [P<1> Syn Q<1>] then |= [(∃x)Px Syn (∃x)Qx]
DR3-3e If |= [P<1> Cont Q<1>] then |= [(∀x)Px Cont (∀x)Qx]
DR3-3f If |= [P<1> Cont Q<1>] then |= [(∃x)Px Cont (∃x)Qx]

3.321 R1—Substitution of Logical Synonyms (SynSUB)

This rule remains as before. The substitutability of alphabetic variants is added as a notational convenience. Also many more pairs of wffs can be proven logically synonymous due to the added Instantiation Rules for Quantifier Introduction, the expanded vocabulary and the expanded capabilities of U-SUB. But the substitution of one of two logically synonymous wffs for the other in a SYN-theorem, is not changed essentially. Given the logical synonymy of two wffs, theoremhood is preserved if one replaces one or more occurrences of the other in any theorem. Besides providing new possibilities for the exercise of SynSUB, the definitions of quantified expressions also provide the link between general and particular forms of expressions.

3.322 DR3-2—Uniform Substitution (U-SUB)

With the introduction of variables and quantifiers, the rule of uniform substitution of predicate schemata for predicate place-holders, is now extended to allow the introduction of complex predicate scehmata (with or without quantifiers) at all occurrences of modes of an abstract atomic predicate schema in a SYN- or CONT-theorem, while preserving the SYN or CONT relation in the result. In R3-2, clause (i) and clause (iv) (which is added to R2-2 of Chapter 2) prevent substitutions which would either introduce expressions which are not wffs, or would not preserve SYN or CONT.

How R3.2 works may be explained as follows:

1. Let 'P' stand for any predicate letter, and the expressions to be replaced be occurrences of modes of $[P<t_1,\ldots,t_n>]$ in some theorem, R.
   Note: 1) The predicate letter 'P' may stand alone. I.e., n=0 and no '$<\ldots>$' follows 'P'. In this case any wff as defined in this chapter, may replace P at all of its occurrences in R. However, this is not the case of interest here.
   2) If $n>0$, then
   a) the subject terms in $<t_1,\ldots,t_n>$, may be either
   (i) individual constants, a, b, c, or
   (ii) arguments position holders, 1, 2, 3, or
   (iii) individual variables, x,y,z,…, [Added]
   b) Different occurrences of '$P_j<t_1,\ldots,t_n>$' may have different subject-terms terms at different positions in '$<t_1,\ldots,t_n>$' in its modes,
   c) all occurrences of $P<t_1,\ldots,t_n>$ are n-adic,—have exactly n argument terms.
   d) Since '$P<t_1,\ldots,t_n>$' occurs only in wffs in R, and since all wffs are closed in A-logic, any individual variables in $P<t_1,\ldots,t_n>$ must be bound to some quantifier in R but outside of $P<t_1,\ldots,t_n>$ (since an atomic predicate schema has no conjunctions, disjunctions or quantifiers in it).

2. Let Q be a wff.;
   Q is suitable to be introduced at all modes of $P<t_1,\ldots,t_n>$ which occurr in R
   iff
   (i) Q is a wff **(hence any variables in Q are bound inQ)**, and
   (ii) Q is h-adic, where $h \geq n$ and $[P<t_1,\ldots,t_n>]$ is n-adic and
   (iii) Q has occurrences of all numerals 1 to n (in any order) in some argument positions in Q, and
   (iv) No individual variable in Q has any other occurrence in the theorem R within which the substitution is to be made.

3. If Q is a suitable predicate schema to substitute for $P<t_1,\ldots,t_n>$ in R,
   that is, if Q satisfies the conditions in 2,
   then Q is introduced at any occurrence of a mode of $P<t_1,\ldots,t_n>$ by
   Step 1) first replacing that occurrence of a mode of $P<t_1,\ldots,t_n>$ by Q, and then
   Step 2) replacing each occurrence of a numeral '**i**' in Q, with the occupant of the **i** th argument position in that mode of $P<t_1,\ldots,t_n>$.[16]

---

16. "Introduction" comes from Quine, *Elementary Logic*, (1965). On page 99 he wrote: "Introduction of a predicate or predicate schema Q at a given occurrence of a predicate letter consists in supplanting that occurrence and the attached string of variables … by an expression which we get from Q by putting the initial variable of the string for '①', the next variable for '②', and so on." In A-logic '$<1>$' is used (instead of '①') etc., as a position-holder for individual constants and numerals as well as individual variables.

4. "Q is <u>substituted</u> for $P<t_1,\ldots,t_n>$ in a wff R", (which means "Q is suitable and Q is <u>introduced</u> at every occurrence of a mode of $P<t_1,\ldots,t_n>$ in wff R")

5. <u>Abbreviation</u>: '$R(P<t_1,\ldots,t_n>/Q)$' for 'the result of substituting Q for $P<t_1,\ldots,t_n>$ in a wff R.

6. The principle of U-SUB in rule R3-2, may be expressed symbolically as,
   If |= [R Syn S] then |= [$R(P<t_1,\ldots,t_n>/Q)$ Syn $S(P<t_1,\ldots,t_n>/Q)$]

The revised U-SUB is the Rule of Inference supported by the principle expressed in 6. This rule says that if [R SYN S] is a theorem, then we may, if we wish, assert as theorem the result of the introducing any suitable Q at each occurrence of a mode of $P<t_1,\ldots,t_n>$ throughout R and S in [R SYN S]. I.e., If [R SYN S] then [$R(P<t_1,\ldots,t_n>/Q)$ SYN $S(P<t_1,\ldots,t_n>/Q)$] may be inferred!

Note that clause (i) of the suitability conditions in step 2 excludes the possibility that the expression Q which is substituted will have a free variable which could be captured by some quantifier elsewhere in R. For by holding that the substituted expression Q must be a wff, and that wffs must have all variables bound, there will be no free variables in Q to be captured. If free variables were allowed in Q, then the introduction might result in moving from a logically synonymous pair to a logically non-synonymous pair.

For example, let Q be '(C<2,1> & F<x>)'
and let us introduce '(C<2,1> & F<x>)'
at each mode of 'P<1,2,>' in [(∀y)(∃x) Pxy SYN (∀x)(∃y) Pyx)]
we get by Step 1) [(∀y)(∃x)(C<2,1> & F<x>) SYN (∀x)(∃y)(C<2,1> &F<x>)]
and by Step 2) the non-theorem [(∀y)(∃x)(C**yx** & Fx) SYN (∀x)(∃y)(C**xy** & Fx)]

This result does not preserve synonymy. For a counter-example in the domain of human relationships, let us interpret 'C<1,2>' as "<1>is a child of <2>" and 'F<1>' as '<1>is a female'; then the wff on the left hand side, '(∀y)(∃x)(Cyx & Fx)', means

"For all y there is some x such that y is child of x, and x is female"

from which "everybody has a mother" follows and is true in fact; while the wff on the right hand side, '(∀x)(∃y)(Cxy & Fx)' means

"For all x there is some y, x is a child of y, and x is female",

from which it follows "everybody is a daughter of someone", which is false. The two wffs can not mean the same if, on instantiating both with the same atomic predicates, one instantiation can be true when the other is false. So synonymy has not been preserved.

We can satisfy all clauses of suitability if we put a constant or an argument-position-holder in place of free variables, and synonymy will be preserved. For example: .

a) If we introduce '(C<2,1> & F<**1**>)' as Q, instead of '(C<2,1> & F<**x**>)', for P<1,2> in [(∀y)(∃x) Pxy SYN (∀x)(∃y) Pyx)]
   we get [(∀y)(∃x)(C**yx** & Fx) SYN (∀x)(∃y)(C**xy** & Fy)] which has as an instantiation
   "'Everyone is the child of some female' Syn 'Everyone is the child of some female' "

b) If we introduce '(C<2,1> & F<**2**>}' as Q, instead of '(C<2,1> & F<**x**>}', for P<1,2> in [(∀y)(∃x) Pxy SYN (∀x)(∃y) Pyx)] we get [(∀y)(∃x)(Cyx & Fy) SYN (∀x)(∃y)(Cxy & Fx)] which has as its instantiation that "Everyone is the child of someone and is female" is synonymous with "Everyone is the child of someone and is female". These are synonymous but false.

Both examples a) and b) preserve logical synonymy, as do other substitutions by R3-2.

Secondly, clause (iv) of the suitability conditions, excludes the possibility of moving from synonymy to non-synonymy because some variable in R becomes captured by quantifier in Q. For clause (iv) prohibits any quantifier in Q which has a variable which occurs elsewhere in R. To see what could happen without clause four, suppose we should introduce .'(∀y)Q<1,y>' for 'P<1>' into the theorem, gotten by Alphabetic Variance, [(∃x)Px SYN (∃y)Py)]:

by step 1) we get '[(∃x)(∀y)Q<1,y> SYN (∃y)(∀y)Q<1,y>)]'
then by step 2) we get '[(∃x)(∀y)Qxy SYN (∃y)(∀y)Qyy)]'

This result is not only is not well-formed (since '(∃y)' has been rendered vacuous); if we drop or ignore the vacuous qunatifier, it has false instances such as that "Somebody loves everybody" is synonymous with "Everybody loves themselves".

If Q contains quantifiers with new variables (i.e., ones that do not occur in R), and satisfies the other clauses, R3-2 preserves Logical Synonymy and Containment. For example, If '(∀z)Q<1,z>' is introduced for 'P<1>' into the theorem, [(∃x)Px SYN (∃y)Py)],

by step 1) we get '[(∃x)(∀z)Q<1,z> SYN (∃y)(∀y)Q<1,z>)]'
then by step 2) we get '[(∃x)(∀y)Qxz SYN (∃y)(∀z)Qyz)]'

and the result preserves synonymy by Alphabetic Variance.

Essentially, R3-2 preserves logically synonymy because the over-all structures of the wffs in the initial SYN or CONT theorem is not changed. All occurrences of an abstract atomic predicate schema are replaced by a more complex or different expression; the result has, or lacks, the same bonds of cross reference to other parts of R that the initial atomic components had.

3.323 DR3-3a and DR3-3b—Quantifier Introduction

The various rules for quantifier introduction are special cases of the general rule of Instantiation derived from theorems and other rules of inference:

R2-3. If |= [P<1>] then |= [Pa] [Instantiation, (INST)]

This says that whatever is logically true of any predicate is logically true of any application of that predicate to any individual. If a predicate has a logical property, every sentence with just that predicate has that logical property. That property belongs to all statements that ascribe that predicate to any object. From this the following rules of quantifier introduction can be derived:

DR3-3a If |= [P<1>] then |= [(∀x)Px] 'CG' (Conjunctive Generalization)
DR3-3b If |= [P<1>] then |= [(∃x)Px] 'DG' (Disjunctive Generalization)
DR3-3c If |= [P<1> Syn Q<1>] then |= [(∀x)Px Syn (∀x)Qx]
DR3-3d If |= [P<1> Syn Q<1>] then |= [(∃x)Px Syn (∃x)Qx]

DR3-3e If $\models$ [P<1> Cont Q<1>] then $\models$ [(∀x)Px Cont (∀x)Qx]
DR3-3f If $\models$ [P<1> Cont Q<1>] then $\models$ [(∃x)Px Cont (∃x)Qx]

A quantified wff means that the predicate of the quantification is ascribed to every individual in some domain, either in a generalized conjunction or a generalized disjunction.

Df '∀'. $[(\forall_k x)Px \text{ SYN}_{df} (Pa_1 \,\&\, Pa_2 \,\&\, \ldots \,\&\, Pa_k)]$
Df '∃'. $[(\exists_k x)Px \text{ SYN}_{df} (Pa_1 \text{ v } Pa_2 \text{ v } \ldots \text{ v } Pa_k)]$

From these definitions and from R2-3 "INST", it follows that if a logical property belongs to a predicate, it belongs to any quantification of that predicate in any domain. Eventually we want to show that no matter what property or relationship is asserted to hold of a predicate, or pair of predicates with an open argument-position-holder, that logical property or relationship will hold of the application of that predicate to any and all individuals.

DR3-3a If $\models$ [P<1>] then $\models$ [(∀x)Px] 'CG'(Conjunctive Generalization)
DR3-3b If $\models$ [P<1>] then $\models$ [(∃x)Px] 'DG'(Disjunctive Generalization)[17]

The rule CG in A-logic is analogous of M-logic's "Universal Generalization" (UG). It generalizes from the logical property of a predicate to a logical property of a quantification of that predicate. The rule DG is analogous in A-logic to M-logic's "Existential Generalization" (EG). It is justified on the same grounds as CG; it proceeds from the logical property of a predicate to a logical property of a disjunctive quantification of that predicate.

It is important to note that these rules fail when factual truth rather than logical truth is involved. In the first place, since predicates are neither true nor false, the principles (using 'T(P<1>)' for ''P<1>' is true'), 'If T[P<1>] then T[(∀x)Px]' and 'If T[P<1>] then T[(∃x)Px]', make no sense; since 'P<1>' is neither true nor false. More important, the statement "if a predicate is factually true of an individual then it is factually true of all individuals" is not valid; instantiations of 'If T[Pa] then T[(∀x)Px]' are clearly often false. The factual truth of a universal proposition does not follow validly from the factual truth of an individual proposition; fallacies of "hasty generalization" fall in this category. CG must not be confused with such a rule. UG, the rule of "Universal Generalization" in M-logic appears similar to such a rule, but in M-logic it has to be hedged with restrictions to prevent its being taken as sanctioning hasty generalizations.

The analogue of DG with factual truth, 'If T[Pa] then T[(∃x)Px]'—which is called 'Existential Generalization" or "EG" in M-logic—is not false or falsifiable, but it can not be based on logical Containment or Synonymy. It is valid in A-logic, but in a limited way. In effect, CG and DG, say that if it is true by logic alone that some logical property holds of some predicate, then it will be true by logic alone that that property holds of a generalized statement with that predicate. Again if it is true that some pair of predicate schemata stand in the relation of logical synonymy or logical containment, then the conjunctive generalization of the two schemata, and the disjunctive generalization of the two schemata, will stand in the same logical relationship.

17. Because of the absence of negation in this chapter this requires two rules. With negation either Rule can be derived from the other, leaving just one rule, DR4-3.

Thus from CG and DG, we derive principles for deriving SYN-theorems and CONT-theorems about quantified expressions, from SYN-theorems and CONT-theorems about unquantified predicate schemata.[18]

If the predicate ‘Syn’ belongs to two predicates with the same set of argument-position-holders by virtue of logic alone, then the predicate ‘Syn’ belongs to the similiar quantifications of the two predicates:

DR3-3c If |= [P<1> Syn Q<1>] then |= [(∀x)Px Syn (∀x)Qx]
DR3-3d If |= [P<1> Syn Q<1>] then |= [(∃x)Px Syn (∃x)Qx]

And since CONT-theorems are reducible to SYN-theorems, the same holds of CONT-theorems:

DR3-3.3e If |= [P<1> Cont Q<1>] then |= [(∀x)Px Cont (∀x)Qx]
DR3-3.3f If |= [P<1> Cont Q<1>] then |= [(∃x)Px Cont (∃x)Qx]

These are all principles of logical inference—conditional statements which say if certain things are logically true of predicates in the antecedents, then certain things will be true of single or multiple applications of that predicate to individuals. These principle do not assert that either the statement in the antecedent or that in the consequent (SYN or CONT) are true logically or in fact. The square brackets in both antecedent and consequent make it clear that they say only that if any result of replacing the variables by permissible expressions were to have certain logical properties then the result of making the indicated replacements in the consequent would have the logical properties or relations ascribed there.

DR3-3c and DR3-3d presuppose that in moving from the logical synonymy of two predicates in [P<1> SYN Q<1>] to [(∀x)Px SYN (∀x)Qx] or to [(∃x)Px SYN (∃x)Qx] the variables of the quantifiers are uniformly substituted for all occurrences of one and the same argument-position-holder (numeral) in P and in Q.

For example, from |= [P<1,2> SYN (P<1,2> v P<1,2>)
we can infer |= [(∀x)P<1,x> SYN (∀x)(P<1,x> v P<1,x>)
or |= [(∀x)P<x,2> SYN (∀x)(P<x,2> v P<x,2>)
but not |= [(∀x)P<1,x> SYN (∀x)(P<x,2> v P<x,2>)
or |= [(∀x)P<x,x> SYN (∀x)(P<x,x> v P<x,x>)

Quine’s *100 says in effect “If A is a tautology then ⊢ (∀x)A” ; that if a wff, A, is tautology, then the closure of its universal quantification, (∀x)A, is a theorem. DR3-3a and DR3- 3b are not exact AL-analogues of *100 or any related theorem in Quine, because 1) without a negation function we do not yet have Quine’s notion of a tautology, 2) we do not count sentential functions (“open sentences” for Quine) as wffs (in place of the free variables in open sentences we have numerals as argument place-holders in predicate schemata), 3) Quine’s *100 allows vacuous quantification, which DR3-3a and DR3-3b do not. The SYN-theorems which follow from them are free of the explicit and implicit references to tautology and quantificational truth in Quine’s *100. Only later, in Ch.5, do we hold that if a **predicate** is

18. Other properties which are determinable by logic alone, and fall under this rule are inconsistency, tautologousness(defined as denials of inconsistencies), contrariety, and others to be discussed in the pages below.

tautologous then any quantification or instantiation of it is tautologous (in the sense of being a denial of an inconsistent expression).

This is the closest we come to an AL-analogue of Quine's *100. Nevertheless, as mentioned above, these rules perform the very important function, also performed by *100 in Quine's system, of preserving theoremhood while introducing quantifiers into wffs which initially lack them.

How DR3-3a to DR3-3f are derived from Df '∀' and Df '∃' and the system of Chapter 1 and 2 is indicated by the following proofs of DR3-3a and DR3-3c. In such proofs we use the expressions '$[\forall_k$-Exp]' and '$[\exists_k$-Exp]' to refer to the abbreviation of a conjunction (or disjunction) in which every conjunct (or disjunct) applies the same predicate, $[P<1>]$, to each member of a set k of individuals (k is finite). In other words, they abbreviate the expressions "For all x in a domain of k individuals" and "For some x in a domain of k individuals", respectively.

DR3-3a If $\models [P<1>]$ then $\models [(\forall x)Px]$

Proof: A. Basis step: For $n=1$, we have $x_1=\{a_1\}$ and:

1) $\models [P<1>]$ [Premiss]
2) $\models [Pa_1]$ [1),.R2-3,MP]
3) $\models [Pa_1 \text{ Syn } (\forall_1 x)Px]$ [$\forall_1$-Exp]
4) $\models [(\forall_1 x)Px]$ [2),3),SynSUB]
5) If $\models [P<1>]$ then $\models [(\forall_1 x)Px]$ [1) to 4),Cond. Proof]

B. Inductive step For $n=k$ we have $\{a_1,a_2,\ldots,a_k\}$, For $n=k+1$ we have $\{a_1,a_2,\ldots,a_k,a_{k+1}\}$,:

6) $\models [(\forall x_k)Px]$ [Premiss]
7) $\models [Pa_{k+1}]$ [1),R2-3]
8) $\models [((\forall x_k)Px \ \& \ Pa_{k+1})]$ [6),7),ADJ]
9) $\models [(\forall x_k)Px \ \& \ Pa_{k+1}) \text{ Syn } ((\forall x_k)Px \ \& \ Pa_{k+1})]$ [T1-11,U-SUB]
10) $\models [(\forall x_k)Px \text{ Syn } (Pa_1 \ \& \ Pa_2 \ \&\ldots\& \ Pa_k)]$ [$\forall_k$-Exp]
11) $\models [(\forall x_k)Px \ \& \ Pa_{k+1}) \text{ Syn } ((Pa_1 \ \& \ Pa_2 \ \&\ldots\& \ Pa_k) \ \& \ Pa_{k+1})]$ [9),10),(SynSUB]
12) $\models [(\forall x_{k+1})Px \text{ Syn } ((Pa_1 \ \& \ Pa_2 \ \&\ldots\& \ Pa_k) \ \& \ Pa_{k+1})]$ [$\forall_{k+}$1-Exp]
13) $\models [(\forall x_{k+1})Px \text{ Syn } ((\forall x_k)Px \ \& \ Pa_{k+1})]$ [12),10),SynSUB]
14) $\models [(\forall x_{k+1})Px]$ [8),13),R1]
15) [If $\models (\forall x_k)Px$ then $\models (\forall x_{k+1})Px$] [By 6)-14), Conditional Proof]

Hence, DR3-3a [If $\models P<1>$ then $\models (\forall x)Px$] [By Steps A and B, Math Ind.]

DR3-3c If $\models [P<1> \text{ Syn } Q<1>]$ then $\models [(\forall x)Px \text{ Syn } (\forall x)Qx]$

Proof: A. Basis step: For $n=1$, we have $x_1=\{a_1\}$ and:

1) $\models [P<1> \text{ Syn } Q<1>]$ [Premiss]
2) $\models [Pa_1 \text{ Syn } Qa_1]$ [1),R2-3]
3) $\models [Pa_1 \text{ Syn } (\forall_1 x)Px]$ [$\forall_1$-Exp]
4) $\models [Qa_1 \text{ Syn } (\forall_1 y)Qy]$ [$\forall_1$-Exp]
5) $\models [Pa_1 \text{ Syn } (\forall_1 y)Qy]$ [2),4),SynSUB]
6) $\models [(\forall_1 x)Px \text{ Syn } (\forall_1 y)Qy]$ [5),3),SynSUB]
7) $\models [(\forall_1 x)Qx \text{ Syn } (\forall_1 y)Qy]$ [Alph.Var.]
8) $\models [(\forall_1 x)Px \text{ Syn } (\forall_1 x)Qx]$ [6),7),SynSUB]
9) If $\models [P<1> \text{ Syn } Q<1>]$ then $\models [(\forall_1 x)Px \text{ Syn } (\forall_1 x)Qx]$ [1) to 8),Cond. Proof]

B. Inductive step: For $n=k$ we have $\{a_1,a_2,\ldots,a_k\}$, for $n=k+1$ we have $\{a_1,a_2,\ldots,a_k,a_{k+1}\}$,:

10) $\models [(\forall x_k)Px \text{ Syn } (\forall x_k)Qx]$ [Assumption]
11) $\models [(Pa_{k+1} \text{ Syn } Qa_{k+1})]$ [1),R2-3]
12) $\models [((\forall x_k)Qx \ \& \ Qa_{k+1}) \text{ Syn } ((\forall x_k)Qx \ \& \ Qa_{k+1})]$ [T1-11,USUB]

13) $\models$ [(($\forall x_k$)Qx & Pa_{k+1}) Syn (($\forall x_k$)Qx & Qa_{k+1})] [12),11),SynSUB]
14) $\models$ [(($\forall x_k$)Px & Pa_{k+1}) Syn (($\forall x_k$)Qx & Qa_{k+1})] [13),10),SynSUB]
15) $\models$ [($\forall x_k$)Px Syn (Pa_1 & Pa_2 &...& Pa_k)] [$\forall_k$-Exp]
16) $\models$ [($\forall x_k$)Qx Syn (Qa_1 & Qa_2 &...& Qa_k)] [$\forall_k$-Exp]
17) $\models$ [($\forall x_k$)Px & Pa_{k+1}) Syn ((Pa_1 & Pa_2 &...& Pa_k) & Pa_{k+1})] [14),15),SynSUB]
18) $\models$ [(($\forall x_k$)Qx & Qa_{k+1}) Syn ((Qa_1 & Qa_2 &...& Qa_k) & Qa_{k+1})] [12),16),SynSUB]
19) $\models$ [($\forall x_{k+1}$)Px Syn ((Pa_1 & Pa_2 &...& Pa_k) & Pa_{k+1})] [$\forall_{k+}$1-Exp]
20) $\models$ [($\forall x_{k+1}$)Qx Syn ((Qa_1 & Qa_2 &...& Qa_k) & Qa_{k+1}))] [$\forall_{k+}$1-Exp]
21) $\models$ [($\forall x_{k+1}$)Px Syn (($\forall x_k$)Px & Pa_{k+1})] [19),15),SynSUB]
22) $\models$ [($\forall x_{k+1}$)Px Syn (($\forall x_k$)Qx & Qa_{k+1})] [14),21),SynSUB]
23) $\models$ [($\forall x_{k+1}$)Qx Syn (($\forall x_k$)Qx & Qa_{k+1})] [20),16),SynSUB]
24) $\models$ [($\forall x_{k+1}$)Px Syn ($\forall x_{k+1}$)Qx] [22),23) SynSUB]

Hence:

25) If $\models$ [($\forall x_k$)Px Syn ($\forall x_k$)Qx] then $\models$ [($\forall x_{k+1}$)Px Syn ($\forall x_{k+1}$)Qx]
[By 10)-24), Conditional Proof]

Hence, DR3-3c If $\models$ [P<1> Syn Q<1>] then $\models$ [($\forall$x)Px SYN ($\forall$x)Qx)]
[By Steps A and B, Math Ind.]

3.4 Quantification Theorems (T3-11 to T3-47)

Quantified wffs represent the Boolean expansions of their matrices in any unspecified domain of reference. Since Boolean expansions eliminate variables and quantifiers, the logical relations asserted in quantification theorems represent relations of logical structures which are completely covered by the quantifier-free axioms and substitution rules of Chapters 1 and 2.

Significant quantificational SYN-theorems all have to do with scope changes for quantifiers; Boolean expansions of the two wffs which are synonymous have syntactically different logical structures. Neither the two wffs nor their Boolean expansions differ in the elementary predicate schemata involved. No quantified wff can be synonymous with another if there is an elementary predicate schema with modes that occur in one but not the other. No quantified wff can logically contain another wff if there is an elementary predicate schema with modes which occur in the latter wff but not the former.

The quantificational theorems of logical synonymy and containment below are divided into three groups: logical synonymies based on re-ordering rules only (T3-11 to T3-18, Section 3.411), logical synonymies based on distribution rules (T3-19 to T3-32, Section 3.412), and theorems of logical containment (T3-33 to T3-47, Section 3.42).

The SYN-theorems of quantification are logically prior to the CONT-theorems. A wff, P, logically contains Q if and only if $\models$ (P SYN (P&Q)). From the latter it does not follow that Q contains P; logical containment differs from logical synonymy though dependent upon it. Thus all candidates for primitive theorems in the axiomatic development are SYN-theorems. The basic theorems used in proofs come from the first two groups. In the system of this chapter, the basic SYN-theorems are T3-19 to T3-32; the CONT-theorems (T3-33 to T3-47) are all derived from them.

The first group of SYN-theorems (T3-11 to T3-18) implicitly involve nothing more than &-orderings based on Ax1-01, Ax1-03 and Ax1-05 or v-orderings based on Ax1-02, Ax1-04 and Ax1-06. The second group (T3-19 to T3-32) yield more sophisticated theorems, including the "axiom theorems". They are based on distribution principles Ax1-07 or Ax1-08 which involve logical structures which mix & and v.

The third group, in Section 3.333, contains only CONT-theorems whose converses are not CONT-theorems. These have the numbers T3-33 to T3-47. (All SYN-theorems and their converses are also

CONT-theorems; but this group contains no CONT-theorems which are also SYN-theorems). The SYN-theorems needed to establish these CONT-theorems are the odd-numbered theorems from T3-13 to T3-29. The even-numbered theorems from T3-14 to T3-30 are the duals of these; none of them yield non-convertible logical containments. Each of the CONT-theorems in this third group is an AL-analogue of one of Quine's theorem schemata in quantification theory. But the dual of a CONT-theorem is not a CONT-theorem unless the two components are SYN, which none in this group are.

From every CONT-Theorem we may deduce two SYN-theorems: first an odd-numbered SYN-theorem, which has the antecedent on the left and the antecedent conjoined with what it contains on the right; and second, an even-numbered SYN-theorem just like the odd-numbered SYN-Theorem except for thorough-going interchange of '&'s and 'v's, and of conjunctive and disjunctive quantifiers.

3.41 Theorem of Quantificational Synonymy (SYN)

For a complete system of theorems we can distinguish three kinds of SYN-theorems: those based solely on Re-Ordering Rules, those also based on Distribution Rules, and those based on logical relationships between modes of a predicate. Basic to them all are the theorems T3-11 and T3-12 which are based directly on the definitions of quantifiers.

T3-11. [((∀$_n$x)Px) SYN (Pa$_1$ & P$_2$ &...& P$_n$)] [Df '(∀x)Px']
T3-12. [((∃$_n$x)Px) SYN (Pa$_1$ v P$_2$ v...v P$_n$)] [Df '(∃x)Px']

3.411 Based on Re-Ordering Rules

The first six quantificational SYN-theorems are based only on rules of &-ordering or v-ordering, and are 'SYN'-for-'≡' analogues of Quine's metatheorems *119, *138, *140, *141, *157 and *160 in his *Mathematical Logic*:

T3-13. [(∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx)]	*140
T3-14. [(∃x)(Px v Qx) SYN ((∃x)Px v (∃x)Qx)]	*141
T3-15. [(∀x)(∀y)Rxy SYN (∀y)(∀x)Rxy]	*119
T3-16. [(∃x)(∃y)Rxy SYN (∃y)(∃x)Rxy]	*138
T3-17. [(∀x)(P & Qx) SYN (P & (∀x)Qx)]	*157
T3-18. [(∃x)(P v Qx) SYN (P v (∃x)Qx)]	*160

The odd-numbered theorems require only [&-ORD]; they are derivable in any finite domain using only Axioms Ax1-01, Ax1-03 and Ax1-05. The even-numbered theorems involve only [v-ORD]; they are derivable in any finite domain using only Axioms Ax1-02, Ax1- 04 and Ax1-06. The first four require only Commutation and Association; the last six require Idempotence as well. None of them require distributions principles Ax1-07 or Ax1-08.

Although is is possible to prove by mathematical induction that each of these six theorems holds in all finite domains, I have put only this sort of proof for T3-13 in APPENDIX III; below I provide only examples showing how proofs of these theorems might go in domains of two or three members. This is because the principle of DR1-03 ("&-ORD") and DR1-04 ("v-ORD") seem intuitively clear and sufficient to establish that theorems based solely on ordering rules that will hold in every domain.

T3-13 and T3-14. The first two theorems are the AL-analogues of Quine's *140 and *141 respectively. Only Commutation and Association are needed.

In a domain of 2, T3-13 is an instance of theorem T1-12 which is based solely on &-commutation and &-association:

1) ((P & Q) & (R & S))	SYN ((P & R) & (Q & S))	[T1-12]
2) ((Pa&Qa) & (Pb&Qb))	SYN ((Pa&Pb) & (Qa&Qb))	[1),U-SUB]
3) $(\forall_2 x)$(Px & Qx)	SYN (($\forall_2$x)Px & ($\forall_2$x)Qx)	[2),$\forall_2$-Abbr]

In a domain of three, T3-13 is established in five steps, using T1-12 twice:

T3-13. (∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx)
Proof for a domain of 3:

1)	($\forall_3$x)(Px & Qx)SYN($\forall_3$x)(Px & Qx)	[T1-11]
2)	(") SYN ((Pa & Qa) & ((Pb & Qb) & (Pc & Qc)))	[1),$\forall_3$-Exp]
3)	(") SYN ((Pa & Qa) & ((Pb & Pc) & (Qb & Qc)))	[2),T1-12,R1]
4)	(") SYN ((Pa & (Pb & Pc)) & (Qa & (Qb & Qc)))	[3),T1-12,R1]
5)	($\forall_3$x)(Px & Qx) SYN (($\forall_3$x)Px & ($\forall_3$x)Qx))	[(5),$\forall_3$-Abbr(twice)]

In any domain of n > 2, T3-13 can be established in n+2 steps, where the steps between Steps 2) and 2n) are n-1 iterated uses of T1-12 with R1 on the Boolean expansion of the left-hand wff (as in the proof for a domain of three).

It seems obvious from the preceding considerations, that T3-13 can be proved in every finite domain. Nevertheless, a proof by mathematical induction is provided in Appendix III. Proofs in any, or all, domains of the dual of T3-13, namely

T3-14. (∃x)(Px v Qx) SYN ((∃x)Px v (∃x)Qx) (The AL-analogue of *141)

are dual proofs of the corresponding proofs of T3-13. In a domain of 2) the proof of T3-14 is a substitution instance of the dual of T1-12 namely,

T1-13. |= [((P v Q) v (R v S)) SYN ((P v R) v (Q v S))]

and in any domain of n > 2, T3-14 is established in n+2 steps, with n-1 iterated uses of T1-13 with R1 between steps 2) and 2n), instead of T1-12 on the Boolean expansion of the left-hand wff.

T3-15 and T3-16. In the next two theorems (the AL-analogues of Quine's *119 and *138) synonymy is preserved in two cases of quantifier position reversal. By "quantifier position reversal" I mean cases in which one quantifier has another quantifier within its immediate scope and their relative positions are changed without any change in their common matrix. In some such cases synonymy is preserved, namely, when all quantifiers are conjunctive, or all quantifiers are disjunctive. This what T3-15 and T3-16 assert. When one is disjunctive and the other conjunctive the best we can get is a containment—see T3-27 and T3-37.[19]

---

19. The non-synonymy is seen in a "truth-table" of the expansions of (∀x)(∃y)Rxy and (∃y)(∀x)Rxy in a domain of two individuals, {a,b}:

(∀x)(∃y)Rxy		(∃y)(∀x)Rxy
((Raa v Rab) & (Rba v Rbb))	SYN	((Raa & Rba) v (Rab & Rbb))
F T T **T** T T F		F F T **F** T F F

However, T3-37 [(∃y)(∀x)Rxy CONT (∀x)(∃y)Rxy)] is proved in Section 3.233; it follows by definition of ' CONT' f rom T3-27, proved in Section 3.232.

T3-15. (∀x)(∀y)Rxy SYN (∀y)(∀x)Rxy (AL-analogue of Quine's *119)
T3-16. (∃x)(∃y)Rxy SYN (∃y)(∃x)Rxy (AL-analogue of Quine's *138)

In a domain of two, T3-15 (like T3-13) is a substitution instance of T1-12:

1) ((P & Q) & (R & S)) SYN ((P & R) & (Q & S)) [T1-12]
2) ((Raa&Rab) & (Rba&Rbb)) SYN ((Raa&Rba) & (Rab&Rbb)) [1),U-SUB]
3) (($\forall_2$y)Ray & ($\forall_2$y)Rby) SYN (($\forall_2$x)Rxa & ($\forall_2$x)Rxb)) [2),($\forall_2$-Abbr.(four times)]
4) ($\forall_2$x)($\forall_2$y)Rxy SYN ($\forall_2$y)($\forall_2$x)xy [3),($\forall_2$-Abbr.(twice)]

In a domain of three members, {a,b,c}, the logical synonymy asserted in T3-15 can be established in 9 steps using T1-12 four times.(To help understand what happens in the steps 4) through 7) I put in **bold** the two conjuncts which interchange positions from the previous step using T1-12 and R1).[20]

T3-15. (∀x)(∀y)Rxy SYN (∀y)(∀x)Rxy *119
<u>Proof in a domain of three {a,b,c}</u>:
1) ($\forall_3$x)($\forall_3$y)Rxy SYN ($\forall_3$x)($\forall_3$y)Rxy [T1-11]
2) (") SYN (($\forall_3$y)Ray & ($\forall_3$y)Rby & ($\forall_3$y)Rcy) [1),$\forall_3$-Exp]
3) (") SYN ((**Raa** & (Rab & Rac))
&(**Rba** & (Rbb & Rbc))
&(**Rca** & (Rcb & Rcc))) [2),$\forall_3$-Exp (three times)]
4) (") SYN ((**Raa** & (**Rba** & **Rca**))
&(Rab & (Rbb & Rcb))
&(Rac & (Rbc & Rcc))) [3),&-0RD,R1]
5) (") SYN (($\forall_3$x) Rxa & ($\forall_3$x) Rxb & ($\forall_3$x) Rxc) [4),$\forall_3$-Abbr(three times)]
6) ($\forall_3$x)($\forall_3$y)Rxy SYN ($\forall_3$y)($\forall_3$x)Rxy [5),$\forall_3$-Abbr]

(Note that step 5) could have gotten (($\forall_3$y) Rya & ($\forall_3$y) Ryb & ($\forall_3$y) Ryc) on the right side so that 6) would assert ($\forall_3$x)($\forall_3$y)Rxy SYN ($\forall_3$x)($\forall_3$y)**Ryx**. But this makes no difference because ($\forall_3$x)($\forall_3$y)**Ryx** and ($\forall_3$y)($\forall_3$x)Rxy are alphabetic variants.)

The number of steps required to establish T3-15 in a domain of n members, if we use the same strategy with T1-12 would be $((n-1)^2 + 5)$. The five steps are to remove and re-introduce quantifiers; the $(n-1)^2$ steps involve iterated uses of T1-12 with R1.

The dual of T3-15 is, T3-16. (∃x)(∃y)Rxy SYN (∃y)(∃x)Rxy. (The AL-analogue of *138)

Proofs of this theorem in any, or all, finite domains would be dual proofs of the proofs T3-15 using T1-13 instead of T1-12.

Thus T3-15 and T3-16 show that if two quantifiers occur immediately next to each other, have the same scope, and both quantifiers are conjunctive or both quantifiers are disjunctive, then logical syn-

---

20. Note that in substituting one side of T3 for the other, the second and third of the four conjuncts interchange position, while the first and fourth stay in place:
1) ((P&Q)&(R&S)) SYN ((P&R)&(Q&S)) [T1-12]
2) (..........) SYN ((P&**R**)&(**Q**&S)) [Premiss]
3) (..........) SYN ((P&**Q**)&(**R**&S)) [2),T1-12,R1]

onymy is preserved through a change in the left-to-right order of occurrence (re-ordering) of the quantifiers.

T3-17 and T3-18. These theorems say that positioning of a component P inside or outside of the scope of a quantifer does not affect sameness of meaning, provided the variable in the quantifier does not occur unbound in P.

T3-17. [(∀x)(P & Qx) SYN (P & (∀x)Qx)]
T3-18. [(∃x)(P v Qx) SYN (P v (∃x)Qx)]

These are ‘Syn’-for-‘≡’ analogues of Quine’s metatheorems,

*157. If x is not free in P then [(∀x)(P & Qx) ≡ (P & (∀x)Qx)]
*158. If x is not free in P then [(∃x)(P v Qx) ≡ (P v (∃x)Qx)]

In A-logic, it is not necessary to state the condition ‘if x is not free in P’; because (i) obviously, P , by itself does not have ‘x’ attached, and (ii) P cannot, by U-SUB, be replaced by an expression with any umbound variable not present in P. To preserve theoremhood only well-formed formulae (in which all variables must be bound) can be used to introduce a replacement for P.

In A-logic, the phrase ‘x is not free in P’ applies to any one of three sorts of expressions which could be parts of a wff with no free occurrences of x:

1) Unquantified wffs, including atomic wffs, and compound wffs without any variables at all.
   E.g., ‘P_1’, ‘Pa’, ‘$P\langle 1\rangle$’, ‘$P_2\langle 1,a\rangle$’, ‘$P_3\langle b,1\rangle$’, ‘$P_2\langle 1,2\rangle$
2) Quantified wffs, without any free variables, ‘x’ or otherwise;
   E.g., ‘$(\exists x)(P_1\langle 1,x\rangle$’, ‘$(\exists x)P_3\langle 1,x,a\rangle$’ and ‘(∃y)(∀x)(Py & Rxy)’ and
3) Parts of a wff having some variable(s) other than ‘x’ but not ‘x’.
   E.g., ‘Py’ in ‘(∃y)(∀x)(Py & Rxy)’.

The expressions under 1) and 2) are wffs of A-logic and have no free variables of any sort. As wffs no variable, x or otherwise, can be free in them. Expressions of the third kind standing alone are not wffs in A-logic precisely because a variable is not bound. Rather, they stand for modes of an abstract predicate within some wff. For example, ‘Py’ in ‘(∃y)(∀x)(Py & Rxy)’ stands for the modes of ‘$P\langle 1\rangle$’, such as ‘Pa’,’Pb’, etc. which occur in the Boolean expansion of ‘(∃y)(∀x)(Py & Rxy)’.

The Rules of Formation in Chapter 2 say that if a predicate letter stands alone it is a wff, and that the result of prefixing a predicate letter to $\langle t_1,\ldots,t_n\rangle$ is a wff if each t_i is either a numeral or an individual constant is a wff. This is unchanged in the Chapter 3 formation rule,

FR3-3. If $P_i \in \{PL\}$ and $\{t_1,\ldots,t_n\} \in \{\{APH\} \cup \{IC\}\}$, then $P_i\langle t_1,t_2,\ldots,t_n\rangle$ is a wff.

FR3-4 introduces variables into a well-formed schemata only with the quantifiers to which they are bound:

FR3-4. If $P_i\langle t_1,\ldots,t_n\rangle$ is a wff and $t_i \in \{APH\}$,
then $(\forall x)P_i\langle t_1,\ldots,t_n\rangle t_i/x$ and $(\exists x)P_i\langle t_1,\ldots,t_n\rangle t_i/x$ are wffs. (Added in Ch. 3)

In R3-2 (U-SUB) parts of wffs (like 'Py' in '(∀y)(∀x)(**Py** & Qx) SYN (∀y)(**Py**&(∀x)Qx)') are not wffs; rather they are stand-ins for a set of modes of the abstract wff 'P<1>'. A well- formed schema, Q, is introduced in a way that insures that no changes in the logical structure of the Boolean expansions of the two wffs will lead from logical synonymy to non-synonymy. The addition of Clause (iv) in R3-2 does not say that 'P_i<t_1,...,t_n> is a wff when one of the terms is a variable. It simply says that parts of that kind in a wff may be replaced by different parts which contain the same variable(s), but is gotten by replacing argument-position-holders of the wff Q by those variables in a certain way which will preserve the initial logical property or relation.

To prove T3-17 and T3-18 it must be proven that if the condition described by "x is not free in ..." is met, then the pairs of wffs in each will be logically synonymous, whereas if it is not met, the two wffs may not be synonymous.

To see what must be avoided consider the following case in which a quantifier captures a variable as the result of changing its scope. Suppose the domain is people and we say, "All (people) are Red-blooded and some (people) are Pacifists and Quakers". This is factually true and it has the logical form depicted by

(∀x)(Rx &(∃x)(Px & Qx))

If we were to substitute 'Px' (in which 'x' is free) for 'P' (in which 'x' is not free) in T3-18 we would get, an expression which is not well-formed on the right,

[(∃x)(Px & Qx) Syn (Px & (∃x)Qx))],

and persisting in our error, from this we would derive, by SynSUB into an instance of T1-11,

(∀x)(Rx &**(∃x)(Px & Qx))** Syn (∀x)(Rx & **(Px & (∃x)Qx))**

in which both expressions are well-formed. But on the interpretation above the right hand side reads "All (people) are Red-blooded Pacifists and some (people) are Quakers" which is false because not all people are pacifists. Obviously, the two wffs are not logically synonymous, since their instances are not. In moving '(∃x)' to the right, 'Px' was released from bondage to '(∃x)' and was captured by '(∀x)'.

What happens becomes clear in the Boolean expansions. In a domain of two, the Boolean expansion of '(∀x)(Rx &**(∃x)(Px & Qx)**)' is:

(Ra & ((**Pa** &Qa)**v**(**Pb**&Qb)) & Rb & ((**Pa** & Qa)**v**(**Pb**&Qb))

while the Boolean expansion of '(∀x)(Rx & **(Px &(∃x)Qx)**)' is:

(Ra & **Pa** & (Qa v Qb)) & (Rb & **Pb** & (Qa & Qb))

The second contains and entails (**Pa & Pb**) (everybody in the domain of two is a pacifist), whereas the first does not. This is how changes in the scope of a quantifier can result in different meanings.

This result was only possible because two quantifiers, '(∀x)' and '(∃x)' used the same variable.If all quantifers use a different variable no quantifier can capture another variable when it changes scope. But there was nothing wrong with having the same variable in two quantifiers. Both expressions were well-formed, but they had different bondage patterns which led to non-synonymy.

Since |= [(∃y)(Py & Qy) Syn (∃x)(Px & Qx)] by Alphabetic Variance, it follows by SynSUB that |= [(∀x)(Rx &(∃x)(Px & Qx)) Syn (∀x)(Rx &(∃y)(Py & Qy)).

Thus '(∀x)(Rx &(∃y)(Py & Qy))', is an alternative symbolism for "All (people) are Red- blooded and some (people) are Pacifists and Quakers". If we were to substtute 'Py' (in which 'y' is free) for 'P' (in which 'y' is not free} in T3-18 we would get, again, an expression which is not well-formed on the right, namely,

[(∃y)(Py & Qy) Syn (Py & (∃y)Qy))],

and persisting in our error, from this we would get, by SynSUB,

(∀x)(Rx &**(∃y)(Py & Qy))** Syn (∀x)(Rx & **(Py & (∃x)Qy))**

In this case, however, we move not from a truth to a falsehood, but from a well-formed wff to one that is not well-formed, since 'Py' is not bound to any quantifier and the whole can not be expanded clearly.

The important point is not that P must be a well-formed formula without any free variable. Rather, a quantifier must not capture the variable of some component which was previously bound to a different quantifier in a way that leads from synonymy to non-synonymy, or change the expression from a meaningful well-formed structure, to one which is not well-formed and as clear in meaning.

The point is not even that shifts in quantifiers' scope should not change the logical structure. All Axioms and significant SYN-theorems of A-logic have differences of logical structure on the two sides of 'SYN', but in theorems the differences do not change the meaning of what is said about any subjects. Rather is should not change the structure in a way that leads from synonymy to non-synonymy.

In contrast to the mis-applications of T3-18 just discussed, if 'P' is replaced by a wff, Pa or P<1>, which has no individual variables, or by any wff in which all variables are bound, synonymy is preserved: The Boolean expansion of '(∀x)(Rx &(∃x)(**Pa** & Qx))' in a domain of 2 is:

((Ra & ((**Pa** &Qa)v(**Pa**&Qb))) & (Rb & ((**Pa** & Qa)v(**Pa**&Qb))))

while the Boolean expansion of '(∀x)(Rx & (Pa &(∃x)Qx))' in a domain of 2 is:

((Ra & **Pa** & (Qa v Qb)) & (Rb & **Pa** & (Qa v Qb)))

and these two expansions are proved synonymous by &v-Distribution.

There are many cases in which a quantifier changes its scope without changing the referential meaning of the overall expression. Consider the the theorem,

|= [(∃y)(∀x)(Rxy & Py) Syn (∃y)(Py & (∀x)Rxy)] which is roughly the form of the statement,
' "There is someone everybody respects and is a Priest"
is referentially synonymous with
"There is someone who is a Priest and everybody respects him" '.

In this case the quantifier '(∀x)' has 'Py' in its scope on the left, but not in its scope on the right. The proof of logical synonymy in this case, begins with T3-17: and finishes using DR3-3d.,

1) [(∀x)(P & Qx) SYN (P & (∀x)Qx)] [T3-17]
2) [(∀x)(Qx & P) SYN (P & (∀x)Qx)] [1),&-COMM, SynSUB]
3) [(∀x) (Rx<1> & P) Syn (P & (∀x)Rx<1>)] [2),U-SUB ‘R<1,2>’ for ‘Q<1>’]
4) [(∀x) (Rx<1> & P <1>) Syn (P<1> & (∀x)Rx<1>)] [3),U-SUB ‘P<1>’ for ‘P’]
5) [(∃y)(∀x)(Rxy & Py) Syn (∃y)(Py & (∀x)Rxy)] [4),DR3-3d]
or 5') [(∀y)(∀x)(Rxy & Py) Syn (∀y)(Py & (∀x)Rxy)] [4),DR3-3c]

Note that DR3-3 requires that we introduce a new variable in step 5.??
To see why (∃y)(∀x)(Py & Rxy) is logically synonymous with ‘(∃y)(Py & (∀x)Rxy)’ compare the Boolean expansions of each in domain of three. (Substitutions for ‘Py’ are in **bold**):

(∃y) (∀x) (Py&Rxy)	(∃y) (Py & (∀x)Rxy)
(((**Pa**&Raa) & (**Pa**&Rba) & (**Pa**&Rca))	((**Pa** & (Raa&Rba&Rca))
v ((**Pb**&Rab) & (**Pb**&Rbb) & (**Pb**&Rcb))	v (**Pb** & (Rab&Rbb&Rcb))
v ((**Pc**&Rac) & (**Pc**&Rbc) & (**Pc**&Rcc)))	v (**Pc** & (Rac&Rbc&Rcc)))

Each pair of first disjuncts, second disjuncts and third disjuncts (by ∃-Exp) in the two wffs are logically synonymous. The two members in each pair contain all and only the same set of elementary wffs, and while there are more substitution instances of ‘P<1>’ in the left-hand expressions than the right, the added occurrences on the left in each row are all the same wff and can be reduced by &-ORD and &-IDEM to the single occurrence on the right. Thus by jumping (∀x) over ‘Py’ the logical structure is altered, but not in a way that affects logical synonymy.

We prove that T3-17 will hold in all domains by a sort of mathematical induction. If the two are SYN in a domain of one individual, and you can prove that if they are SYN in a domain of n individuals then they are SYN in a domain of n+1 individuals, then you have shown that they SYN in all finite domains.

In a domain of one individual, T3-17. [(∀x)(P & Qx) SYN (P & (∀x)Qx)] is simply
[(Pa& Qa) SYN (Pa & Qa)], which is a U-SUB instance of T1-11.

In a domain of two members T3-17 becomes simply an instance of the converse of T1-14:

1) (P & (Q & R)) SYN ((P & Q) & (P & R)) [T1-14]
2) ((P & Q) & (P & R)) SYN (P & (Q & R)) [1),DR1-01]
3) ((P & Qa) & (P & Qb)) SYN (P & (Qa&Qb)) [2),U-SUB]
4) $(\forall_2 x)$(P & Qx) SYN (P & $(\forall_2 x)$Qx) *157 [3),$\forall_2$-Abbr]

In a domain of three the proof requires two applications of T1-14:

T3-17. [(∀x)(P & Qx) SYN (P & (∀x)Qx)] *157
<u>Proof</u> in a domain of 3, {a,b,c},

1) $(\forall_3 x)$(P & Qx) SYN $(\forall_3 x)$(P & Qx) [T1-11]
2) $(\forall_3 x)$(P & Qx) SYN ((P & Qa) & ((P & Qb) & (P & Qc))) [1),$\forall_3$-Exp]
3) $(\forall_3 x)$(P & Qx) SYN ((P & Qa) & (P & (Qb & Qc))) [2),T1-14,R1]
4) $(\forall_3 x)$(P & Qx) SYN (P & (Qa & Qb & Qc)) [3),T1-14,R1]
5) $(\forall_3 x)$(P & Qx) SYN (P & $(\forall_3 x)$Qx) [4),$\forall_3$-Abbr]

In any domain of n > 2, T3-17 can be established in n+2 steps, where the steps between Steps 2) and 2n) are n-1 iterated uses of T1-14 with R1 on the Boolean expansion of the left-hand wff (as in the proof above for a domain of three). Using T1-14 in this manner, it is proved by mathematical induction to hold in all finite domains.

The dual of T3-17 (i.e., the AL-analogue of Quine's *158) is T3-18:

T3-18. [(∃x)(P v Qx) SYN (P v (∃x)Qx)] *158

Its proof is similar to that of T3-17, exept that each step is the dual of the corresponding step in the proof oT3-17. Thus T1-15 replaces T1-14, $\exists_3$-Exp replaces $\forall_3$-Exp, etc. Whereas T3-17 and T3-18, and all follow from them are based on the ordering axioms, Ax.1-01 to Ax.1-06, we shall see in the next section how T3-19 and T3-20 are based on distribution axioms 1-07 and 1-08.

3.412 Based on Distribution Rules

In this section we develop ten SYN-theorems of quantification theory. Only two of these, T3-19 and T3-20, are 'SYN'-for-'≡' analogues of metatheorems presented in Quine's *Mathematical Logic*. The others, whose analogues Quine did not dignify with names, are introduced to facilitate proofs of the CONT-theorems in Section 3.3 which are analogous to more metatheorems of Quine.

T3-19. [(∃x)(P & Qx) SYN (P & (∃x)Qx)] *158
T3-20. [(∀x)(P v Qx) SYN (P v (∀x)Qx)] *159
T3-21. [(∀x)Px SYN ((∀x)Px & (∃x)Px)]
T3-22. [(∃x)Px SYN ((∃x)Px v (∀x)Px)]
T3-23. [(∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Px)]
T3-24. [(∀x)(Px v Qx) SYN ((∀x)(Px v Qx) v (∀x)Px)]
T3-25. [((∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx))]
T3-26. [((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)(Px v Qx))]
T3-27. [(∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]
T3-28. [(∀y)(∃x)Rxy SYN ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]

We take up these theorems by pairs of duals, in the listed order.

Given the definitions of quantificational wffs and the principles of mathematical induction, all of the theorems of standard quantification theory are developed from the principles of logical synonymy and containment presented in Chapter 1, together with the definitions and rules of this chapter. In effect they all use the definition of quantification in terms of the concept of Boolean expansions. However, such proofs using mathematical induction are laborious and difficult. Further, mathematical induction is technique for justifying fixed theorems one at a time, rather than a rule of inference which generates theorems with new logical structures from previous theorems; therefore it can be of no help in establishing completeness with respect to theorems not yet presented.

The possiblity of a completeness proof is presented, if we can 1) establish a small number of quantificational theorems by mathematical induction, then 2) derive all of the other quantificational theorems we need directly from that small group, without any more mathematical inductions.

Many quantificational theorems can be derived from the principles of Chapter 1 using the Instantiation Principles
DR3-3c If |= [P<1> Syn Q<1>] then |= [(∀x)Px Syn (∀x)Qx]
DR3-3d If |= [P<1> Syn Q<1>] then |= [(∃x)Px Syn (∃x)Qx]
DR3-3e If |= [P<1> Cont Q<1>] then |= [(∀x)Px Cont (∀x)Qx]
DR3-3f If |= [P<1> Cont Q<1>] then |= [(∃x)Px Cont (∃x)Qx]

Others can be derived using U-SUB as developed in Section 3.321. The definitions of quantifiers can yield another large class of theorems. But to get the full variety of theorems something more, employing SynSUB, is required.

In subsequent sections we prove four quantificational theorems by mathematical induction. These, with their duals, are sufficient by themselves to derive all negation-free quantificational SYN-theorems and CONT-theorems. After negation is introduced we can derive the duals of these four plus all other SYN-and CONT-theorems essential to standard quantification theory using Double Negation and some standard definitions which involve negation. In Chapter 5, with the definitions of Inconsistency and Tautologies, the quantificational theorems or standard logic are derived as tautologies of A-logic.

The fact that these theorems are derivable from the theory of logical synonymy and containment, and that all derivations use only rules of inference from that theory, shows that quantification theory in general can be based upon the semantic theory of negation-free logical containment independently of truth-functional semantics and the concepts of tautology and inconsistency.

The next ten theorems are shown to hold in all finite domains by mathematical induction, and this is sufficient to derive axiomatically all possible SYN- and CONT-theorems in negation-free quantification theory.

T3-19 and T3-20. The first two basic-theorems in this section are rules of passage T3-19 and T3-20, are based on v&-Distribution (Ax1-07) and &v-Distribution (Ax1-08) respectively.[21]
They are 'SYN'-for-'≡' analogues of Quine's

*158 If x is not free in P then [(∃x)(P & Qx) SYN (P & (∃x)Qx)]
and *159 If x is not free in P then [(∀x)(P v Qx) SYN (P v (∀x)Qx)]

except that the condition, 'x is not free in P' is dropped for reasons given in the previous section.

The first, T3-19. [(∃x)(P & Qx) SYN (P & (∃x)Qx))] is established in any finite domain by iterated uses of Ax1-08. In a domain of 2 members, its Boolean expansion is simply an instance of the converse of &v-Distribution, i.e., Ax1-08:

T3-19. [(∃x)(P & Qx) SYN (P & (∃x)Qx))]
Proof in a domain of 2:
1) [((P & Qa) v (P & Qb)) SYN (P & (Qa v Qb))] [Ax1-08,DR1-01)]
2) $(\exists_2 x)$(P & Qx) SYN (P & $(\exists_2 x)$Qx)) [1),$\exists_2$-Abbr]

---

21. Jacques Herbrand introduced the term "Rules of Passage". Cf. Logical Writings, Jacques Herbrand, edited by Warren Goldfarb, Harvard University Press,1971, p 225). He included both (1) the rules above for moving wffs with no occurrences of a given variable in and out of the scope of any quantifiers with that variable, and (2) Rules of Quantifier Interchange (based on DeMorgan Theorems, Double Negation and Quantifier defintions). I use "Rules of Passage" only for the former; only those rules bring about changes of positions in logical structures.

The dual of T3-19 in a domain of 2 is simply an instance of v&-Distribution, i.e., Ax1-07:

T3-20. [(∀x)(P v Qx) SYN (P v (∀x)Qx)) *159
Proof in a domain of 2:
1) ((P v Qa) & (P v Qb)) SYN (P v (Qa & Qb)) [Ax1-07,DR1-01]
2) ($\forall_2$x)(P & Qx) SYN (P & ($\forall_2$x)Qx)) [1),$\forall_2$-Abbr]

The proof that T3-19 holds in all finite domains is by mathematical induction:

T3-19. (∀x)(P & Qx) SYN (P & (∃x)Qx)
Proof: Let x_n= Ua_n={a_1,a_2,...,a_k,a_{k+1},...}, ^{nN}.
A. Basis step:
For n=1, we have x_1={a_1} and:
1) ($\exists_1$x)Qx SYN Qa_1 [$\exists_1$-Exp]
2) ($\exists_1$x)(P & Qx) SYN (P & Qa_1) [$\exists_1$-Exp]
3) ($\exists_1$x)(P & Qx) SYN (P & ($\exists_1$x)Qx) [2),1),R1]
B. Inductive step: For n=k we have x_n=x_k={a_1,a_2,...,a_k},
for n=k+1 we have x_n=x_{k+1}={a_1,a_2,...,a_k,a_{k+1}}, and:
1) ($\exists_k$x)(P & Qx) SYN (P & ($\exists_k$x)Qx) [Assumption]
2) ($\exists_k$x)Qx SYN (Qa_1 v Qa_2 v...v Qa_k) [$\exists_k$-Exp]
3) ($\exists_k$x)(P&Qx) SYN ((P&Qa_1)v(P&Qa_2)v...v(P&Qa_k)) [$\exists_k$-Exp]
4) ($\exists_{k+1}$x)Qx SYN (Qa_1 v Qa_2 v... v Qa_k v Qa_{k+1}) [$\exists_{k+1}$-Exp]
5) ($\exists_{k+1}$x)Qx SYN ((Qa_1 v Qa_2 v... v Qa_k) v Qa_{k+1}) [4),v-ORD]
6) ($\exists_{k+1}$x)Qx SYN ((E_kx)Qx v Qa_{k+1}) [5),2),R1]
7) ($\exists_{k+1}$x)(P & Qx) SYN ((P&Qa_1)v(P&Qa_2)v...v(P&Qa_k)v(P&Qa_{k+1})) [$\exists_{k+1}$-Exp]
8) ($\exists_{k+1}$x)(P & Qx) SYN (((P&Qa_1)v(P&Qa_2)v...v(P&Qa_k))v(P&Qa_{k+1})) [7),v-ORD]
9) ($\exists_{k+1}$x)(P & Qx) SYN (($\exists_k$x)(P & Qx)v(P & Qa_{k+1})) [8),3),R1]
10) ($\exists_{k+1}$x)(P & Qx) SYN ((P & ($\exists_k$x)Qx) v (P & Qa_{k+1})) [9),1),R1b]
11) ($\exists_{k+1}$x)(P & Qx) SYN (P & (($\exists_k$x)Qx v Qa_{k+1})) [10),Ax.1-08,R1]
12) ($\exists_{k+1}$x)(P & Qx) SYN (P & ($\exists_{k+1}$x)Qx) [11),6),R1]
13) If ($\exists_k$x)(P & Qx) SYN (P & ($\exists_k$x)Qx)
then ($\exists_{k+1}$x)(P & Qx SYN (P & ($\exists_{k+1}$x)Qx) [1)-13),Conditional Proof]
Hence, |= [(∃x)(P & Qx) SYN (P & (∃x)Qx)] [Steps 1 & 2, Math Induction]

The proof by mathematical induction that T3-20 holds in all finite domainss similar step-by-step and justification-by-justification, except that (i) all occurrences of '∃' are replacd by occurrences of '∀', (ii) '&'s and 'v's are interchanged, i.e., each occurrence of '&' is replaced by 'v' and each occurrence of 'v' is replaced by '&', and (iii) each of the two occurrences of 'v-ORD' is replaced by '&-ORD', and (iv) the one occurrence of 'Ax.1-08' is replaced by 'Ax.1-07'. Thus for example, the base ste looks like this:

T3-20. [(∀x)(P v Qx) SYN (P v (∀x)Qx)]
Proof: Let x_n= Ua_n={a_1,a_2,...,a_k,a_{k+1},...},$^{n\in N}$.
A. Basis step:
For n=1, we have x_1={a_1} and:
1) ($\forall_1$x)Qx SYN Qa_1 [$\forall_1$-Exp]
2) ($\forall_1$x)(P v Qx) SYN (P v Qa_1) [$\forall_1$-Exp]
3) ($\forall_1$x)(P v Qx) SYN (P v ($\forall_1$x)Qx) [2),1),R1]

T3-21 and T3-22. The Boolean expansions of the theorems,

T3-21. $(\forall x)Px$ SYN $((\forall x)Px \& (\exists x)Px)$
T3-22. $(\exists x)Px$ SYN $((\exists x)Px \vee (\forall x)Px)$

are instances of Ax1-01 and Ax1-02 in a domain of 1, and of T1-18 and T1-19, respectively, in a domain of 2 members. In larger domains, the Boolean expansions on the right are derived from the Boolean expansions on the left by successive applications of T1-18 and &-ORD in the case of T3-21 and T1-19 and v-ORD in the case of T3-22. Since T1-18 is derived from Ax1-08 and &- ORD and T1-19 is derived from Ax1-07 and v-ORD, those are the axiomatic sources of T3-21 and T3-22.

Thus in a domain of 1, the Boolean expansion of T3-21 is simply a substitution instance of Ax.1-01 which is &-IDEM:

T3-21. $(\forall_1 x)Px$ SYN $((\forall_1 x)Px \& (\exists_1 x)Px)]$

Proof:	1) $(\forall_1 x)Px$ SYN $(\forall_1 x)$ Px	[T1-11]
	2) Pa SYN Pa	[1),B_1Exp]
	3) Pa SYN (Pa & Pa)	[2), Ax.1-01,R1b]
	4) $(\forall_1 x)Px$ SYN $((\forall_1 x)Px \& (\exists_1 x)Px)$	[7),Q_1-Abbr]

In a domain of 2 it is simply an instance of T1-18 which is based on Ax1-08, &v-Distribution.

T3-21. $(\forall_2 x)Px$ SYN $((\forall_2 x)Px \& (\exists_2 x)Px)$

Proof in a domain of 2:

1) (Pa & Pb) SYN ((Pa & Pb) & (Pa v Pb))	[2),T1-18,U-SUB]
2) $(\forall_2 x)Px$ SYN ($(\forall_2 x)Px \& (\exists_2 x)Px$)	[3),Q_2-Abbr]

In a domain of 3, iterations of T1-18 with &-ORD, establish the theorem:

Proof in a domain of 3,

1)	$(\forall_3 x)Px$	SYN $(\forall_3 x)Px$	[T1-11]
2)	(Pa&(Pb&Pc))	SYN (Pa&(Pb&Pc))	[1),B_3-Exp]
3)	(Pa&(Pb&Pc))	SYN (Pa&((Pb&Pc)&(PbvPc)))	[2),T1-18,R1]
4)	(Pa&(Pb&Pc))	SYN ((Pb&Pc)&(Pa&(PbvPc)))	[3),&-ORD]
5)	(Pa&(Pb&Pc))	SYN ((Pb&Pc)&((Pa&(PbvPc))&(Pav(PbvPc))))	[4),T1-18,R1]
6)	(Pa&(Pb&Pc))	SYN (((Pb&Pc)&(Pa&(PbvPc)))&(Pav(PbvPc)))	[5,&-ORD]
7)	(Pa&(Pb&Pc))	SYN ((Pa&(Pb&Pc))&(Pav(PbvPc)))	[6),4),R1]
8)	$(\forall_3 x)Px$	SYN ($(\forall_3 x)Px \& (\exists_3 x)Px$)	[7),Q_3-Abbr]

By successive application of T1-18 followed by &-ORD in the manner of steps 3) to 4) and again of steps 5) to 6), one can get a proof of T3-21 in any domain greater than 2 in (2(n-1) + 4) steps. (E.g., in a domain of 4, there are 2(4-1) +4) = 10 steps, in a domain of 3, 8 steps). The dual of T3-21, T3-22 $(\exists x)P$ SYN $((\exists x)P \vee (\forall x)P)$, is proved by a dual proof, using the duals, T1-19 and v-ORD, of T1-18 and &-ORD in all intermediate steps.

To prove that T3-21 and T3-22 hold in every finite domain, and thus establish theoremhood, we prove by mathematical induction that T3-21 will hold in every finite domain:

<u>T3-21.</u> <u>$(\forall x)Px$ SYN $((\forall x)Px$ & $(\exists x)Px$</u>
<u>Proof</u>: Let $x_n = Ua_n = \{a_1,a_2,\ldots,a_k,a_{k+1},\ldots\}$
A. <u>Basis step</u>:
For $n=1$, we have $x_1=\{a_1\}$ and:

1) $(\exists_1 x)Px$ SYN Pa_1 [$\exists_1$-Exp]
2) $(\forall_1 x)Px$ SYN Pa_1 [$\forall_1$-Exp]
3) $((\forall_1 x)Px \;\&\; (\exists_1 x)Px)$ SYN $((\forall_1 x)Px \;\&\; (\exists_1 x)Px)$ [T1-11]
4) $((\forall_1 x)Px \;\&\; (\exists_1 x)Px)$ SYN $((\forall_1 x)Px \;\&\; Pa_1)$ [3),1),R1]
5) $((\forall_1 x)Px \;\&\; (\exists_1 x)Px)$ SYN $(Pa_1 \;\&\; Pa_1)$ [4),2),R1]
6) $((\forall_1 x)Px \;\&\; (\exists_1 x)Px)$ SYN Pa_1 [5),Ax1-01,R1]
7) $(\forall_1 x)Px$ SYN $((\forall_1 x)Px \;\&\; (\exists_1 x)Px)$ [2),6),R1b]

B. <u>Inductive step</u>: For $n=k$ we have $x_n=x_k=\{a_1,a_2,\ldots,a_k\}$,
for $n=k+1$ we have $x_n=x_{k+1}=\{a_1,a_2,\ldots,a_k,a_{k+1}\}$, and:

1) $(\forall_k x)Px$ SYN $((\forall_k x)Px\&(\exists_k x)Px)$ [Assumption]
2) $(\forall_k x)Px$ SYN $(Pa_1\&Pa_2\&\ldots\&Pa_k)$ [$\forall_k$-Exp]
3) $(\exists_k x)Px$ SYN $(Pa_1 v Pa_2 v\ldots v Pa_k)$ [$\exists_k$-Exp]
4) $(\forall_{k+1} x)Px$ SYN $(Pa_1\&Pa_2\&\ldots\&Pa_k\&Pa_{k+1})$ [$\forall_{k+1}$-Exp]
5) $(\forall_{k+1} x)Px$ SYN $((\forall_k x)Px\&Pa_{k+1})$ [7),3),R1]
6) $(\exists_{k+1} x)Px$ SYN $(Pa_1 v Pa_2 v\ldots v Pa_k v Pa_{k+1})$ [$\exists_{k+1}$-Exp]
7) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((\forall_{k+1} x)Px \;\&\; (\exists_{k+1} x)Px)$ [T1-11]
8) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((\forall xk+1)Px\&(Pa_1 v Pa_2 v\ldots v Pa_k v Pa_{k+1}))$ [7),6),R1]
9) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((Pa_1\&Pa_2\&\ldots\&Pa_k\&Pa_{k+1})$
$\& (Pa_1 \text{ v } Pa_2 v\ldots. v\; Pa_k \text{ v } Pa_{k+1}))$ [8),4),R1]
10) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(((\forall_k x)Px\&Pa_{k+1})$
$\& (Pa_1 \text{ v } Pa_2 v\ldots v\; Pa_k v\; Pa_{k+1}))$ [9),2),R1]
11) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(((\forall_k x)Px\&Pa_{k+1})\&((\exists_k x)Px \text{ v } Pa_{k+1}))$ [10),6),R1]
12) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((((\forall_k x)Px\&Pa_{k+1} \& (\exists_k x)Px) \text{ v } (((\_{}_k x)Px\&Pa_{k+1})\& Pa_{k+1}))$
[11),Ax1-08(R1b),R1]
13) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN$((((\forall_k x)Px\&(\exists_k x)Px\&Pa_{k+1})$
$\text{v } ((\forall_k x)Px\&(Pa_{k+1}\& Pa_{k+1})))$ [12),&-ORD,twice]
14) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(((\forall_k x)Px \;\&\; Pa_{k+1})$
$\text{v } (\forall_k x)Px \;\&\; (Pa_{k+1} \;\&\; Pa_{k+1})))$ [13),1),R1]
15) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(((\forall_k x)Px\&Pa_{k+1}) v((\forall_k x)Px \;\&\; Pa_{k+1}))$ [14),Ax1-01,R1]
16) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((\forall_k x)Px\&Pa_{k+1}$ [15),Ax1-05,R1]
17) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(\forall_{k+1} x)Px$ [16),5),R1]
18) $(\forall_{k+1} x)Px$ SYN $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ [17),DR1-01]
19) If $(\forall_k x)Px$ SYN $((\forall_k x)Px\&(\exists_k x)Px)$
then $(\forall_{k+1} x)Px$ SYN $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ [1)-18),Conditional Proof]

Hence, $\models (\forall x)Px$ SYN $((\forall x)Px \;\&\; (\exists x)Px)$ [<u>Steps</u> 1 & 2, Math Induct]

The proof by mathematical induction of the dual of T3-21, namely

T3-22. $(\exists x)Px$ SYN $((\exists x)Px \text{ v } (\forall x)Px)$

is the dual of this proof, step-for-step: '∀' and '∃' and '&' and 'v' are interchanged throughout, while each axiom cited in a justification in brackets is replaced by its dual—Ax1-01 by Ax1-02, Ax1-03 by Ax1-04, Ax1-05 by Ax1-06, Ax1-08 by Ax1-07, and Ax1-04 by Ax1-03. All else is the same.

T3-23 and T3-24. Consider: "If something is both P and Q, then something must be P". The logical truth of this statement is grounded in the analytic containment theorem, |= [∃x)(Px & Qx) CONT (∃x)Px], which is immediately derivable by Df 'CONT' from

T3-23. (∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Px)

which is is basically a generalization of T1-28.

From T3-23 with Df 'CONT' we can derive

T3-47. |= [(∃x)(Px & Qx) CONT ((∃x)Px & (∃x)Qx)]

which is the AL-analogue of Quine's *156. I have not given the simpler theorem, |= [∃x)(Px & Qx) CONT (∃x)Px], a number since Quine does not give its truth-functional analogue a number in *Mathematical Logic*, although it is easily provable there. In a domain of 2 the Boolean expansion of T3-23 is simply an instance of T1-28:

1) ((P & Q) v (R & S)) SYN (((P & Q) v (R & S)) & (P v R)) [T1-28]
2) ((Pa&Qa) v (Pb&Qb)) SYN (((Pa&Qa) v (Pb&Qb)) & (PavPb)) [1),U-SUB]
3) (∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Px) [2),$\exists_2$-Abbr(thrice)]

For a domain of 3, T3-23 is proved in eight steps:

1) (∃x)(Px & Qx) SYN (∃x)(Px & Qx) [T1-11]
2) ((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) SYN ((Pa&Qa) v ((Pb&Qb)v(Pc&Qc))) [1),$\exists_3$-Exp]
3) ((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) SYN ((Pa&Qa) v (((Pb&Qb)v(Pc&Qc))&(PbvPc))) [2),T1-28,R1]
4) ((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) SYN (((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) &**((Pa&Qa)v(PbvPc))**) [3),Ax1-07,R1b]
5) ((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) SYN (((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) &(**(Pav(PbvPc))&(Qav(PbvPc))**))) [4),Ax1-07,R1b]
6) ((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) SYN ((((Pa&Qa)v((Pb&Qb)v(Pc&Qc))) &((Pav(PbvPc))&(Qav(PbvPc)))**&(PavPbvPc)**) [5),&-ORD]
7) ((Pa&Qa)v((Pb&Qb)v(Pc&Qc)))SYN(((Pa&Qa)v((Pb&Qb)v(Pc&Qc)))&(Pav(PbvPc))) [6),5),R1]
8) ∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Px) [7),$\exists_3$-Abbr]

One method of proving T3-23 in any domain of any n > 2 members is described as follows: The first two steps and the last two are essentially the same as in a domain of 3. The third step applies T1-28 to the right most disjunct in the right hand wff. Then follow n-1 applications of Ax1-08, which culminate in a wff which has a Boolean expansion of ($\exists_n$x)Px as a conjunct, followed by the use of &-ORD as in step 6) above and ending in two more steps (as in 7) and 8) above) with the conclusion. Such a proof takes n + 5 steps.

The proof by mathematical induction that T3-23 holds for all finite domains is given in Appendix III, and is sufficient ground for introducing it as an "axiom theorem" for axiomatic proofs of other theorems.

For each proof of T3-23, in any or all finite domains, there is a proof for its dual, T3-24. (∀x)(Px v Qx) SYN ((∀x)(Px v Qx) v (∀x)P). The dual proofs uses T1-29 in place of T1-28, Ax1-07 in place of Ax1-08, v-ORD in place of &-ORD and Q4-Abbr,in place of Q3-Abbr. Otherwise they are the same step by step. The proof of T3-24 in a domain of two is as follows:

|= [($\forall_2$x)(Px v Qx) SYN (($\forall_2$x)(Px v Qx) v ($\forall_2$x)Px)]
Proof:
1) ((P v Q) & (R v S)) SYN (((P v Q) & (R v S)) v (P & R)) [T1-29]
2) ((PavQa) & (PbvQb)) SYN (((PavQa) & (PbvQb)) v (Pa&Pb)) [1),U-SUB]
3) ($\forall_2$x)(Px v Qx) SYN (($\forall_2$x)(Px v Qx) v ($\forall_2$x)Px) [2),Q_2-Abbr]

T3-25 and T3-26. Consider next: "If everything is P and something is Q, then at least one thing must be both P and Q". The grounds for the logical truth of this statement lies in the analytic containment theorem T3-45:

T3-45. ((∀x)Px & (∃x)Qx) CONT (∃x)(Px & Qx)

which is an AL-analogue of Quine's *154, and is gotten using Df 'CONT' from theorem T3-25:

T3-25. (∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx))

In a domain of 2, T3-25 appears as an instance of T1-30:

1) ((P & Q) & (R v S)) SYN ((P & Q) & ((P & R) v (Q & S))) [T1-30]
2) ((Pa&Pb) & (QavQb)) SYN ((Pa&Pb) & ((Pa&Qa) v (Pb&Qb))) [1),U-SUB]
3) ((∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx)) [2),Q_2-Abbr]

Theorem T3-26. ((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)(Px v Qx)) is dual of T3-25. In a domain of 2, it is an instance of the dual of T1-30, namely T1-31:

1) ((P v Q) v (R & S)) SYN ((P v Q) v ((P v R) & (Q v S))) [T1-31]
2) ((PavPb) v (Qa&Qb)) SYN ((PavPb) v ((PavQa) & (PbvQb))) [1),U-SUB]
3) ((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)(Px v Qx)) [2),Q_2-Abbr]

T1-30 was established by means of Ax1-08 and &-ORD; T3-25 in any domain is established by a repetitive procedure using the same axioms. More precisely: in any domain of k ($k>1$), T3-25 may be established in $3k+3$ steps. There are three introductory, then k-1 applications of Ax1-08 with R1b, each followed by an application of &-ORD; after all of this there are k-1 applications of Ax1-08, then the final three steps. This procedure is illustrated in the proof in a domain of 3:

T3-25. $(\forall_3 x)Px \;\&\; (\exists_3 x)Qx)$ SYN $((\forall_3 x)Px \;\&\; (E_3 x)(Px \;\&\; Qx))$

Proof:

1) $((\forall_3 x)Px \;\&\; (\exists_3 x)Qx)$ SYN $((\forall_3 x)Px \;\&\; (\exists_3 x)Qx)$ [T1-11]
2) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (\exists_3 x)Qx)$ SYN $((Pa_1\&Pa_2\&Pa_3) \;\&\; (\exists_3 x)Qx)$ [1),$\forall_k$-Exp]
3) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ [2), $\exists_k$-Exp]
4) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $(((Pa_1\&Pa_2\&Pa_3)\&Qa_1)$
$v((Pa_1\&Pa_2\&Pa_3)\&(Qa_2 v Qa_3))$ [3),Ax1-08,R1b]
5) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $(((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_1\&Qa_1))$
$v\; ((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_2 v Qa_3)))$ [4),&-0RD]
6) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $(((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_1\&Qa_1))$
$v(((Pa_1\&Pa_2\&Pa_3)\&Qa_2)$
$v((Pa_1\&Pa_2\&Pa_3)\&Qa_3)))$ [5),Ax1-08,R1b]
7) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $(((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_1\&Qa_1))$
$v\; ((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_2\&Qa_2))$
$v\; ((Pa_1\&Pa_2\&Pa_3)\&Qa_3)))$ [6),&-ORD]
8) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $(((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_1\&Qa_1))$
$v\; ((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_2\&Qa_2))$
$v\; ((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_3\&Qa_3)))$ [7),&-ORD]
9) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v Qa_2 v Qa_3))$ SYN $(((Pa_1\&Pa_2\&Pa_3) \;\&\; (Pa_1\&Qa_1))$
$v\; ((Pa_1\&Pa_2\&Pa_3) \;\&((Pa_2\&Qa_2)v(Pa_3\&Qa_3)))$
[8),Ax1-08,R1]
10) $((Pa_1\&Pa_2\&Pa_3) \;\&\; (Qa_1 v(Qa_2 v Qa_3))$ SYN
$((Pa_1\&Pa_2\&Pa_3)\& ((Pa_1\&Qa_1)v(Pa_2\&Pa_3)v(Qa_2\&Qa_3)))$ [9),Ax.1-08,R1]
11) $((\forall_3 x)Px \;\&\; (Qa_1 v Qa_2\&Qa_3))$ SYN
$((\forall_3 x)Px \;\&\; ((Pa_1\&Qa_1)v(Pa_2\&Pa_3)v(Qa_2\&Qa_3)))$ [10), $\forall_3$-Abbr]
12) $((\forall_3 x)Px \;\&\; (\exists_3 x)Qx\;)$ SYN $((\forall_3 x)Px \;\&\; (\exists_3 x)(Px \;\&\; Qx))$ [11),$\exists_3$-Abbr]

The proof that T3-25 holds in all finite domains is established by mathematical induction in Appendix III. A proof of the dual of T3-25, namely, T3-26, $((\exists x)Px \; v \; (\forall x)Qx)$ SYN $((\exists x)Px \; v \; (\forall x)(Px \; v \; Qx))$ can be established in any and all finite domains by a proof dual to that of T3-25 using Ax1-07 and v-ORD in place of Ax1-08 and &-ORD.

T3-27 and T3-28. Consider: "If somebody loves everyone, then for everyone there is somebody who loves them". This has the logical form, "If $(\exists x)(\forall y)Lxy$ then $(\forall y)(\exists x)Lxy$". In Analytic Logic the logical truth of this statement is grounded in the logical containment theorem

T3-27c. $\models [(\exists x)(\forall y)Rxy$ CONT $(\forall y)(\exists x)Rxy)]$

which is gotten by Df 'CONT' from the logical synonymy theorem, T3-27.

T3-27. $\models [(\exists x)(\forall y)Rxy$ SYN $((\exists x)(\forall y)Rxy \;\&\; (\forall y)(\exists x)Rxy)]$

In a domain of 2, T3-27 is simply an instance of T1-34:

1) $((P \;\&\; Q) \; v \; (R \;\&\; S))$ SYN $(((P \;\&\; Q)v(R \;\&\; S))\&((P \; v \; R) \;\&\; (Q \; v \; S)))$ [T1-34]
2) $((Raa\&Rab)v(Rba\&Rbb))$ SYN $(((Raa\&Rab)v(Rba\&Rbb))\&((Raa v Rba)\&(Rab v Rbb)))$ [U-SUB]
3) $(\exists_2 x)(\forall_2 y)Rxy$ SYN $((\exists_2 x)(\forall_2 y)Rxy \;\&\; (\forall_2 y)(\exists_2 x)Rxy)$ [2),Q2-Abbr]

The dual of T3-27 is T3-28, (∀y)(∃x)Rxy SYN ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy)). In a domain of 2 it is an instance of the dual of T1-34, namely T1-35.

In domains larger than 2, T3-27 is established by repetitive uses of **Ax1-07, &-ORD and v-ORD,**—which are the axioms needed to prove T1-34. T3-28 is established by dual proofs, as in other examples. I will not present a general procedure for establishing T3-27 or T3-28 in any given finite domains of k members; this can be left to the reader. However, a proof that T3-27 holds in all finite domains using mathematical induction is given in Appendix III. A dual proof of its dual, T3-28, is easily constructed from that proof.

The key step in the proof of T3-27 by mathematical induction (See Appendix III), is Step 10) in the sequence of sub-steps 9), 10) and 11) in the Inductive Step. In 10) we use the principle of Generalized Distribution [Gen.Dist] which says that a continuous disjunction is synonymous with a conjunction of itself with any disjunction with includes at least one conjunct from each non-disjunctive disjunct in the disjunction. Indeed it is synonymous with itself conjoined with any number of such disjunctions. In the case below, the derivative disjunctions selected and conjoined are each equivalent to a conjunctive (universal) quantification in which 'y' is replaced by a constant. Each i^{th} row preceded by an '&', is a derived disjunction of the i^{th} left-most components (the columns) in the disjunction of conjunctions comprising the first five lines). To see how steps 9), 10), 11) work, consider a proof in a domain of 3:

T3-27. [($\exists_3$y)($\forall_3$x)Rxy SYN (($\exists_3$y)($\forall_3$x)Rxy & ($\forall_3$x)($\exists_3$y)Rxy))]
<u>Proof in a domain of three, i.e., n =(k+1)=3</u>

	1) ($\exists_3$y)($\forall_3$x)Rxy SYN ($\exists_3$y)($\forall_3$x)Rxy]	[T1-11]
	2) ($\exists_3$y)($\forall_3$x)Rxy SYN (($\forall_3$x)Rxa v ($\forall_3$x)Rxb v ($\forall_3$x)Rxc)	[($\exists_3$-Exp]
<u>Step 9):</u>	3) ($\exists_3$y)($\forall_3$x)Rxy SYN ((Raa & Rba & Rca) v(Rab & Rbb & Rcb) v(Rac & Rbc & Rcc))	[$\forall_3$-Exp,3 times]
<u>Step 10):</u>	4) ($\exists_3$y)($\forall_3$x)Rxy SYN (((**Raa** & Rba & Rca) v(**Rab** & Rbb & Rcb) v(**Rac** & Rbc & Rcc)) &(**(Raa v Rab v Rac)** & (Rba v Rbb v Rbc) & (Rca v Rcb v Rcc)))	[3),GEN v&-DIST]
<u>Step 11):</u>	5) ($\exists_3$y)($\forall_3$x)Rxy SYN (($\exists_3$y)($\forall_3$x)Rxy &((Raa v Rab v Rac) &(Rba v Rbb v Rbc) &(Rca v Rcb v Rcc)))	[4),3),R1]
<u>Step 16):</u>	6) ($\exists_3$y)($\forall_3$x)Rxy SYN (($\exists_3$y)($\forall_3$x)Rxy & (($\exists_3$y)Ray & ($\exists_3$y)Rby & ($\exists_3$y)Rcy))	[5),$\exists_3$-Abbr(3 times)]
<u>Step 17):</u>	7) ($\forall_3$x)($\exists_3$y)Rxy SYN (($\exists_3$y)Ray & ($\exists_3$y)Rby & ($\exists_3$y)Rcy)	[$\forall_3$-Exp]
<u>Step 18):</u>	8) ($\exists_3$y)($\forall_3$x)Rxy SYN (($\exists_3$y)($\forall_3$x)Rxy & ($\forall_3$x)($\exists_3$y)Rxy))	[6),7),R1]

The dual of T3-27 is the theorem T3-28. (∀x)(∃y)Rxy SYN ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy). It is of little interest since it does not yield a containment, but it can be proved by the dual proof of the proof of T3-27.

The next theorem is an important one for Mathematical Logic. It is the basis in A- logic of the M-logic theorem which frequently occurs as an axiom of quantification theory, e.g., of Quine's *101. [(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))].[22]

T3-29 [(∀x)(Px v Qx) SYN ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx))]
Proof:

1) (∃x)Px v (∀x)Qx) SYN (((∃x)Px v (∀x)(Px v Qx)) [T3-26]
2) (∀x)(Px v Qx) SYN (∀x)(Px v Qx) v (∀x)Px) [T3-24]
3) (∀x)(Px v Qx) SYN (∀x)(Px v Qx) v ((∀x)Px & (∃x)Px) [2),T3-21,R1b]
4) (∀x)(Px v Qx) SYN ((∀x)(Px v Qx)v(∀x)Px) & ((∀x)(Px v Qx) v (∃x)Px)) [3),Ax1-07,R1b]
5) (∀x)(Px v Qx) SYN ((∀x)(Px v Qx)v(∀x)Px) & ((∃x)Px v (∀x)Qx)) [4),1),R1]
5) (∀x)(Px v Qx) SYN ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx)) [5),2),R1]

In a domain of 2 members T3-29 it is simply an instance of

T1-32. ((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) & (PvRv(Q&S))

which is a 'SYN'-for-'≡' analogue of the theorem Russell identified as Leibnitz' "Praeclarum Theorem".[23] In a domain of 2 members, {a,b}, T3-29 has the following Boolean expansion,

1) ((P v Q)&(R v S)) SYN (((P v Q)&(R v S))&((P v R)v(Q & S))) [T1-32]
2) ((PavQa)&(PbvQb)) SYN (((PavQa)&(PbvQb)) & ((PavPb)v(Qa&Qb))) [1),U-SUB]
3) $(\forall_2 x)$(Px v Qx) SYN ($(\forall_2 x)$(PxvQx) & ((PavPb)v$(\forall_2 x)$Qx))) [2),∀-Abbr]
4) $(\forall_2 x)$(Px v Qx) SYN ($(\forall_2 x)$(PxvQx) & (($(\exists_2 x)$Pxv$(\forall_2 x)$Qx))) [3),∃-Abbr]

The dual of this theorem is,

T3-30. [(∃x)(Px&Qx) SYN ((∃x)(Px&Qx) v ((∀x)Px&(∃x)Q)),

It is similarly related in a domain of 2 to T1-33, the dual of T1-32. It can be derived by a dual proof (using T3-22, T3-23 and T3-25 instead of T3-21, T3-24 and T3-26), but this theorem is of little interest as it yields no containment theorems.

3.42 Theorems of Quantificational Containment (CONT)

All of the containment theorems in this section, except the first three, are derived axiomatically from the theorems just discussed. The logical containment theorems which will be established in this section are:

T3-33. [(∀x)Px CONT Pa_i] ($1 \underline{>} i \underline{>} n$) [Compare to *103]
T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx]
T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx]
T3-36. [(∀x)Px CONT (∃x)Px)] *136

22. *101 is derived from T4-37, by DR 5-4c to get
T5-437c. TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))] in Chapter 5.
23. Cf. *Principia Mathematica,* 1925, *3.45. I have replaced the horseshoe by (~AvB) then eliminated the negations by substitution and double negation.

T3-37. [(∃y)(∀x)Rxy CONT (∀x)(∃y)Rxy]	*139
T3-38. [((∀x)Px v (∀x)Qx) CONT (∀x)(Px v Qx)]	*143
T3-39. [(∀x)(Px v Qx) CONT ((∃x)Px v (∀x)Qx))]	*144
T3-40. [(∀x)(Px v Qx) CONT ((∀x)Px v (∃x)Qx))]	*145
T3-41. [((∀x)Px v (∃x)Qx) CONT (∃x)(Px v Qx)]	*146
T3-42. [((∃x)Px v (∀x)Qx) CONT (∃x)(Px v Qx)]	*147
T3-43. [(∀x)(Px & Qx) CONT ((∃x)Px & (∀x)Qx)]	*152
T3-44. [(∀x)(Px & Qx) CONT ((∀x)Px & (∃x)Qx)]	*153
T3-45. [((∀x)Px & (∃x)Qx) CONT (∃x)(Px & Qx)]	*154
T3-46. [((∃x)Px & (∀x)Qx) CONT (∃x)(Px & Qx)]	*155
T3-47. [(∃x)(Px & Qx) CONT ((∃x)Px & (∃x)Qx)]	*156

Except for the first three, each of these CONT-theorems is a 'CONT'-for-'⊃' analogue of a metatheorem of Quine's of the form ├ [A ⊃ B]. Quine's asterisked number is on the right. The first three of these CONT-theorems, however, are based on subordinate Modes of Predicates, and are not derived as generalizations on the principles in Chapter 1.

3.421 Quantification on Subordinate Modes of Predicates

All of the theorems T3-13 to T3-30 are based simply on generalizations of the Ax1-01 to Ax1-08, the most interesting of them (T3-19 to T3-30) being based on distribution theorems.

But there is a third kind of theorem that can not be derived from those principles alone. They depend on the nature of quantified dyadic predicates and their subordinate modes. In its simplest form, this is the principle which says that 'everybody loves everybody' logically implies that 'everybody loves themselves' is logically true,

One of its simplest forms is T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx]. The left-hand Q-wff logically contains a Q-wff which quantifiies a mode of its predicate. The predicate of the Q-wff on the left has fewer variables attached to it than the predicate of the Q-wff on the right. Most instantiations of the first Q-wff do not contain an instantiation of the second; e.g., 'Rab' does not contain 'Raa'; "Al loves Barby" does not contain "Al loves Al". Yet the quantification of 'Rxy' with two conjunctive quantifiers contains a quantification of 'Rxx'. More generally—a quantification of '$R<x_1,x_2,\ldots,x_k>$' may logically contain a quantification of '$R<x_1,x_1,\ldots,x_k>$', or of '$R<x_2,x_2,\ldots,x_k>$', or of '$R<x_1,x_2,\ldots,x_1>$', etc.

In contrast, an examination of the theorems presented in Sections 3.411 and 3.412 shows that none have less variables attached to some predicate letter on one side of 'SYN' than on the other. The principles presented in this section can not be derived from any of those theorems. Thus we need to establish a different kind of theorem, peculiar to quantification theory.

Theoremhood in such cases requires that the quantifiers be the right kind and arranged in the right order. Thus

'(∃x)(∃y)Rxy' does not contain '(∃x)Rxx'.
("Somebody shot somebody" does not contain "Somebody shot himself")

there must be at least one universal quantifier. And although we have the theorem

T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx], ("Someone shot everyone" contains "Somebody shot himself"),

'(∀y)(∃x)Rxy' does not contain '(∃x)Rxx', ("Every one shot someone" does not contain "someone shot himself").

If a disjunctively quantified Q-wff is to contain a quantification of one of its predicate's modes, it must have a conjunctively quantified wff in its normal scope.

Further, there are many modes of a predicate, and if the second is not a subordinate mode of the first, then the quantification of the first will not contain the quantification of the second. For example, '(∀x)Rxx' does not contain '(∃x)(∃y)Rxy': in a domain of 2, (Raa & Rbb) does not contain ((RaavRab)v(RbavRbb)). 'R<1,1>' is subordinate to 'R<1,2>', but not conversely.[24]

Again, 'Rab' and 'Rba' are modes of R<1,2>, but neither is subordinate to the other. Thus (∀x)(∃y)Rxy does not contain (∀x)(∃y)Ryx. On murderer's row "Every one killed someone" is true, but it does not contain "Every one is such that someone killed them".

Theorems with quantified reflexivity like T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx] and T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx] are based on the special kinds of conjunction and disjunction which are represented by the quantifiers '∀' and '∃'. In every domain, if all x's and all y's are such that Rxy, then all x's are such that Rxx, and all y's are such that Ryy. These results come, not merely from the structure of and's and or's but from the concept of the application of an n-adic predicate to all members in a domain of n or more individuals whenever n is greater than 1. In short,

T3-31. [(∀x)(∀y)Rxy SYN ((∀x)(∀y)Rxy & (∀x)Rxx)]
hence, T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx]

T3-32. [(∃x)(∀y)Rxy SYN ((∃x)(∀y)Rxy & (∃x)Rxx)]
hence, T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx]

T3-31 and T3-34 hold of all domains:
In a domain of 2: '(∀x)(∀y)Rxy' means '((**Raa** & Rab) & (Rba & **Rbb**))'
which contains [Raa & Rbb], or, (∀x)Rxx.
In a domain of 3: '(∀x)(∀y)Rxy' means '[(**Raa**&Rab&Rac) & (Rba&**Rbb**&Rbc) & (Rca&Rcb&**Rcc**)]'
which contains [Raa & Rbb & Rcc] or (∀x)Rxx.
And so on in every domain.

T3-31. [(∀x)(∀y)Rxy SYN ((∀x)(∀y)Rxy & (∀x)Rxx)]
Proof: 1) (∀x)(∀y)Rxy SYN ($(R<a_1,a_1> \& R<a_1,a_2> \&...\& R<a_1,a_n>)$
$\& (R<a_2,a_1> \& R<a_2,a_2> \&...\& R<a_2,a_n>)$
& …
$\& (R<a_n,a_1> \& R<a_n,a_2> \&...\& R<a_n,a_n>))$ [Df '(∀x)Px']
2) (∀x)(∀y)Rxy SYN ($(\mathbf{R<a_1,a_1>} \& R<a_1,a_2> \&...\& R<a_1,a_n>)$
$\& (R<a_2,a_1> \& \mathbf{R<a_2,a_2>} \&...\& R<a_2,a_n>)$
& …
$\& (R<a_n,a_1> \& R<a_n,a_2> \&...\& \mathbf{R<a_n,a_n>}))$
$\& (\mathbf{R<a_1,a_1> \& R<a_2,a_2> \&...\& R<a_n,a_n>})$ [1),&-ORD]
3) (∀x)(∀y)Rxy CONT (∀x)(∀y)Rxy & $(R<a_1,a_1> \& R<a_2,a_2> \&...\& R<a_n,a_n>)$ [1),2),SynSUB]
4) $(R<a_1,a_1> \& R<a_2,a_2> \&...\& R<a_n,a_n>)$ SYN (∀x)Rxx [Df '(∀x)Px']
5) (∀x)(∀y)Rxy SYN ((∀x)(∀y)Rxy & (∀x)Rxx) [4),3),SynSUB]

---

24. In analytic truth-logic, T(Raa & Rbb) **A-implies** T((Raa v Rab) v (Rba v Rbb)), and T(∀x)Rxx **A-implies** T((∃x)(∃y)Rxy). But in A-logic proper, without the T-operator, containment fails so there are not such implications. See Sect. 7.423

The proof of T3-31 is basically a matter of repetition and re-arrangement of conjuncts in a conjunction. The proof of T3-32 involves distribution principles as well. Such derivations can be used in proofs by mathematical induction to show that these theorem hold in all finite domains.

T3-32. [(∃x)(∀y)Rxy SYN ((∃x)(∀y)Rxy & (∃x)Rxx)

<u>Proof</u>:

1) (∃x)(∀y)Rxy SYN (($R<a_1,a_1>$ & $R<a_1,a_2>$ &...& $R<a_1,a_n>$)
   v ($R<a_2,a_1>$ & $R<a_2,a_2>$ &...& $R<a_2,a_n>$)
   v ...
   v ($R<a_n,a_1>$ & $R<a_n,a_2>$ &...& $R<a_n,a_n>$)) [Df '(∀x)Px']
2) (∃x)(∀y)Rxy SYN (((**$R<a_1,a_1>$** & $R<a_1,a_2>$ &...& $R<a_1,a_n>$)
   v ($R<a_2,a_1>$ & **$R<a_2,a_2>$** &...& $R<a_2,a_n>$)
   v ...
   v ($R<a_n,a_1>$ & $R<a_n,a_2>$ &...& **$R<a_n,a_n>$**))
   & (**$R<a_1,a_1>$ v $R<a_2,a_2>$ v...v $R<a_n,a_n>$**)) [1),GEN(∀&-DIST)]
3) (∃x)(∀y)Rxy Syn ((∃x)(∀y)Rxy & ($R<a_1,a_1>$ v $R<a_2,a_2>$ v...v $R<a_n,a_n>$)) [2),1),SynSUB]
4) ($R<a_1,a_1>$ v $R<a_2,a_2>$ v...v $R<a_n,a_n>$) SYN (∃x)Rxx [Df '(∃x)Px']
5) (∃x)(∀y)Rxy SYN ((∃x)(∀y)Rxy & (∃x)Rxx) [4),3),SynSUB]

The duals of SYN-theorems T3-31, [(∃x)(∃y)Rxy SYN ((∃x)(∃y)Rxy v (∃x)Rxx)] and T3-32. [(∀x)(∃y)Rxy SYN ((∀x)(∃y)Rxy v (∀x)Rxx)] are easily proven but are not very interesting as they give rise to no containments.

The relevant CONT-theorems follow immediately from T3-31 and T3-32**:.**

T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx]

<u>Proof</u>: 1) [(∀x)(∀y)Rxy SYN ((∀x)(∀y)Rxy & (∀x)Rxx)] [T3-31]
2) (∀x)(∀y)Rxy CONT (∀x)Rxx [1),Df 'Cont']

T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx]

<u>Proof</u>: 1) [(∃x)(∀y)Rxy SYN ((∃x)(∀y)Rxy & (∃x)Rxx)] [T3-32]
2) (∃x)(∀y)Rxy CONT (∃x)Rxx [1),Df 'Cont']

Various other SYN-theorems are gotten by alphabetic variance, changing the order of similar quantifiers and re-lettering carefully, etc.. If A-logic is to include a complete M-logic quantification theory, it must cover all polyadic predicates, not just binary predicates. For example, it must be able to prove

|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)Rxxxx
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)(∀z)Rxyzz
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)(∀z)Rxyzy
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)(∀z)Rxyzx
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)Rxyyy
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)Rxyxy
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)Rxyyx
|= [(∀x)(∀y)(∀z)(∀w)Rxyzw CONT (∀x)(∀y)Rxyxx

and many others, without getting as theorems,

[(∀x)(∀y)(∀z)Rxy**zz** CONT (∀x)(∀y)Rxy**xy**]
or [(∀x)(∀y)(∀z)(∀w)Rx**zz**yw CONT (∀x)(∀y)Rx**xy**yy], etc..

In the first of these non-theorems, every instantiation of 'Rxyzz' will have the same two constants in the two right-most positions; none will have different constants in those positions as do some instantiations of 'Rxyxy' or 'Rxyyx'; hence the right-hand side is not contained in the left. Similarly for positions 2 and 3 in the second case). Why each of these pairs will be synonymous, or not, in every domain, may be understood by examining their Boolean expansions in domains with as many individuals as there are quantifiers. The un-numbered SYN-theorems above are derivable, with the help of U-SUB and the two theorems, T3-34 and T3-35.

Theorems T3-34 and T3-35 are related to, but not the same as, Quine's axiom schema *103, in *Mathematical Logic*, this reads:

*103 If φU is like φ except for containing free occurrences of α' where
φ contains free occurrences of α,
then the closure of [(α)φ ⊃ φ'] is a theorem.

This is related to T3-34 by the puttng 'Rxy' for 'φ', 'Ryy' for 'φU', 'x' for 'α' and 'y' for 'αU':

If 'Ryy' is like 'Rxy' except for containing free occurrences of 'y' wherever
'Rxy' contains free occurrences of 'x',
then the closure of [(∀x)Rxy ⊃ Ryy] is a theorem. I.e., [(∀y)((∀x)Rxy ⊃ Ryy)] is a theorem.

Once this step (Step 2 in the derivations below) is gotten, Quine can derive the '⊃'-for-'CONT' analogue of T3-34, T3-35, and other such analogues as follow:

⊢ [(∀α)φ ⊃ φU]		
1) ⊢ [(∀x)Rxy ⊃ Ryy]	[by *103]	
2) ⊢ [(∀y)((∀x)Rxy ⊃ Ryy)]	[the closure of 1)]	
3) ⊢ [(∃y)((∀x)Rxy ⊃ Ryy)]	[2),*136,*104]	
4) ⊢ [(∀y)(∀x)Rxy ⊃ (∀y)Ryy]	[2),*101,*104]	⊨ [(∀x)(∀y)Rxy CONT (∀y)Ryy]
5) ⊢ [(∀y)(∀x)Rxy ⊃ (∀x)Rxx]	[4),*171]	Cf. T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx]
6) ⊢ [(∀y)(∀x)Rxy ⊃ (∃x)Rxx]	[3),*142]	⊨ [(∀x)(∀y)Rxy CONT (∃x)Rxx]
7) ⊢ [(∃y)(∀x)Rxy ⊃ (∃x)Rxx]	[6),*161].	Cf. T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx]

There are no 'CONT'-for-'⊃' analogues of steps 1), 2) and 3). For neither '[(∀x)Rxy CONT Ryy]' nor '[(∀y)((∀x)Rxy CONT Ryy)]' nor '[(∃y)((∀x)Rxy CONT Ryy)]' are well-formed. But in Chapter 5 we establish Step 2) as a tautology (but not a containment). As Rosser showed, this can be used instead of Quine's *103. It, with surrogates for Quine's other axiom-schemata, permits a completeness proof for the M-logic fragment in A-logic. (See Sect. 5.343).

Basic to theorems in this area is the principle of Universal Instantiation (UI): a conjunctive (universal) quantification logically contains each instance of its predicate. We use 'a_i', or 'a' without a subscript, stand for any individual constant in a Boolean expansion of the quantification.

<u>T3-33. $[(\forall x)Px \text{ CONT } Pa_i]$</u> $(1 \leq i \leq n)$

<u>Proof</u>: 1) $[(\forall x)Px \text{ SYN } (Pa_1 \&\ldots\& Pa_i \&\ldots\& Pa_n)$ [Df '$(\forall x)Px$']
2) $[(\forall x)Px \text{ SYN } ((Pa_1 \&\ldots\& (Pa_i \& Pa_i) \&\ldots\& Pa_n)$ [1),&-IDEM]
3) $[(\forall x)Px \text{ SYN } ((Pa_1 \&\ldots\& Pa_i \&\ldots\& Pa_n) \& Pa_i)$ [2),&-ORD]
4) $[(\forall x)Px \text{ SYN } ((\forall x)Px \& Pa_i)$ [3),1),SynSUB]
5) $[(\forall x)Px \text{ CONT } Pa_i$ [4),Df 'CONT']

From this with U-SUB we can get partial or complete instantiations of any conjunctively quantified Q-wff. Some examples:

<u>$[(\forall x)Rxx \text{ CONT } Ra_ia_i]$</u>

<u>Proof:</u> 1) $[(\forall x)Px \text{ SYN } (Pa_1 \& Pa_2 \&\ldots\& Pa_n)$ [Df '$(\forall x)Px$']
2a) in place of '$P<1>$' introduce '$R<1,1>$':
2) $(\forall x)Rxx \text{ SYN } (Ra_1a_1 \& Ra_2a_2 \&\ldots\& Ra_na_n)$ [1),U-SUB]
3) $(\forall x)Rxx \text{ SYN } ((Ra_1a_1 \& Ra_2a_2 \&\ldots\& Ra_na_n) \& Ra_ia_i)$ $(1 \leq i \leq n)$ [2),&-ORD]
4) $(\forall x)Rxx \text{ CONT } Ra_ia_i$ [3),Df 'CONT']

<u>$[(\forall x)R<x,a_i> \text{ CONT } R<a_i,a_i>]$</u>

<u>Proof</u>: 1) $(\forall x)Px \text{ CONT } Pa_i)$ [T3-33]
2a) In place of '$P<1>$' introduce '$R<1,a>$':
2) $(\forall x)R<x,a_i> \text{ CONT } R<a_i,a_i>$ [1),U-SUB]

<u>$[(\forall x)(\forall y)R<x,y> \text{ CONT } (\forall y)R<a_i,y>]$</u>

<u>Proof</u>: 1) $(\forall x)Px \text{ CONT } Pa_i)$ [T3-33]
2a) In place of '$P<1>$' introduce '$(\forall y)R<1,y>$':
2) $(\forall x)(\forall y)R<x,y> \text{ CONT } (\forall y)R<a_i,y>$ [1),U-SUB]

<u>$[(\forall x)(\forall y)R<x,y> \text{ CONT } (\forall x)R<x,a_i]$</u>

<u>Proof</u>: 1) $(\forall x)Px \text{ CONT } Pa_i)$ [T3-33]
2a) In place of '$P<1>$' introduce '$(\forall y)R<y,1>$':
2) $(\forall x)(\forall y)R<y,x> \text{ CONT } (\forall y)R<y,a_i>$ [1),U-SUB]
3) $(\forall y)R<y,a_i> \text{ SYN } (\forall x)R<x,a_i>$ [Alpha.Var]
4) $(\forall x)(\forall y)R<x,y> \text{ CONT } (\forall x)R<x,a_i>$ [3),T3-15,SynSUB]

<u>$[(\forall x)(\forall y)R<x,y> \text{ CONT } R<a_j,a_i>]$</u>

<u>Proof</u>: 1) $(\forall x)(\forall y)R<x,y> \text{ CONT } (\forall y)R<y,a_i>$ [preceding Theorem]
2) $(\forall y)Py \text{ CONT } Pa_i)$ [T3-33,Alpha.Var.]
3a): In place of '$P<1>$' introduce '$R<1,a>$':
3) $(\forall y)R<y,a_i> \text{ CONT } R<a_i,a_i>$ [2),U-SUB]
4) $(\forall x)(\forall y)R<x,y> \text{ CONT } R<a_i,a_i>$ [1),3),CONT-Syll.]

3.422 Derived from the Basic Theorems

All of the containment theorems in this section will be derived axiomatically from the basic theorems established in Section 3.32.

A proof by mathematical induction could be given for all theorems of analytic quantification theory, but such proofs can be avoided by deriving them from theorems already established by such proofs. The logical containment theorems remainimg in this section are:

T3-36. [(∀x)Px CONT (∃x)Px)] *136
T3-37. [(∃y)(∀x)Rxy CONT (∀x)(∃y)Rxy] *139
T3-38. [((∀x)Px v (∀x)Qx) CONT (∀x)(Px v Qx)] *143
T3-39. [(∀x)(Px v Qx) CONT ((∃x)Px v (∀x)Qx))] *144
T3-40. [(∀x)(Px v Qx) CONT ((∀x)Px v (∃x)Qx))] *145
T3-41. [((∀x)Px v (∃x)Qx) CONT (∃x)(Px v Qx)] *146
T3-42. [((∃x)Px v (∀x)Qx) CONT (∃x)(Px v Qx)] *147
T3-43. [(∀x)(Px & Qx) CONT ((∃x)Px & (∀x)Qx)] *152
T3-44. [(∀x)(Px & Qx) CONT ((∀x)Px & (∃x)Qx)] *153
T3-45. [((∀x)Px & (∃x)Qx) CONT (∃x)(Px & Qx)] *154
T3-46. [((∃x)Px & (∀x)Qx) CONT (∃x)(Px & Qx)] *155
T3-47. [(∃x)(Px & Qx) CONT ((∃x)Px & (∃x)Qx)] *156

In Chapter 5, from each of these CONT-theorems we will derive a metatheorem of Quine of the form |- [A ⊃ B] with the asterisked number on the right.

The proof of each of the theorems above proceeds by establishing first a theorem |= [A SYN (A&B)], then getting |= [A CONT B] by Df 'CONT'. All containment theorems except the first, are gotten from odd-numbered quantificational SYN-theorems from T3-13 to T3-27. None of the even numbered theorems T3-14 to T3-28 can yield CONT-theorems because they have disjunctions rather than conjunctions as right-hand components. The CONT-theorems, T3-36 to T3-45 are all in effect abbreviations of odd-numbered SYN-theorems. The duals of these CONT-theorems are not theorems—though the duals of the SYN-theorems which these CONT-theorems abbreviate are.

T3-36. (∀x)Px CONT (∃x)Px) Cf.*136
<u>Proof</u>: 1) (∀x)Px SYN ((∀x)Px & (∃x)Px) [T3-21]
2) (∀x)Px CONT (∃x)Px) [1),Df 'CONT']

T3-37. (∃y)(∀x)Rxy CONT (∀x)(∃y)Rxy Cf.*139
<u>Proof</u>: 1) (∃x)(y)Rxy SYN ((∃x)(y)Rxy & (y)(∃x)Rxy) [T3-27]
2) (∃y)(∀x)Rxy CONT (∀x)(∃y)Rxy [1),Df 'CONT']

T3-38. ((∀x)Px v (∀x)Qx) CONT (∀x)(Px v Qx) Cf.*143
<u>Proof:</u>
1) ((∀x)(Px v Qx)) SYN ((∀x)(Px v Qx) v (∀x)Px) [T3-24]
2) ((∀x)(Px v Qx)) SYN ((∀x)(Px v Qx) v (∀x)Qx) [T3-24]
3) ((∀x)(Px v Qx)) SYN (((∀x)(Px v Qx) v (∀x)Px) v (∀x)Qx) [2),1),R1b]
4) ((∀x)(Px v Qx)) SYN (((∀x)Px v (∀x)Qx) v (∀x)(Px v Qx)) [3),v-ORD]
5) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx) v ((∀x)Px & (∀x)Qx) [T1-19]
6) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx) v (∀x)(Px & Qx)) [5),T3-13,R1]
7) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx) v (∀x)(Px & Qx & (Px v Qx))) [6),T1-18,R1]
8) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx) v (∀x)((Px & Qx) & (Px v Qx))) [7),&-ORD]
9) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx) v ((∀x)(Px & Qx) & (∀x)(Px v Qx)))
[8),T3-13,R1b]
10) ((∀x)Px v (∀x)Qx)) SYN ((((∀x)Px v (∀x)Qx) v (∀x)(Px & Qx))
&(((∀x)Px v (∀x)Qx) v (∀x)(Px v Qx)) [9),Ax1-07,R1b]

11) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx)
&(((∀x)Px v (∀x)Qx) v (∀x)(Px v Qx)) [10),6),R1]
12) ((∀x)Px v (∀x)Qx)) SYN (((∀x)Px v (∀x)Qx) & (∀x)(Px v Qx)) [11),4),R1]
13) ((∀x)Px v (∀x)Qx)) CONT (∀x)(Px v Qx) [12),Df 'CONT']

T3-39. Consider, "If everyone either was murdered or died of old age, then either someone was murdered or all died of old age." The logical truth of this statement is grounded in the analytic containment theorem, T3-39. (∀x)(Px v Qx) CONT ((∃x)Px v (∀x)Qx)). T3-39 is the basis, in A-logic, of Quine's (and others') axiom for quantification theory,

(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx). It is also the 'Cont'-for-'⊃' analogue of Quine's *144.

T3-39. (∀x)(Px v Qx) CONT ((∃x)Px v (∀x)Qx)) Cf.*144
Proof: 1) (∀x)(Px v Qx) SYN ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx)) [T3-29]
2) (∀x)(Px v Qx) CONT ((∃x)Px v (∀x)Qx) [7),Df 'CONT']

Other 'Cont'-for-'⊃' analogues for Quine's "metatheorems" are:

T3-40.(∀x)(Px v Qx) CONT ((∀x)Px v (∃x)Qx)) Cf.*145
Proof: 1) (∀x)(Px v Qx) CONT ((∃x)Qx v (∀x)Px) [T3-29,U-SUB]
2) ((∃x)Qx v (∀x)Px) SYN ((∀x)Px v (∃x)Qx)) [Ax1-04,U-SUB]
3) (∀x)(Px v Qx) CONT ((∀x)Px v (∃x)Qx) [2),1),SynSUB]

T3-41. ((∀x)Px v (∃x)Qx) CONT (∃x)(Px v Qx) Cf.*146
Proof: 1) (∀x)Px v (∃x)Qx) SYN ((∀x)Px v (∃x)Qx) [T1-11]
2) (∀x)Px v (∃x)Qx) SYN (((∀x)Px & (∃x)Px) v (∃x)Qx) [1),T3-21,R1]
3) ((∀x)Px v (∃x)Qx) SYN (((∀x)Px v (∃x)Qx) & ((∃x)Px v (∃x)Qx)) [2),Ax1-08,R1b]
4) ((∀x)Px v (∃x)Qx) SYN (((∀x)Px v (∃x)Qx) & (∃x)(Px v Qx) [3),T3-14,R1]
5) ((∀x)Px v (∃x)Qx) CONT (∃x)(Px v Qx) [4),Df 'CONT']

T3-42. ((∃x)Px v (∀x)Qx) CONT (∃x)(Px v Qx) Cf.*147
Proof: 1) ((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)Qx) [T1-11]
2) ((∀x)Qx SYN ((∃x)Qx & (∀x)Qx)) [T3-21,Ax1-02,R1]
3) ((∃x)Px v (∀x)Qx) SYN ((∃x)Px v ((∃x)Qx & (∀x)Qx) [1),2),R1b]
4) ((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)Qx)) & ((∃x)Px v (∃x)Qx)) [3),Ax1-07,R1b]
5) ((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)Qx) & (∃x)(Px v Qx)) [4),T3-14,R1]
6) ((∃x)Px v (∀x)Qx) CONT (∃x)(Px v Qx) [5),Df 'CONT']

T3-43. (∀x)(Px & Qx) CONT ((∃x)Px & (∀x)Qx) Cf. *152
Proof: 1) (∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx) [T3-13]
2) (∀x)(Px & Qx) SYN ((∀x)Px & (∃x)Px) & (∀x)Qx) [1),T3-21,R1b]
3) (∀x)(Px & Qx) SYN (((∀x)Px & (∀x)Qx) & ((∃x)Px & (∀x)Qx) [2),DR3(&-ORD]
4) (∀x)(Px & Qx) SYN ((∀x)(Px & Qx) & ((∃x)Px & (∀x)Qx)) [3),1),R1]
5) (∀x)(Px & Qx) CONT ((∃x)Px & (∀x)Qx) [5),Df 'CONT]

T3-44. (∀x)(Px & Qx) CONT ((∀x)Px & (∃x)Qx) Cf. *153

Proof: 1) (∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx) [T3-13]
2) ((∀x)Qx & (∃x)Qx) SYN (∀x)Qx [T3-21,DR1-01]
3) (∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx & (∃x)Qx) [1),2),R1]
4) (∀x)(Px & Qx) SYN (((∀x)Px & (∀x)Qx) & ((∀x)Px & (∃x)Qx)) [3),&-ORD]
5) (∀x)(Px & Qx) SYN ((∀x)(Px & Qx) & ((∀x)Px & (∃x)Qx)) [4),1),R1]
6) (∀x)(Px & Qx) CONT ((∀x)Px & (∃x)Qx) [5),Df 'CONT]

T3-45. ((∀x)Px & (∃x)Qx) CONT (∃x)(Px & Qx) Cf.*154

Proof: 1) ((∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx)) [T3-25]
2) (∃x)(Qx & Px) SYN ((∃x)(Qx & Px) & (∃x)Qx) [T3-23,U-SUB]
3) (∃x)(Px & Qx) SYN (∃x)(Qx & Px) [T1-11,Ax1-03,R1]
4) (∃x)(Px & Qx) SYN ((∃x)(Qx & Px) & (∃x)Qx) [3),2),R1b]
5) (∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Qx) [4),3),R1]
6) ((∀x)Px & (∃x)Qx) SYN ((∀x)Px & ((∃x)(Px & Qx) & (∃x)Qx)) [1),5),R1b]
7) ((∀x)Px & (∃x)Qx) SYN (((∀x)Px & (∃x)Qx) & (∃x)(Px & Qx)) [6),&-ORD]
8) ((∀x)Px & (∃x)Qx) CONT (∃x)(Px & Qx) [7),Df 'CONT']

T3-46: ((∃x)Px & (∀x)Qx) CONT (∃x)(Px & Qx))) Cf.*155

Proof: 1) ((∃x)Px & (∀x)Qx) SYN ((∀x)Qx & (∃x)Px)) [Ax1-03]
2) ((∀x)Qx & (∃x)Px) SYN ((∀x)Qx & (∃x)(Qx & Px)) [T3-25,U-SUB]
3) ((∃x)Px & (∀x)Qx) SYN ((∀x)Qx & (∃x)(Qx & Px)) [1),Ax1-03,R1]
4) (∃x)(Qx & Px) SYN (∃x)(Px & Qx) [T1-11,Ax1-03,R1]
5) ((∃x)Px & (∀x)Qx) SYN ((∀x)Qx & (∃x)(Px & Qx)) [3),4),R1]
6) ((∃x)Px & (∀x)Qx) SYN ((∀x)Qx & ((∃x)(Px & Qx) & (∃x)Px)) [5),T3-23,R1b]
7) ((∃x)Px & (∀x)Qx) SYN ((∃x)Px & (∀x)Qx) & (∃x)(Px & Qx)) [6),&-ORD]
8) ((∃x)Px & (∀x)Qx) CONT (∃x)(Px & Qx) [7),Df 'CONT']

T3-47. (∃x)(Px & Qx) CONT ((∃x)Px & (∃x)Qx) Cf.*156

Proof: 1) (∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Px)] [T3-23]
2) (∃x)(Qx & Px) SYN ((∃x)(Qx & Px) & (∃x)Qx)] [T3-23,U-SUB]
3) (∃x)(Px & Qx) SYN (∃x)(Qx & Px) [T1-11,Ax1-03,R1]
4) (∃x)(Px & Qx) SYN ((∃x)(Qx & Px) & (∃x)Qx) [3),2),R1b]
5) (∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Qx) [4),3),R1]
6) (∃x)(Px & Qx) SYN (((∃x)(Px & Qx) & (∃x)Px) & (∃x)Qx) [5),1),R1b]
7) (∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & ((∃x)Px) & (∃x)Qx)) [6),&-ORD]
8) (∃x)(Px & Qx) SYN ((∃x)Px) & (∃x)Qx) [7),Df 'CONT']

Quine discusses six "chains" of successively weaker theorems:

v-chains: <*143,*145,*146> and <*143,*144,*147>;
⊃-chains: <*148,*101,*151> and <*148,*149,*150>.
&-chains: <*153,*154,*156> and <*152,*155,*156>;

Given the transitivity of containment (DR1-19) this means that from each chain of three CONT-theorems, an additional chain of three theorems will be derivable. Within each chain the antecedent of each member will contain the consequent of any other member which comes after it in the chain.

The analogues of Quine's chains of metatheorems which distribute quantifiers around disjunctions are as follows. On the right, to facilitate understanding, I show Boolean expansions in a domain of two, {a,b} of each component.

v-chain: <*143,*145,*146>		If Domain = 2,	((∀x)Pxv(∀x)Qx) = ((Fa&Fb)v(Ga&Gb))	
T3-38(*143).	((∀x)Px v (∀x)Qx)	CONT	(∀x)(Px v Qx)	| CONT ((FavGa)&(FbvGb))
T3-40(*145).	(∀x)(Px v Qx)	CONT	((∀x)Px v (∃x)Qx)	| CONT ((Fa&Fb)v(GavGb))
T3-41(*146).	((∀x)Px v (∃x)Qx)	CONT	(∃x)(Px v Qx)	| CONT ((FavGa)v(FbvGb))

v-chain: <*143,*144,*147>		If Domain = 2,	((∀x)Pxv(∀x)Qx) = ((Fa&Fb)v(Ga&Gb))	
T3-38(*143).	((∀x)Px v (∀x)Qx)	CONT	(∀x)(Px v Qx)	| CONT ((FavGa)&(FbvGb))
T3-39(*144).	(∀x)(Px v Qx)	CONT	((∃x)Px v (∀x)Qx)	| CONT ((FavFb)v(Ga&Gb))
T3-42(*147).	((∃x)Px v (∀x)Qx)	CONT	(∃x)(Px v Qx)	| CONT ((FavGa)v(FbvGb))

Chains of metatheorems which distribute quantifiers around conjunctions are as follows:

&-chain: 153,154,156;		If Domain = 2,	(∀x)(Px & Qx) = ((Fa&Ga)&(Fb&Gb))	
T3-44(*153).	(∀x)(Px & Qx)	CONT	((∀x)Px & (∃x)Qx)	| CONT ((Fa&Fb)&(GavGb))
T3-45(*154).	((∀x)Px & (∃x)Qx)	CONT	(∃x)(Px & Qx)	| CONT ((Fa&Ga)v(Fb&Gb))
T3-47(*156).	(∃x)(Px & Qx)	CONT	((∃x)Px & (∃x)Qx)	| CONT ((FavFb)&(GavGb))

&-chain: 152,155,156		If Domain = 2,	(∀x)(Px & Qx) = ((Fa&Ga)&(Fb&Gb))	
T3-43(*152).	(∀x)(Px & Qx)	CONT	((∃x)Px & (∀x)Qx)	| CONT ((FavFb)&(Ga&Gb))
T3-46(*155).	((∃x)Px & (∀x)Qx)	CONT	(∃x)(Px & Qx)	| CONT ((Fa&Ga)v(Fb&Gb))
T3-47(*156).	(∃x)(Px & Qx)	CONT	((∃x)Px & (∃x)Qx)	| CONT ((FavFb)&(GavGb))

Quine's ⊃-chains, <*148,*101,*151> and <*148,*149,*150> and their components are established after negation is introduced in Chapter 4.

Finally we give a theorem which is useful in establishing an axiom which Rosser used in lieu of Quine's *103. It quantifies over the CONT relation, but this can be shown to be acceptable.

T3-48. [(∀y) **(**(∀x)Pxy CONT Pyy**)**]

Proof: 1) [(∀x)Pxa_1 CONT Pa_1a_1]

2) [(∀x)Pxa_2 CONT Pa_2a_2]

. . .

n) [(∀x)Pxa_n CONT Pa_na_n]

n+1) [((∀x)Pxa_1 CONT Pa_1a_1) & ((∀x)Pxa_2 CONT Pa_2a_2) & … & ((∀x)Pxa_n CONT Pa_na_n)]

n+2) [(∀y) ((∀x)Pxy CONT Pyy)]

3.43 Rules of Inference

At this point, we have produced AL-analogues of 22 of the Quine's 56 metatheorems in Quantification theory, including the metatheorems, *170 and *171 which are covered by Alphabetic Variance theorems. This leaves 34 ML metatheorems to be accounted for.

We may, however add three rules of inference which have ‘CONT’-for-‘⊃’ analogues or ‘SYN’-for-‘⊃’ analogues in Quine’s *Mathematical Logic*:

DR1-20. If |= [P_1 CONT P_2], |= [P_2 CONT P_3],…., and |= [P_{n-1} CONT P_n], then |= [P_1 CONT P_n].
is the ‘CONT’-for-‘⊃’ analogue of :
***112**. If ├ [$P_1 \supset P_2$], ├ [$P_2 \supset P_3$],…., and ├ [$P_{n-1} \supset P_n$], then ├ [$P_1 \supset P_n$].

DR1-20b. If |= [P_1 SYN P_2], |= [P_2 SYN P_3],…, |= [P_2 SYN P_k] then |= [P_1 SYN P_k],
is the ‘SYN’-for- ‘≡’ analogue of:
***113**. If ├ [$P_1 \equiv P_2$], ├ [$P_2 \equiv P_3$],…., and ├ [$P_{n-1} \equiv P_n$], then ├ [$P_1 \equiv P_n$].

R1-1 If |= [P SYN Q], then |= [R SYN R(P//Q)] [= R1, SynSUB]
is the ‘SYN’-for-‘≡’ analogue of:
***123**. If ├ [P ≡ Q] and RU is formed from R by putting Q for some occurrences of P, then ├ [R ≡ R’]

Thus 25 of Quine’s 56 metatheorems have ‘SYN’for-‘≡’ or ‘CONT’-for-‘⊃’ analogues in the logic of synonymy and containment among wffs built up from conjunction, denial, and quantifiers in Chapters 1 through 4. The rest will be established or accounted for in Chapters 4 and 5. But these twenty five analogues include everything that is essential for the logic of SYN and CONT.

3.5 Reduction to Prenex Normal Form

A prenex normal form wff is one in which all quantifiers are on the left of the wff with scopes covering all atomic and molecular components. The preceding theorems provide all rules and premisses necessary to insure that any negation-free Q-wff can be reduced to another Q-wff which is in prenex normal form and is logically synonymous to the first wff. Q-wffs are transformed into logically synonymous prenex normal form wffs as follows.

If a quantifier is prefixed to a conjunction or disjunction and one of the conjuncts or disjuncts has no occurrence of the quantifier’s variable, then that quantified conjunction or disjunction is synonymous (and TF-equivalent) to a conjunction or disjunction in which the quantifier is prefixed only to the component which which has one or more occurrences of the variable. Thus either of the two expressions may replace the other *salve sens*. The relevant SYN- theorems, called “Rules of Passage” are:

	or, by commutation:
T3-17. [(∀x)(P & Qx) SYN (P & (∀x)Qx)]	T3-17'. [(∀x)(Qx & P) SYN ((∀x)Qx & P)]
T3-18. [(∃x)(P v Qx) SYN (P v (∃x)Qx)]	T3-18'. [(∃x)(Qx v P) SYN ((∃x)Qx v P)]
T3-19. [(∃x)(P & Qx) SYN (P & (∃x)Qx)]	T3-19'. [(∃x)(Qx & P) SYN ((∃x)Qx & P)]
T3-20. [(∀x)(P v Qx) SYN (P v (∀x)Qx)]	T3-20'. [(∀x)(Qx v P) SYN ((∀x)Qx & P)]

If a quantifier is prefixed to a conjunct or disjunct of a wff and the other conjunct or disjunct has an occurrence of the variable in the quantifier, then by Alphabetic Variance, the variable of the quantifier and the expressions bound to it can be replaced by a variable which does not occur elsewhere, and then the rules of passage applied. Thus given: (∀y)(Py & (∃y)Qy), we get (∀y)(Py & (∃x)Qx) by Alphabetic Variance, and then (∀y)(∃x)(Py & Qx) by T3-19.

Thus whenever a Quantifier occurs in the scope of a conjunction or disjunction sign in negation-free logic, it can be moved to the left, out of the scope of that conjunction or disjunction sign so that the whole conjunction or disjunction lies in the scope of the quanitifier. It follows that every wff in negation-free quantification theory can be replaced by a logically synonymous wff in prenex normal form .

In the next chapter it is shown that the introduction of negation into the symbolic language does not affect this result; every quantificational wff can be reduced to a logically synonymous wff in prenex normal form by means of Double Negation, Demorgan theorems, and definitions.

Chapter 4

Negation

4.1 Introduction: Negation and Synonymy

4.11 The Negation Sign

The sign, ‘$\sim$’, placed to the immediate left of any sentence or predicate is called the negation sign, and the meaning or concept conveyed by it, negation. Logicians often suggest ‘$\sim$P’ should be read as ‘not-P’; this is some help, but does not tell us what ‘not-’ means’. Perhaps the best reading in ordinary language is “It is not the case that…” but this too leaves much to be explained.

Negation by itself is not a predicate. It is neither verb, nor adjective, nor noun phrase. It is not predicated directly of any individual object; in the language of A-logic, ‘$\sim <1>$’, ‘$\sim <2>$’, ‘$\sim a_1$’, ‘$\sim a_2$’ and ‘$(\exists x) \sim x_1$’ are not well-formed schemata. Instantiation of such symbols in ordinary language, e.g., ‘It is not the case that $<1>$’, or ‘It is not the case that Julius Caesar’ or ‘For some x, it is not the case that x’ are either nonsensical, or else elliptical for an expression with an unexpressed predicate; e.g.,”It is not the case that Julius Caesar existed’ or ‘For some x, it is not the case that x exists in the actual world’.[1]

The negation sign is a predicate operator; the result of prefixing it to a predicate or to a sentence is a new predicate, or a new sentence with a new and different meaning. Like ‘and’ and ‘or’ it is syncategorematic—it gives an added meaning to what it operates on, but it does not have a referent and does not convey an idea which can stand apart from anything else.

4.12 The Meaning of the Negation Sign

What meaning does a negation sign add to what it operates on? In A-logic the intended meaning is not the same as either ‘It is not true that…’ or ‘It is false that…’ although both of these expressions certainly involve negation. Both of these involve the presence, or lack of a correspondence relation between the meaning of sentence and some reality which the sentence purports to describe. M-logic, because it

---

1. In contrast to early versions of M-logic, A-logic does not interpret ‘$(\exists x)$’ as ‘There exists an x’; it treats ‘$<1>$exists’ and ‘$<1>$exists in field of reference$<2>$’ as perfectly good predicates.

focuses on sentences which are either true or false, confuses negation with ‘it is false that…’ and the latter with ‘It is not true that…’. In Analytic logic these are three distinct and different concepts, symbolized in three distinct ways.[2]

Negation is involved not only in cases where truth is at issue, but also where the agents’ acts of accepting or rejecting proposals for action, judgments of what ought to be or what is beautiful, are involved. Some philosophers and logicians have interpreted negation as denial as opposed to affirmation or acceptance.[3] These are not the same as non-truth or falsehood, since an affirmation, like a denial, can be either true or false. Affirmation and denial are acts of agents. Logic, which is used to reason about real things, is not well served by including attitudes in which agents may differ among its primitive notions.

Perhaps the concept of negation is best described as the “non-applicability” of a predicate. Suppose we are creating a story about an imaginary being, Porfo, and we say ‘Porfo was not angry’. We are not asserting truth or falsehood, since Porfo is not assumed to exist in any domain of real entities. We are painting a picture, and saying the image we are trying to convey is that of an entity without the property of being angry; ‘is angry’ does not apply to the image we are trying to convey.

Added insight into the concept as used in A-logic will be gained by looking at the role the negation sign plays in the formal logic. Although we shall usually read ‘~P’ as ‘it is not the case that P’, or as ‘not-P’, these expressions do not add much clarification. Clarity is sharpened some by distinquishing different ways negation operates in different modes of discourse. Despite difficulty in providing an intuitively clear synonym, the negation sign ‘~’ has meaning, as do ‘&’ and ‘v’. Like ‘&’ and ‘v’, it is a syncategorematic locution. Its meaning lies in how it changes the meaning of what it operates on, not in any substantive content of its own. To know how it changes or affects the meaning of what it operates on is to know its meaning. It is used in all disciplines and contexts—it is not subject specific. It is syncategorematic. It is therefore a purely logical constant.

Given any meaningful predicate or sentence P, the result of prefixing a negation sign, [~P], has a different meaning than the unnegated expression. Different predicate schemata, ‘P’ and ‘Q’, do not <u>necessarily</u> represent predicates which differ in meaning, though they may. But the meaning of ‘~’ assures us that ‘P’ and ‘~P’ <u>will</u> differ in meaning. An expression which is negated always has a meaning which is “opposite” to the meaning of the same expression unnegated. More precisely, in logic we say a negated predicate (or sentence) contradicts, or “is the contradictory of “, the same predicate (or sentence) by itself.

Compare the two predicates, a) ‘<1>is round’ and ‘<1>is square’, with the two predicates b) ‘<1>is round’ and ‘<1>is not round’. In both pairs, the two expressions have opposing meanings. But in the first pair this opposition <u>can not</u> be displayed in formal logic; the best we can do is represent the difference by using different predicate letters, as in ‘P_1<1>’ and ‘P_2<1>’. The latter are only placeholders for two predicates which may or may not be opposed and may or may not even be different. The second pair, however, signifies a firm opposition by the purely formal symbols, ‘P_1<1>’ and ‘~P_1<1>’. We know that ‘P_1’ is a placeholder for some predicate, though it is open what predicate that might be. We also know exactly how predicates represented by ‘P_1<1>’ and ‘~P_1<1>’ will

2. In Chapter 7, the differences between negation or ‘it is not the case that’, negation ‘it is false that…’ and ‘it is not true that…’ are displayed in 3-valued truth-tables, where ‘0’ means neither true nor false: ‘It is false that P’ symbolized by ‘FP’ is defined as ‘T~P’, so that; It is not false that P’ is ‘~T~P ‘ and ‘It is neither true nor false that P, symbolized by ‘0P’ Syn$_{df}$ ‘(~TP & ~FP) ‘.

P	~P	TP	~TP	T~P	0P
0	0	F	T F	F0	T
T	F	T	F T	FF	F
F	T	F	T F	TT	F

3. In *Begriffschrift*, 1879, Frege first associated his sign for negation not with truth and falsity, but with denial (wird verneint), as opposed to affirmation (wird bejaht).

differ; they will be contradictories. Part of the meaning of the '$\sim P_i$<1>' is negation, but it is only the syncategorematic, non-substantive part signified by '∼'.

4.13 POS and NEG Predicates

Is 'immortal' positive, or negative? The terms 'positive' and 'negative' have no single clear meaning in common usage, logic, or linguistics. Frege saw no logical use for the concept of negative judgments as distinct from positive (or "affirmative") judgments[4] and Russell held that there is no formal test of whether a proposition is negative or positive.[5]

In A-logic a rigorous formal distinction is drawn between wffs which are 'POS' and those that are 'NEG'. Every wff schema is *prima facie* either POS or NEG, exclusively. Every unambiguous expression or sentence translated into the language of M-logic or A-logic will be POS or NEG exclusively. The distinction relies on the concept of atomic predicates as POS expressions, and on differences of scope among negation signs.

It does not follow, of course, that every wff gotten by substitution in a POS wff is POS, for unless it is restricted, U-SUB allows any atomic wff to be replaced by a NEG wff. But some POS (or NEG) wffs are such that all wffs derived by U-SUB from them are POS(or NEG) and these are of special interest to logic.

Ordinary language does not always make clear whether the predicate used is POS or NEG as we define it. This is illustrated by 'The present king of France is not bald', which can be read two ways, distinguishable by their logical form: "There is a present king of France and he is not bald" and "It is not the case that there is a present king of France that is bald". They have the logical forms, '(∃x)(PKFx & ∼Bx)'and '∼(∃x)(PKFx & Bx)' respectively. These differ in the scope of the negation sign and in meaning. The first is a POS statement, the second is a NEG statement. However, in A-logic the distinction between POS and NEG does not depend on quantifiers; it lies in the logical structure of a predicate regardless of what the quantifier may be: '(P<1> & ∼Q<1>)' is POS and '∼(P<1> & Q<1>)' is NEG. In Q-wffs, it is established in the matrix of the prenex normal forms and if a wff is POS (or NEG) so are all of its logical synonyms, including any synonym with a negation sign prefixed to one or more quantifiers.

Intuitively, a predicate is POS if it can be viewed as describing at least one distinct, specific property or relation that might conceivably be present in some object or n-tuple of objects. If it is NEG, it does not contain or entail, by itself, the idea of the presence of any one property or relation in any object; it conveys only the idea of the non-presence or absence of one or more positive properties or relations.

The formal definition of 'POS' and 'NEG' assumes that atomic predicates are by definition POS, and their denials are NEG. E.g., '<1>is red' is POS, 'it is not the case that <1>is red', viewed all by itself, is NEG. Also '<1>is a member of<2>' is POS, and '∼(<1>is a member of <2>)' is NEG. Whether a compound predicate is POS or NEG depends on the connective used. A conjunction is POS if and only if one or more of its conjuncts are POS; otherwise it is NEG. A disjunction is POS if and only if all of its disjuncts are POS, otherwise it is NEG. If a predicate is NEG, its negation is POS. Thus the definition of POS and NEG is a generative definiton:

---

4. See Frege's essay, "Negation" (1919), in Geach and Black's Translations, especially pages 125-126.
5. See Russell's discussion of negative facts, in "The Philosophy of Logical Atomism" (1918), especially pp 211-216 in his *Logic and Knowledge*.

Df 'POS': '<1>is POS' Syn_{df} '<1>is a wff and either (i)<1>is atomic,
or (ii) <1>is [P & Q] and P is POS,
or (iii) <1>is [P v Q] and P is POS & Q is POS,
or (iv) <1>Syn Q and Q is POS.'

Df 'NEG': '<1>is NEG' Syn_{df} '<1>is a predicate and [<1>Syn ~P] and P is POS.'

The following principles of inference follows from these definitions:

1 If [P] is POS and [P SYN Q] then Q is POS
2 If [P] is NEG and [P SYN Q] then Q is NEG
3 If [P] is atomic, then [P] is POS
4 If [P] is atomic, then [~P] is NEG
5 If [P] is POS then [P&Q} is POS
6 If [P] is NEG, then [PvQ] is NEG
7 If [P] is POS and [Q] is POS, [PvQ] is POS
8 If [P] is NEG and [Q] is NEG, [P&Q] is NEG
9 If [P] is POS then [~P] is NEG
10 If [P] is NEG, then [~P] is POS
11 If [PvQ] is POS, then Q is POS
12 If [P & Q] is NEG, then Q is NEG

POS and NEG are properties of wffs. Asserting that a wff is POS or NEG, differs from asserting that every expression that can be gotten by substitution from that wff is POS or NEG. This distinction is expressed symbolically by the difference between asserting that a wff is POS and asserting that a quasi-quoted wff is POS:

(i) " '(P<1> & ~Q<2>)' is POS", is *prima facie* true by our definition, but

(ii) "[P<1> &~Q<2>] is POS", is neither true nor false, since it means "a result of replacing 'P<1>' by a predicate or predicate schema and 'Q<2>' by a predicate or predicate schema in '(P<1> &~Q<2>)' is POS". The results of such a replacement may be either POS or NEG depending on whether the replacements for 'P' and 'Q' are POS or NEG. If there are n different atomic predicates in a wff, there are 2^n distinct possible cases.

The concept of a wff's being logically POS, or 'L-POS', applies to some quasi-quoted wffs. A wff is L-POS if and only if all instantiations of every wff gotten by U-SUB from that wff are POS. Correspondingly, a wff is L-NEG if and only if all instantiations of every wff gotten by U-SUB from that wff is NEG. A wff which is neither L-POS not L-NEG is Contingent.

With respect to quantification, a Q-wff is POS if and only if the matrix of its prenex-normal form is *prima facie* POS; otherwise it is NEG. Its expansion in every domain retains the property of POS or NEG which belongs to that matrix.[6] Thus if the matrix of a prenex normal form Q-wff is NEG, then the Boolean-expansion of that Q-wff in every domain is NEG. This is illustrated by expansion in a domain of 3:

Example 1: '(∀x)~Px' Syn '(~Pa & ~Pb & ~Pc)'
Both '(∀x)~Px' and ' (~Pa & ~Pb & ~Pc)' are *prima facie* NEG

Example 2: '(∃x)Px' Syn '(Pa v Pb v Pc)'
Both '(∃x)Px' and ' (Pa v Pb v Pc)' are *prima facie* POS.

The property of being *prima facie* POS (or NEG) is preserved in all synonyms. For example, if a predicate schema is NEG, every prenex form of a quantification of it is NEG. The matrix of the prenex normal form of '(∀x)(∃y)(∀z)(~Fyz v Fxz))' is NEG and its logical synonym by T4-31 and

---

6. It is proved later in this chapter that all quantified wffs in which negation signs occur are reducible to prenex normal form.

SynSUB,'(∀x)(∃y)(∀z) ~ (Fyz & ~Fxz))' is also NEG, as is its synonym by Q-Exchange and SynSUB '~(∃x)(∀y)(∃z)(Fyz & ~Fxz))'. Its synonym by Rules of Passage also preserve POS/ NEG; e.g., '(∃x)(∀y (∃z) ~ (Fxx & ~Fyx &(Fxz v ~Fyz))' is *prima facie* NEG, as is its synonym, '(∃x)(~ Fxx &(∀y)(~ Fyx & (∃z)(Fxz v ~Fyz)))'.

The denial of a POS Q-wff is NEG, and the denial of a NEG Q-wff is POS. Neither the conjunctive quantifier nor the disjunctive quantifier are intrinsically POS and NEG; these properties are determined essentially by the positioning of negation signs in a predicate.

As is the case with unquantified wffs, if a Q-wff is *prima facie* POS, it does not follow that POS applies to the same Q-wff in quasi-quotes. For the results of using U-SUB to replace atomic components by POS or NEG expressions yields different results in different cases. For example, the expansion of '(∀x) ~ Px' in any domain is *prima facie* NEG, but when put it in quasi-quotes [(∀x) ~ Px], its expansions [~ Pa_1 & ~ Pa_2 & ...] are not L-NEG or L-POS, but CONTINGENT because '[(∀x) ~ Px]' allows the possibility of either

a) '((∀x) ~ (Px v ~Px))' [U-SUB '(P<1> v ~P<1>)' for 'P<1>'] a) would be L-POS.
or b) '((∀x) ~ (Px & ~Px))' [U-SUB '(P<1> & ~P<1>)' for 'P<1>'] b) would be L-NEG.
or c) '((∀x) ~ (Px & ~Qx))' [U-SUB '(P<1> & ~Q<1>)' for 'P<1>'] c) " CONTINGENT.

However, if a) and b) are put in quasi-quotes, then unlike '[(∀x) ~ Px]', the results would be L-POS: and L-NEG respectively:

a') '[(∀x) ~ (Px v ~Px)]' is L-POS, because no matter what is substituted by U-SUB for its atomic component,'P<1>', the result in all domains is POS, and
b') '[(∀x) ~ (Px & ~Px)]' is L-NEG, for no matter what is substituted by U-SUB for its atomic component,'P<1>', the result in all domains is NEG.
c') '[(∀x) ~ (Px & ~Qx)]' on the other hand is CONTINGENT; By U-SUB on its two atomic components, it could be made either L-POS, L-NEG, or neither.

All theorems in M-logic's quantification theory are quasi-quoted Q-wffs that are provably L-NEG because the predicate of the quantification is L-NEG. Any wff gotten by U-SUB from such a quasi-quoted wff is such that it, and any expansion of it in any domain is L-NEG. These are essentially quantifications of L-NEG predicates.

But it is not enough for theoremhood in M-logic, that a wff be L-NEG. The denial of a theorem must be inconsistent. Some Q-wffs are inconsistent or tautologous in smaller domains but not in larger domains. For example, [Pa v ~Pb] is L-NEG but it is not a theorem, and [(∃x)(∀y)(~ Rxy v Ryx)] is L-NEG but not a theorem because it is tautologous in domains of one or two members but is not tautologous in domains of three or more members. Thus being L-NEG is a necessary, but not sufficient, condition for theoremhood in M-logic. To understand the essential character of theorems of M-logic, we need the concepts of inconsistency and tautology developed in Chapter 5.

In analytic logic, SYN- and CONT-statements themselves (which are 2nd-level statements about language) are always POS since 'SYN' and 'CONT' are atomic predicates and their denials are NEG statements. But their component subject-terms—whether wffs, wffs in quasi- quotes, or actual predicates or statements—may be either POS or NEG viewed by themselves. The latter are what we are talking about in A-logic.

Using the definitions of POS and NEG, and the principles of inference, 1 to 12, which followed from them, one can establish by direct inspection whether any given wff in M-logic or A-logic is *prima facie* POS or NEG. For example, working outward from atomic components, both pure predicates, quantificational predicates and expansions of quantificational predicates can be determined to be prima facie POS or NEG Hence (letting ‘p’ mean ‘POS’ and ‘n’ mean’NEG’)

	‘$(\exists_3 x)$	(Px	&	~	Qx)’ is *prima facie* POS.
	p	p	**p**	n	p
by principles on page 174,	7	3	5	4	3

and If ‘(P<1> & ~Q<1>)’ is POS, then ‘((Pa & ~Qa) v (Pb & ~Qb) v (Pc & ~Qc))’ is POS

	P<1>	&	~	Q<1>		Pa	&	~	Qa	v	Pb	&	~	Qb	v	Pc	&	~	Qc
	p	**p**	n	p		p	p	n	p	p	p	p	n	p	**p**	p	p	n	p
by principles	3	**5**	4	3		3	5	4	3	7	3	5	4	3	7	3	5	4	3

The same principles can be used to construct the “POS/NEG” table of a quasi-quoted wff. It tells us how the results of substituting POS or NEG expressions by U-SUB for different atomic components, will make a compound expression POS or NEG. Each row in the POS/NEG table represents one alternative, and shows how that replacement of atomic predicates with POS or NEG expressions, will make the resulting whole expression POS or NEG in that case:[7]

	[P]	[~P]	[Q]	[~Q]	[P & Q]	[P v Q]	[P ⊃ Q] [~P v Q]	[P ≡ Q] [(~PvQ) & (~QvP)]	[P v ~P]
Row 1:	n	p	n	p	n **n** n	n **n** n	p **n** n	n n n	n n p n
Row 2:	p	n	n	p	p **p** n	p **n** n	n **n** n	n p p	p n n p
Row 3:	n	p	p	n	n **p** p	n **n** p	p p p	p p n	n n p n
Row 4:	p	n	p	n	p **p** p	p **p** p	n **n** p	n n n	p n n p

The properties of POS and NEG apply to both predicates and sentences (saturated predicates). These properties of Q-wffs belong to sentences iff they belong to their over-all predicates.

If a quasi-quoted wff is an unquantified predicate schema or unquantified sentence schema, one can determine by POS/NEG tables whether it is L-NEG or not, and this is enough to establish it as a theorem of M-logic. But if it is quantified, for reasons mentioned, finding the wff L-NEG is not sufficient to prove that it is a theorem.

If [P CONT Q] and the antecedent, P, is NEG, then the consequent,Q, must be NEG.
I.e., If [P CONT Q] and NEG(P), then NEG(Q))]

Proof: 1) P is NEG, then the MOCNF(P) is NEG
(from Lemmas: If A SYN B, then POS(A) iff POS(B)
If A SYN B, then NEG(A) iff NEG(B)
2) If P CONT Q then P SYN (P&Q) [Df’CONT’]
3) If NEG(P) and P SYN (P&Q) then NEG(P&Q)

---

7. POS-NEG tables are related to the two-valued truth-tables of mathematical logic in the following respects: In propositional logic, if ‘T’ is replaced by ‘N’ (for ‘NEG’) throughout the truth-table of a wff and ’F’ is replaced throughout by ‘P’ (for ‘POS’), the truth-table is converted into the POS/NEG table of the same wff, and vice versa As we shall see in Chapter 5, an expression is inconsistent only if it is POS in the final column of every row, and tautologous only if NEG in all rows. In analytic truth-logic, two-valued truth-tables are rejected in favor of three-valued truth-tables for truth-logic.The POS/NEG distinction among predicates identifies both ‘(Pa v ~Pa)’ and ‘(Pa v ~Pb)’ as NEG, but does not distinguish the tautologousness of the first statement form from the contingency of the second. This is the crucial distinction for M-logic, and one that can be made by truth-values. See Chapters 7 & 8.

4) If NEG(P) and (P CONTQ) then NEG(P&Q) [3),2),SynSUB]
5) If NEG(P&Q) then NEG(Q)
6) If P CONT Q & NEG(P) then NEG(Q) [4),5),HypSYLL]

It follows that if P is NEG and Q is POS, then P does not CONT Q. However, if P is POS and [P CONT Q], then Q may be NEG—e.g., '(Pa & ∼Qb)' is POS and logically contains '∼Qb' which is NEG. But if Q is POS and P CONT Q, then P is POS.

Although the POS/NEG distinction does not settle all problems of theoremhood in M-logic, it is useful in the development of a theory of definition which will avoid certain paradoxes which have beset M-logic. Since definitions are within our control, we may set the following rule:

Rule of Definition:
In any definition, the definiendum and the definiens should both be POS.

If this rule is followed, then the following will be true: If [P Syn_{df} Q] then POS[P] and POS[Q]

This is sufficient for purposes of language. Nothing is lost. For, we can negate any POS expression to get a NEG one. Whatever was an allowable NEG expression using NEG definienda, can be introduced by negating a POS expression. Just as many NEG expressions are allowed each way. What is avoided are only the man-made paradoxes that would otherwise make a contradiction follow logically from a definition.

The most famous example for logicians, is the source of Russell's paradox. Let '$\sim(<1>\in<1>)$' mean 'it is not the case that $<1>$ is a member of $<1>$' or, "$<1>$ is not a member of itself". When we have a POS predicate as definiens and a NEG predicate as definiendum, in certain cases a contradiction follows logically, by the rule of U-SUB(Universal Instantiation). Let 'W' mean 'the class of elements that are not members of themselves':

1) $<1>\in W\ Syn_{df} \sim <1>\in<1>$ Or, in M-logic: 1) $(x\in W \equiv \sim x\in x)$
   P n p p n p
2) W∈W Syn ∼W∈W [1),U-SUB,'W'for '<1>] 2) $W\in W \equiv \sim W\in W$ [1),UI]

This violates the principle that if two expressions are Synonymous, they are both POS or both NEG. Hence we prohibit defining a POS predicate by a NEG definiendum.

There is no problem if definiens and definiendum are both POS. Suppose we let 'W' stand for "the class of entities that are members of themselves". Then by U-SUB, If [$<1>W\ Syn_{df} <1>\in<1>$] then [W∈W Syn W∈W]. And in fact, what Quine and other mathematical logicians have done to avoid the paradox is to define both classes and elements of classes with a POS defninens and a POS definiendum. E.g., letting '{x:Px}' stand for the class of things that are P, 'y∈{x:Px}' is defined as '(∃z)(y∈z & (∀x)(∼x∈y v Px))' and a class is a member of some class is defined as follows: '{x:Px}∈<1>' Syn_{df} '(∃y)(y∈<1> & y={x:Px})'. In both cases the definiens and definiendum are both POS. Further difficulties arise because the definition of class identity is based in M-logic on the TF-biconditional, which is NEG, thus what looks like a positive relation, 'a is identical with b', is instead a NEG predicate: ('a=b' Syn_{df} '(∀x)(a∈x ≡ b∈x)' Syn '(∀x)(∼(a∈x & ∼b∈x) & ∼(b∈x & ∼a∈x))'. Thus Quine's class of all elements, 'V', has NEG predicate to define the notion of element, which one would think would be a POS notion.

Of course the POS/NEG distinction drawn here is not ordinary usage. Ordinary usage is unclear on 'positive" and "negative" and this is not unclear. This theory of the POS/NEG distinction is a prescriptive proposal for logical purposes. Just as M-logic provided a new system for putting universal and particular propositions into a canonical form, this provides a method for putting atomic predicates, sentences and their definitions, into a canonical POS form, so that the negation sign remains a linguistic operator, but never a predicate. Nothing is lost and the concepts of POS and NEG are useful in disambiguating statements and predicates in both logic and ordinary language.

The expression '<1> is immortal' is normally taken to apply to a living being (imaginary or not). One way to interpret it, is as saying "<1> doesn't die". In M-logic this may be rendered either

'(<1>is a person & ~(<1>is mortal))'
or '~ (<1>is a person & <1>is mortal)'.

The first is POS because its first conjunct is POS, the second is NEG.

Considered by itself, if '<1>is immortal' means '<1>does not ever die' it is NEG, and expressible as '(∀t)~(<1>dies at t)' or, "<1> does not die at any time", or synonymously, '~(∃t)(<1>dies at t)' i.e., "It is not the case that there is a time at which <1>dies". But "immortal" may be intended to mean "lives forever", in which case it is expressible as POS statement: '(∀t)<1>is alive at t)' i.e., "<1>is alive at all times" or '~(∃t)~ <1>is alive at t)' i.e., "It is not the case that there is a time when <1>is not alive" which is also POS. Being dead is not the same as not being alive. The positive description of a death involves a change from the vital signs of life to the cessation of all such signs. Stones are never alive so they don't die. Thus, individuals may mean different things in ascribing immortality; the POS/NEG distinction provides a way to disambiguate and make clear what they mean.

In the next chapters we will examine relationships between the POS and NEG distinctions and inconsistency, tautology and contingency, and how POS/NEG tables can be used to determine which wffs are logically inconsistent, logically tautologous, or simply contingent. In Chapter 6, we examine how it applies to C-conditionals. In Chapter 7 it is related to concepts of being true, false not-true or not-false. In Chapter 8 we examine the connection between the POS/NEG distinction and Validity in truth-logic.

4.14 Roles of the Negation Sign in Logic

We may distinguish two important roles for the negation operator in logic. The first is that of a logical operator which can create samenesses and differences in meanings, and thus affect logical synonymy and logical containment. The second is that, with conjunction and disjunction, it can be used to explicate the meanings of the logical predicates '<1>is inconsistent' and '<1>is tautologous'.

In this chapter we deal only with the first of these two logical roles: the ways in which negation affects meanings, logical synonymy and containment among disjunctive and conjunctive sentences. Tautology and inconsistency are neither used, nor presupposed in what follows. SYN- theorems and CONT-theorems in A-logic do not depend on these concepts.The second role is developed in Chapter 5; there the logical concepts of tautology and inconsistency are developed and provide a semantic criterion for all and only the theorems and anti-theorems of mathematical logic.

In semantical definitions generally, negation facilitates definition by differentiation as in,

'<1>is a bachelor' syn_{df} '(<1>is male & ~ <1>is married)'

Or, in chemistry we may define,

'<1>is an element' syn_{df}
'<1>is a substance & ~(<1>can be broken down into simpler substances by chemical means)'

These are POS in definiens and definiendum; they have the form [P_1<1> syn (P_2<1> & ~P_3<1>)] where P_1 and P_2 are atomic, hence are POS. In logic, since predicate letters are not substantive predicates, we get forms of differentiation but no differentiation of actual non-logical predicates.

Thus one principle of definitions, useful to avoid paradoxes, is that definiens and definiendum should both be POS, or both be NEG. Russell's paradox and other paradoxes come from a violation of this. For example, from "x∈A =df ~x∈x", where A is supposed to be the class of classes which are not members of themselves, yields '(∀x)(x∈A ≡~x∈x) hence, by UI, ⊢ {A∈A ≡ ~A∈A).

4.15 The Negation Sign and Logical Synonymy

The role of negation in with respect to logical synonymy is radically different from that of conjunction and disjunction.

The SYN-axioms of conjunction and disjunction display the most basic ways in which pairs of logical structures can be logically synonymous while differing in repetition, order, or grouping. What is essential and constitutive of logical synonymy and containment is that meaning is preserved despite differences in the position, order and grouping of similar components in compound symbols—i.e., it is preserved across certain differences in logical structure.

Negation does not effect the internal logical structure of what it is prefixed to at all. It does not introduce or eliminate any repetitions of elementary wffs. It does not alter the position of any elementary wff in the left-to-right order in which elementary wffs occur. It does not introduce or alter any grouping of components. As a unary operator, it has no rules for grouping and ordering. Components of a negated expression, large or small, can be moved about within the logical structure only by rules for idempotence, commutation, association, or distribution of conjunction and/or disjunction without changing the meaning of the negated expression as a whole.

Thus negation, though essential to theoremhood or inconsistency in M-logic, has no essential or constitutive role in determining logical synonymy. As an operator which can signify a difference in meanings or, by double negation or definitions, express the same meaning in a different way, it is like predicate letters, which as occurrences of different letters stand for expressions with meanings which may be different (but need not be), or as occurrences of the same letter represent the same meanings. However, a negated predicate letter must represent an expression with a different meaning than the expression represented by the predicate letter alone.

Whether the meaning represented by a component is differentiated from some other component by prefixing a negation sign, or by using a different predicate letter makes no difference to logical synonymy. All wffs are synonymous with normal form wffs which have negation signs prefixed only to atomic wffs. Logically synonymous pairs always have all and only the same set of basic elementary components. In all significant cases (excluding instances of the triviality, ⊨ [P SYN P], and excluding definitional or abbreviational synonymies) logical synonymy involves wffs with the same ultimate components on each side, but with differences in their order of occurrence, repetition or grouping—i.e., in the patterns of cross reference among them. Thus negation's role with respect to logical synonymy is subordinate to that of conjunction and disjunction.

Negation changes components' meanings; conjunction alone and disjunction alone do not. When any expression is negated, the meaning of the resulting expression is completely different.

When two different predicates are conjoined or disjoined, the result does not change the meanings of its components. Of course the meanings of the new whole—the conjunction or disjunction—is not merely the meaning of one of the predicate listed alongside the meaning of the other. The conjunction or disjunction of two distinct predicates has a meaning over and above the meanings of the two predicates listed separately. Truth-tables and POS-NEG tables make clear how the meaning of a conjunction differs from that of disjunction. If a single predicate is conjoined or disjoined with itself, the new whole means the same as the predicate all by itself (idempotence). But in all cases involving only conjunction and disjunction the meanings of the components are embedded, unchanged in the compound.

The meaning of a conjunctive or disjunctive predicate can not apply to any state of affairs unless the independent meaning of one or both of the components does. In contrast the meaning of a negated predicate applies if and only if the meaning of its component predicate does not apply.

4.16 Synonymies Due to the Meaning of the Negation Sign

Though negation has no essential role in transformations of logical structure, there are synonymies due to the specific way a predicate's meaning differs from that of its negation.

With the introduction of negation, two new kinds of SYN-theorems are possible. These do not involve any change in the sequential order or grouping of atomic wffs (as in COMM or ASSOC), nor do they change the number of occurrences of atomic wffs (as in IDEM or DIST).

One kind is due to the principle of double negation: for any P, P is synonymous with ∼∼P. Negation is self-canceling; the negation of the negation of an expression means the same as the expression alone. Replacing '∼∼P' by 'P' does not change the order or the grouping of any atomic components and does not affect the logical structure. One may prefix as many negations signs as one wishes to any formula; the result is synonymous with P if an even number of negations signs are prefixed and it is synonymous with ∼P if an odd number of negation signs are prefixed.

The other new kind of SYN-theorem is based on DeMorgan theorems. A compound expression with negation and conjunction is synonymous with another compound without conjunction but with disjunction instead. Again there is no change in order or grouping of atomic components. In certain cases it is held that negation combined with another logical constant is SYN with, a third logical constant. Following DeMorgan, 'P or Q' is referentially synonymous with "not(not-P and not-Q)' and 'P and Q' is referentially synonymous with "not(not P or not-Q)'. I.e., the definition D4-5 of disjunction, or Df'v', is: [(P v Q) SYN_{df} ∼(∼P & ∼Q)].

By defining '(P v Q)' as an abbreviation of '∼(∼P & ∼Q)', we can get, using double negation and U-SUB, all eight basic Demorgan theorems, six of which we prove as theorems:

Df 'v'.	[(P v Q) SYN ∼(∼P & ∼Q)]	T4-11.	[(P & Q) SYN ∼(∼P v ∼Q)]
T4-16.	[(∼P v Q) SYN ∼(P & ∼Q)]	T4-15.	[(∼P & Q) SYN ∼(P v ∼Q)]
	[(P v ∼Q) SYN ∼(∼P & Q)]		[(P & ∼Q) SYN ∼(∼P v Q)]
T4-18.	[(∼P v ∼Q) SYN ∼(P & Q)]	T4-17.	[(∼P & ∼Q) SYN ∼(P v Q)]

The truth-functional "conditional" can also be defined as abbreviating a compound using negation and conjunction D4-6. [(P ⊃ Q) SYN_{df} ∼(P & ∼Q)] [Df '⊃'] and further DeMorgan theorems can be established with it.

In Quantification theory it follows that the disjunctive quantifier, '(∃x)', will be referentially synonymous with '∼(∀x)∼', and the conjunctive quantifier, '(∀x)', will be synonymous with '∼(∃x)∼'. These are known as laws of quantifer exchange or Q-Exch. Once again, the resulting synonymies do not change the order or grouping of any atomic components.

4.17 Outline of this Chapter

With this chapter we get a complete axiomatization of logical synonymy and containment with respect to wffs of mathematical logic. But we do not get mathematical logic itself. We get that fragment of Analytic logic which deals only with the wffs of M-logic. The only basic novelty is the introduction of negated wffs by U-SUB and the use of SynSUB with definitional synonymies or the axiom of double negation. All previous principles of synonymy and containment apply.

Section 4.2 deals with consequences for the object language of introducing a negation operator. Section 4.3 deals with negation in sentential logic; Section 4.4 deals with negation in quantification theory. Section 4.5 establishes the soundness and completeness of this fragment of analytic logic with respect to logical synonymy and containment among wffs of M-logic.

There are three consequences of adding '$\sim$' to the language of the previous chapter: (1) an additional class of wffs in the symbolic object-language; (2) expansion of the Axiomatic System and its theorems, (3) revision of the SYN-metatheorems and their proofs. In Section 4.2, we show how the introduction of negation and the definition '(P v Q)' SYN_{df} '$\sim$($\sim$P&$\sim$Q)' affects the set of well-formed wffs, increasing enormously their numbers, and the number of pairs of wffs which stand in the relations SYN or CONT. In Section 4.3, we add one axiom (Double Negation). This allows elimination of the four even-numbered axioms of Chapter 1 which can now be derived as theorems from the five axioms which are left. On introducing a negation operator we must reformulate the axiomatic base of analytic logic so that its theorems will include all and only logically SYN or CONT pairs of the wffs in standard logic. In Section 4.4 theorems are derived from the new axiom set and the new definition of 'v'. These include DeMorgan theorems, theorems of Quantifier-exchange (which are 'SYN'-for-'≡' analogues of Quine's *130 to *133), new proofs of of SYN-theorems of Chapter 3 utilizing definitions employing negation, and nine SYN- or CONT-theorems which are analogues of Quine's theorems in *Mathematical Logic* (ML) with truth-functional biconditionals or conditionals:

T4-33.	[(∃x)(Px ⊃ Qx) SYN ((∀x)Px ⊃ (∃x)Qx)]	ML*142
T4-34.	[(∃x)(Px ⊃ Q) SYN ((∀x)Px ⊃ Q)]	ML*162
T4-35.	[(∀x)(Px ⊃ Q) SYN ((∃x)Px ⊃ Q)]	ML*161
T4-36.	[((∃x)Px ⊃ (∀x)Qx) CONT (∀x)(Px ⊃ Qx)]	ML*148
T4-37.	[(∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)]	ML*101
T4-38.	[(∀x)(P ⊃ Qx) SYN (P ⊃ (∀x)Qx)))]	Cf. ML*159
T4-39.	[(∀x)(Px ⊃ Qx) CONT ((∃x)Px ⊃ (∃x)Qx)]	ML*149
T4-40.	[((∃x)Px ⊃ (∃x)Qx) CONT (∃x)(Px ⊃ Qx)]	ML*150
T4-41.	[((∀x)Px ⊃ (∀x)Qx) CONT (∃x)(Px ⊃ Qx)]	ML*151

Section 4.5 will show that the re-formulated principles expressed in SYN-metatheorems 1 to 13, are essentially unaffected by the introduction of negation, and that the soundness and completeness results in Chapter 1 can be extended to the wffs of quantification theory in mathematical logic. A primary objective of this chapter is to demonstrate that the principles of logical synonymy and containment are independent of negation.

4.2 Additions to the Logistic Base

The introduction of negation adds a new primitive, making some of the primitives in earlier chapters derivable from definitions using negation and thus becoming non-primitive. It also involves an additional axiom, "double negation", and an additional derived rule of inference.

4.21 Well-formed Formulae with Negation, Defined

By adding a negation sign to the language of negation-free logic, the set of wffs is expanded to become the same as the set of wffs in classical PM-type logic. The class of pairs which are SYN is enlarged exponentially.[8] New definitions using negation permit the set of primitive symbols to be revised and simplified, and new symbols are added. As a result of these changes adjustments must be made in previous concepts of an elementary wff, of basic normal forms, and of duals.

Formally, the symbolic language is now described as follows:

I.Primitive symbols: [**Adds ' ~ '**; **drops 'v' and 'E'**]

1. Logical constants: & ~ ∀
2. Grouping devices:) (> <
3. Predicate letters: P_1 P_2 P_3 … (The class, {PL})
4. Argument Position Holders: 1 2 3 … (The class, {APH})
5. Individual constants: a_1 a_2 a_3 … (The class, {IC})
6. Individual variables: x_1 x_2 x_3 … (The class, {IV}) [9]

   [We shall also use 't_1', 't_2'… for "subject terms",
   i.e., members of the class, ST = the class **{**{APH}∪{IC}∪{IV}**}**]

II. Rules of Formation [Drops ' [P v Q]' in FR1-2 adds ' [~P]' in FR4-5]

FR4-1. If P_i e{PL}, then P_i e wff.
FR4-2. If P and Q are wffs, [P&Q] is a wff, **[AvB] dropped]**
FR4-3. If P_i e{PL} and t1,…,ti e**{**{APH} ∪ {IC}**}**, then [P_i<t1…ti>]e wff.
FR4-4. If t_i e{IV}, then (∀t_i)P<…t_i…> e wff, **[(∃x)P] dropped]**
FR4-5. If P is a wff, then [~P] is a wff. [Added]

Obviously, the introduction of ' ~ ' required an addition to the Rules of Formation, in FR4-5. If A is a wff, then [~A] is a wff. This rule allows ' ~ ' to be prefixed to any well-formed wff, or mode of same, simple or compound.

III. Definitions:

D1. (P&Q&R) SYN_{df} (P&(Q&R)) (**D1b dropped**)
D2. (V_kx)Px SYN_{df} (Pa_1 & Pa_2 & … & Pa_k) (**D2b dropped**)
D3. (P v Q) SYN_{df} ~(~P & ~Q) ["DeM1", Added]
D4. (P ⊃ Q) SYN_{df} ~(P & ~Q) [Added]
D5. (P ≡ Q) SYN_{df} ((P ⊃ Q)&(Q ⊃ P)) [Added]
D6. (∃x)Px SYN_{df} ~(∀x)~Px [Added]

4.211 Increased Variety of Wffs

The term "elementary wff" is now changed to cover both atomic wffs and their negations. The predicate 'is an atomic wff' will only apply to predicate letters alone or predicate letters prefixed to ordered n-tuples

---

8. E.g., Without negation the two binary connectives,'&' and 'v', can yield only four distinct truth-functions using two variables; with negation added they yield 16 distinct truth-functions. If two expressions are synonymous, then they must express similar truth-functions. Thus the increase in expressible truth-functions represents an increase in distinguishable kinds of synonymous pairs, even though pairs with similar truth-functions are not necessarily synonymous.

9. 'PL' abbreviates 'predicate letters', 'APH' abbreviates 'argument position holders', 'IC' abbreviates 'individual constants', and 'IV' abbreviates 'Individual Variables'. At times for convenience we use x,y,z instead of x1 x2 x3; a,b,c instead of a1 a2 a3 … and P,Q,R instead of P1 P2 P3.

of argument position holders, individual constants or individual variables. The predicate “is an elementary wff” will apply only to an atomic wff and to atomic wffs with negation signs prefixed. Thus atomic wffs with or without negation signs prefixed to them comprise the class of elementary wffs.

The variety of compound wffs is increased by the inclusion of negations of any wffs in negation-free language with ‘&’ and ‘v’ . Compound wffs in addition include denials of conjunctions or disjunctions; e.g., ‘ ~ (P&Q)’, ‘ ~ (Pv ~ Q)’.

But they are also increased by the use of new abbreviations made possible by negation. As in standard logic, disjunction can be defined in terms of conjunction and denial. Thus we drop ‘v’ as a primitive symbol and define it as the abbreviation of expression using ‘ ~ ’ and ‘&’:

D5. [(P v Q) SYN$_{df}$ ~ (~ P & ~ Q)]

In the same vein, we add the usual definitions of ‘⊃’(the truth-functional “if…then”) and of ‘**w**’(the truth-functional “if and only if”), thus adding two new symbols which abbreviate expressions using only conjunction and denial in primitive notation:

D6. [(P ⊃ Q) SYN$_{df}$ ~ (P & ~ Q)]
D7. [(P ≡ Q) SYN$_{df}$ ((P ⊃ Q)&(Q ⊃ P))]

Though as abbreviations these definitions increase notations without adding meanings beyond those of the primitive terms, since this chapter adds a new primitive term the new symbols do represent meanings not expressible in the negation-free wffs of Chapters 1 to 3. These additional meanings do not, however, portend changes in the concepts of logical synonymy or containment.

4.212 Adjusted Definitions of Basic Normal Form Wffs

The addition of negation and these definitions requires changes in the characterization of the Basic Normal Forms, which were previously defined only for negation-free wffs. By usual definitions, normal form wffs contain only the logical constants, ‘ ~ ’, ‘&’ and ‘v’, but in previous chapters there were no negation signs. Within a Basic Normal Form the elementary components may now include negated predicate letters and negated atomic wffs. The addition of negated atomic wffs requires amendments to the definition of the ‘alphabetic order’ of elementary wffs in the definition of Ordered Basic Normal Forms. But also, in the reduction to a normal form, the newly defined constants ‘⊃’ and ‘≡’ in D6 and D7 must be eliminated by definitional substitution.

Thus additional steps must be added to the procedure of reduction to Basic Normal Forms. Some steps move negations signs inward, others eliminate defined connectives other than ‘&’ and ‘v’, and we must supplement the rule of alphabetic order with some clause, e.g., that an atomic wff immediately precedes its own negation alphabetically and that an ordered conjunction or disjunction which begins with an unegated atomic wff, precedes any such conjunction or disjunction that begins with the negation of that wff. Such a clause would mean that ‘((P ⊃ P) v (~ (~ R&Q) & P))’ is SYN to the following basic normal form wffs (the names of its Basic Normal Forms are underlined):

CNF: ((~ PvPv ~ QvR) & (~ PvPvP))
MCNF: ((~ PvPv ~ Q) & (~ PvPv ~ QvR) & (~ PvP) & (Pv ~ PvR))
MOCNF: ((Pv ~ P) & (Pv ~ Pv ~ Q) & (Pv ~ PvR) & (Pv ~ Pv ~ QvR))
MinOCNF: ((Pv ~ P) & (Pv ~ Qv ~ PvR))

DNF: $(\sim P \vee P \vee (P\&R) \vee (P\&\sim Q))$
MDNF: $(\sim P \vee P \vee (\sim P\&P) \vee (P\&R) \vee (P\&\sim Q) \vee (\sim P\&\sim Q) \vee (\sim P\&R)$
$\vee (P\&\sim P\&\sim Q) \vee (P\&\sim P\&R) \vee (P\&\sim Q\&R) \vee (\sim P\&\sim Q\&R)$
$\vee (P\&\sim P\&\sim Q\&R))$
<u>MODNF</u>: $(P \vee \sim P \vee (P\&\sim P) \vee (P\&\sim Q) \vee (P\&R) \vee (\sim P\&\sim Q) \vee (\sim P\&R)$
$\vee (P\&\sim P\&\sim Q) \vee (P\&\sim P\&R) \vee (P\&\sim Q\&R) \vee (\sim P\&\sim Q\&R)$
$\vee (P\&\sim P\&\sim Q\&R))$
<u>MinODNF</u>: $(P \vee \sim P \vee (P \& \sim P \& \sim Q \& R))$

This completes the changes in the definitions and extensions of 'elementary wff', 'compound wff', 'normal form wffs' and 'Basic Normal Form wffs', needed to accommodate the introduction of the negation operator. For a quantified expression this clause would apply to the matrix of the prenex normal form in getting to a basic normal form of that expression.

4.213 Negation, Synonymy and Truth-values

In any given context, a sentence represented by '$\sim$P' will talk about all and only the same entities as P, and will have all and only the same elementary unnegated predicates. But 'P' and '$\sim$P' are not, and can never be, symonymous; '$\sim$P' does not say the same things about those entities as 'P'. If P represents a proposition, the states of affairs which by existing would make a proposition P true are different from the states of affairs which by existing would make [$\sim$P] true. The truth-conditions of P and $\sim$P are different, though they talk about the same entities and use the same positive predicate terms. Consequently, to keep track of sameness of meaning, sentence letters and their negations must be treated separately as non-Synonymous meaning components in compound expressions.

Standard truth-tables, which assign T's and F's only to atomic wffs, and making the truth or falsehood of $\sim$P a function of the falsehood or truth of P do not serve the purposes of the system of logical synonymy. Sentences which differ in meaning come out to be "logically equivalent" (i.e., truth-functional equivalent) if they have the same truth-values for all possible assigments of T or F to sentence letters (making all tautologies TF-equivalent and all inconsistencies TF-equivalent, as well as establishing Absorption as TF-equivalent).

If two sentences are synonymous, they must have the same truth-conditions and be composed of elementary sentences which have the same truth-conditions (i.e.,they must be made true or false, by the existence or non-existence of the same specific states of affairs). Thus for logical synonymy each negated atomic sentence or sentence letter must be treated as if it had a unique set of truth-conditions, just as each sentence letter must be viewed as (possibly) standing for a unique set of truth-conditions. Negations of any given sentence letter are treated as distinct sentence letters are treated. To preserve samness of meaning, analytic truth-tables must treat occurrences of elementary wffs 'P' and '$\sim$P' as if they were completely different variables.

The Primitives, Rules of Formation and Definitions above yield the same set of wffs as those in Mathematical Logic, except that they are interpreted as ranging over predicates in general, rather than only of saturated predicates, i.e., indicative sentences. The same points just made by reference to truth-tables, can be made by reference to POS-NEG tables for predicates, which in themselves are neither true nor false. The 'N's and 'P's in the POS-NEG tables may be thought of as representing possible entities, including the contents of our ideas, of which a predicate, simple or complex, is or is not applicable. (Cf. Section 4.13)

4.22 Axioms and Rules of Inference with Negation and Conjunction

For purpose of analytic logic, the introduction of negation permits the addition of one new SYN-Axiom schema:

Ax.4-05. [P SYN ∼∼P] [DN—Double Negation]

This says that any indicative sentence P, talks about the same entities and says the same things about those entities as a sentence which negates the negation of P. For example, 'Joe died' is referentially synonymous with 'it is not the case that it is not the case that Joe died'; both talk about Joe, say of him that he died, and are true or not under exactly the same circumstances. Double-negations are negation-canceling negations.

With Double Negation and the definition, D5, of disjunction, it is possible to drop four disjunctive axiom schemata which were needed for the negation-free logic, namely,

Ax.1-02. P SYN (PvP)	[v-IDEM]
Ax.1-04. (PvQ) SYN (QvP)	[v-COMM]
Ax.1-06. (Pv(QvR)) SYN ((PvQ)vR)	[v-ASSOC]
Ax.1-08. (P&(QvR)) SYN ((P&Q)v(P&R))	[&v-DIST]

These are proved as theorems by SynSUB, from the axiom of Double Negation, the definition disjunction, Df 'v', and the intitial negation-free axioms 1-01, 1-03, 1-05, and 1-07, which are renumbered as Axioms 4-01, 4-02, 4-03 and 4-04. In addition many other new theorem-schemata involving negation can also be derived.

Thus the Axioms and Rules of Inference for Sentential Analytic Logic with the primitive connectives, '&', '∼', and '∀x' are:

IV. Axiom Schemata: [Drops: Ax.1-02, Ax.1-04, Ax.1-06, Ax.1-08; adds: Ax.4-05.

Ax.4-01. [P SYN (P&P)]	[&-IDEM1]	
Ax.4-02. [(P&Q) SYN (Q&P)]	[&-COMM]	
Ax.4-03. [(P&(Q&R)) SYN ((P&Q)&R)]	[&-ASSOC1]	
Ax.4-04. [(P&(QvR)) SYN ((P&Q)v(P&R))]	[&v-DIST1]	
Ax.4-05. [P SYN ∼∼P]	**[DN]**	**[Added]**

We retain just the three rules of inference, U-SUB, SynSUB, and INST, along with the principle of Mathematical Induction for quantified wffs and the Rule of Alphabetic Variance.

R3-0. If P and Q are alphabetic variants, |= [P] Syn |= [Q].

V. Transformation Rules

R4-1. From [P Syn Q] and [R Syn S],
it may be inferred that [P Syn Q(S//R)] **[SynSUB]**

This is the same as R1-1, except that the new formation rules allow wffs to include expressions with negation signs. It covers definitional Synonymy: From [P SYN Q] and [R Syn_{df} S] infer that [P SYN Q(S//R)]). As before, we can derive R4-1b. [If (P SYN Q) and (R SYN S), then (P SYN Q(**R**//**S**)). For simplicity, we will use 'R1' and 'R1b' instead of 'R4-1' and 'R4-1b' in proofs which follow.

R4-2. If [R SYN S] and (i) $P_i<t_1,\ldots,t_n>$ occurs in R,
and (ii) Q is an h-adic wff, where h $\geq$ n,
and (iii) Q has occurrences of all numerals 1 to n,
and (iv) no individual variable in Q occurs in R or S,
then it may be inferred that [R($P_i<t_1,\ldots,t_n>$/Q) SYN S($P_i<t_1,\ldots,t_n>$/Q)]. **[U-SUB]**

Though apparently the same as R3-2, in view of the new concept of wffs with negation, this has new consequences, to be discussed below after the discussion of R4-2.

R4-3. If $\models$ [P<1>] then $\models$ [Pa_i] **[INST]**

The new versions of SynSUB and INST are essentially no different than R2-1 and R2-3, except that they deal with a new class of well-formed formulae with occurrences of negated wffs.

The Rule of Uniform Substitution, R4-2 also appears the same as the rule R3-2, except that wffs now include expressions with negation signs. But the introduction of negation signs makes it possible for inconsistencies to be introduced by R4-2 into theorems which had no inconsistencies, thus to create theorems of a differnent logical kind. This rule preserves the well-formedness of wffs. It also will preserve Logical Synonymy and Logical Containment, thus facilitating the proofs of many SYN- and CONT-theorems with new components in which negation occurs. In Chapter 5 we shall see that it also preserves Logical Inconsistency and Logical Tautology as properties of certain wffs and statements, and in Chapter 7 it is shown to preserve assertions of factual Truth and Falsehood.

However, as it stands, R4-2 allows us to move from theorems which have no inconsistent components to theorems with one or more inconsistent components. We will later require that if a conditional or an argument is to be valid according to A-logic, the premiss and conclusion (antecedent and consequent) must be jointly consistent. It is therefore of interest to devise ways to preserve consistency in the wffs we use.

In this chapter—and later on in Chapters 6 and 8—we will make use of a restricted sub-rule of R4-2, which preserves whatever degree of consistency a wff may have by either first, permittng any **negation-free** predicate schema, Q, to be introduced at all occurrences of any predicate letter P that occurs only POS in a wff R in accordance with R4-2, **provided** Q does not contain any predicate letter that occurs NEG elsewhere in R, or second permitting the introduction of negation only through replacing predicate letters by their own denials. This rule, R4-2ab, (or "U-SUBab") might be called "Restricted U-SUB with respect to inconsistency-introduction", and is intended to preserve what ever degree of consistency exists in an expression R, thus to prevent the introduction of new inconsistent components into a theorem by means of U-SUB. It may be expressed formally as,

R4-2ab ("Restricted U-SUB" or "U-SUBab")

If [R SYN S]
and **either a)** (i) $P_i<t_1,\ldots,t_n>$ occurs only POS in R,
and (ii) Q is an h-adic negation-free wff, where h $\geq$ n,
and (iii) Q has occurrences of all numerals 1 to n,
and (iv) no predicate letter in Q occurs NEG in R or S.
and (v) no variable in Q occurs in R or S,
or b) (i) $P_i<t_1,\ldots,t_n>$ occurs in R and S,
(ii) Q is '$\sim P_i<t_1,\ldots,t_n>$'
then it may be inferred that [R($P_i<t_1,\ldots,t_n>$/Q) SYN S($P_i<t_1,\ldots,t_n>$/Q)]. **[U-SUBab]**

Alternatively, R4-2ab may be presented as two separate sub-rules of R4-2. Using the two rules separately will make it clearer that we are abiding by the restrictions implicit in R4-2ab.

R4-2a ("U-SUBa")

If [R SYN S] and (i) $P_i<t_1,\ldots,t_n>$ occurs only POS in R and S,
and (ii) Q is an h-adic **negation-free** wff, where h $\geq$ n,
and (iii) Q has occurrences of all numerals 1 to n,
and (iv) no predicate letter in Q occurs NEG in R or S,
and (v) no individual variable in Q occurs in R or S,
then it may be inferred that $[R(P_i<t_1,\ldots,t_n>/Q)$ SYN $S(P_i<t_1,\ldots,t_n>/Q)]$. **[U-SUBa]**

R4-2b ("U-SUBb")

If [R SYN S] and (i) $P_i<t_1,\ldots,t_n>$ occurs in R and S, (POS or NEG)
and (ii) Q is '$\sim P_i<t_1,\ldots,t_n>$'
then it may be inferred that $[R(P_i<t_1,\ldots,t_n>/Q)$ SYN $S(P_i<t_1,\ldots,t_n>/Q)]$. **[U-SUBb]**

Obviously Rules R4-2a and R4-2b are each special cases of the unrestricted U-SUB, i.e., R4-2. The decision to restrict the use of U-SUB to these two cases is an option open to us, and one which will be employed in this chapter and in Chapter 6 and 8, when we wish to avoid adding new inconsistent elements in theorems.

The result of confining the use of R4-2 to these two cases is that no predicate letter in R' (the result of applying R4-2a and/or R4-2b to R) will have both POS and NEG occurrences of a predicate letter unless there was already both a POS and a NEG occurrence of that letter in R. Logical inconsistencies and logical tautologies can be present in components of a wff only if some predicate letter occurs both POS and NEG.

If a proof begins with any SYN- or CONT-theorem in chapters 1 through 3 (all of which are negation-free), then all derivations which use only the restricted rule of substitution, R4-2ab, will produce SYN- and CONT-theorems in which no atomic wff occurs both POS and NEG. This means that no theorem so produced will contain any wff whose MODNF (maximum ordered disjunctifve normal form) contains any atomic wff which occurs both negated and unnegated in some disjunct. This will hold of all SYN- and CONT-theorems with quantifiers, and all Boolean expansions of wffs within such theorems. Let us call such theorems "**absolutely** consistent", meaning that none of their basic components are inconsistent. As long as we confine U-SUB to R4-2ab in this chapter, all resulting theorems will be absolutely consistent.

If any predicate letter occurs both POS and NEG in a wff, there will be at least one disjunct of its MODNF form which will be inconsistent, having both that predicate letter and its denial conjoined in the maximum disjunct. However, a wff may be logically consistent even though one or more of its components are inconsistent. For example, '((P&~P)vQ)' is a consistent wff, capable of being either true or false when 'Q' is replaced by a consistent statement. Such wffs may be called "Partially Consistent", and 'not-INC'. A logically **inconsistent** wff (an 'INC'wff) is one such that every disjunct of its MODNF contains both an atomic wff and its denial. U-SUBab preserves consistency even of partially consistent wffs like '((P&~P)vQ)'. Although consistency is required for validity in A-logic, **absolute** consistency is not required.

The range of possibilities is illustrated as follows:

(i) '(P&(PvQ))' is absolutely consistent; its MODNF is '(Pv(P&Q))'.
(ii) '(P&(Qv ~ Q))' is partly consistent; its MODNF is '((P&Q)v(P& ~ Q)v(P&**Q& ~ Q**))'.
(iii) '(Pv(Q& ~ Q))' is partly consistent; its MODNF is '(Pv(P&Q)v(P& ~ Q)v(**Q& ~ Q**)v(P&**Q& ~ Q**))'.
(iv) '(P&(Q& ~ Q))' is INC (logically inconsistent); in this case, its MODNF is '(P& **Q& ~ Q**)'
while its MOCNF is '(P & **Q & ~ Q** & (PvQ) & (Pv ~ Q) & (Qv ~ Q) & (PvQv ~ Q))'.

Note that '(P ⊃ Q)' (which is SYN to '(~ P v Q)') is absolutely consistent, and every application of U-SUBab to '(P ⊃ Q)' will be absolutely consistent. But (P ⊃ P) and the truth-functional biconditional '(P ≡ Q)' are <u>not</u> absolutely consistent. For [(P ⊃ P) SYN (~ PvPv(~ P&P))] and '(P ≡ Q)' is synonymous with '(~ PvQ) & (~ QvP)" which has the MODNF form,

((P& ~ P) v (P&Q) v (~ P& ~ Q) v (Q& ~ Q)
v (P& ~ P&Q) v (P& ~ P& ~ Q) v (P&Q& ~ Q) v (~ P&Q& ~ Q)
v (P& ~ P&Q& ~ Q)).

No application of U-SUB or U-SUBab to '(P ⊃ P)' or to a truth-functional biconditional can be absolutely consistent. Some of its MODNF's disjuncts will be inconsistent in every application of U-SUB or U-SUBab. By contrast, in Chapter 6 and 8 we see that '(P => P)' and '(P <=> Q)'—defined as '((P => Q) & Q => P))' with C-conditionals—<u>are</u> absolutely consistent, and U-SUBab can preserve that absolute consistency, though unr**e**stricted U-SUB does not.

If U-SUB is confined to R4-2ab, it will be possible to get every possible wff which is absolutely consistent; i.e., every wff in which each atomic wff occurs either POS or NEG but never both POS and NEG. In the basic normal forms, this class consists of every possible normal form wff which has all of its atomic wffs either occurring negated or unnegated, but none which occur both negated and unnegated.

The theorems in this chapter and named 'T4-…', will be only those derivable using the restricted rule, R4-2ab (or "U-SUBab"). Initially we use R4-2a and R4-2b separately to show how R4-2ab works at any given step. From the proof of T4-24 on, however, we will cite R4-2ab leaving it to the reader to see how it was applied in any given step, including cases in which several applications are telescoped into one. For example, If we have a step 5) in some proof which says,

R R'
"5) [(~ (~ Q v P) ⊃ ~ P) SYN ~ (~ (~ Q v P) & ~ ~ P)] [Df '⊃', U-SUBab]

this asserts that we can get from Df '⊃' to step 5) by a sequence which uses only U-SUBa and U-SUBb and the result will have no more inconsistent disjuncts in its DNF than Df '⊃'. This we can do by the following sequence of sub-steps (atomic wffs occurring POS are in **Bold**):

4a). (P ⊃ **Q**) SYN ~ (P & ~ **Q**)	[Df '⊃']
4b) (P ⊃ **S**) SYN ~ (P & ~ **S**)	[4a),U-SUBa 'S'for 'Q']
4c) (~ **P** ⊃ **S**) SYN ~ (~ **P** & ~ **S**)	[4b),U-SUBb ' ~ P'for 'P']
4d) (~ (**Q** v **R**) ⊃ **S**) SYN ~ (~ (**Q** v **R**) & ~ **S**)	[4c),U-SUBa '(QvR)'for 'P']
4e) (~ (~ **Q** v **P**) ⊃ **S**) SYN ~ (~ (~ **Q** v **P**) & ~ **S**)	[4d)U-SUB,a ' ~ Q'for 'Q']
4f) (~ (~ **Q** v **P**) ⊃ ~ **P**) SYN ~ (~ (~ **Q** v **P**) & ~ ~ **P**)	[4e)U-SUBb ' ~ P'for 'S]
5) (~ (~ Q v **P**) ⊃ ~ **P**) SYN ~ (~ (~ Q v **P**) & ~ ~ **P**)	[4f),U-SUBb ' ~ P'for 'P']

In proofs with quantificational wffs, R4-2a permits the introduction of any negation-free wff which satisfies its conditions while R4-2b only allows negation to be introduced as a prefix to single predicate letters. That all theorems listed in this chapter are absolutely consistent and are provable using only the restricted forms of U-SUB can be verified by seeing that none have a predicate letter occurring both POS and NEG either implicitly (with '⊃') or explicitly.

Returning to the Rule of Instantiation and rules which are derivable from it, "INST" or R4-3, "If |= [P<1>] then |= [Pa_i]", appears to be the same as R2-3, but since the Rule of U-SUB is changed to allow negation-introduction, R4-3 with U-SUB_b allows the derived rule,

"If |= [~P<1>] then |= [~Pa_i]". and the Derived Instantiation Rules of Chapter 3 still hold, but are augmented in similar fashion to allow:

DR4-3a. If |= [~P<1>] then |= [(∀x)~Px] 'CG' (Conjunctive Generalization) [R3-3a,R4-2b]
DR4-3b. If |= [~P<1>] then |= [(∃x)~Px] 'DG' (Disjunctive Generalization) [R3-3b,R4-2b]

Referential synonymy is preserved when both synonyms are negated. The next derived rule multiplies the number of previous SYN-theorems by two. It is made possible by U-SUBab.

DR4-1. If |= [P SYN Q] then |= [~P SYN ~Q]
<u>Proof</u>: 1) |= [P SYN Q] [Assumption]
2) |= [~P SYN ~P] [T1-11,R4-2b' ~P'for'P']
3) |= [Q SYN P] [1),DR1-01]
4) |= [~P SYN ~Q] [2),3),R1]
5) If |= [P SYN Q] then |= [~P SYN ~Q] [1) to 4),Cond. Proof]

This fails if 'CONT' is put in place of 'SYN', and thus certain transposition rules fail.[10] However, by DR4-1 and R4-2 the following are derivable:

DR4-3c If |= [P<1> Syn Q<1>] then |= [(∀x)~Px Syn (∀x)~Qx]
DR4-3d If |= [P<1> Syn Q<1>] then |= [(∃x)~Px Syn (∃x)~Qx]
DR4-3e If |= [P<1> Cont Q<1>] then |= [(∀x)~Px Cont (∀x)~Qx]
DR4-3f If |= [P<1> Cont Q<1>] then |= [(∃x)~Px Cont (∃x)~Qx]

The rules SynSUB and U-SUB and INST are necessary and sufficient to derive the pairs of schemata whose instances will be synonymous by virtue of their logical form and the logical constants of this language. All pairs which are SYN, are also truth-functionally equivalent.

4.3 Theorems of Analytic Sentential Logic with Negation and Conjunction

We next give proofs of some important SYN-theorems known as DeMorgan Equivalences which do not appear in negation-free logic, and after that proofs of the theorem-schemata which replace the axiom schemata, Ax.1-02, Ax.1-04, Ax.1-06 and Ax.1-08 of Negation-Free Analytic Sentential Logic. These

---

10. We can <u>not</u> derive |= [~B CONT ~A] from |= [A CONT B], or the converse. This would lead to violations of the rule that if |= [A CONT B], all elementary components of B must occur in A. Though |= [(A&B) CONT B), it is not the case that |= [~B CONT ~(A&B)] . However, see Section 7.4231, with Ti7- 80 [~TP Impl ~T(P&Q)]. The inference "(P is not true), therefore it is not true that (P & Q)" is valid based on the meaning of 'true' and the presuppositions of truth-logic.

proofs are the minimum needed to establish that the SYN-metatheorems established in the negation-free logic of '&' amd 'v' still obtain after the introduction of negation. Its theorem-schemata are 'SYN'-for-'≡' analogues or 'CONT'-for-'⊃'-analogues of the sub-set of theorem-schemata of standard sentential logic, which have only logical synonymy or containment relations between the components of the '⊃' or '≡'.

We start the numbering of SYN and CONT theorems added by the system with negation at 'T4-11'. The first five theorems of Chapter 1, T1-11 to T1-15 and their proofs, as well as derived rules DR1-01 and DR1-02, have the same proofs as appeared in negation-free Analytic &v-Logic.

T1-11. [P SYN P]
DR1-01. If [P SYN Q] then [Q SYN P].
DR1-02. If [P SYN Q] and [Q SYN R] then:[P SYN R]

To reduce to normal forms we can use the converses of D5 and D6:

[~(P&~Q) SYN (P ⊃ Q)]	[Df '⊃', DR1-01]
[(~(P&~Q)&~(Q&~P)) SYN (P ≡ Q)]	[Df '≡', DR1-01]

In addition the following "DeMorgan Equivalences" can be used to move negations signs from outside to inside a pair of parentheses.

T4-11. [(P&Q) SYN ~(~Pv~Q)]	[DeM1]	T4-15. [(~P&Q) SYN ~(Pv ~Q)]	[DeM5]
T4-12. [(PvQ) SYN ~(~P&~Q)] [Df 'v']	[DeM2]	T4-16. [(~PvQ) SYN ~(P& ~Q)]	[DeM6]
T4-13. [(P&~Q) SYN ~(~PvQ)]	[DeM3]	T4-17. [(~P&~Q) SYN ~(PvQ)]	[DeM7]
T4-14. [(Pv~Q) SYN ~(~P&Q)]	[DeM4]	T4-18. [(~Pv~Q) SYN ~(P&Q)]	[DeM8]

The proofs follow:

T4-11. [(P & Q) SYN ~(~P v ~Q)] [DeM1]

Proof: 1)	(P&Q) SYN ~~(P&Q)	[Ax.4-05, R4-2a,'(P&Q)'for'P']
2)	(P&Q) SYN ~~(~~P&~~Q)	[1),Ax.4-05,R1(twice)]
3)	(~Pv~Q) SYN ~(~~P&~~Q)	[Df'v', R4-2b,(twice)]
3)	(P&Q) SYN ~(~Pv~Q)	[2),3),R1]

T4-12. [(P v Q) SYN ~(~P & ~Q)] [DeM2] [Df 'v']

T4-13. [(P &~Q) SYN ~(~P v Q)] [DeM3]

Proof: 1)	(P&Q) SYN ~~(P&Q)	[Ax.4-05, R4-2a]
2)	(P&~Q) SYN ~~(P&~Q)	[1), R4-2b]
3)	(P&~Q) SYN ~~(~~P&~Q)	[2),Ax.4-05,R1]
4)	(~PvQ) SYN ~(~~P&~Q)	[Df 'v',R4-2b'~P'for'P']
5)	(P&~Q) SYN ~(~PvQ)	[3),4),R1]

T4-14. [(P v~Q) SYN ~(~P & Q)] [DeM4]

Proof: 1)	(P v Q) SYN ~(~P&~Q)	[Df 'v', R4-2a]
2)	(P v ~Q) SYN ~(~P&~~Q)	[1),R4-2b,'~Q'for'Q']
3)	(P v ~Q) SYN ~(~P& Q)	[2),Ax.4-05,R1]

T4-15. [(~P & Q) SYN ~(P v ~Q)] [DeM5]

Proof: 1) (P & Q) SYN ~~(P&Q) [Ax.4-05,R4-2a]
2) (~P & Q) SYN ~~(~P&Q) [1),R4-2b,'~P'for'P']
3) (~P & Q) SYN ~~(~P&~~Q) [2),Ax.4-05,R1b]
4) (Pv ~Q) SYN ~(~P&~~Q) [Df 'v',R4-2b]
5) (~P & Q) SYN ~(P v ~Q) [3),4),R1]

T4-16. [(~P v Q) SYN ~(P & ~Q)] [DeM6]

Proof: 1) (~PvQ) SYN ~(~~P&~ Q) [Df 'v', R4-2b,'~P'for'P']
2) (~PvQ) SYN ~(P&~Q) [2),Ax.4-05,R1]

T4-17. [(~P&~Q) SYN ~(PvQ)] [DeM7]

Proof: 1) (~P&Q) SYN ~(Pv~Q) [T4-15]
2) (~P&~Q) SYN ~(Pv~~Q) [1),R4-2b,'~Q'for'Q']
3) (~P&~Q) SYN ~(PvQ) [2),Ax.4-05,R1]

T4-18. [(~Pv ~Q) SYN ~(P&Q)] ["DeM8"]

Proof: 1) (Pv Q) SYN ~(~P&~Q) [Df 'v']
2) (~Pv Q) SYN ~(~~P&~Q) [1),R4-2b]
3) (~Pv ~Q) SYN ~(~~P&~~Q) [2),R4-2b]
4) (~Pv ~Q) SYN ~(P&Q) [3),Ax.4-05,R1(twice)]

The next four theorems re-establish Ax.1-02 (v-IDEM), Ax.1-04 (v-COMM), Ax.1-06(v-ASSOC) and Ax.1-08(&v-DIST) of Chapter 1, as theorems T4-19 to T4-22.

T4-19. [P SYN (PvP)] ["v-IDEM1",previously Ax.1-02]

Proof: 1) (PvP) SYN ~(~P&~P) [Df 'v',R4-2a 'P'for'Q']
2) ~P SYN (~P&~P) [Ax.4-01,R4-2b]
3) (PvP) SYN ~~P [1),2),R1]
4) (PvP) SYN P [3),Ax.4-05,R1]
5) P SYN (PvP) [4),DR1-01]

T4-20. [(PvQ) SYN (QvP)] ["v-COMM",previously Ax.1-04]

Proof: 1) (PvQ) SYN ~(~P&~Q) [Df 'v']
2) (~P&~Q) SYN (~Q & ~P) [Ax.4-02,R4-2b(twice)]
3) (PvQ) SYN ~(~Q&~P) [1),2),R1b]
4) (RvQ) SYN ~(~R&~Q) [3),R4-2a,'R'for'P']
5) (RvP) SYN ~(~R&~P) [4),R4-2a,'P'for'Q']
6) (QvP) SYN ~(~Q&~P) [5),R4-2a'Q'for'R']
7) (PvQ) SYN (QvP) [3),6),R1]

T4-21. [(Pv(QvR)) SYN ((PvQ)vR)] ["v-ASSOC1",previously Ax.1-06]

Proof: 1) (Pv(QvR)) SYN ~(~P&~(QvR)) [Df 'v',R4-2a '(QvR)'for'Q']
2) ~(PvR) SYN (~P&~R) [T4-17, R4-2a 'R'for'Q']
3) ~(QvR) SYN (~Q&~R) [3), R4-2a 'Q'for 'P']
4) (Pv(QvR)) SYN ~(~P&(~Q&~R)) [1),3),R1b]

5) (~P&(~Q&~R)) SYN ((~P&~Q)&~R) [Ax.4-03,U-SUBb]
6) (Pv(QvR)) SYN ~((~P&~Q)&~R) [4),5),R1b]
7) (Pv(QvR)) SYN ~(~~(~P&~Q)&~R) [6),Ax.4-05,R1b]
8) (Pv(QvR)) SYN ~(~(PvQ)&~R) [7),Df 'v',R1]
9) (Pv(QvR)) SYN ((PvQ)vR) [8),Df 'v',R1]

T4-22. [(P&(QvR)) SYN ((P&Q) v (P&R))] ["&v-DIST3",previously Ax.1-08]
Proof: 1) (P&(QvR)) SYN (P&(QvR)) [T1-11]
2) (P&(QvR)) SYN ~~(P&(QvR)) [1),Ax.4-05,R1b]
3) (P&(QvR)) SYN ~(~Pv ~(QvR)) [2),T4-18,R1]
4) (P&(QvR)) SYN ~(~Pv(~Q&-R)) [3),T4-17,R1]
5) (~Pv(~Q&-R)) SYN ~((~Pv ~Q)&(~Pv ~R)) [Ax.4-04,U-SUBb]
6) (P&(QvR)) SYN ~((~Pv ~Q)&(~Pv ~R)) [4),5),R1b]
7) (P&(QvR)) SYN ~(~(P&Q)&~(P&R)) [6),T4-18,R1b(twice)]
8) ((P&Q) v (P&R)) SYN ~(~(P&Q)&~(P&R)) [Df 'v',U-SUBa]
9) (P&(QvR)) SYN ((P&Q) v (P&R)) [7),8),R1]

Having re-established these theorems, we can use the same proofs we used in Chapter 1 to get T1-11 to T1-38. The proofs differ only in the justifications to the right of each step,

'Ax.4-01' replaces 'Ax.1-01' 'T4-19' replaces 'Ax.1-02'
'Ax.4-02' replaces 'Ax.1-03' 'T4-20' replaces 'Ax.1-04'
'Ax.4-03' replaces 'Ax.1-05' 'T4-21' replaces 'Ax.1-06'
'Ax.4-04' replaces 'Ax.1-07' 'T4-22' replaces 'Ax.1-08'

Before leaving the field of unquantified theorems, we mention one which leads to a special problem for A-logic.

T4-23. [(P&~Q) CONT ~(Q&~P)]
Proof: 1) (P&Q) CONT (PvQ) [T1-38].
2) (P&~Q) CONT (Pv~Q) [1),U-SUBb,'~Q'for'Q']
3) (P&~Q) CONT ~(~P&~~Q [2),T4-12,R1]
4) (P&~Q) CONT ~(~P&Q) [3),Ax.4-05]

Using unrestricted U-SUB, we can derive '(P&~P) CONT ~(P & ~P)' by R4-2 putting '~P' for 'Q'. This says a contradiction contains its own denial and shows one consequence of introducing inconsistency into CONT theorems and SYN theorems using unrestricted U-SUB. These are true SYN or CONT-theorems, but they differ in kind from theorems which are free of inconsistency.

Thus we have added 16 new theorems to the set derivable in the negation-free &/v language of Chapter 1. Four take the place of axiom schemata Ax.1-02, Ax.1-04, Ax.1-06 and Ax.1-08 and 12 are new theorems in which negation occurs. A net total of 44 theorems plus the five axiom schemata and three definitions are sufficient to establish SYN-metatheorems 1 to 13 either for all absolutely consistent wffs gotten by restricted U-SUBab, or (dropping restrictions on R4-2) for all wffs in the language of Standard Logic.

4.4 Theorems of Analytic Quantification Theory with Negation and Conjunction

The formation rule which admits negated expressions as well-formed formulae, covers negations of quantified wffs as well as negations of atomic, elementary and compound wffs. This rule makes possible the Laws of Quantifer Interchange. DR4-1 allows the move from [P SYN Q] to [~Q SYN ~P]. With this the Laws of Quantifier Interchange make every logical synonymy derivable from its dual. This cuts the set of basic SYN-theorems of Section 3.2 in half, since the dual of each SYN-Theorem is derivable through definitions D5-D8 with Ax.4-05 (Double Negation) and DR4-1.

On the other hand, the duals of CONT theorems are not necessarily theorems; CONT- theorems are derivable only from a dual which has a conjunction (rather than a disjunction) as one component—e.g., from (P SYN P&Q) rather than from (P SYN (PvQ)). Having derived each of what were Ax.1-02, Ax.1-04, Ax.1-06 and Ax.1-08, in Chapter 1, we need derive only one member of each dual; the other member will then be derivable from the first without further reference to negation using the same proofs as in Chapter 1 (with new theorem names).

4.41 Laws of Quantifier Interchange

The principles customarily known as laws of quantifier interchange follow from the definition of '(∃x)' and double negation, Ax.4-05, and their converses. (Where applicable Quine's numbers for metatheorems which are '≡' for 'SYN' analogues, or '⊃' for 'CONT' analogues, are given on the right.)

In Section 19 of Quine's *Mathematical Logic*, entitled Existential Quantification, are listed the definition of '(∃x)', D8. (∃x) for ~(∀x)~ [Df '(∃x)'] and four laws of quantifier interchange:

*130. $\vdash [\sim(\forall x)\phi \equiv (\exists x)\sim\phi]$
*131. $\vdash [\sim(\exists x)\phi \equiv (\forall x)\sim\phi]$
*132. $\vdash [\sim(\forall x_1)\ldots(\forall x_n)\phi \equiv (\exists x_1)\ldots(\exists x_n)\sim\phi]$
*133. $\vdash [\sim(\exists x_1)\ldots(\exists a_n)\phi \equiv (\forall x_1)\ldots(\forall x_n)\sim\phi]$

The 'SYN' for '≡' analogues of these in A-logic are the converses of:

T4-24. $\models [(\exists x)\sim Px$ Syn $\sim(\forall x)Px]$ [cf.*130]
T4-25. $\models [(\forall x)\sim Px$ Syn $\sim(\exists x)Px]$ [cf.*131]
T4-26. $\models [(\exists x_1)\ldots(\exists x_n)\sim P\langle x_1,\ldots,x_n\rangle$ Syn $\sim(\forall x_1)\ldots(\forall x_n)P\langle x_1,\ldots,x_n\rangle]$ [cf.*132]
T4-27. $\models [(\forall x_1)\ldots(\forall x_n)\sim P\langle x_1,\ldots,x_n\rangle$ Syn $\sim(\exists x_1)\ldots(\exists x_n)P\langle x_1,\ldots,x_n\rangle]$ [cf.*133]

We use T4-24 to T4-27, rather than their converses, because with the present form of SynSUB, R1, they are more useful in reducing Q-wffs to prenex and intranex normal forms by moving quantifiers out from the scope of negation signs.

T4-24. [(∃x)~Px SYN ~(∀x)Px]
<u>Proof</u>: 1) [(∃x)Px SYN ~(∀x)~Px] [Df '(∃x)']
2) [(∃x)~Px SYN ~(∀x)~~Px] [1),U-SUBb]
3) [(∃x)~Px SYN ~(∀x)Px] [2),Ax.4-05,R1]

T4-25. $[(\forall x){\sim}P]$ SYN $\sim(\exists x)P]$

Proof: 1) $[(\exists x)Px]$ SYN $\sim(\forall x){\sim}Px]$ [D8]
2) $[\sim(\exists x)Px]$ SYN $\sim\sim(\forall x){\sim}Px]$ [1),DR4-1]
3) $[\sim(\exists x)Px]$ SYN $(\forall x){\sim}Px]$ [2),Ax.4-05,R1]
4) $[(\forall x){\sim}Px]$ SYN $\sim(\exists x)Px]$ [3),DR1-01]

T4-26. $[(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $\sim(\forall x_1)\ldots(\forall x_n)P\langle x_1,\ldots,x_n\rangle]$

Proof: 1) $[(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle]$ [T1-11]
2) $[(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\exists x_1)\ldots(\exists x_{n-1}){\sim}(\forall x_n)P\langle x_1,\ldots,x_n\rangle]$ [1),T4-24,R1b]
3) $[(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\exists x_1)\ldots{\sim}(\forall x_{n-1})(\forall x_n)P\langle x_1,\ldots,x_n\rangle]$ [2),T4-24,R1b]
....
n) $[(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\exists x_1){\sim}(\forall x_2)\ldots(\forall x_{n-1})(\forall x_n)P\langle x_1,\ldots,x_n\rangle]$ [n-1),T4-24,R1b]
n+1) $[(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $\sim(\forall x_1)(\forall x_2)\ldots(\forall x_{n-1})(\forall x_n)P\langle x_1,\ldots,x_n\rangle]$ [n),T4-24,R1]

T4-27. $[(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $\sim(\exists x_1)\ldots(\exists x_n)P\langle x_1,\ldots,x_n\rangle]$

Proof: 1) $[(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle]$ [T1-11]
2) $[(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\forall x_1)\ldots(\forall x_{n-1}){\sim}(\exists x_n)P\langle x_1,\ldots,x_n\rangle]$ [1),T4-25,R1b]
3) $[(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\forall x_1)\ldots{\sim}(\exists x_{n-1})(\exists x_n)P\langle x_1,\ldots,x_n\rangle]$ [2),T4-25,R1b]
....
n)$[(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $(\forall x_1){\sim}(\exists x_2)\ldots(\exists x_{n-1})(\exists x_n)P\langle x_1,\ldots,x_n\rangle]$ [n-1),T4-25,R1b]
n+1) $[(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle$ SYN $\sim(\exists x_1)(\exists x_2)\ldots(\exists x_{n-1})(\exists x_n)P\langle x_1,\ldots,x_n\rangle]$ [n),T4-24,R1]

If it is desired to prefix single negation signs to quantifications which are initially unnegated, it is easy to do so by a step using one of the following theorems with R1.

$[\sim(\forall x){\sim}Px$ SYN $(\exists x)Px]$ [the converse of Df'‘$(\exists x)$’] [Df ‘$(\exists x)$’,DR1-01]

$[\sim(\exists x){\sim}Px$ SYN $(\forall x)Px]$

Proof: 1) $[(\exists x){\sim}Px$ SYN $\sim(\forall x)Px]$ [T4-24]
2) $[\sim(\exists x){\sim}Px$ SYN $\sim\sim(\forall x)Px]$ [1),DR4-1]
3) $[\sim(\exists x){\sim}Px$ SYN $(\forall x)Px]$ [2),Ax.4-05,R1]

$[\sim(\forall x)Px$ SYN $(\exists x){\sim}Px]$ ML*130] [T4-24,DR1-01]
$[\sim(\exists x)Px$ SYN $(\forall x){\sim}Px]$ ML*131] [T4-25,DR1-01]
$[\sim(\forall x_1)\ldots(\forall x_n)P\langle x_1,\ldots,x_n\rangle$ SYN $(\exists x_1)\ldots(\exists x_n){\sim}P\langle x_1,\ldots,x_n]$ ML*132] [T4-26,DR1-01]
$[\sim(\exists x_1)\ldots(\exists a_n)P\langle x_1,\ldots,x_n\rangle$ SYN $(\forall x_1)\ldots(\forall x_n){\sim}P\langle x_1,\ldots,x_n\rangle]$ ML*133] [T4-27,DR1-01]

With the introduction of negation, we are able to derive the duals of any odd-numbered theorem from T3-13 to T3-31 using the new axioms, definitions and rules. Thus of the ten “basic” theorems T3-19 to T3-28, we may take only the odd-numbered theorems as “basic” theorems” reducing the number of theorems in this category to five.

Finally,T4-28 is Theorem *3.47 in Russell and Whitehead’s *Principia Mathematica*. There it is said that an analogue for classes of this theorem was proven by Leibniz, “and evidently pleased him,since he calls it “Praeclarum Theorema””.

T4-28) ((P ⊃ R) & (Q ⊃ S)) CONT ((P&Q) ⊃ (R&S)) Leibniz' "Praeclarum Theorema"
Proof: 1) ((P v T) & (R v S)) CONT ((PvR)v(T&S))) [T1-40, U-SUBa,'T'for Q']
2) ((P v T) & (Q v S)) CONT ((PvQ)v(T&S))) [1),U-SUBa,'Q'for'R']
3) ((P v R) & (Q v S)) CONT ((PvQ)v(R&S))) [2),U-SUBa,'R'for'T']
4) ((~PvR) & (~Q v S)) CONT ((~Pv ~Q)v(R&S)) [3),U-SUBb,'~P'for'P','~Q'for'Q']
5) ((~PvR) & (~Q v S)) CONT (~ (P&Q)v(R&S)) [4),T4-18, R1b]
6) ((P ⊃ R) & (Q ⊃ S)) CONT ((P&Q) ⊃ (R&S)) [5),Df '⊃'(thrice)]

This theorem plays an important part in establishing quantificational axioms of M-logic. (See T3-29, T3-39, and T4-37.)

4.42 Quantificational Re-ordering

In Chapter 3 the odd-numbered quantificational theorems from T3-13 to T3-19, are derivable by mathematical induction from the ordering Axioms 1-01, 1-03 and 1-05 (&-IDEM, &-COMM and &-ASSOC) alone. In this chapter these same axioms are re-named Ax.4-01, Ax.4-02, and Ax.4-03. The quantificational even-numbered theorems, T3-14, T3-16 and T3-18 are now provable using the rules, definitions and the smaller set of axioms of Chapter 4.

The Axioms 1-02, 1-04, 1-06, and 1-08 have been re-established as the theorems,

T4-19. [P SYN (PvP)]	[v-IDEM]	[Same as Ax.1-02]
T4-20. [(PvQ) SYN (QvP)]	[v-COMM]	[Same as Ax.1-04]
T4-21. [(Pv(QvR)) SYN ((PvQ)vR)]	[v-ASSOC]	[Same as Ax.1-06]
T4-22. [(P&(QvR)) SYN ((P&Q) v (P&R))]	[&v-DIST]	[Same as Ax.1-08]

Therefore the proofs of T3-13.[(∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx)]
T3-15.[(∀x)(∀y)Rxy SYN (∀y)(∀x)Rxy]
T3-17.[(∀x)(P&Qx) SYN (P&(∀x)Qx)]
and T3-19.[(∃x)(P & Qx) SYN (P & (∃x)Qx)],
which were proved from Axioms 1-02,1-04, 1-06 and 1-08 respectively in Chapter 3, are now re-proved in precisely the same way, except that the axioms cited in the proofs in Chapter 3 are re-named and cited as the theorems T4-19,T4-20, T4-21 and T4-22 of Chapter 4. Thus we can use T3-13, T3-15, T3-17 and T3-19 within this chapter to prove their duals, T3-14, T3-16, T3-18 and T3-20, with the help of the new rules, axiom and definitions in Chapter 4:

Thus from T3-13, the analogue of ML*140, we derive its dual T3-14,analogue of ML*141.

T3-14. [(∃x)(Px v Qx) SYN ((∃x)Px v (∃x)Qx)] ML*141
Proof: 1) (∀x)(~Px & ~Qx) SYN ((∀x)~Px & (∀x)~Qx) [T3-15,U-SUBb,'~P'for'P','~Q'for'Q']
2) ~(∀x)(~Px & ~Qx) SYN ~((∀x)~Px & (∀x)~Qx) [1),DR4-1]
3) ~(∀x)(~Px & ~Qx) SYN (~(∀x)~Px v ~(∀x)~Qx) [2),T4-18,R1]
4) ~(∀x)(~Px & ~Qx) SYN ((∃x)Px v (∃x)Qx) [3),Df '(∃x)',R1(twice)]
5) ((∃x)Px v (∃x)Qx) SYN ~(∀x)(~Px & ~Qx) [4),DR1-01]
6) (~P<1> &~Q)<1>) SYN ~(P<1> v Q<1>) [T4-17]
7) (∀x)(~Px &~Qx) SYN (∀x)~(Px v Qx) [6), DR3-3c]
8) ((∃x)Px v (∃x)Qx) SYN ~(∀x)~(Px v Qx) [5),7),R1b]

9) ((∃x)Px v (∃x)Qx) SYN (∃x)(Px v Qx) [6),Df '(∃x)',R1]
10) (∃x)(Px v Qx) SYN ((∃x)Px v (∃x)Qx) [7),DR1-01]

In a similar manner, 1) from T3-15, the 'SYN'-for-'≡' analogue to Quine's *119, we derive the 'SYN'-for-'≡' analogue of Quine's *138, T3-16, [(∃x)(∃y)Pxy SYN (∃y)(∃x)Pxy], and

2) From T3-17, the first "Rule of Passage" and 'SYN'-for-'≡' analogue of Quine's *157, we derive the 'SYN'-for-'≡' analogue of Quine's *160, T3-18. [(∃x)(P v Qx) SYN (P v (∃x)Qx)].

T3-16 and T3-18 are re-ordering principles, as we saw in Chapter 3. T3-20 is proven in the next section since it is based on distributive principles.

4.43 Quantificational Distribution

Distribution principles are not changed by negation. But negation makes possible the proof of half of the distribution principles, from their duals, by means of D5, DeMorgan Theorems and Quantifier Interchange. It also vastly increases the number of distinct distribution principles by dealing with symbols with prefixed negation signs. These theorems may be seen as intermediate steps of the proofs below.

From the third "Rule of Passage" (T3-19) which is the 'SYN'-for-'≡' analogue of ML*158, we derive the 'SYN'-for-'≡' analogue of ML *159, T3-20:

T3-20 (∀x)(P v Qx) SYN (P v (∀x)Qx)) ML*159

<u>Proof</u>: 1) (∃x)(~P&~Qx) SYN (~P&(∃x)~Qx) [T3-19,U-SUBb,'~P'for'P','~Q<1>'for'Q<1>']
2) ~(∃x)(~P&~Qx) SYN ~(~P&(∃x)~Qx) [1),DR4-1]
3) ~(∃x)(~P&~Qx) SYN (~~P v ~(∃x)~Qx) [2),T4-17,R1]
4) ~(∃x)(~P&~Qx) SYN (~~P v (∀x)Qx) [3),D6,R1]
5) ~(∃x)(~P&~Qx) SYN (P v (∀x)Qx) [4),Ax.4-05,R1]
6) (P v (∀x)Qx) SYN ~(∃x)(~P & ~Qx) [5),DR1-01]
7) (P v (∀x)Qx) SYN ~(∃x)~(P v Qx) [6),T4-18,R1b]
8) (P v (∀x)Qx) SYN (∀x)(P v Qx) [7),D6,R1]
9) (∀x)(P v Qx) SYN (P v (∀x)Qx) [8),DR1-01]

In a similar way, from T3-21 we derive T3-22, [(∃x)Px SYN ((∃x)Px v (∀x)Px)];
from T3-23 we derive T3-24, [(∀x)(Px v Qx) SYN ((∀x)(Px v Qx) v (∀x)Px)];
from T3-25 we derive T3-26, [((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)(Px v Qx))];
and from T3-27 we derive T3-28, [(∀y)(∃x)Pxy SYN ((∀x)(∃y)Pxy v (∃y)(∀x)Pxy))].

Thus the only "basic theorems" needed are T3-13, T3-15, T3-17, T3-19,T3-21, T3-23, T3-25 and T3-27. The first three of these are based simply on re-ordering; The last five are sufficient grounds for making significant structural changes.

4.44 Containment Theorems with Truth-functional Conditionals

In addition to the laws of interchange, ML includes metatheorems in which connectives in the components are truth-functional conditionals. Since '⊃' and '≡' were not definable in the first negation-free three chapters,'SYN'-for-'≡' analogues and 'CONT'-for-'⊃' analogues expressions with truth-functional conditionals did not occur. With the introduction of negation, they are now expressible and derivable using the definition of '⊃', D6, along with D5-8, Ax.4-05 (Double negation) and DR4-1. These analogues are:

T4-30. [(P ⊃ Q) SYN ~(P&~Q)] [D6, Df.'⊃']
T4-31. [(~PvQ) SYN (P ⊃ Q)]
T4-32. [(P ⊃ Q) SYN (~Q ⊃ ~P)]
T4-33. [(∃x)(Px ⊃ Qx) SYN ((∀x)Px ⊃ (∃x)Qx)] ML *142
T4-34. [(∃x)(Px ⊃ Q) SYN ((∀x)Px ⊃ Q)] ML*162
T4-35. [(∀x)(Px ⊃ Q) SYN ((∃x)Px ⊃ Q)] ML*161
T4-36. [((∃x)Px ⊃ (∀x)Qx) CONT (∀x)(Px ⊃ Qx)] ML*148
T4-37. [(∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)] ML*101
T4-38. [(∀x)(P ⊃ Qx) SYN (P ⊃ (∀x)Qx))) .Cf. ML*159
T4-39. [(∀x)(Px ⊃ Qx) CONT ((∃x)Px ⊃ (∃x)Qx)] ML*149
T4-40. [((∃x)Px ⊃ (∃x)Qx) CONT (∃x)(Px ⊃ Qx)] ML*150
T4-41. [((∀x)Px ⊃ (∀x)Qx) CONT (∃x)(Px ⊃ Qx)] ML*151

The asterisked number on the right is that of the ML-metatheorem of which it is an analogue. Proofs of these 'CONT'-for-'⊃' analogues are given below.

T4-30. [(P ⊃ Q) SYN ~(P&~Q)] [Df '⊃']

T4-31. [(~PvQ) SYN (P ⊃ Q)]
Proof: 1) (~PvQ) SYN ~(P&~Q) [T4-16]
2) (~PvQ) SYN (P ⊃ Q) [1),Df '⊃']

T4-32. [(P ⊃ Q) SYN (~Q ⊃ ~P)]
Proof: 1) (P ⊃ Q) SYN ~(P&~Q) [Df '⊃']
2) (P ⊃ Q) SYN ~(~Q&P) [1),&-COMM]
3) (P ⊃ Q) SYN ~(~Q&~~P) [2),DN]
4) (P ⊃ Q) SYN (~Q ⊃ ~P) [3),Df '⊃']

T4-33. [(∃x)(Px ⊃ Qx) SYN ((∀x)Px ⊃ (∃x)Qx)] ML*142
Proof: 1) (∃x)(Px v Qx) SYN ((∃x)Px v (∃x)Qx) [T3-14]
2) (∃x)(~Px v Qx) SYN ((∃x)~Px v (∃x)Qx) [1),U-SUBb]
3) (∃x)(~Px v Qx) SYN (~(∀x)Px v (∃x)Qx) [2),T4-24(DR1-01),R1]
4) (∃x)(Px ⊃ Qx) SYN ((∀x)Px ⊃ (∃x)Qx) [3),T4-31]

From the theorems expressing "Rules of Passage" in Chapter 3,

T3-17. [(∀x)(P & Qx) SYN (P & (∀x)Qx)] "Rules of Passage" ML*157
T3-18. [(∃x)(P v Qx) SYN (P v (∃x)Qx)] ML*160
T3-19. [(∃x)(P & Qx) SYN (P & (∃x)Qx)] ML*158
T3-20. [(∀x)(P v Qx) SYN (P v (∀x)Qx)] ML*159

we derive two more "Rules of Passage" with truth-functional conditionals:

T4-34. [(∃x)(Px ⊃ Q) SYN ((∀x)Px ⊃ Q)] ML*162
Proof: 1) [(∃x)(~~Q v ~Px) SYN (~~Q v(∃x)~Px)] [T3-18,U-SUBab]
2) [(∃x)(~Px v ~~Q) SYN ((∃x)~Px v ~~Q)] [1),v-COMM,R1(twice)]
3) [(∃x)(~Px v Q) SYN ((∃x)~Px v Q)] [2),DN,R1(twice)]

4) [(∃x)(~Px v Q) SYN (~(∀x)Px v Q)] [3),Q-Exch]
5) [(∃x)(Px ⊃ Q) SYN ((∀x)Px ⊃ Q)] [4),T4-31,(twice)]

T4-35. [(∀x)(Px ⊃ Q) SYN ((∃x)Px ⊃ Q)] ML*161
<u>Proof:</u> 1) [(∀x)(PvQx) SYN (Pv(∀x)Qx)] [T3-20]
2) [(∀x)(~~Q v ~Px) SYN (~~Q v (∀x)~Px)] [1),U-SUBab]
2) [(∀x)(~Px v ~~Q) SYN ((∀x)~Px v ~~Q)] [1),T4-20,SynSUB]
3) [(∀x)(~Px v Q) SYN ((∀x)~Px v Q)] [2),DN,SynSUB]
4) [(∀x)(~Px v Q) SYN (~(∃x)Px v Q) [5),Q-Exch,SynSUB]
5) [(∀x)(Px ⊃ Q) SYN ((∃x)Px ⊃ Q) [6),T4-31(twice)]

T4-36. [((∃x)Px ⊃ (∀x)Qx) CONT (∀x)(Px ⊃ Qx)] ML*148
<u>Proof</u>: 1) ((∀x)Px v (∀x)Qx) CONT (∀x)(Px v Qx) [T3-38]
2) ((∀x)~Px v (∀x)Qx) CONT (∀x)(~Px v Qx) [1),U-SUBb]
3) (~(∃x)Px v (∀x)Qx) SYN ((∀x)~Px v (∀x)Qx) [T1-11,T4-25,R1]
4) (~(∃x)Px v (∀x)Qx) CONT (∀x)(~Px v Qx) [3),2),R1b]
5) ((∃x)Px ⊃ (∀x)Qx) CONT (∀x)(Px ⊃ Qx) [4),T4-31]

The 'CONT'-for-'⊃' analogue of Quine's axiom-Schema ML*101, which is

⊢ [(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)], is T4-37, [(∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)].

The latter follows easily from the CONT-theorem T3-39:

T4-37. [(∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)] ML*101
<u>Proof:</u> 1) (∀x)(Px v Qx) CONT ((∃x) Px v (∀x)Qx) [T3-39]
2) (∀x)(~Px v Qx) CONT ((∃x)~Px v (∀x)Qx) [1),U-SUBa]
3) (∀x)(~Px v Qx) CONT (~(∀x)Px v (∀x)Qx) [2),T4-24,R1b]
4) (∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)) [4),Df'‘⊃']

The 'CONT'-for-'⊃' analogue of another "Rule of Passage" often used as an axiom of quantification theory, and closely related to Quine's ML*159, is derived from T3-20 and T4-30:

T4-38. [(∀x)(P ⊃ Qx) SYN (P ⊃ (∀x)Qx)] .Cf. ML*159
<u>Proof:</u> 1) (∀x)(P v Qx) SYN (P v (∀x)Qx)) [T3-20]
2) (∀x)(~P v Qx) SYN (~P v (∀x)Qx)) [1),U-SUBb'~P'for'P']
3) (∀x)(P ⊃ Qx) SYN (P ⊃ (∀x)Qx)) [2),T4-30,R1b (twice)]

The 'CONT'-for-'⊃' analogue of Quine's ML*149 is derivable from T3-40:

T4-39. (∀x)(Px ⊃ Qx) CONT ((∃x)Px ⊃ (∃x)Qx)) ML*149
<u>Proof</u>: 1) (∀x)(~Px v Qx) CONT ((∀x)~Px v (∃x)Qx)) [T3-40,U-SUBb]
2) (∀x)(~Px v Qx) CONT (~(∃x)Px v (∃x)Qx)) [1),T4-24,R1b]
3) (∀x)(Px ⊃ Qx) CONT ((∃x)Px ⊃ (∃x)Qx)) [2),T4-31,R1]

Using T4-24 or T4-25 and Df '⊃', we derive from T3-41, the 'CONT'-for-'⊃' analogue of *150, which is T4-40; and from T3-42, the 'CONT'-for-'⊃' analogue of *151, which is T4-41:

T4-40. [((∃x)Px ⊃ (∃x)Qx) CONT (∃x)(Px ⊃ Qx)] ML*150
<u>Proof</u>: 1) ((∀x) ~ Px v (∃x)Qx) CONT (∃x)(~ Px v Qx) [T3-41,U-SUBb]
2) (~ (∃x)Px v (∃x)Qx)) CONT (∀x)(~ Px v Qx) [1),T4-25,R1b]
3) ((∃x)Px ⊃ (∃x)Qx)) CONT (∀x)(Px ⊃ Qx) [2),T4-31,R1]

T4-41. ((∀x)Px ⊃ (∀x)Qx) CONT (∃x)(Px ⊃ Qx) ML*151
<u>Proof</u>: 1) ((∃x) ~ Px v (∀x)Qx) CONT (∃x)(~ Px v Qx) [T3-42,U-SUBb]
2) (~ (∀x)Px v (∃x)Qx)) CONT (∃x)(~ Px v Qx) [1),T4-24,R1b]
3) ((∀x)Px ⊃ (∃x)Qx)) CONT (∃x)(Px ⊃ Qx) [2),T4-31,R1]

At this point, we have produced AL-analogues of 37 of the Quine's 56 metatheorems in Quantification theory, including the metatheorems, *170 and *171 which are covered by Alphabetic Variance theorems.[11]

This leaves 19 ML metatheorems to be accounted for.[12] Thus 37 of Quine's 56 metatheorems have 'SYN'for'≡', or 'CONT'-for-'⊃', analogues in the Analytic logic of synonymy and containment among wffs built up from conjunction, denial, and quantifiers in Chapters 1 through 4 using U-SUBab.

Five of the 19 of Quine's metatheorems which do not have analogues will have no place in analytic logic since they sanction vacuous quantifiers. The analogue of *116, requires unrestricted U-SUB and is proved in Chapter 5. Others either depend on definitions of inconsistency and tautology introduced in the next chapter, Chapter 5, or are shown in that chapter to be unnecessary for the soundness or completeness of the fragment of analytic logic which is equivalent to mathematical logic.

4.5 Soundness, Completeness, and Decision Procedures

4.51 Re: A Decision Procedure for [A CONT C] with M-logic's Wffs

Is there a general decision procedure such that given any two well-formed formulae, A and C, as defined in this chapter, we can find out in specifiable finite number of steps that [A CONT C] is true, or else, that it is false?

---

11. Namely, *101[T4-37], *119[T3-15], *130[T4-24], *131[T4-25], *132[T4-26], *133[T4-27], *136[T3-36], *138[T3-16], *139[T3-37], *140[T3-13], *141[T3-14], *142[T4-33], *143[T3-38], *144[T3-39], *145[T3-40], *146[T3-41], *147[T3-42], *148[T4-36], *149[T4-39], *150[T4-40], *151[T4-41], *152[T3-43], *153[T3-44], *154[T3-45], *155[T3-46], *156[T3-47], *157[T3-17], *158[T3-19], *159[T3-20], *160[T3-18], *161[T4-35], *162[T4-34] and *170,*171 [Alpha.Var.] and Rules *112, *113 and *123. This list plus *116 is spelled out in detail in Ch. 5, Section 5.3431 .

12. They are: *116, :the 7 Rules of Inference: *100,*104,*111,*115,*117,*124,*163,
8 Conditioned Theorem Schemata: *103, *134, *102,*118,*137, and Substitution Rules *120,*121,*122.
and 3 Theorem Schemata: *110,*114,*135,

We will leave the question open after explaining what it is and suggesting the possibility of steps towards a solution. Leaving it open requires argument because Church's theorem asserts there is no such decision procedure for theoremhood of '⊃'-for-'CONT' analogues, [A ⊃ C], in M-logic.

Whenever a predicate, such as '<1>is a theorem', is introduced to distinguish certain kinds of entities from others in the same class, it is appropriate to ask for a procedure by which one can determine whether that predicate applies or does not apply. We call such a procedure a *general decision procedure* if, for any given entity (or n-tuple of entities) in the class, it will guarantee a definite yes, or else a definite no, after a specifiable finite number of steps. We will call such a procedure *a partial decision procedure* if it provides such a procedure for getting a yes but not a no, or a no but not a yes.

A formal axiomatic system, S, like a good definition, provides at least a partial decision procedure for the predicate '<1>is a theorem of S'. The theorems in the axiomatic system of this chapter are expressions of the form '(A SYN B)' or '(A CONT B)' where A and B are wffs. Thus a decision procedure for this system must be a procedure which will decide in a specifiable finite number of steps that any expression in one of these two forms is, or is not, a theorem, i.e., is derivable from the axioms by the rules, or is not derivable.

Since all SYN-theorems are also CONT-theorems (but not the reverse), and ((A CONT B) & (B CONT A)) is provable if and only if (A SYN B), we may limit the question to that of a decision procedure for [A CONT B]. The axiom system of this chapter gives a way to decide that [A CONT B] has been derived (hence is derivable), but no way to decide that it is not derivable.

Any formal axiomatic system may be viewed as simply a set of procedures for manipulating symbols. Viewing it this way is the best way to test it for rigor. Viewed this way, the "definition of subject matter" defines only a class of symbols to which attention will be confined. The concept of a "proof" from axioms by rules, is replaced by that of the predicate '<1>is a deriv**ed** wff in S', and the concept of a theorem, is replaced by that of the predicate '<1>is a deriv**able** wff of S'. The two parts of the system constitute a full decision procedure for '<1>is a wff of S', and (superimposed on that) a partial decision procedure for the predicate—'<1>is a wff & <1>is derivable in S'.

Thus viewing the axiomatic system as purely a set of rules for writing down symbols, it consists of the two parts:

First, rigid rules which define a kind of symbol (wffs) which 1) we can produce at will, and which 2) provide a procedure for deciding whether any string of marks in front of us is, or is not, a symbol of that kind (a wff).

Second, a method which distinguishes certain wffs (let us call them 'X-wffs' rather than 'theorems') from other wffs. This method consists in (i) listing one or more wffs as X-wffs; these may be called the "initial X-wffs", and (ii) provides rules whereby one can start with an initial X-wff and perform operations on it that produce other wffs which, are also, by definition, to be called X-wffs, as well as the results of applying the rules to these results; any such possible result may be called "a derivable X-wff" to distinguish it from the "initial" X-wffs.

Viewing the system this way allows us to think of the symbols called theorems solely in terms of what could be seen on the written page:

1) Definition of symbols which constitute the subject-matter in the System ("wffs"):
   1a) the simplest signs to be used; none have parts; some are wffs. ["Primitives"]
   1b) the rules for forming certain compound signs "wffs" by ordered (left to right) and/or (grouped) arrangements of simple signs. ["Formation Rules"]
   1c) abbreviations: alternative symbols for certain compound wffs. ["Definitions"]

2) A partial definition of a sub-Species of wffs,'X-wffs',in system S:.
   2a) One or more initial X-wffs which are members of the subject-matter domain and belong to the sub-Set at issue, ["Axioms"]
   2b) Rules for forming new X-wffs by operations on the axioms, and/or by applying these rules to X-wffs previously derived from the axioms by the rules. ["Rules of Inference"]

Clauses 1a), 1b) and 1c) yield a rigorous, full decision procedure for the predicate which covers the subject terms ["<1>is a wff"]. Clauses 2)a and 2)b on top of 1a),1b), and 1c), constitute full decision procedure for "<1>is a derivation of an X-wff" (in place of "<1>is a proof in S") but only a partial decision procedure for '<1>is a deriv**able** wff in S' (in place of "<1>is a theorem of S"). The latter is only partial, because it gives no method for deciding that a given wff is not derivable. One can try again and again to derive a wff from the axioms and fail; failure does not prove there is no proof.

The decision problem for the axiom system of this chapter is whether there is a full decision procedure for the question 'is '[A CONT C] deriv**able**?' where A and C are any two wffs. In other words is there any procedure whereby one can take any two randomly chosen wffs, A and C, and get either a 'yes' or 'no' within a specifiable finite number of steps, on whether [A CONT C] is derivable from the axioms, rules and definitions given in this chapter?

In M-logic there is a decision procedure for whether any arbitrary wff is derivable wthin a axiomatic formulation of the sentential calculus. It is easily proven that a wff of the sentential calculus will have a proof if and only if it has a truth-table with only T's in the final column. The rules for assigning T's and 'F's are clear and whether they have been correctly assigned or not is fully decidable. The finite number of steps required to test any wff can be computed from the finite number of distinct sentence letters in any given wff.

The use of T's and F's and the word "truth-table" are inessential. The truth-table method may be viewed as simply another set of rules for working with symbols. If we replace 'T' by '#' and 'F' by '7' and 'truth-tables' by '#-tables'. The #-table method is simply a method whereby it is assumed that every sentence letter can be assigned either one, but never to both '#' and '7', and rules are given for assigning '#' or '7' to compound expressions based on values that have been assigned to its components. The final column of a #-table shows whether # or 7 is assigned to the whole compound for each assignment of # or 7 to its components. Then it is proven that (i) initial X-wffs ("axioms") and any derived X-wff will have only #'s in the final column of its "#-table", and (ii) if any wff has only #'s in the final column of its "#-table" then a derivation of that wff from the initial X-wffs can be constructed. Thus a general decision procedure is provided:

<1>is an X-wff if and only if <1>is a wff which has only #'s in the final column of its "#-table".

According to Church's theorem there is no such general decision procedure for whether a quantificational wff is derivable in M-logic.

In A-logic we have seen that there are general decision procedures for determining whether any two unquantified wffs, A and C, are such that [A SYN B] or [A CONT C], is derivable. One can use "analytic truth-tables", or "POS/NEG tables", as well as reduction to basic normal forms.

The decision problem at this point is whether, if A and/or C are Q-wffs, there is a general decision procedure for whether [A CONT C] is derivable in the fragment of A-logic in this chapter.

In light of Church's theorem, an affirmative answer would depend on differences between the decision problem in M-logic and in A-logic. Some of these differences are made clear by consideration of a reduction of the problem in M-logic by Janos Suranyi. Suranyi stated that the decision problem for M-logic can be solved if there is a general decision procedure for determining the derivability of wffs of the form

I. $[(\forall x)(\forall y)(\forall z)A\langle x,y,z\rangle \supset (\exists x)(\exists y)(\forall z)C\langle x,y,z\rangle]$,

where $A\langle x,y,z\rangle$ and $C\langle x,y,z\rangle$ are quantifier-free matrices and contain none but binary predicate letters.[13]

The class of pairs of wffs $\langle P,Q\rangle$ such that [P CONT Q] is a subclass of the pairs such that $\vdash [P \supset Q]$. Assuming Suranyi is correct, would it follow that if a general decision procedure were found for

II. $[(\forall x)(\forall y)(\forall z)A\langle x,y,z\rangle$ CONT $(\exists x)(\exists y)(\forall z)C\langle x,y,z\rangle]$,

then a the decision problem for derivability of $[P \supset Q]$ could be solved? This is far from clear. It is not even clear that a general decision procedure for [P CONT Q] would follow. Nevertheless, it is instructive to consider whether there could be a general decision procedure for every statement of the form II. For, asking that question brings out marked differences in the conditions that must be satisfied.

1) Let P be any Q-wff of M-logic without individual constants or vacuous quantifiers such that P is a quantificational wff of A-logic.
   and let $\langle P,Q_1\rangle$ be the class of pairs of wffs such that $\vdash [P \supset Q_1]$,
   and let $\langle P,Q_2\rangle$ be the class of pairs of wffs such that $\models [P \text{ Cont } Q_2]$.

2) $\langle P,Q_2\rangle$ has a subclass of pairs not included in $\langle P,Q_1\rangle$ such that no components of both P or Q_2 occur NEG. For none of these $\vdash [P \supset Q]$. Any pair such that [P CONT Q] is derivable by substitution in these.

3) The the class $\langle P,Q_1\rangle$ is much larger than that of $\langle P,Q_2\rangle$, due to constraints imposed by the definition of 'CONT':
   a) Members of any pair in $\langle P,Q_2\rangle$ must be such that Q_2 has no predicate letter which does not occur in P. Any pair in $\langle P,Q_1\rangle$ (i.e., if $\vdash [P \supset Q_1]$), Q_1 may have occurrences of any finite number of predicate letters not in P.

   b) Members of any pair in $\langle P,Q_2\rangle$ must be such that no predicate letter occurs NEG in Q_2 unless it occurs NEG in P and such that no predicate letter occurs POS in Q_2 unless it occurs POS in P. If $\vdash [P \supset Q_1]$, Q_1 may have occurrences of any finite number of predicate letters occurring NEG or POS regardless of whether or how they occur in P.

   c) If $\models [P \text{ Cont } Q_2]$, Q_2 must be such that every conjunct in the MOCNF(Q_2) must be a conjunct in the MOCNF(P). If $\vdash [P \supset Q_1]$, then given any R whatever, $\vdash [(P \& {\sim}Q_1) \supset R]$ even if MOCNF(R) has no conjuncts which are conjuncts of MOCNF(P) or of MOCNF $(P \& {\sim}Q_1)$.

---

13. Suranyi, Janos, *Acta Mathematica Academiae Scientiarum Hungaricae*, Vol 1 (1950), pp 261-271. Cited in Church, IML, p 279

This shows that given any matrix, A<x,y,z>, there will be a vast number of matrices, C<x,y,z> in M-logic theorems of form I, which are not admissible in A-logic theorem of form II which have A<x,y,z> as the matrix of its antecedent. The excluded versions of C<x,y,z> are excluded because of constraints in the definitions of SYN and CONT.

If a wff of the form I is to be a theorem of M-logic the normal forms of the theorem must have at least one predicate letter that occurs both POS and NEG. If a wff of the form II is to be a theorem of A-logic, it is not necessary that any predicate letter occurs both POS and NEG int the antecedent and/or the consquent. In fact, all theorems of Form II in A-logic can be gotten by substitution in theorems with no components occurring NEG at all.

Thus to get a general decision procedure for whether

II. [(∀x)(∀y)(∀z)A<x,y,z> CONT (∃x)(∃y)(∀z)C<x,y,z>],

is derivable or not,

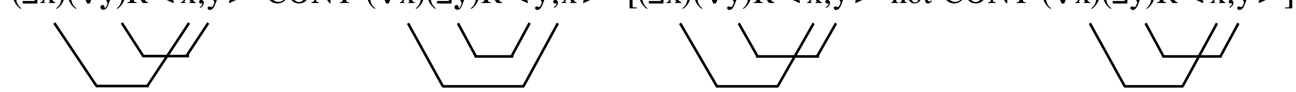
Step 1: (To simplify the procedure) uniformly replace each predicate letter which occurs NEG in P and/or Q_2, by a new predicate letter which does not occur elsewhere, and occurs POS.

Step 2: Reduce the results on A<x,y,z> and C<x,y,z> to MOCNF wffs.
Let us call the results MOCNF(C') and MOCNF(A').

Step 3: We get a negative decision if any of the following are not satisfied:
2.1 All predicate letters in MOCNF(C') also occur in MOCNF(A');[14]
2.2 All conjuncts of MOCNF(C') are conjuncts of MOCNF(A');
2.3 Every mode of a predicate which occurs in C' occurs as a mode of that predicate in A'.

However, this does not make a general decision procedure. Surviving pairs of wffs are affected by patterns of cross-reference from quantifiers to atomic components. Consider, with the above conditions met,

Somebody loves everybody logically CONTAINS Everybody is loved by someone	Somebody loves everybody does not logically contain Everybody loves somebody
(∃x)(∀**y**)R<x,y> CONT (∀x)(∃**y**)R<y,x>	[(∃x)(∀y)R<x,y> not-CONT (∀x)(∃y)R<x,y>]



These schemata have the quantifiers occurring in the alphabetic order of their variables in both antecedent and consequent, while the variables in the matrix occur in a different order.

By alphabetic variance, with no change of meaning, quantifiers' variables can occur in non-alphabetic order in one wff and in alphabetic order in the other, while variables within the two matrices are in the same alphabetic order:

---

14. If |= [A CONT C] then (i) No predicate letter in C occurs NEG unless it also occurs NEG in A, and (ii) no predicate letter in C occurs POS unless it also occurs POS in A. But Step 2, which replaces NEG-occurring letters uniformly with new letters, together with Step 3.1 assures accountbility to this requirement.

Somebody loves everybody
logically CONTAINS
Everybody is loved by someone
(∃y)(∀x)R<y,x> CONT (∀x)(∃y)R<y,x>

Somebody is loved by everybody
does not logically contain
Everybody is loved by somebody
(∃y)(∀x)R<x,y> not-CONT (∀x)(∃y)R<x,y>

In Boolean expansions the differences in cross-reference disappear; they are creatures of abbreviation. Instead we have disjunctions of conjunctions, or conjunctions of disjunctions, and the question (re [A CONT C]) is whether all conjuncts in a basic conjunctive normal form of C are conjuncts in the MOCNF(A).

Whether the lettering in the quantifiers is the same or not is not decisive. (∃x)(∀y)Rxy yields the same Boolean expansion in any domain as (∃u)(∀x)Rux or (∃z)(∀w)Rzw. However, a general decision procedures can will be facilitated by relettering the Q-wffs so that quantifiers occur in alphabetic order of their variables in both antecedent and consequent (using principles of alphabetic variance).

As the table below shows, the order of the quantifiers makes a difference, even when the matrices are the same or differ only in the order of variables within atomic wffs of the matrix.

Boolean expansions in a domain of 2:

(∀x)(∀y)Rxy = ((R11&R12) & (R21& R22))
(∀x)(∃y)Rxy = ((R11v R12) & (R21vR22))
(∃x)(∀y)Rxy = ((R11&R12) v (R21&R22))
MinOCNF[(∃x)(∀y)Rxy] SYN (**(R11v R21)** & (R11v R22)& (R12v R21) & **(R12v R22)**)
if D = 2, [(∃x)(∃y)Rxy SYN ((R11vR12) v (R21vR22))]

Also the number of quantifiers in the consequent may be greater, the same as, or smaller than, the number in the antecedent; the number will not determine whether A CONT C or not.

(∃x)(∀y)Rxy CONT (∃x)Rxx,	(∃x)(∀y)Rxy does not CONT (∀x)Rxx
(∃x)(∀y)Rxy CONT (∀x)(∃y)Ryx	(∃x)(∀y)Rxy does not CONT (∀x)(∃y)Rxy
(∃x)(∀y)Rxy CONT (∀x)(∃y)(∃z)(Ryy & Rxz)	(∃x)(∀y)Rxy does not CONT (∀x)(∃y)(∃z)(Ryy & Rzx)

Theoretically, there is a decision procedure for every finite domain: test the Boolean expansion in that domain. But to be a logical containment, the antecedent must contain the consequent in every domain no matter how large. Is there a formula which will select, for any two wffs, a finite number n, such that n is the size of the finite domain such that if [P CONT Q_2] in that domain, then [P CONT Q_2] in all larger domains?

If such a method is to work, the formula must select a domain which has at least as diverse a set of ordered tuples as would be produced by the number of quantifiers in Q_2 and the modes covered by the predicates in Q_2. To this end the domain chosen must have at least as many members as the largest number of quantifiers needed by either the antecedent or the consequent, for the number of quantifiers affects the diversity of the ordered n-tuples in the two wffs, even when they the same predicate letters in the same mode. E.g., [(∀x)(∀y)(Rxy v Ryx) CONT (∀x)(∀y)(∀z)(Rxy v Ryz)] in a domain of 2, but not in a domain of 3.

Returning to the Suranyi reduction class, Does (∀x)(∀y)(∀z)A(x,y,z) CONT (∃x)(∃y)(∀z)C(x,y,z)? Obviously, if [A(x,y,z)] SYN [C(x,y,z)] the answer will be yes in all domains, for when all quantifiers in the antecedent are conjunctive (universal), all disjunctive quantifications of the same matrix will be contained in the antecedent, since any conjunction contains all possible disjunctions of its conjuncts. Thus it seems the problem should be decidable by reference to the expansion of both A<x,y,z> and C<x,y,z> in a domain of 3.

Does Bx3,MinOCNF[(∀x)(∀y)(∀z)A(x,y,z)]
(((A111 & A112 & A113) & (A121 & A122 & A123) & (A131 & A132 & A133))
& ((A211 & A212 & A213) & (A221 & A222 & A223) & (A231 & A232 & A233))
& ((A311 & A312 & A313) & (A321 & A322 & A323) & (A331 & A332 & A333)))
CONT Bx3,MinOCNF[(∃x)(∃y)(∀z)C(x,y,z)]**?**
(((C111 & C112 & C113) v (C121 & C122 & C123) v (C131 & C132 & C133))
v ((C211 & C212 & C213) v (C221 & C222 & C223) v (C231 & C232 & C233))
v ((C311 & C312 & C313) v (C321 & C322 & C323) v (C331 & C332 &C333))))

The decision will vary with variations in what is put for A and C. All elementary components occurring in the consequent must occur in the antecedent, and all conjuncts of the consequent must be conjunctions or disjunctions of conjuncts in the antecedent. This part of the decision procedure falls under the general decision procedure for unquantified wffs. To decide whether ⊢ [A ⊃ C] holds in this case one must decide whether [(∀x)(∀y)(∀z)A(x,y,z)] is inconsistent internally, and whether [(∃x)(∃y)(∀x)C<x,y,z>] is tautologous by itself. Neither case is relevant to whether ⊨ [A Cont C].

Even if there is a decision procedure for the 'CONT'-for-'⊃' analogue of Suranyi's reduction class, this would not prove that there is a general decision procedure for CONT-theorems, nor would it prove that a finite domain can be found for any A and C, such that A logically contains C in that domain if and only if it does so in all domains. Nevertheless, the discussion above highlights important differences in the decision problems.

It may help to compare the suggested kind of decision procedure for [A CONT C], with the "tree method" used to prove ⊢ [A ⊃ C] a theorem in A-logic. The latter is complete, in the sense that it will give a proof of any theorem of M-logic, but it is not a decision procedure because it can not reach a decision in all cases in which a Q-wff is not a theorem.

In the example below, testing logical containment in a Boolean expansion is decidable either yes or no, in a case in which the tree method leads to an endless number of steps.

Tree Method:			Testing for Logical Containment
	If everybody loves somebody then some love themselves		If everybody loves somebody then some love themselves
	⊢ ((∀x)(∃y)Rxy ⊃ (∃x)Rxx)		⊨ [(∀x)(∃y)Rxy CONT (∃x)Rxx]
1)	(∀x)(∃y)Rxy	[Premiss]	In a domain of 2, (∀x)(∃y)Rxy
2)	(∀x)~Rxx	[Denied conclusion]	is: ((Raa v Rab) & (Rba v Rbb))
3)	(∃y)Ray	[1),UI]	This <u>does not</u> CONTAIN,
4)	Rab	[3),EI]	i.e., does not have as a conjunct,
5)	~Rbb	[2),UI]	(∃x)Rxx, which is: (Raa v Rbb)

6)	(∃y)Rby	[1),UI]	Hence the containment does not
7)	Rbc	[6),EI]	Hold in all domains,
8)	~Rcc	[2),UI]	
9)	(∃y)Rcy	[1),UI]	
	… no end		

If |= [A Cont C] is decided affirmatively by the method suggested, then |- [A ⊃ C] is decided affirmatively by the tree method.

Tree Method:		Testing for Logical Containment
If somebody loves everybody		If someone loves everybody
then someone loves ones self		then someone loves himself
|- [(∃x)(∀y)Rxy ⊃ (∃x)Rxx]		|= [(∃x)(∀y)Rxy Cont (∃x)Rxx]
1) (∃x)(∀y)Rxy	[Premiss]	In a domain of 2, ((∃x)(∀y)Rxy
2) (∀x)~Rxx	[Denied conclusion]	is ((Raa & Rab)v(Rba & Rbb))
3) (∀y) Ray	[1),EI]	SYN ((Raa v Rba) & **(Raa v Rbb)**
4) Raa	[3),UI]	&(Rab v Rba) & (Rab v Rbb))
5) ~Raa	[2),UI]	which CONT (∃x)Rxx, i.e, **(Raa v Rbb)**

The preceding discussion is intended to make clear that the lack of a general decision procedure for theorems of M-logic does not entail the lack of a decision procedure for CONT-theorems in A-logic. It does not solve the decision problem for [A CONT C], but suggests further investigation.

4.52 Soundness and Completeness re: [A CONT C]

There are ordinary senses of the terms "soundness" and "completeness" which are central, though not without real problems.

Suppose we believe the central concept of concern in logic is the concept of something "following logically from" something else. In evaluating a system of logic for soundness the question then is, are there any pairs of expressions A and B such that the system of logic says "B follows logically from A" when in fact B does not follow logically from A? For completeness, the question is, are there any pairs of expressions A and B such that although B follows logically from A in fact, there is no theorem in the logic that states that B follows logically from A.

In this chapter we do not give give a formal account of "following logically". Rather we are building the ground work for it. We hold that the concepts of 'SYN' and 'CONT' are essential ingredients in the concept of "following logically from" though neither is identical with it.

So we may ask at this point whether the system of logic so far developed is sound and complete with respect to "is logically synonymous with". The terms 'logical' and 'synonymous' are both well established words of ordinary language, so we might hold that there is one rational, ordinary meaning for "logically synonymous". Assuming this, the question of soundness is whether there any pairs of wffs A and B, such that 'A SYN B' occurs as a theorem, but A is not logically synonymous with B in ordinary discourse? For completeness the questions is, are there any pairs of expressions A and B, such that A is logically synonymous with B, but 'A SYN B' can not be derived in the system.

There are two basic problems with this "ordinary sense" approach to soundness and completeness. One is that in ordinary language many different meanings and usages of "synonymous" and "logical" occur, and there is therefore no absolute fact with respect to any ordinary-language assertion that A is logically synonymous with B. Indeed the compound term "logically synonymous" is not widely used. Second, if a system of logic is to be rigorous with a proof of completeness or soundness, the logician must formalize one unambiguous meaning each for "logical", "synonymous" and "logically synonymous". In providing such meanings for A-logic the logician may add elements which are not recognized in ordinary discourse. He does this to make his terms more precise and fixed than anything available in ordinary discourse. All attempts to formalize logic have these problems.

The logician's definitions of his terms are, then, semi-artificial; they do not fall out precisely from pre-existing "natural language". This does not mean they are not good. The cotton gin was an artifical contrivance, but it helped people to do things better than they could do them "naturally". The logician's definitions should be evaluated in terms of their contribution to the things we want logic to do. Though "logically follows" is a rarely used term, there are abundant examples in mathematics, science, business, most special areas of knowledge, and common sense, of cases in which it would be generally agreed, if asked, that specific conclusions follow logically from their premisses or assumptions. We want formal logic to give a systematic account which explains why as many as possible of these cases follow logically, without being forced also into saying that some obvious non-sequiturs are cases of following logically.

The initial ordinary sense of the questions remains central, but takes the more amorphous form of asking whether a system of logic does the jobs we want logic to do. We want it to account for '<1> logically implies <2>', '<2> follows logically from <1>', '<1> is a valid argument', and other expressions. Where its results fail to coincide with strong ordinary usage, we look for a correction. Where it fails to account for some of the things we expect in logic we look for a revision. The "semi-artificial" definitions the logician introduces should be judged in terms of their role in the final product. However, terms used should be chosen so as to produce results as close as possible to at least one recognizable ordinary usage of the term.

Given the necessary semi-artificiality of basic definitions in logic, the questions of soundness and completeness take two forms. One is relative to ordinary usage just mentioned, the other is more rigorus and technical.

The technical proofs of soundness and completeness for logic are often proofs relative to mathematical models. But the soundness and completeness of a formal system of logic with respect to ordinary language, is not necessarily established by proving completeness or soundness relative to a mathematical model. The latter are only as good for soundness and completeness in the ordinary sense, as the completeness or soundness in the ordinary sense of the non-mathematical interpretation of the mathematical models.

The technical question of the soundness and completeness of A-logic with respect to [A CONT B] is different than the technical question of the completeness of M-logic with respect to $\vdash [A \supset B]$, because the definitions of terms used in the two logics differ at certain points.

The propositional logic and quantification theory of M-logic (the "first order predicate calculus") is proven sound and complete in the following sense (called "semantic completeness"):

1) it is sound in the sense that all instantiations of its theorem-Schemata are true when interpreted according to the semantic theory which treats 'and' and 'not' as truth-functions and '(∀x)Px' as true if and only if for every interpretation of its predicate letters and variables, there are no false cases.

2) It is complete if there is no wff that is true under all interpretations, but is not a theorem.

It is also consistent (sound) and complete in several technical senses based on properties of the mathematical models being used.[15]

A-logic is like M-logic in its definition of "logical structure" (except that it has a different binary connective for 'if…then'). But its semantic theory does not define logical truth in terms of truth-values, or truth-functions, of object-language expressions. Rather it is concerned with properties and relationships of meanings of expressions. The truth-values of expressions can be the same when meanings are different. The predicates '<1>is true' and '<1>is false' are extra- logical predicates. The predicates '<1>is inconsistent' and '<2>is tautologous" as well as '<1>SYN<2>' and '<1>CONT<2>' are logical predicates—predicates which it is the primary business of logic to study.

Thus the semantic completeness and soundness of A-logic, will depend in part on the completeness of 'A SYN B' (from which 'A CONT B' is defined). The relationship of this soundness and completeness to ordinary meanings of 'A is synonymous with B' that A-logic is trying to capture is expressed as follows: 'A is synonymous with B' means that A and B are linguistic expressions with meanings, and B refers to or talks about all and only those things that A talks about, and B ascribes about all and only the same properties and relations to things it talks about that A does, and all and only those things entailed or implied by A is entailed or implied by B. The complete formalization of this can not be given until later when 'implies' and 'entails' as well as other terms have been explicated. But the defining symbolic features and relationships which convey 'A is logically synonymous with B' or 'A SYN B' are:

(i) all and only those individual constants which occur in A occur in B;
(ii) all and only those predicate letters which occur in A occur in B;
(iii) each predicate letter in B applies to all and only those n-tuples of individual constants that the same predicate letter applies to in A;
(iv) Every predicate letter occurs POS in A iff it occurs POS in B; Every predicate letter occurs NEG in A iff it occurs NEG in B;
(v) All and only conjuncts of MOCNF(A) are conjuncts of MOCNF(B).

The adverb 'logically' signifies that 'A' and 'B' in this definition refers only to logical structures as displayed in wffs, and that the synonymy at issue is only synonymy so far as it is determined by pure logical structures.

A proof of technical soundness for the system of this chapter with respect to |= [A SYN B] would consist in a proof that whenever two wffs are proven to be SYN, they will have just the syntactical properties and relations stated in (i) to (v); and that if they fail to meet any one of them, a proof that they are SYN can not go through. The technical completeness of the system with respect to [A SYN B], consists in a proof that if A and B are any two wffs, as wffs are defined in this chapter, which satisfy all of the properties and relationships in (i) thru (v), then a proof of |= [A SYN B] is derivable.

These kinds of technical proofs are necessary to establish the rigor and universality of a formal system; these are the features which make a formal system an improvement over ordinary language. But any statement that A-logic is more sound and more complete, than M-logic, is not an assertion of

15. E.g., M-logic is **consistent in the sense of Post** iff a wff consisting of a sentence letter alone is not a theorem. M-logic is **complete in the sense of Post**: iff every sentence or propositional form, B, is either derivable, or its addition as an axiom would render the system of logic inconsistent in the sense of Post. [See Church, A, IML, Section 18.] These definitions are based on presuppositions of M-logic. A-logic does not use 'consistent' or 'complete' with respect to a system in these senses; e.g., a negation-free logic can't be inconsistent .

technical soundness and completeness alone. Both should be equally sound and complete in the technical senses. But even if they are not, the important deciding question remains that of soundness and completeness in an ordinary sense. This can be decided only by considering whether the final products of A-logic (i) retain all that is sound in M-logic, while (ii) covering more cases which intelligent people would ordinarily call cases of one thing's following logically from another than M-logic, and/or (iii) excluding cases which M-logic sanctions but which intelligent people would ordinarily call illogical or non-Sequiturs.

The soundness and completeness of A-logic with respect to [A CONT C] where A and C are wffs as defined in this chapter, is an essential requirement for the soundness and completeness of A-logic as a whole. It is presupposed in the next chapter where we show that A-logic retains what is sound in M-logic, while excluding a variety of *non sequiturs*. It is presupposed in Chapters 5 through 8 as we deal with the more central logical concepts of "following logically", "logically valid inference" and "logically valid conditionals", which require in addition to Containment, the property of being consistent or satisfiable, i.e., not being inconsistent.

Section B

Mathematical Logic

Chapter 5

Inconsistency and Tautology

The purpose of this chapter is to present a fragment of A-logic which is sound and complete with respect to M-logic. This fragment consists of the set of all and only those theorems of A-logic which ascribe tautologousness to a wff of M-logic. To accomplish this purpose, we must remove the restrictions in U-SUBab, and use the unrestricted U-SUB of Chapter 3. The resulting system of tautologies stands in 1-to-1 correspondence with the theorems and non-theorems of M-logic; but it is deliberately not complete with respect M-logic's rules of inference, since some of these sanction *non sequiturs* according to A-logic.

5.1 Introduction

Formal Logic has two objectives: 1) to formulate reliable rules of logical inference, and 2) (secondarily) to formulate rules for identifying inconsistencies. Mathematical logic errs in respect to the first, but is a great advance towards the second objective, since what it identifies is tautologies, i.e., denials of inconsistencies which become inconsistencies upon being negated.

The first objective requires a concept of a relation of "following from", which may or may not hold between one idea or assertion and a second one. In A-logic this relation is based on concepts of logical synonymity and containment that are not recognized in M-logic. Negation is not required for synonymy and containment; that is, it is not essential to their definition. In contrast, the second objective requires negation, which is present implicitly or explicitly in all inconsistencies and tautologies.

It is a fact of our existence that ideas, and predicates or sentences which express ideas, can be inconsistent. The common concept of an objective reality (of which certain predicates or sentences might be true or false), is the concept of a complex entity that is never inconsistent, either as a whole or in any of its parts. This notion is reified as the "Law of Non- contradiction". Most human problems are problems of fixing our ideas about processes and facts in objective reality—ideas which are consistent and confirmable in experience. The ability to detect inconsistencies (and eliminate them from thought and discourse) has a basic utility in human problem-solving, but it is not all there is to logic nor is it the most important part of logic.

In the first four chapters we dealt with logical synonymy and containment, including (in Chapter 4) synonymies and containments based on the meaning of negation. They laid some foundations for making reliable determinations of what "follows logically" from what; this concept will be developed further in Chapters 6, 8 and 9.

In this chapter we focus on the concepts of inconsistency and its complement, tautology, which are the central concepts of M-logic. Inconsistency is defined in terms of synonymy, negation and conjunction. We define a tautology simply as a negation of an inconsistency.[1]

Since inconsistencies and tautologies are only possible in wffs which contain contradictories as components (i.e., have one or more atomic components occurring both NEG and POS in the wff), proofs of tautologousness or inconsistency require that we be allowed, by unrestricted U-SUB, to introduce NEG occurrences of one or more atomic wffs into wffs which already have POS occurrences of them (or vice versa). This means removing the restrictions in U-SUBab which has, in effect, governed the use of U-SUB in the first four chapters with the result that all theorems in those chapters were contradictory-free.

In Section 5.21 we give generative definitions of two logical predicates; 'INC<1>' for "<1>is logically inconsistent" and 'TAUT<1>' for "<1>is a tautology" (or, by our definition, "<1>is a denial of an inconsistency"). These are defined in terms of CONTainment and operations with negation, conjunction and disjunction.[2] In Section 2.6 two derived rules of inference, which say that SynSUB preserves INC and TAUT, are added to the logistical base of the previous chapter. Many derived rules of inference follow from these definitions and rules. With the logistic base of the previous chapter, this provides a set of wffs, axioms and rules, from which we can derive all and only the theorems of standard logic as TAUT-theorems within analytic logic.

To prove the completeness of A-logic with respect to M-logic, in Section 5.3 we look at three axiomatizations of M-logic (Section 5.34), interpreting theoremhood there both in terms of tautology and its contradictory, inconsistency. We then prove the soundness and completeness of A-logic (as thus far developed) with respect to the theorems of M-logic, by deriving the primitive rule of inference and axioms of these systems of M-logic from the axioms and rules of A-logic. Since its rules of inference are different, A-logic establishes M-logic's theorems without having any of its "paradoxical" rules of inference.[3] It also avoids some inessential, vacuous, or problematic results found in some formulations of M-logic.

5.2 Inconsistency and Tautology

5.21 Definitions: INC and TAUT

Logical inconsistency (abbr: 'INC') is inconsistency so far as it is determined solely by logical structure, i.e., by the meanings of syncategorematic logical connectives like 'and' and 'not' and 'or' and the

---

1. This account of 'Tautology' accords with Wittgenstein's dictum in *Tractatus Logico-Philosophicus*, "6.1 The propositions of logic are tautologies", where by "propositions of logic" he meant the theorems in Russell and Whitehead's *Principia Mathematica*. It differs from Quine's use of "tautology" in *Mathematical Logic*, where he limits it to "statements which are true solely by virtue of truth-functional modes of composition", i.e., theorems of the sentential calculus, excluding theorems of quantification theory.

2. In Chapter 6 these definitions are augmented by clauses defining inconsistency and tautology of C-conditionals.

3. Including "From inconsistent premisses any conclusion whatever may be logically inferred", and "Every theorem of logic, is validly inferred from any premiss whatever".

syntactical structure. The simplest examples of logically inconsistent predicates have the form '(P& ~ P)'. This is a subspecies of inconsistency. In a broader sense of inconsistency, the expressions,

'<1>is a is square plane figure and <1>is a triangle'
and '<2>is a brother of George and <2>is female'

are recognized as inconsistent predicates, by virtue of certain widely accepted meanings of their content words. But they are only inconsistent if their inconsistency can be derived from the logical structure of meanings or definitions of the words, 'square' and 'triangle' and 'brother' and 'female' acceptable to the parties engaged in discourse. Inconsistency in this broader sense is definable in terms of logical consistency together with principles for defining non-logical predicates.

The symbolism we use to talk about inconsistencies and tautologies may be summarized as follows: We abbreviate the predicate '<1>is inconsistent' in the broad sense, as simply 'Inc<1>'. The concept of *logical* inconsistency is conveyed by '⊨ Inc<1>', and abbreviated as 'INC<1>'.

'⊨ Inc<1>' means 'It is true in A-logic, by virtue of its logical form, that <1> is inconsistent'. Thus '⊨ Inc[P & ~ P]' means 'It is true in A-logic, by virtue of its logical form, that [P& ~ P] is inconsistent'.

Analysed symbol by symbol:

'⊨' means "It is true in A-logic that, by virtue of its logical form…";
'Inc' stands for the predicate of logic, "<1>is inconsistent", and
'[P& ~ P]' stands for "a result of replacing all occurrences of 'P' in '(P& ~ P)' by some well-formed formula or ordinary language predicate or sentence".

The abbreviation of '⊨ Inc' by 'INC', expressed by ('INC<1>' Syn$_{df}$ '⊨ Inc<1>'), makes 'INC[P & ~ P]' stands for the complex statement, written out in full,

"It is true in A-logic that, *by virtue of its logical form*, that a result of replacing all occurrences of 'P' in '(P& ~ P)' by a well-formed formulas, or by any ordinary language predicate or sentence, is inconsistent".

This particular expression, 'INC[P& ~ P]', is a theorem of A-logic, i.e., a true statement of that logic. In contrast, 'INC[P&Q]' and 'INC[P]' are not theorems. Taken as assertions by themselves they are neither true nor false. However, as schemata, they occur in principles of logical inference, such as "If INC[P] then INC[P&Q]", which means **if** the result of replacing 'P' by some wff or predicate or sentence is inconsistent, **then** the result of making the same replacement for 'P' in '(P&Q)' will make [P&Q] inconsistent."

In A-logic, an expression is said to be tautologous if and only if its denial is inconsistent. There are other ways of defining 'tautology' but this definition is especially useful for our present purposes, since it describes a property possessed by all and only those wffs of M-logic which are its theorems.

Df 'Taut'. 'Taut[P]' Syn$_{df}$ 'Inc[~ P]'

This definition holds both in M-logic, formal A-logic, and in A-logic broadly construed (which includes inconsistencies provable by reference to definitions of substantive, non-"logical" words). It follows that formal expressions which are *logically* tautologous are the complements of expressions which are *logically* inconsistent:

Df ‘TAUT’. ‘TAUT[P]’ Syn_{df} ‘INC[~P]’

Thus in A-logic, ‘TAUT[~(P&~P)]’ means in full, “It is true in that, *by virtue of its logical form*, a result of replacing all occurrences of ‘P’ in ‘~(P&~P)’ by some wff, or by any meaningful ordinary language predicate or sentence, is tautologous”. ‘TAUT[~(P&~P)]’ is a theorem of A-logic, but ‘TAUT[P&Q]’ and ‘TAUT[P]’ are not. However, the principles of inference in A-logic include ‘If TAUT[P&Q] then TAUT[P]’ which is A-valid, though neither component is true by itself.

For every INC-theorem, there is a corresponding TAUT-theorem. For every Principle of Inference to an inconsistency, there is a correponding Principle of Inference to a tautology. Since, we take inconsistency as the more primitive notion, we will assign the same names to INC-theorems and principles as to TAUT-theorems and principle, except that the latter will have an apostrophe appended.

So far we have not said what ‘<1>is inconsistent’ means, or provided criteria which if met, would allow its application. We begin with a set of three critera for logical inconsistency, or ‘INC<1>’ which apply both in A-logic and M-logic.

If P is any wff in the language of M-logic ‘INC[P]’ can be generatively defined by a disjunction of three conditionals each of which provides a sufficient condition for predicating ‘INC<1>’ of a predicate or sentence, P.

Df ‘INC’. ‘INC[P]’ Syn_{df} ‘(i) If [(P SYN (Q&~R)) & (Q CONT R)], then INC[P],
or (ii) if [(P SYN (Q&R)) & INC(Q)], then INC[P],
or (iii) if [(P SYN (QvR)) & INC(Q) & INC(R)], then INC[P].’

This will not be a full definition of ‘inconsistent’ since it will not cover inconsistencies in extra-logical expressions, including inconsistencies in future extensions of logic. Here it defines the narrow kind, logical inconsistency—it only covers inconsistencies based on the meanings of the syncategorematic words in abstract logical forms of wffs of M-logic. However, this definition will help define inconsistency in the broader sense when supplemented by principles for defining non-logical terms. Given the Df ‘INC’ and Df ‘TAUT’ it follows that

[TAUT[P] Syn (Either (i) P SYN (Q ⊃ R) & Q CONT R
or (ii) TAUT[Q] & [P SYN (QvR)]
or (iii) [(P SYN (Q&R)] & TAUT(Q) & TAUT(R))]’

The proof is as follows:

1) TAUT[P] Syn_{df} INC[~P] [Df ‘TAUT’]
2) INC[~P] SYN (i) [(~P SYN (Q&~R)) & (Q CONT R)],
or (ii) [(~P SYN (Q&R)) & INC(Q)],
or (iii) [(~P SYN (QvR)) & INC(Q) & INC(R)]’ [Df ‘INC’,U-SUB ‘~P’for’P’]
3) Case (i) [(~P SYN (Q&~R)) & (Q CONT R)],
1) Q CONT R [Assumption]
2) (~P SYN (Q&~R))
3) (~~P SYN ~(Q&~R)

4) (P SYN ∼(Q&∼R))
5) (P SYN (Q ⊃ R))
6) P SYN (Q ⊃ R) & Q CONT R

4) Case (ii) [(∼P SYN (Q&R)) & INC(Q)],
1) (∼P SYN (∼Q&∼R)) & INC(∼Q)]
2) (∼P SYN (∼Q&∼R))
3) (∼P SYN∼(QvR))
4) (P SYN (QvR))
5) INC(∼Q)]
6) TAUT[Q] & [P SYN (QvR)]

5) Case (iii) [(∼P SYN (QvR)) & INC(Q) & INC(R)]
1) [(∼P SYN (∼Qv∼R)) & INC(∼Q) & INC(∼R)]
2) [∼P SYN (∼Qv∼R)) &TAUT(Q) & TAUT(R)]
3) [∼P SYN (∼Qv∼R)]
4) [∼∼P SYN ∼(∼Qv∼R)]
5) [(P SYN ∼(∼Qv∼R)]
6) [(P SYN (Q&R)]
7) [(P SYN (Q&R)] & TAUT(Q) & TAUT(R)]

6) 'TAUT[P]' Syn 'Either (i) P SYN (Q ⊃ R) & Q CONT R
or (ii) TAUT[Q] & [P SYN (QvR)] [By Cases (i),(ii) and (iii)]
or (iii) [(P SYN (Q&R)] & TAUT(Q) & TAUT(R)]'

5.211 Derived Rules: *INC and TAUT with SYN and CONT*

Starting from Df 'Inc' and Df 'Taut', we can derive many principles of inference. There are a variety of principles of inference which which support rules for moving from one expression to the predicate of inconsistency or tautology of some expression. We present many of these principles as metatheorems, named 'MT5-…', rather than derived rules, since they will not be used in proofs outside of this section. Those that are the basis of inference rules in future pages, are named 'DR5-…' to indicate they will be used as derived rules.

The primary metatheorems provide sufficient conditions for predicating inconsistency. But every predication of inconsistency is synonymous with another expression which predicates tautology. In particular, 'Inc(P & ∼Q)' is synonymous with 'Taut(P ⊃ Q)'. The latter is derived from the former by inserting a double negation and applying the definition of '⊃' and of 'Taut'. The latter is named by the same number and letter as the former, but with an apostrophe attached, as in DR5-5 and DR5-5', and DR5-5a and DR5-5a', and MT5-21 and MT5-21'.

In these derivations we presuppose principles of truth-logic which will not be established until Chapter 8. We are in effect applying such principles to logic itself. The principles of inference involved will be mentioned on the right, though the proofs of these principles do not come until later.

The first principle, DR5-5, is a special case of SynSUB; INC (logical inconsistency) stands for a property of expressions which is preserved on substitution of logical synonyms for one another in that expression.

DR5-5. If |= [P Syn Q] and Inc[R] then Inc[R(P//Q)]

Many derivations which follow depend on the principle that if a synonym replaces a synonym in any inconsistent expression then the resulting expression is also inconsistent (with a similar principle for tautologies). In other words, SynSUB preserves the property of Inc. Thus also logical inconsistency and logical tautology are preserved by SynSUB (i.e., SYNSUB preserves INC and TAUT). The following semantic principle is also presupposed: If the antecedent is true, the consequent is true. With 'T' for "it is true that…",

If [T((P Syn Q) and Inc(R)], then T(Inc[(R(P//Q)]
and from this we get the principle restricted to purely formal logical truths:

DR5-5. If [((P SYN Q) & INC(R)) then INC(R(Q//P)]

The substitution of synonyms does not preserve all properties of expressions. If an expression is polysyllabic or composed of three words, the properties of "being polysyllabic", or "composed of three words" are usually not preserved upon substitution of synonyms. Thus DR5-5 is a principle which does not flow from the meaning of 'and', 'or' and 'not' alone. It is justified by the special property of "being inconsistent". From DR5-5 and Df 'Taut', we derive the principle that substitution of synonyms in a tautology preserves tautology.

DR5-5'. If [(P SYN Q) & TAUT(R)] then TAUT(R(Q//P)]

Proof:	1) [(P SYN Q) & TAUT(R)]	[Premiss]
	2) [P SYN Q]	[1),SIMP]
	3) TAUT[R]	[1),SIMP]
	4) INC[~R]	[4),Df 'TAUT']
	5) [(P SYN Q) & INC(~R]	[2),4),ADJ]
	6) If [(P SYN Q) & INC(~R] then INC[~R(Q//P)	[DR5-5,U-SUB]
	7) INC[~R(Q//P)	[5),6),MP]
	8) TAUT[R(Q//P]	[7),Df 'TAUT']
	9) If [(P SYN Q) & TAUT(R)] then TAUT[R(Q//P)	[1) to 9),Cond.Pr.]

The next two principles of inference, based on DR5-5, present analogues of *modus tollens* and *modus ponens*.

|= [If P CONT Q & INC[Q] then INC[P]

Proof:	1) P CONT Q & INC[Q]	[Premiss]
	2) P CONT Q	[1),SIMP]
	3) INC Q	[2),SIMP]
	4) If INC Q then INC(P&Q)]	[3),Df 'INC'(Clause ii)]
	5) INC[P&Q]	[3),4),MP]
	6) (P SYN (P&Q))	[2),Df 'CONT']
	7) INC[P]	[6),5),DR5-5]
	8) If P CONT Q & INC[Q] then INC[P]	[1) to 7),Cond.Pr.]

|= [If P CONT Q & TAUT[P] then TAUT Q]

Proof:	1) P CONT Q & TAUT[P]	[Premiss]
	2) P CONT Q	[1),SIMP]
	3) TAUT[P]	[1),SIMP]

4) (P SYN (P&Q))	[2),Df 'CONT']
5) TAUT[P&Q]	[4),3),DR5-5']
6) If TAUT[P&Q) then TAUT(P) & TAUT(Q)]	[3),Df 'TAUT'(Clause ii)]
7) TAUT(P) & TAUT(Q)	[5),6),MP]
8) TAUT(Q)	[7),SIMP]
9) If P CONT Q & TAUT[P] then TAUT[Q]	[1) to 8),Cond.Pr.]

This principle is very close to the principle of Taut-Detachment, which is the basic rule used in proofs of theorems by most axiomatizations of M-logic. Taut-Detachment is the same as the truth-funtional version of Modus Ponens which Quine used in his *Mathematical Logic*.[4] Most other axiomatizations of M-logic used this as the basic rule of inference. TAUT-detachment differs only in replacing 'P CONT Q' by 'TAUT(P ⊃ Q)', but since these are not synonymous, we defer the more complicated proof of TAUT-Detachment until Section 5.341.

Secondly, we derive several principles which tell how to derive Inconsistent expressions and Tautologous expressions from theorems of containment or synonymity. All inconsistent statements and all tautologous statements (including all theorems of M-logic) can be derived from Synonymy theorems and SynSUB. These principles involve transformation from statements about the logical relations of synonymy and logical containment into statements about the radically different properties of being inconsistent or tautologous. We name these principles DR5-5a to DR5-5d', since they are the basis of derived rules which will be used often in future pages.

These principles are based on Df 'Inc'. There are three clauses in the definition of INC, and another three from Df 'TAUT' from each of which various derived rules follow. The following principles, based on clause (i), will be used widely in later chapters and in the proofs of completeness in Section 5.3.

DR5-5a. If [P CONT Q], then INC[P & ~Q]
DR5-5a′. If [P CONT Q], then TAUT[P ⊃ Q]
DR5-5b. If [P SYN Q], then INC[P & ~Q]
DR5-5b′. If [P SYN Q], then TAUT[P ⊃ Q]
DR5-5c If [P SYN Q], then INC(Q&~P))
DR5-5c′. If [P SYN Q], then TAUT[Q ⊃ P]
DR5-5d If [P SYN Q], then INC((P &~Q) & (Q&~P))
DR5-5d′. If [P SYN Q], then TAUT[P ≡ Q]

Clause (i) of Df 'Inc' bases inconsistency on the containment relation. Some conjunctions with a form (R&~S) are inconsistent although R does not contain S, but all such inconsistent conjunctions will be synonymous with at least one inconsistent conjunction of the form (P&~Q), such that P CONT Q. For example, ((P&~P)&~Q) is INC although ((P&~P) does not contain ~Q; however, ((P&~P)&~Q) SYN ((P&~Q)&~P), and (P&~Q) CONT P. The inconsistency of ((P&~P)&~Q) is derived in A-logic from the inconsistency of ((P&~Q)&~P), plus SynSUB.

4. Quine's version is *104 If ⊢ [$\phi \supset \psi$] and ⊢ ϕ then ⊢ ψ. In A-logic this is expressed as If TAUT(P ⊃ Q) and TAUT(P) then TAUT(Q), where 'P' and 'Q' are placeholders for indicative sentences or sentence schemata.

DDR5-5a If [P CONT Q] then INC[P&~Q]
Proof: 1) [Q CONT R] [Premiss]
2) [(Q&~R) SYN (Q&~R)] [T1-11,U-SUB]
3) [(Q&~R) SYN (Q&~R)] & [Q CONT R] [1),2),Adj]
4) If [(Q&~R) SYN (Q&~R)] & [Q CONT R] then INC[Q&~R] [Df 'INC'clause(i),U-SUB:'Q&~R'for'P']
5) INC[Q&~R] [3),4),MP]
6) If [Q CONT R] then INC[Q&~R] [1) to 5),Cond.Pr]
7) If [P CONT Q] then INC[P&~Q] [6) Re-lettering]

DR5-5a′ If [P CONT Q] then TAUT[~PvQ]
Proof: 1) If [P CONT Q] then INC[P&~Q] [DR5-5a]
2) If [P CONT Q] then INC[~~(P&~Q)] [1),DN,R1b]
3) If [P CONT Q] then TAUT[~(P & ~Q)] [2),Df 'Taut']
4) If [P CONT Q] then TAUT[(P ⊃ Q)] [3),Df '⊃',R1b]

DR5-5b If [P SYN Q] then INC[P&~Q]
Proof: 1) [P SYN Q] [Premiss]
2) [P CONT Q] [1),DR1-11]
3) INC[P&~Q] [2),DR5-5a]
4) If [P SYN Q] then INC[P&~Q] [1) to 3),Cond.Pr.]

DR5-5b′ If P SYN Q then TAUT[P ⊃ Q]
Proof: 1) P SYN Q [Premiss]
2) [P CONT Q] [1),DR1-11]
3) TAUT[(P ⊃ Q)] [2),DR5-5a′,MP]
4) If [P SYN Q], then TAUT[P ⊃ Q] [1) to 4), Cond.Pr.]

DR5-5d If [P SYN Q], then INC((P &~Q) & (Q&~P))
Proof: 1) [P SYN Q] [Premiss]
2) [(P CONT Q) & (Q CONT P)] [1),DR1-14]
3) [P CONT Q] [2),SIMP]
4) [Q CONT P] [2),SIMP]
5) INC[P & ~Q] [3),Df 'INC'clause(i)]
6) INC[Q & ~P] [4),Df 'INC'clause(i)]
7) INC[(P& ~Q) & (Q & ~P)] [6),Df 'INC'clause(ii)]
8) If [P SYN Q], then INC((P &~Q) & (Q&~P)) [1) to 7), Cond. Pr.]

DR5-5d′ If P SYN Q then TAUT[P ≡ Q]
Proof: 1) P SYN Q [Premiss]
2) TAUT[P ⊃ Q] [1),DR5-5b′]
3) [Q CONT P] [1),DR1-12]
4) TAUT[Q ⊃ P] [3),DR5-5a′]
5) TAUT[P ⊃ Q] & TAUT[Q ⊃ P]] [4),5),ADJ]
6) TAUT[(P ⊃ Q) & (Q ⊃ P)] [5),Df 'Taut'(Clause iii)]
7) TAUT[(P ≡ Q)] [6),Df '≡']
8) If [P SYN Q] then TAUT[P ≡ Q] [1) to 7),Cond.Pr.]

Though rooted in Clause (i), DR5-5d′. is also based on Clause (iii) in Df 'Inc' (see step 6)).

5.212 Metatheorems Regarding INC and TAUT with 'and' and 'or'

The logic of inconsistency (hence of tautology) is based on clauses in the definition of inconsistency. MT5-21 and MT5-21' are based directly on clause (ii) in Df 'Inc' which says the inconsistency of one conjunct is sufficient to make the whole conjunction inconsistent.

MT5-21 If INC[P] then INC[P&Q]

Proof: 1) INC[Q] [Premiss]
2) [(Q&R) SYN (Q&R)] [T1-11,U-SUB]
3) [(Q&R) SYN (Q&R)] & INC[Q] [1),2),Adj]
4) If [((Q&R) SYN (Q&R)) & INC(Q)], then INC[Q&R],
[Df 'INC',clause(ii),U-SUB:'Q&R'for'P']
5) INC[Q&R] [3),4),MP]
6) If INC[Q] then INC[Q&R] [1) to 5),Cond.Pr]
7) If INC[P] then INC[P&Q] [6) Re-lettering]

MT5-21' If TAUT[P] then TAUT[PvQ]

Proof: 1) If INC[~P] then INC[~P&~Q] [MT5-21,U-SUB:'~P'for'P','~Q'for'Q']
2) If INC[~P] then INC[~~(~P&~Q)] [1),DN,SynSUB]
3) If INC[~P] then INC[~(PvQ)] [2),Df'v',SynSUB]
4) If TAUT[P] then TAUT[(PvQ)] [3),Df 'Taut']

Clause (iii) of Df 'INC' says that the inconsistency of both disjuncts is necessary and sufficient to make the whole disjunction inconsistent.

MT5-22. If INC[Q v R] then INC[Q] & INC(R)

Proof: 1) (INC[Q v R]) [Premiss]
2) (INC[Q v R]) Syn
((If [((QvR) SYN (Q&~R)) & (Q CONT R)], then INC[QvR])
v (if [((QvR) SYN (Q&R)) & INC(Q)], then INC[QvR])
v (if [((QvR) SYN (QvR)) & INC(Q) & INC(R)], then INC[QvR))
[Df 'Inc',U-SUB'(QvR)'for'P']
3) ((If [((QvR) SYN (Q&~R)) & (Q CONT R)], then INC[QvR])
v (if [((QvR) SYN (Q&R)) & INC(Q)], then INC[QvR])
v (if [((QvR) SYN (QvR)) & INC(Q) & INC(R)], then INC[QvR)) [1),2),SynSUB]
4) Not-[If (<u>(QvR) SYN (Q&~R)</u>) & (Q CONT R)], then INC[QvR]) [non-truth of antecedent]
5) Not-(if [<u>((QvR) SYN (Q&R))</u> & INC(Q)], then INC[QvR]) [non-truth of antecedent]
6) (If [((QvR) SYN (QvR)) & INC(Q) & INC(R)], then INC[QvR)) [3),4),5),Alt.Syll]
7) [((QvR) SYN (QvR)) & INC(Q) & INC(R)] & INC[QvR)] [8),T[INC(Q) & INC(R)]
9) If (INC[QvR] then T(INC[Q] & INC[R]) [1) to 9) Cond Proof]

MT5-22' If TAUT[Q&R] then TAUT[Q] and TAUT[R]

Proof: 1) If INC(~Qv~R) then (INC(~Q) & INC(~R)) [MT5-22,U-SUB]
2) If INC~(~~Q&~~R) then (INC(~Q) & INC(~R)) [1),Df.'v']

3) If INC ~ (Q&R) then (INC(~ Q) & INC(~ R)) [2),DN]
4) If TAUT(Q&R) then (TAUT(Q) & TAUT(R)) [3),Df 'TAUT']

MT5-23 If (INC[P] & INC[Q]) then INC[PvQ]
Proof: 1) INC[Q] & INC[R] [Premiss]
2) [(Q&R) SYN (Q&R)] [T1-11,U-SUB]
3) [(Q&R) SYN (Q&R)] & INC[Q] & INC[R] [1),2),Adj]
4) If [(QvR) SYN (QvR)) & INC(Q) & INC(R)], then INC[QvR] [Df 'INC',clause(iii),U-SUB:'QvR'for'P']
5) INC[QvR] [3),4),MP]
6) If INC[Q] & INC[R] then INC[QvR] [1) to 5),Cond.Pr]
7) If INC[P] & INC[Q] then INC[PvQ] [6) Re-lettering]

MT5-23' If (TAUT[P] & TAUT[Q]) then TAUT[P&Q]
Proof: 1) If INC[~ P] & INC[~ Q] then INC[~ Pv ~ Q] [MT5-23,U-SUB:' ~ P'for'P',' ~ Q'for'Q']
2) If INC[~ P] & INC[~ Q] then INC-[P&Q] [1),DeM,SynSUB]
3) If TAUT[P] & TAUT[Q] then TAUT[(P&Q)] [2),Df 'Taut']

MT5-24 INC[P] & INC[Q] iff INC[PvQ]
Proof: [MT5-22,MT5-23,Df 'iff']

MT5-24' TAUT[P] & TAUT[Q] iff TAUT[P&Q]
Proof: [MT5-22',MT5-23',Df 'iff']

Two other principles for deriving INC-theorems from INC-theorems and TAUT-theorems from TAUT-theorems, are,

DR5-5f. If INC[P] and INC[~ P&Q] then INC[Q] "INC-Det."
DR5-5f'. If TAUT[P] and TAUT[P ⊃ Q] then TAUT[Q]. "TAUT-Det."

The latter is the same as the detachment rule which is generally called "Modus Ponens" in M-logic, and is frequently taken as the sole basic rule of inference there. In A-logic it is called "Tautology Detachment" and is view as a species of disjunctive syllogism, since '(P ⊃ Q)' is synonymous with '(~ P v Q)'. The proofs of these two principle are more involved than those of MT5-21 to MT5-24' and are presented in Section 5.341.

5.213 Derived Rules: *INC and TAUT with Instantiation and Generalization*

Next we extend the principles for arriving at INC- and TAUT- theorems to cover quantified statements and statements about individuals.

If a predicate P<1> is inconsistent, then any statement which predicates that predicate of some individual is inconsistent. This is like the basic rule of instantiation and generalization (R2-3) presented in Chapter 2, except that here it says that the logical property of being inconsistent, if present in any predicate, is carried over to any statement in which that predicate is applied to an individual or individuals. In Chapter 2, 3, and 4, the logical relations of synonymy and containment were carried over from predicates to statements with those predicates. Here it is the property of inconsistency, and derivatively tautology, which is carried over from the predicate to statements with that predicate.

DR5-30. If Inc[P<1>] then Inc[Pa]

Putting '~P<1>' for 'P<1>' in DR5-30, and applying the definition of 'Taut', we get:

DR5-30'. If Taut[P<1>] then Taut[Pa]
Proof: [DR5-30, U-SUB,'~P' for 'P',Df 'Taut']

The converse of DR5-30 also holds. If a particular application of a predicate to an individual is **logically** inconsistent, then it must be because the predicate of that statement is inconsistent by virtue solely of its logical form. This is presented below as MT5-35 :

MT5-35. If INC[Pa] then INC[P<1>]

DR5-30 and MT5-35 together yield the biconditional principles, MT5-52 and MT5-52'.

Note that this biconditional is not the same as a synonymy. For 'INC[Pa]' and 'INC[P<1>]'assert the predicate 'INC' to two different objects, one of which is a statement, and the other a predicate without a subject. The first conditional, 'If INC[P<1>] then INC[Pa] says that if the predicate is inconsistent this makes the statement inconsistent. The second conditional recognizes that if a statement is inconsistent, it is because the predicate is inconsistent—it doesn't say that the statement is what makes the predicate inconsistent.

Many metatheorems are provable. Some of them look like special cases of what is usually called "Universal Generalization" (UG) or "Existential Gneneralization" (EG) in M-logic. Rigorous proofs within the system of A-logic, will be given in Chapter 8.

MT5-31 says, if a predicate [P<1>] is inconsistent, then every application of that predicate will be inconsistent: (INC[Pa_1] & INC[Pa_2] &...& INC[Pa_n]). This warrants MT5-31.

MT5-31. If Inc[P<1>] then (∀x)Inc[Px]
Proof: 1) If Inc[P<1>] then Inc[Pa_1]
2) If Inc[P<1>] then Inc[Pa_2]
3) If Inc[P<1>] then Inc[Pa_1] & Inc[Pa_2]
4) If Inc[P<1>] then Inc[Pa_3]
5) If Inc[P<1>] then Inc[Pa_1] & Inc[Pa_2] & Inc[Pa_3]
... etc
2n+1) If Inc[P<1>] then Inc[Pa_1] & Inc[Pa_2] &...& Inc[Pa_n]
2n+2) If Inc[P<1>] then ($\forall_n$x)Inc[Px]

MT5-31' If Taut[P<1>] then (∀x)Taut[Px]
Proof: 1) If Inc[~P<1>] then (∀x)Inc[~Px] [MT5-31,U-SUB(P/~P)]
2) If Taut[P<1>] then (∀x)Taut[Px] [1),Df 'Taut']

From MT5-35 and MT5-31, by the principle of the hypothetical syllogism, we derive MT5-36, If INC[Pa] then (∀x)INC[Px] which says if a statement about an individual is inconsistent, then every application of that predicate to any individual is logically inconsistent. It follows that if a statement about an individual is logically tautologous, then every application of that predicate to any individual is logically tautologous. This, the dual, MT5-36'. If TAUT[Pa] then (∀x)TAUT[Px], is part of Quine's principle that if a statement is logically true, then every statement which has the same logical form is true; in this case the difference in content is merely the difference in the individuals of which the predicate is predicated.[5]

If a predicate P<1> is inconsistent, then the statement that it applies to at least one thing in some field of reference, is inconsistent. This is based on clause (iii) of the definition of 'Inc', and the fact that a disjunctive quantifier represents a disjunction in which the same predicate is applied to all individuals in the field of reference. The definition of inconsistency for a disjunction requires that each disjunct be inconsistent. Thus the definition of an inconsistency for a disjunctive quantification requires that every disjunct—every application of the predicate of the quantification to an individual be inconsistent. But we have already seen by MT5-31, that this will be the case if the application of that predicate of the quantification to any individual is inconsistent. If [P<1>], and every application of [P<1>], is inconsistent, then [(∃x)Px] is inconsistent, for by clause (iii) of Df 'INC',

|= (INC[Pa_1] & INC[Pa_2] &...& INC[Pa_n]) CONT INC[(∃x)Px]

Proved formally,

DR5-32 If INC[P<1>] then INC[(∃x)Px]

Proof: 1) If INC[P<1>] then INC[Pa_1]

2) If INC[P<1>] then INC[Pa_2]

3) If INC[P<1>] then INC[Pa_1] & INC[Pa_2]

4) If INC[P<1>] then INC[Pa_1 v Pa_2] [3),Df 'INC'(iii)]

5) If INC[P<1>] then INC[Pa_3]

6) If INC[P<1>] then INC[Pa_1 v Pa_2] & INC[Pa_3]

7) If INC[P<1>] then INC[Pa_1 v Pa_2 v Pa_3] [6),Df 'INC'(iii)]

... etc

2n+1) If INC[P<1>] then INC[Pa_1 v ... vPa_n] [2n),Df 'INC'(iii)]

2n+2) If INC[P<1>] then INC[(E_nx)Px]

Putting '~P' for P in DR5-32, and applying Q-exch and the definition of 'TAUT', we get:

DR5-32' If TAUT[P<1>] then TAUT[(∀x)Px]

Proof: 1) If INC[~P<1>] then INC[(∃x)~Px] [DR32.U-SUB,'~P'for'P]

2) If INC[~P<1>] then INC[~(∀x)Px] [1),Q-Exch]

3) If TAUT[P<1>] then TAUT[(∀x)Px] [2), Df 'TAUT']

There are six basic forms of statements which assert inconsistency of a predicate, its instantiation, or a generalization:

INC[P<1>], INC[Pa], (∀x)INC[Px], INC[(∃x)Px], (∃x)INC[Px], INC[(∀x)Px].

We focus first on the INC-Principles from which all TAUT-Principles can be derived. There are 36 possible ordered pairs of the six kinds of INC-Statements which could yield the antecedent and consequent of a conditional principle of inference. Six of these are redundancies, which we will ignore (e.g., If (INC[Pa] then INC[Pa]). Of the remaining 30, 8 are invalid and 22 are valid.

---

5. Quine's logical truths are all tautologies (as defined here). Quine says of logically true statements, "Their characteristic is that they not only are true but stay true even when we make substitutions upon their component words and phrases as we please, provided that the so-called "logical" words...stay undisturbed." p. 4, *Methods of Logic*, (4th Ed.).

DR5-30.	If INC[P<1>] then INC[Pa]	**.:** INC[P<1>] **iff** INC[Pa]	[by MT5-35]
MT5-31.	If INC[P<1>] then (∀x)INC[Px]	**.:** INC[P<1>] **iff** (∀x)INC[Px]	[by MT5-40]
DR5-32.	If INC[P<1>] then INC[(∃x)Px]	**.:** INC[P<1>] **iff** INC[(∃x)Px]	[by MT5-45]
MT5-33.	If INC[P<1>] then (∃x)INC[Px]	But not: If (∃x)INC[Px] then INC[P<1>]),	
DR5-34.	If INC[P<1>] then INC[(∀x)Px]	But not: If INC[(∀x)Px] then INC[P<1>]),	
MT5-35.	If INC[Pa] then INC[P<1>]	**.:** INC[Pa] **iff** INC[P<1>]	[by DR5-30]
MT5-36.	If INC[Pa] then (∀x)INC[Px]	**.:** INC[Pa] **iff** (∀x)INC[Px]	[by MT5-41]
MT5-37.	If INC[Pa] then INC (∃x)[Px]	**.:** INC[Pa] **iff** INC(∃x)[Px]	[by MT5-46]
MT5-38.	If INC[Pa] then (∃x)INC[Px]	But not If (∃x)INC[Px] then INC[Pa]	
MT5-39.	If INC[Pa] then INC[(∀x)Px]	But not: [If INC(∀x)Px then INC(Pa)]	
MT5-40.	If (∀x)INC[Px] then INC[P<1>]	**.:** (∀x)INC[Px] **iff** INC[P<1>]	[by MT5-31]
MT5-41.	If (∀x)INC[Px] then INC[Pa]	**.:** INC[Pa] **iff** (∀x)INC[Px]	[by MT5-36]
MT5-42.	If (∀x)INC[Px] then INC[(∃x)Px]	**.:** INC[(∃x)Px] iff (∀x)INC[Px]	[by MT5-47]
MT5-43.	If (∀x)INC[Px] then (∃x)INC[Px]	But not: If (∃x)INC[Px] then(∀x)INC[Px]	
MT5-44.	If (∀x)INC[Px] then INC[(∀x)Px]	But not If INC[(∀x)Px] then (∀x)INC[Px]	
MT5-45.	If INC(∃x)[Px] then INC[P<1>]	**.:** INC[(∃x)Px] **iff** INC[P<1>])	[by DR5-32]
MT5-46.	If INC(∃x)[Px] then INC[Pa]	**.:** INC[Pa] **iff** INC[(∃x)Px])	[by MT5-37]
MT5-47.	If INC[(∃x)Px] then (∀x)INC[Px]	**.:** INC[(∃x)Px] **iff** (∀x)INC[Px])	[by MT5-42]
MT5-48.	If INC[(∃x)Px] then (∃x)INC[Px]	But not: If (∃x)INC[Px] then INC[(∃x)Px]	
MT5-49.	If INC[(∃x)Px] then INC[(∀x)Px]	But not: If INC[(∀x))Px] then INC[(∃x)Px]	
MT5-50.	If (∃x)INC[Px] then INC[(∀x)Px]	**:.** INC[(∀x)Px] **iff** (∃x)INC[Px]	[by MT5-51]
MT5-51.	If INC[(∀x)Px] then (∃x)INC[Px]	**:.** INC[(∀x)Px] **iff** (∃x)INC[Px]	[by MT5-50]

From these we derive seven biconditionals, and the converse of each holds:

MT5-52.	INC[Pa] **iff** INC[P<1>]	[MT5-35, DR5-30, Df' 'iff']
MT5-53.	(∀x)INC[Px]**iff** INC[P<1>]	[MT5-40,MT5-31, Df' 'iff']
MT5-54.	INC[(∃x)Px] **iff** INC[P<1>]	[MT5-45,DR5-32, Df' 'iff']
MT5-55.	INC[Pa] **iff** (∀x)INC[Px]	[MT5-36,MT5-41, Df' 'iff']
MT5-56.	INC[Pa] **iff** INC(∃x)[Px]	[MT5-37,MT5-46, Df' 'iff']
MT5-57.	INC[(∃x)Px] **iff** (∀x)INC[Px]	[MT5-42,MT5-47, Df' 'iff']
MT5-58.	INC[(∀x)Px] **iff** (∃x)INC[Px]	[MT5-50,MT5-51, Df' 'iff']

Each of the first six basic forms of statements, which assert inconsistency of a predicate, its instantiation or a generalization, is synonymous respectively to a TAUT-principle for the negation of the predicate, gotten by Df 'TAUT', by double negating the wff involved and, where the fourth and sixth cases occur, using Q-exch.

TAUT[~P<1>], TAUT[~Pa], (∀x)TAUT[~Px], TAUT[(∀x)~Px], (∃x)TAUT[~Px],TAUT[(∃x)~Px],

The TAUT-Principle which is the dual of any INC-Principle can be gotten by simply negating the wff or matrix, using Q-exch where the fourth and sixth cases occur and applying

Df 'TAUT'. This kind of proof was given above for DR5-32' and DR5-34'.

DR5-30'. If TAUT[P<1>] then TAUT[Pa] **[DR5-30**,U-SUB(P/~P),Df ‘TAUT’]

MT5-31'. If TAUT[P<1>] then (∀x)TAUT[Px] [MT5-31,U-SUB(P/~P),Df ‘TAUT’]

DR5-32'. If TAUT[P<1>] then TAUT[(∀x)Px] **[DR5-32**,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-33'. If TAUT[P<1>] then (∃x)TAUT[Px] [MT5-33,U-SUB(P/~P),Df ‘TAUT’]

DR5-34'. If TAUT[P<1>] then TAUT[(∃x)Px] **[DR5-34**,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-35'. If TAUT[Pa] then TAUT[P<1>] [MT5-35,U-SUB(P/~P),Df ‘TAUT’]

MT5-36'. If TAUT[Pa] then (∀x)TAUT[Px] [MT5-36,U-SUB(P/~P),Df ‘TAUT’]

MT5-37'. If TAUT[Pa] then TAUT (∃x)[Px] [MT5-37,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-38'. If TAUT[Pa] then (∃x)TAUT[Px] [MT5-38,U-SUB(P/~P),Df ‘TAUT’]

MT5-39'. If TAUT[Pa] then TAUT[(∃x)Px] [MT5-39,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-40'. If (∀x)TAUT[Px] then TAUT[P<1>] [MT5-40,U-SUB(P/~P),Df ‘TAUT’]

MT5-41'. If (∀x)TAUT[Px] then TAUT[Pa] [MT5-42,U-SUB(P/~P),Df ‘TAUT’]

MT5-42'. If (∀x)TAUT[Px] then TAUT[(∃x)Px] [MT5-42,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-43'. If (∀x)TAUT[Px] then (∃x)TAUT[Px] [MT5-43,U-SUB(P/~P),Df ‘TAUT’]

MT5-44'. If (∀x)TAUT[Px] then TAUT[(∃x)Px] [MT5-44,U-SUB(P/~P), Q-Exch, Df ‘TAUT’]

MT5-45'. If TAUT(∀x)[Px] then TAUT[P<1>] [MT5-45,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-46'. If TAUT(∀x)[Px] then TAUT[Pa] [MT5-46,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-47'. If TAUT[(∀x)Px] then (∀x)TAUT[Px] [MT5-47,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-48'. If TAUT[(∀x)Px] then (∃x)TAUT[Px] [MT5-48,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-49'. If TAUT[(∀x)Px] then TAUT[(∃x)Px] [MT5-49,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-50'. If (∃x)TAUT[Px] then TAUT[(∃x)Px] [MT5-50,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-51'. If TAUT[(∀x)Px] then (∃x)TAUT[Px] [MT5-51,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-52'. TAUT[Pa] **iff** TAUT[P<1>] [MT5-52, U-SUB(P/~P).Df ‘TAUT’]

MT5-53'. (∀x)TAUT[Px]**iff** TAUT[P<1>] [MT5-53,U-SUB(P/~P).Df ‘TAUT’]

MT5-54'. TAUT[(∀x)Px] **iff** TAUT[P<1>] [MT5-54,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-55'. TAUT[Pa] **iff** (∀x)TAUT[Px] [MT5-55,U-SUB(P/~P).Df ‘TAUT’]

MT5-56'. TAUT[Pa] **iff** TAUT[(∀x)Px] [MT5-56,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-57'. TAUT[(∀x)Px] **iff** (∀x)TAUT[Px] [MT5-57,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

MT5-58'. TAUT[(∃x)Px] **iff** (∃x)TAUT [Px] [MT5-58,U-SUB(P/~P),Q-Exch, Df ‘TAUT’]

The eight conditionals which are invalid as INC-Principles, with counter-examples, are:

Not: If (∃x)INC[Px] then INC[P<1>] (**Converse of** MT5-33. If INC[Pa] then (∃x)INC[Px]}
Counter-example: (∃x)INC(Pxb & ~Pbb) & not INC(P<1,b> & ~Pbb)
Counter-example: (∃x)INC(Px & ~Pa_2) & not INC(P<1> & ~Pa_2)

Not: If INC[(∀x)Px] then INC[P<1>] (**Converse of** DR5-34. If INC[Pa] then INC[(**α**x)Px])
Counter-example: INC(∀x)(Pxa_2 & ~Pa_2a_2) & not INC(P<1,a_2>& ~Pa_2a_2)
Counter-example: INC(∀x)(Px & ~Pa_2) & not INC(P<1> & ~Pa_2)

Not If (∃x)INC[Px] then INC[Pa] (**Converse of** MT5-38. If INC[Pa] then (∃x)INC[Px]}
Counter-example: (∃x)INC(Pxb & ~Pbb) & not INC(Pab & ~Pbb)
Counter-example: (∃x)INC(Px & ~Pa_2) not INC(Pa_1 & ~Pa_2)

Not: If INC[(∀x)Px] then INC[Pa] (**Converse of** MT5-39. If INC[Pa] then INC[(∀x)Px])
Counter-example: INC(∀x)(Pxa_2 & ~Pa_2a_2) & not INC(Pa_1a_2 & ~Pa_2a_2)
Counter-example: INC(∀x)(Px & ~Pa_2) & not INC(Pa_1 & ~Pa_2)

Not If (∃x)INC[Px] then (∀x)INC[Px] (**Converse of** MT5-43.If ((∀x)INC[Px] then (∃x)INC[Px])
Counter-example: (∃x)INC(Pxa & ~Paa) & not (∀x)INC(Pxa & ~Paa)
Counter-example: (∃x)INC(Px & ~Pa_2) & not (∀x)INC(Px & ~Pa_2)

Not:If INC[(∀x)Px] then (∀x)INC[Px] (**Converse of** MT5-44.If (∀x)INC[Px] then INC[(∀x)Px])
Counter-example: INC(∀x)(Pxa & ~Paa) & not (∀x)INC(Pxa & ~Paa)
Counter-example: INC(∀x)(Px & ~Pa_2) & not (∀x)INC(Px & ~Pa_2)

Not If (∃x)INC[Px] then INC[(∃x)Px] (**Converse of** MT5-48.If INC[(∃x)Px] then (∃x)INC[Px])
Counter-example: (∃x)INC(Pxb & ~Pbb) & not INC(∃x)(Pxb & ~Pbb)
Counter-example: (∃x)INC(Px & ~Pa_2) not INC(∃x)(Px & ~Pa_2)

Not: If INC[(∀x)Px] then INC[(∃x)Px] (**Converse of** MT5-49.If INC[(∃x)Px] then INC[(∀x)Px])
Counter-example: INC(∀x)(Px & ~Pa_2) not INC(∃x)(Px & ~Pa_2)
[U-SUB: P<1>/(P<1>&~Pa_2)]
Counter-example: INC(∀x)(Pxa_2 & ~Pa_2a_2) & not INC(∃x)(Pxa_2 & ~Pa_2a_2)
[U-SUB: P<1>/P<1,a_2>]

The duals of these conditionals, which are also not valid, are, respectively,

Not: If (∃x)TAUT[Px] then TAUT[P<1>] (Converse of MT5-33')
Not: If TAUT[(∃x)Px] then TAUT[P<1>] (Converse of DR5-34')
Not: If (∃x)TAUT[Px] then TAUT[Pa] (Converse of MT5-38')
Not: If TAUT(∃x)Px] then TAUT[Pa] (Converse of MT5-39')
Not: If (∃x)TAUT[Px] then (∀x)TAUT[Px] (Converse of MT5-43')
Not:: If TAUT[(∃x)Px] then (∀x)TAUT[Px] (Converse of MT5-44')
Not: If (∃x)TAUT[Px] then TAUT[(∀x)Px] (Converse of MT5-48')
Not: If TAUT[(∃x)Px] then TAUT[(∀x)Px] (Converse of MT5-49')

A Note on Quantification of Polyadic Predicates. The greatest contribution of M-logic lies in its treatment of relations. Relations are described by polyadic predicates. Aristotle's logic was a great advance in the logic of monadic predicates, but was absolutely inadequate in dealing with predicates having more than one subject. This type of predicate is central to mathematics, e.g., ("…is equal to ___)", "(…is less than ___)". For this reason Aristotle's theory was inadequate for mathematics.

In M-logic the logic of monadic predicates is fundamentally different than the logic of polyadic predicates. Among other things, if Church's thesis is true, the former has a decision procedure; the latter does not. Church's thesis may be stated as "Every effectively decidable predicate is general recursive."[6] Church's theorem proves that if this thesis is true, then polaydic predicate logic is not decidable. Whether Church's thesis is true or not, the proof that monadic predicate logic is decidable by general recursive function theory while polyadic predicate logic is not, marks a real difference between these two parts of M-logic.

6. Kleene, *Introduction to Metamathematics*, D.Van Nostrand, 1952, p 300 - 306.

5.3 Completeness of A-logic re:Theorems of M-logic

To show that Mathematical Logic is completely included in Analytic Logic, we first show that the semantic concepts of INC and TAUT, as applied to the wffs of Mathematical Logic coincide with the concepts of inconsistency and tautology used in Mathematical Logic with respect to the semantic devices of truth-table and truth-trees. (Section 5.31) Then we provide an axiomatization for the system of logic so far as it deals only with wffs of Mathematical Logic, and show that all and only the essential wffs which are theorems of three axiomatizations of Mathematical Logic are derivable as TAUT-theorems of A-logic. (Sections.5.32 thru 5.34).

5.31 The Adequacy of the Concepts of 'INC' and 'TAUT' for M-logic

For the unquantified wffs of M-logic, the adequacy of Df 'INC' is easily proved: 1) Every wff is either a conjunction of wffs, a disjunction of wffs, or a negation of a wff. 2) If P CONT Q, Q is a conjunct in the MOCNF of P and the result of conjoining Q with the denial of one of its conjuncts will be inconsistent; for a conjunction is inconsistent (and has an F in every row of its truth-table) if and only if one or more of its conjuncts is SYN with the negation of another conjunct (Df 'Inc', clause (i)). 3) A conjunction is therefore inconsistent if it has at least one inconsistent conjunct (Df 'Inc', clause (ii)). 4) A disjunction is inconsistent (and takes all F's in its truth-table) if and only if all of its disjuncts are inconsistent (Df 'Inc', clause (iii)). 5) A negation is inconsistent if and only if it is SYN with an inconsistent conjunction or disjunction (no elementary wff is logically inconsistent). Inconsistency and tautology are preserved in synonyms (Df 'Inc', clause (iv)) which is a subclass of TF equivalents.

The use of the terms 'CONT' and 'SYN' are permissible in this proof for M-logic, because it follows from DR5-5a' and DR5-5d' that $\vdash$ [P ⊃ Q] and $\vdash$ [P ≡ Q] are necessary conditions respectively of $\models$ [P CONT Q] and $\models$ [P SYN Q] if P and Q are wffs of M-logic. The smaller extensions of the latter two terms does not affect the generality of the argument.

Returning to the proof of adequacy of Df 'INC'; the use of truth-tables with T's and 'F's, is not essential in the proof above. Ordinarily 'T' and 'F' are interpreted as truth and falsehood; properties which belong only to indicative sentences.[7] But the values in the 2-valued tables may be interpreted T and F, or as NEG and POS, or just as '0' and '1'. The set of wffs which are inconsistent by virtue of their logical form is coextensive with the set of wffs which have all F's in their truth-table. It is also coextensive with all wffs which have all POS in their POS/NEG table (i.e., are L-POS) or with all '0's in their 0/1 table, if 1 is the "designated value" and appropriate functions are assigned to '&', 'v' and '∼'. For A-logic POS/NEG tables provide a better device than truth-tables for determining whether unquantified wffs are inconsistent, tautologous or contingent (exclusively). In A-logic INC is treated first as a property of predicates and derivatively of sentences (saturated predicates). Predicates are neither true nor false, but they are either POS or NEG (exclusively) in a language which has negation. Statements are POS or NEG if and only if their predicates are. Thus the proof above for the adequacy of Df 'INC' holds in A-logic with 'POS/NEG table' replacing 'truth-table' and 'POS' replacing 'F' throughout.

The adequacy of the definition of INC and TAUT for quantificational wffs of M-logic, follows from the completeness of the tree-method in quantification theory. The tree method establishes theoremhood in M-logic by testing the denial of a wff for inconsistency (if the denial is inconsistent, the undenied wff is a theorem (and TAUT by our definition). The inconsistencies come down to cases where an unquantified

7. For saturated predicates - sentences - we can view the two values as 'true and 'not true' in accordance with the truth-table method; this allows a law of bivalence which does not, in Chapter 7, conflict with have three "truth-values" including one for 'not true and not-false'.

atomic wff and its negation both occur in the same line of development in a tree, which is equivalent to clause (ii) of Df 'INC', which says that if any conjunct of a conjunction co-occurs with its denial in the conjunction, then the whole conjunction is inconsistent. This method has been proven complete with respect to the theorems of first order M-logic. From this it follows that the definition of 'INC' and 'TAUT' are adequate with respect to inconsistency in M-logic.

The definitions of INC and TAUT coincide with other methods used in M-logic to determine whether one of its wffs is a theorem (i.e., is tautologous) or is inconsistent. One can check for inconsistency by reducing the wff to a disjunctive normal form, then determine whether every disjunct is a conjunction with at least one elementary wff and its denial as conjuncts. The test for tautology is to reduce the wff to a conjunctive normal form then determine whether every conjunct is a disjunction with at least one elementary wff and its denial as disjuncts.

These methods, somewhat tightened, are available in A-logic. Algorithms were given in Chapter 1 for reducing any wff of standard sentential logic to a logically synonymous conjunctive normal form or disjunctive normal form. Using clauses in the definitions above for INC and TAUT, one can then determine by inspection whether the resulting wff, and thus its synonyms, are INC, TAUT or neither. Methods using bivalent tables, whether of "truth-tables" or POS/NEG tables, coincide with the method of reduction to normal forms. Using bivalent tables one can also prove that a system as a whole is consistent, i.e., that no wff is such that both it and its negation is inconsistent - or that no wff is such that both it and its negation is a tautology. The consistency of the system as a whole is different than the consistency of a wff. The formal system of M-logic can be interpreted (as we shall see) as a system which yields only inconsistent wffs; but it is consistent as a whole in the sense that no wff is such that both it and its negation is a wff. The proof of the consistency of a system is sometimes spoken of as a proof of its soundness.

The co-extensiveness of (i) a property of a bivalent model with respect to a wff (e.g., having all F's) with (ii) the property of being inconsistent, should not be confused with the a definition of the meaning of 'inconsistency'. (Similarly for 'tautology' or 'logical truth'.) In A-logic, the inconsistency and tautology are defined in terms of logical forms, not by non-logical "values" assigned to wffs. They are defined by logical containment, negation, conjunction, and disjunction.

Comparing T and F with Taut and Inc: Summary of Major Differences

The properties of being tautologuous and inconsistent behave in some ways like truth and falsehood; but they behave differently in other ways. Inconsistency and tautology are independent of the notions of truth and falsehood and can not be defined in terms of truth and falsehood alone. Predicates that are neither true nor false can be inconsistent or tautologous, and comands or directives that are neither true nor false can be inconsistent or tautologous.

(i) They appear to behave similiarly in the following cases

If INC[P<1>] then INC[(∀x)Px]	UG	If False[Pa] then False[(∀x)Px]
If TAUT[P<1>] then TAUT[(∃x)Px]	EG	If True[Pa] then True[(∃x)Px]

(ii) They are not similar in the following cases, for, since 'P<1>' represents an unsaturated predicate which is neither true nor false, the conditionals on the right below do not hold.

If INC[P<1>] then INC[Pa],	Not: If False[P<1>] then False[Pa]
If TAUT[P<1>] then TAUT[Pa]	Not: If True[P<1>] then True[Pa]
If INC[P<1>] then INC(∃x)Px,	Not: If False[P<1>] then False(∃x)Px
If TAUT[P<1>] then TAUT[(∃x)Px] **EG**	Not: If True[P<1>] then True[(∃x)Px]

If INC[P<1>] then INC(∀x)Px — Not: If False[P<1>] then False[(∀x)Px]
If TAUT[P<1>] then TAUT[(∀x)Px] **UG** — Not: If True[P<1>] then True[(∀x)Px]

(iii) In the following cases, if the predicate of the matrix of a prenex normal form wff is inconsistent (or tautologous), then the conjunctive (and/or disjunctive) quantifications of that matrix will be inconsistent (or tautologous).

If TAUT[P<1>] then TAUT[(∀x)Px]	UG	Not: If True[Pa] then True(∀x)Px,
If INC[P<1>] then INC[(∃x)Px]	EG	Not: False[Pa] then False(∃x)Px,

5.32 The Logistic Base of Inconsistency and Tautology in A-logic

For purposes of comparing M-logic with A-logic and deriving all theorems of M-logic from the base of A-logic, we add the following to the system as it developed through Chapter 4:

Df 'INC'. 'INC[P]' Syn_{df} '(i) If [(P SYN (Q&~R)) & (Q CONT R)], then INC[P],
or (ii) if [(P SYN (Q&R)) & INC(Q)], then INC[P],
or (iii) if [(P SYN (QvR)) & INC(Q) & INC(R)], then INC[P].'
Df 'TAUT'. 'TAUT[P]' Syn_{df} 'INC[~P]'.

Thus the Axiomatic system is expanded to recognize INC and TAUT as logical properties of expressions, so that they will be preserved by the principles SynSUB, U-SUB and Instantiation and Generalization. The primitive symbols are not altered, nor are the rules of formation of the object-langauge wffs. The axioms and all previous theorems remain. Only the rules of transformation are broadened to allow the derivation of statements affirming that certain wffs have the property of being inconsistent, or tautologous.

I. Primitive symbols
1. Logical constants: & ~ ∀
2. Grouping devices:) (> <
3. Predicate letters: P_1 P_2 P_3 … (The class, {PL})
4. Argument-position-holders: 1 2 3… (The class, {APH})
5. Individual Constants: a_1 a_2 a_3 … (The class, {IC})
6. Individual Variables: x_1 x_2 x_3 … (The class, {IV})

We also use 't_1', 't_2',… for "Subject Terms", i.e., the class {ST} = {{APH}∪{IC}∪{IV}}

II. Rules of Formation
FR1. If $P_i \in$ {PL} then $[P_i]$e {wff}.
FR2. If A and B are wffs, [A&B] is a wff.
FR3. If $P_i \in$ {PL} and t1,…,ti ∈{{APH} ∪ {IC}, then $[P_i\langle t1 \ldots ti\rangle] \in$ {wff}.
FR4. If $t_i \in$ {APH} and $t_j \in$ {IV}, then $[(\forall t_1)P\langle \ldots t_i \ldots\rangle(t_i/t_j)] \in$ {wff}.
FR5. If P is a wff, then [~P] is a wff.

III. Definitions
Df5-1. $[(P \,\&\, Q \,\&\, R)\ \text{SYN}_{df}\ (P \,\&\, (Q \,\&\, R))]$
Df5-2. $[(\forall_k x)Px\ \text{SYN}_{df}\ (Pa_1 \,\&\, Pa_2 \,\&\, \ldots \,\&\, Pa_k)]$
Df5-3. $[(P \vee Q)\ \text{SYN}_{df} \sim(\sim P \,\&\, \sim Q)]$
Df5-4. $[(P \cup Q)\ \text{SYN}_{df} \sim(P \,\&\, \sim Q)]$
Df5-5. $[(P \equiv Q)\ \text{SYN}_{df}\ ((P \supset Q) \,\&\, (Q \supset P))]$
Df5-6. $[(\exists x)Px\ \text{SYN}_{df} \sim(\forall x)\sim Px]$

IV. Axiom Schemata:
Ax.5-1. [P SYN (P&P)] [&-IDEM1]
Ax.5-2. [(P&Q) SYN (Q&P)] [&-COMM]
Ax.5-3. [(P&(Q&R)) SYN ((P&Q)&R)] [&-ASSOC1]
Ax.5-4. [(P&(QvR)) SYN ((P&Q)v(P&R))] [&v-DIST1]
Ax.5-5. [P SYN ∼∼P] [DN]

V. Transformation Rules

DR5-5. If [P SYN Q] and INC[R] then INC[R(P//Q)] "SynSUB$_{INC}$"
DR5-20. If (i) INC[R] "U-SUB$_{INC}$"
and (ii) $P_i<t_1,\ldots,t_n>$ occurs in R,
and (iii) Q is an h-adic wff, where h > n,
and (iv) Q has occurrences of all numerals 1 to n,
and (v) no individual variable in Q occurs in R or S,
then INC[R($P_i<t_1,\ldots,t_n>$/Q)]
DR5-30. If INC[P<1>] then INC[Pa] "INSTANT$_{INC}$"
Alpha.Var

5.33 Selected TAUT-Theorems

For purposes of this chaper we focus on TAUT-theorems which have been presented as axioms or axiom-schemata in certain axiomatizations of M-logic. We will concentrate mostly on axioms used by Quine, Rosser and Thomason. There are several ways that tautologies may be derived from the logistic base of A-logic.

Many tautologies which constitute the TAUT-theorems of M-logic are derivable in one step from the contradictory-free CONT- and SYN-theorems proven in the preceding four chapters, by way of
DR5-5a′. If [P CONT Q], then TAUT[P ⊃ Q]
or DR5-5b′. If [P SYN Q], then TAUT[P ⊃ Q]
or DR5-5d′. If [P SYN Q], then TAUT[P ≡ Q]

Examples are Rosser's Axioms 1 and 2, and Thomason's Axioms 4,5 and 6::

T5-501: TAUT[P ⊃ (P & P)] (Rosser's Axiom 1)
Proof: 1) [P SYN (P & P)] [Ax.5-01]
2) TAUT[P ⊃ (P & P)] [1),DR5-5b′]

T5-136c. TAUT[((P&Q) ⊃ P) (Rosser's Axiom 2)
Proof: 1) (P&Q) CONT P [T1-36]
2) TAUT[((P&Q) ⊃ P) [1),DR5-5a′]

T5-438f. TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)] (Thomason's Axiom 4)
Proof: 1) [(∀x)(P ⊃ Qx) SYN (P ⊃ (∀x)Qx)] [T4-38]
2) TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)] [1),DR5-5b′]

T5-437c TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))] (Quine's *101,Thomason's Axiom 5)
Proof: 1) [(∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx))] [T4-37]
2) TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)) [1),DR5-5a′]

T5-333c: TAUT[(∀x)(Px ⊃ Pa)] (Thomason's Axiom 6)
Proof: 1) [(∀x)Px CONT Pa] [T3-33]
2) TAUT[(∀x)Px ⊃ Pa] [1),DR5-5a′]

In the sections which follow we use DR5-5a′, DR5-5b′ and DR5-5d′ frequently to establish other TAUT-theorems of A-logic which correspond to various theorems of M-logic.

In many other cases, one first uses DR5-5a′ or DR5-5b′ or DR5-5d′, then uses U-SUB and/or SynSUB to arrive at the M-logic theorem. Often in these cases, the major connective, '⊃' in the result, cannot be replaced by 'CONT'. For example, though 'TAUT[P ⊃ (Q ⊃ P)]' is a theorem '[P CONT (Q ⊃ P)]' is not. The Taut-theorem requires re-ordering of components in the CONT-theorem, until the relation of logical containment not longer holds between the two major components :

T5-02. TAUT[P ⊃ (Q v P)] ("Addition")
Proof: 1) [(P&Q) CONT P] [T1-36]
2) INC[(P & Q) & ∼P] [1),DR5-5a]
3) INC[(P & (Q & ∼P))] [2),&-ASSOC]
4) INC[(P & (∼Q & ∼P)] [3),U-SUB(Q/∼Q]
5) INC[(P & ∼(Q v P)] [4),T4-17,]
6) INC[∼∼(P & ∼(Q v P)] [5),DN]
7) INC[∼(P ⊃ (Q v P)] [6),Df '⊃']
8) TAUT[P ⊃ (Q v P)] [7),Df 'TAUT']
(Note: the "antecedent" does not logically contain the "Consequent")

T5-03. TAUT[P ⊃ (Q ⊃ P)] (Thomason's Axiom 1)
Proof: 1) TAUT[P ⊃ (∼Q v P)] [T5-02,U-SUB(Q/∼Q)]
2) TAUT[P ⊃ (Q ⊃ P)] [1),T4-31,SynSUB]
(Here also, the "antecedent" does not CONTain the "Consequent".)

T5-04. TAUT[(∼P ⊃ ∼Q) ⊃ (Q ⊃ P)] (Thomason's Axiom 3)
Proof: 1) [∼(∼P&∼∼Q) SYN ∼(∼P&∼∼Q)] [A1,U-SUB]
2) [∼(∼P&∼∼Q) SYN ∼(∼∼Q&∼P))] [1),&-ORD]
3) [∼(∼P&∼∼Q) SYN ∼(Q&∼P)] [2),DN,SynSUB]
4) [(∼P ⊃ ∼Q) SYN (Q ⊃ P)] [3),Df '⊃' thrice)]
5) [(∼P ⊃ ∼Q) CONT (Q ⊃ P)] [4),Df 'CONT']
6) TAUT[(∼P ⊃ ∼Q) ⊃ (Q ⊃ P))] [5),DR5-5a′]

T5-05. TAUT[(P ⊃ Q) ⊃ (∼(Q&R) ⊃ ∼(R&P))] (Rosser Axiom 3)
Proof: 1) [(P & (QvR)) CONT ((P&Q) v R)] [T1-39]
2) INC[(P & (QvR)) & ∼((P&Q) v R)] [1),Df 'INC'Clause (i)]
3) INC[P & (QvR) & ∼(P&Q) & ∼R)] [2),DeM]
4) INC[(P & ∼(Q&R) & ∼(P&∼Q) & ∼∼R] [2),DeM]
5) INC[(P & ∼(Q&R)) & ∼(P&∼Q) & R] [3),DN]
6) INC[∼(P&∼Q) & ∼(Q&R) & (R&P))] [4),&-ORD]
7) INC[∼∼(∼(P&∼Q) & ∼∼(∼(Q&R) & ∼∼(R&P))] [5),Ax.4-05,R1b]
8) INC[∼((P ⊃ Q) ⊃ (∼(Q&R) ⊃ ∼(R&P))] [6),D6,R1(thrice)]
9) TAUT[(P ⊃ Q) ⊃ (∼(Q&R) ⊃ ∼(R&P)] [7),Df 'TAUT']
(Here also, the main "antecedent" does not CONTain the "Consequent".)

T5-06. TAUT[(P ⊃ Q) ⊃ ((Q ⊃ R) ⊃ (P ⊃ R))] (“Hypothetical Syllogism”}
<u>Proof</u>: 1) TAUT[(P ⊃ Q) ⊃ (~(Q&R) ⊃ ~(R&P)] [7),Df ‘TAUT’]
2) TAUT[(P ⊃ Q) ⊃ (~(Q&~R) ⊃ ~(~R&P)] [1),U-SUB(R/~R)]
3) TAUT[(P ⊃ Q) ⊃ (~(Q&~R) ⊃ ~(P & ~R)] [2),&-COMM]
4) TAUT[(P ⊃ Q) ⊃ ((Q ⊃ R) ⊃ (P ⊃ R)] [3),Df ‘⊃’ (twice)]
(Here also, the main “antecedent” does not CONTain the “Consequent”.)

T5-07. TAUT[((P ⊃ (Q ⊃ R) ⊃ ((P ⊃ Q) ⊃ (P ⊃ R)))] (Thomason’s Axiom 2)
<u>Proof</u>: 1) [(P & (~Pv~QvR)) SYN ((P&~P)v(P&~Q)v(P&R))] [Gen&v-DIST]
2) [(P & (~Pv~QvR)) SYN ((PvPvP) & (PvPvR) & (Pv~QvP) & (Pv~QvR)
&(~PvPvP) & **(~PvPvR)** & (~Pv~QvP) & **(~Pv~QvR)**)] [1),Gen.v&-DIST]
3) [(P & (~Pv~QvR)) CONT (**(~PvPvR)** & **(~Pv~QvR)**)] [2),Df ‘CONT’]
4) [((~PvR)vP) & ((~PvR)v~Q)) SYN ((~PvR) v (P&~Q))] [v&-DIST,U-SUB]
5) [(P & (~Pv~QvR)) CONT ((~PvR) v (P&~Q))] [3),4),SYNSUB]
6) INC[(P & (~Pv~QvR)) & ~((~PvR) v (P&~Q))] [5),Df ‘INC’]
7) INC[(P & (~Pv~QvR) & P & ~R & (~PvQ)] [6),DeM]
8) INC[(~Pv(~QvR)) & (~PvQ) & (P&~R)] [7),&-ORD(IDEM)]
9) INC[~~((~Pv(~QvR)) & ~~((~PvQ) & ~~(P & ~R)))] [8),Ax.4-05,(thrice)]
10) INC[~~(~(P&~~(Q&~R)) & ~~(~(P&~Q) & ~~(P & ~R)))] [9),T4-16(twice)]
11) INC[~~((P ⊃ (Q ⊃ R)) & ~~((P ⊃ Q) & ~(P ⊃ R)))] [10),Df ‘⊃’ (4 times)]
12) INC[~(((P ⊃ (Q ⊃ R)) ⊃ ((P ⊃ Q) ⊃ (P ⊃ R)))] [11),Df ‘⊃’ twice)]
13) TAUT[(((P ⊃ (Q ⊃ R)) ⊃ ((P ⊃ Q) ⊃ (P ⊃ R)))] [12),Df ‘TAUT’]

T5-08 TAUT[(∀y)((∀x)Px ⊃ Py)]
<u>Proof</u>: 1) (∀x)Px CONT (∀x)Px [T1-11,U-SUB,DR1-11]
2) (∀x)Px CONT (∀y)Py [1),Alph.Var.]
3) INC[(∀x)Px & ~(∀y)Py] [2)Df ‘INC’(i)]
4) INC[(∀x)Px & (∃y)~Py] [3),Q-Exch]
5) [(∃y)((∀x)Px & ~Py) SYN ((∀x)Px & (∃y)~Py)] [T3-19,U-SUB(Q<1>/~P<1>,P/(∀x)Px)]
6) INC[(∃y)((∀x)Px & ~Py)] [4),5),SynSUB]
7) INC[~~(∃y)((∀x)Px & ~Py)] [6),DN]
8) TAUT[~(∃y)((∀x)Px & ~Py)] [7),Df ‘TAUT’]
9) TAUT[(∀y)~((∀x)Px & ~Py)] [8),Q-Exch]
10) TAUT[(∀y)((∀x)Px ⊃ Py)] [9),Df ‘⊃’]

T5-09. TAUT[(∀y)((∀x)Pxy ⊃ Pyy)] (Rosser’s Axiom 6)
<u>Proof</u>: 1) INC[((∀x)Px & ~Pa)] [T5-333a]
2) INC[((∀x)Pxa & ~Paa)] [1),U-SUB(‘P<1,a>’ for ‘P<1>’]
3) INC[(∃y)((∀x)Pxy & ~Pyy)] [2), MT5-37,MP]
4) INC[~(∀y)~((∀x)Pxy & ~Pyy)] [3),Q-Exch]
5) INC[~(∀y)((∀x)Pxy ⊃ Pyy)] [4),Df ‘⊃’]
6) TAUT[(∀y)((∀x)Pxy ⊃ Pyy)] [5),Df ‘TAUT’]

Finally, there are some TAUT-theorems of M-logic which can not be derived directly from **contradictory-free** CONT-theorems of the previous chapters because the antecedent requires the presence of contradictories. Such is the case if the antecedent is a truth-functional biconditional. The only prototype of a metatheorem in Chapter 2 of Quine's *Mathematical Logic* with this characteristic is ML *116, [(∀x)(Px ≡ Qx) ⊃ ((∀x)Px ≡ (∀x)Qx))], which is proved below. The 'CONT'-for-'⊃' analogue of this is '[(∀x)(Px ≡ Qx) CONT ((∀x)Px ≡ (∀x)Qx)))]'. This can not be derived without allowing contradictories; for '(P≡ Q)' SYN '((P ⊃ Q) & (Q ⊃ P))' which is SYN to '((∼P v Q) & (∼Q v P))', in which the contradictories, 'P' and '∼P', and 'Q' and '∼Q' all occur. Therefore this CONT-Theorem can not be derived starting frcm any contradictory-free CONT-theorems derived in the first four chapters using only U-SUBab .

In this chapter unrestricted U-SUB is not only allowed, but is constantly **required** to get INC-theorems and TAUT-theorems. The definition of 'Inc' is a way of introducing contractiories into wffs. The proof of this prototype of ML*116 is derived from a CONT-theorem, using unrestricted U-SUB in several steps, as follows:

T5-10. TAUT[(∀x)(Px ≡ Qx) ⊃ ((∀x)Px ≡ (∀x)Qx))]
Proof: 1) [(∀x)(Sx & Rx) SYN ((∀x)Sx & (∀x)Rx))] [T3-13,U-SUBab (P<1>/S<1>, Q<1>/R<1>)]
2) [(∀x)((∼Px v Qx) & Rx) SYN ((∀x)(∼Px v Qx) & (∀x)Rx))] [1),U-SUBab (S<1>/∼P<1> v Q<1>)]
3) [(∀x)((∼Px v Qx) & (Rx v Sx)) SYN ((∀x)(∼Px v Qx) & (∀x)(Rx v Sx)] [2),U-SUBab]
4) [(∀x)((∼Px v Qx) & (∼Qx v Sx)) SYN ((∀x)(∼Px v Qx) & (∀x)(∼Qx v Sx)] [3), U-SUB(R<1>/∼Q<1>]
5) [(∀x)((∼Px v Qx) & (∼Qx v Px)) SYN ((∀x)(∼Px v Qx) & (∀x)(∼Qx v Px)] [4), U-SUB(S<1>/∼P<1>]
6) [(∀x)((Px ⊃ Qx) & (Qx ⊃ Px)) SYN ((∀x)(Px ⊃ Qx) & (∀x)(Qx ⊃ Px)] [T4-31,SynSUB(4 times)]
7) [(∀x)(Px ≡ Qx) SYN ((∀x)(Px ⊃ Qx) & (∀x)(Qx ⊃ Px)] [4),Df '≡',SynSUB]
8) [(∀x)(Px ≡ Qx) CONT (∀x)(Px ⊃ Qx)] [7), Df 'CONT']
9) [(∀x)(Px ≡ Qx) CONT (∀x)(Qx ⊃ Px)] [7), Df 'CONT']
10) [(∀x)(Px ⊃ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)] [T4-37]
11) [(∀x)(Qx ⊃ Px) CONT ((∀x)Qx ⊃ (∀x) Px)] [T4-37,U-SUBab]
12) [(∀x)(Px ≡ Qx) CONT ((∀x)Px ⊃ (∀x)Qx)] [6),8),DR1-19]
13) [(∀x)(Px ≡ Qx) CONT ((∀x)Qx ⊃ (∀x) Px)] [7),9),DR1-19]
14) [(∀x)(Px ≡ Qx) CONT (((∀x)Px ⊃ (∀x)Qx) & ((∀x)Qx ⊃ (∀x) Px) [10),11),DR1-15]
15) [(∀x)(Px ≡ Qx) CONT ((∀x)Px ≡ (∀x)Qx)] [2),Df '≡',U-SUB, SynSUB]
16) TAUT [(∀x)(Px ≡ Qx) ⊃ ((∀x)Px ≡ (∀x)Qx)] [13) DR5-5a′]

The definition of 'wff' (well-formed formulae) places no limit on the inclusion of contradictories as components. Among the wffs, contradictory components occur in all TAUT- and INC-theorems, and in all theorems of M-logic. Unrestricted U-SUB is essential to a complete system of INC- and TAUT-theorems, and to a complete set of theorems of M-logic. However, there is also an interest in distinguishing those wffs which are *prima facie* contradictory-free from those that are not, and an interest in distinquishing wffs which are *prima-facie* not-logically inconsistent from those that are INC. In Chapters 6, 8 and 9, we shall find avoidance of contradictory components by means of U-SUBab an important means of preserving the properties of consistency and validity. But for our present task, exploring tautologies and inconsistencies, unrestricted U-SUB remains essential.

5.34 Completeness of A-logic Re: Three Axiomatizations of M-logic

A-logic is sound and complete with respect to the theorems of M-logic. To prove this one must (i) show that the wffs of A-logic include all wffs necessary for M-logic and (ii) prove that among the set of wffs in any axiomatic system of M-logic, all and only those wffs which constitute the essential theorems of M-logic, will be tautologous wffs in the "TAUT-theorems" of A-logic.[8] For example, if and only if '⊢ [Pv ~ P]' appears in M-logic, '|= TAUT[Pv ~ P]' must be provable in A-logic; and since '(P&Q)' is not a theorem in M-logic, it must not be tautology in A-logic.

For a proof of completeness it is sufficient to take any system which has been proven a sound and complete axiomatization of M-logic, and prove that its axioms and rules of inference are derivable in A-logic. For soundness one must prove that no wffs of M-logic which are not theorems of M-logic would be TAUT-theorems of A-logic.

Actually, we will examine and compare three different systems of M-logic to make clear what the essential theorems are. TAUT-theorems and INC-theorems of A-logic will be derived as needed.

There are many different axiomatic systems for mathematical logic, from Frege's first formulation in 1879, through that of Russell and Whitehead's *Principia Mathematica* in 1910, Hilbert and Ackermann (1928), Godel (1930), Quine's *Mathematical Logic* (1940), and many others before and since. They differ in the symbols they use, in their choice of primitives and definitions of logical constants, in their rules of formation for "well-formed formulae" of their system, as well as in their choices of axioms and rules of inference. These variations are usually chosen to sharpen unclear distinctions, or suit a particular interest or purpose.

Despite the seemingly enormous variety, there is a common core; a minimal set of theorems of the "first order predicate calculus", all of which must be shown to be provable, and none of which must be unprovable, if the axiomatization is to be acceptable as an axiomatization of Mathematical Logic. Given any two of these systems, this set of theorems is derivable in both of them, and despite all variations every theorem is translatable from one system to the other. The theorems of M-logic stand in a one-to-one relation to a sub-set of theorems in A-logic which have the form 'TAUT[P]', where P is a wff of M-logic built up from '&', 'v', ' ~ ' and quantifiers.

An interesting and significant facet of all of these formal systems is that each one could be coherently and consistently interpreted in English as systems for generating logical inconsistencies without changing a single symbol. See Section 5.5. After relating A-logic to the three axiomatizations discussed below, we will show how and why this is true.

The axiomatization of M-logic in Richmond Thomason's *Symbolic Logic* (1970) is closes t to A-logic in one respect. His system is unusual in having—in addition to individual variables and individual constants (which are used in many axiomatizations)—a third category of "individual parameters" which behave very much like the "argument-position-holders" in A-logic and serve some of the same purposes. This makes translation of principles from M-logic to A-logic much easier, particularly in quantification theory. We use this system first to prove the completeness of A-logic relative the theorems of M-logic. The use of "individual parameters" allows us to avoid the clauses concerning "free occurrences" of variables which abound in Quine's formalization.

At another extreme is the axiomatization of Quantification Theory in Quine's *Mathematical Logic,* which was designed, like *Principia Mathematica*, to derive mathematics from logic. It did so more

8. The term 'essential' is prefixed to allow exclusion of the wffs in some systems including Quine's and Rosser's, which have components, like vacuous quantifiers, which have no effect on truth or falsehood and can be added and or eliminated without altering any logical properties.

economically, more rigorously, and more accurately, than its predecessor. In this work, upper-case greek letters stand initially for statements, and later for matrices (propositional functions, “open sentences”) as well. These symbols do the job which P_1, P_2,…etc, do for A-logic as placeholders for predicates which are saturated (statements) or unsaturated (matrices). In *Mathematical Logic* Quine’s objective was to develop mathematics from set theory, following Russell and Whitehead. For this he needed just one actual primitive predicate, namely, the binary predicate ‘ϵ’ for ‘<1>is a member of<2>’. But for a general logic, which must deal with many different predicates, the logical form of different expressions is best clarified by having one set of letters to represent distinctions between different polyadic predicates, another set to represent single, fixed individual entities, to which predicates may be applied, and another set to mark places in a predicate that could be occupied by one or another designator of an individual entity.

Quine’s notation, sparse primitives, and choice and phrasing of axioms, make translation of many of his metatheorems into the present version of A-logic difficult. But the comparison of his system with others will help clarify what is essential and what is not in M-logic.

To ease the transition to Quine’s system we make use of a third system, from J.Barkley Rosser’s *Logic for Mathematicians* (1953), which is essentially the same as Quine’s but closer in notation, format and the choice of axioms, to A-logic.

5.341 M-logic’s “modus ponens”, A-logic’s TAUT-Det.

Quine, Rosser, and Thomason all use the same primitive rule of inference, so we begin with a derivation of this rule in A-logic, where it is called TAUT-detachment (“TAUT-Det”). The derivation is followed by a discussion of the relation of this rule to the concept of “following from”.

Thomason calls this primitive rule of inference, “modus ponens”, as do Quine and Rosser. Thomason describes it very simply as follows,

> “If in a proof steps A and (A ⊃ B) have occurred,
> then the result of writing down B as another step is still a proof,”[9]

Since each step in a proof is a theorem for Thomason, this is equivalent to “If [P] is a theorem and [P ⊃ Q] is a theorem then [Q] is a theorem,” which is essentially the same as Quine’s formulation.[10]

The corresponding principle in A-logic is gotten by replacing the predicate ‘<1>is a theorem’ by the predicate ‘<1>is logically tautologous’ or ‘TAUT<1>’. These two predicates are co-extensive with respect a certain class of wffs in M-logic since they are applied in the respective logics to all and only the same members of that class.[11] Thus the equivalent A-logic principle is: “If [P] is Tautologous and [P ⊃ Q] is Tautologous, then [Q] is Tautologous”. This will hereafter be called the Principle of Tautology Detachment, or TAUT-Det,

DR5-5f′. TAUT-Det. If TAUT[P] and TAUT[P ⊃ Q] then TAUT[Q].

This principle should be distinguished from TF-modus ponens, which says “if P is true and [P ⊃ Q] is true, then Q is true.” Being true and being tautologous are not the same. (See Section 7.42124, T7-46⊃, and T8-746⊃ for TF-modus ponens in A-logic)

---

9. Thomason, Richmond, *Symbolic Logic; An Introduction* (1970), MacMillan, London p. 4.
10. In *Mathematical Logic*, *104 is “If [P ⊃ Q] and P are theorems, so is Q.” in my notation.
11. Using our definition of ‘tautologous’.

To prove that TAUT-Det is reliable in A-logic—that we will always pass from the two Tautologies, TAUT[P] and TAUT[P ⊃ Q] to a Tautology [Q]—we prove first the principle from which it is derived; the Rule of Inconsistency-detachment:

DR5-5f. INC-Det. If INC[P] and INC[~P&Q] then INC[Q].[12]
Proof: By hypothesis INC[P] and INC[~P & Q].
By rules of the ordinary M-logic truth-tables, both [P] and [~P & Q] have all F's in the final column of their truth-table.
Since [P]is INC it has all F's and [~P] will have all T's in its truth-table.
Thus, if Q has a T in any row of its truth-table, then [~P & Q] would have a T in that row of its truth-table.
But [~P & Q] is INC and thus has no T's in its truth-table by Hypothesis.
Hence Q has no T's. Hence Q has all F's.
Hence, since every expression that has all F's is INC, |= Inc[Q], i.e., INC[Q].

Three versions of the usual TAUT-detachment rule can be derived in A-logic from the primary principle, INC-Det:

Proofs: 1)	If INC[P] and INC[~P & Q] then INC[Q]	[INC-Det]
2)	If INC[~P] and INC[~~P & ~Q] then INC[~Q]	[1),U-SUB,P/~P,Q/~Q]
3)	If INC[~P] and INC[~~(~~P & ~Q)] then INC[~Q]	[2),Ax.4-05]
4)	If INC[~P] and INC[~(~P v Q)] then INC[~Q]	[3),Df 'v']
5)	If INC[~P] and INC[~~(P & ~Q)] then INC[~Q]	[3),Ax.4-01,R1b]
6)	If INC[~P] and INC[~(P ⊃ Q)] then INC[~Q]	[5),Df '⊃']
7)	**If TAUT[P] and TAUT[~P v Q] then TAUT[Q]**	[4),Df 'TAUT'(3 times]
8)	**If TAUT[P] and TAUT[~(P &~Q] then TAUT[Q]**	[5),Df 'TAUT'(3 times)]
9)	**If TAUT[P] and TAUT[P ⊃ Q] then TAUT[Q]**	[6),Df 'TAUT'(3 times]

Other versions of TAUT-Det, such as

If TAUT[~Q] and TAUT[(P ⊃ Q)] then TAUT[~P]
If TAUT[~P] and TAUT[(P v Q)] then TAUT[Q]
If TAUT[P] and TAUT[~(P&Q)] then TAUT[~Q]

are easily derived using U-SUB and/or re-lettering.

The use of bivalent truth-tables in the proof of INC-Det is not essential; truth-tables were used above to facilitate the passage from familiar procedures of M-logic to those of A-logic. POS-NEG tables work for predicates as well as for statements. In unquantified logic, replacing 'F' by 'POS' and 'T' by 'NEG' throughout, the proof of INC-Det above leads to the same result:

12. To help to dispel the illusory resemblance of this rule to the rule of ***modus ponens*** in traditional logic, I take this principle, from which alternative syllogisms and TF-modus ponens are alike derivable in analytic logic, as the more informative principle.

<u>INC-Det</u>. If INC[P] and INC[~P&Q] then INC[Q].

<u>Proof</u>: By hypothesis INC[P] and INC[~P & Q].
Hence both [P] and [~P & Q] have all POS's in their POS/NEG-table.
But since [P] is INC it has all POS's and [~P] will have all NEG's in its POS-NEG table.
Thus, if Q has NEG in the final column of any row in its POS-NEG-table,
then [~P & Q] would have a NEG in that row of its truth table.
But [~P & Q] is INC and thus has no NEG's in its POS-NEG-table by Hypothesis.
Hence Q has no NEG's. Hence Q has all POS's in its POS-NEG-table.
But, every expression that has all POS's in their POS-NEG-table is INC,
Hence, INC[Q].

Or, a parallel proof can be given to establish TAUT-Det directly.

<u>TAUT-Det</u>. If TAUT[P] and TAUT[~PvQ] then TAUT[Q].

<u>Proof</u>: By hypothesis TAUT[P] and TAUT[~P v Q].
Hence both [P] and [~P v Q] have only NEG's in their POS-NEG-tables.
Since [P] has all NEG's, [~P] will have all POS's in its POS-NEG table.
Thus, if Q has a POS in the final column of any row in its POS-NEG-table,
then [~P v Q] would have a POS in that row of its POS-NEG-table.
But by Hypothesis [~P v Q] has no POS in its POS-NEG-table.
Hence Q has no POS's. Hence Q has all NEG's in its POS-NEG-table.
But, every expression that has all NEG's in its POS-NEG-table is TAUT,
Hence, TAUT[Q].

It does not follow from these proofs that the principle of inference, which Quine, Rosser, Thomason (and many other logicians) take as a sole, sufficient, primitive vehicle in all derivations, is a fundamental principle of logical inference. In A-logic it is not; it is a subordinate, secondary principle of logic for preserving tautology, not a general rule of logical inference.

Fundamental rules of logical inference, must be such that an instantiation of the conclusion always "follows logically" from the premiss. The rule of TAUT-Detachment violates this by sanctioning non sequiturs such as "((P&Q) ⊃ Q) follows logically from (R ⊃ R)". Neither TAUT-Det nor INC-Det are fundamental principles of logical inference. Neither establishes that the <u>wff</u> which is asserted to be tautologous (or inconsistent) in the conclusion, "follows logically" from the <u>wffs</u> asserted to be tautologous (or inconsistent) in the premisses. All they establish is that if the premisses in these rules have the property of being tautologous (inconsistent), its conclusion must be tautologous (inconsistent).

TAUT-Det is a useful, reliable procedure for establishing that one wff is logically tautologous when given the fact that two prior wffs are tautologous. When stated carefully, as Quine and Thomason (and many others) do, it assures the reliability of a very useful short-cut for establishing theorems of M-logic, i.e., tautologies. INC-Det is similarly a useful and reliable short-cut for proving a logical inconsistency from two other inconsistencies.

Consider the following analogy: In arithmetic the rule "If A is Odd and (A times C) is Odd, then C is Odd" is valid, but it is not a fundamental principle of arithmetic. It moves from the property of one number and the property of a product of that number with another number, to a property of the other number. Being odd and being even, are mathematical properties each of which belong to some but not all natural numbers. The fundamental principles of elementary arithmetic are principles about equalities (='s of ='s are =, x=x, (x+y)=(y+x), etc.)—not principles about oddness or eveness. The relation of numerical equality is basic and essential to arithmetic.

Analogously, in formal logic the rule "If A is Tautologous and (A ⊃ C) is tautologous, then C is tautologous", is valid, but is not a fundamental principle of logic. It moves from the property of one expression and the property of a certain compound of that expression with second expression, to a property of the second expression. In this case, the property which is tranferred is tautology. Being tautologous and being inconsistent, are logical properties each of which belong to some but not all expressions. The fundamental principles of formal logic are principles grounded in synonymy [If (P CONT Q & Q CONT R) then P CONT R], [P SYN P], [(P v Q) SYN (Q v P)], etc.)—not principles about tautology or inconsistency. The relation of logical synonymy is basic and essential to logic. Being tautologous is an important property of some expressions, but does not reduce to the fundamental logical relation.

Often the rule of TF-Modus Ponens, is expressed loosely as "If P and (P ⊃ Q), then Q". At one point Rosser, for example, writes:

> '*Modus Ponens*: If P and (P ⊃ Q), then Q, where P and Q are statements'.[13]

There are two things wrong with this looser formulation. The first is that it is not Q which follows from something, but the tautologousness (or theoremhood) of Q, which follows from the tautologousness (or theoremhood) of P and of (P ⊃ Q). Similarly, we may have the principle

"If P is true and (P ⊃ Q) is true, then Q is true",[14] but if so, it is not Q which follows from something but the <u>truth</u> of Q, which follows from the truth of P and the truth of (P ⊃ Q). In A-logic such entailments depend on the meanings of the predicates '<1>is taut', or '<1>is true'; lacking such predicates, the conjunction of P and (~P v Q) does not logically contain Q, since Q does not occur as a conjunct in the MOCNF of (P & (P ⊃ Q)) or any of its logical synonyms.

The second thing wrong lies in interpreting '(P ⊃ Q)' as "if P then Q". In M-logic '(P ⊃ Q)' is read as "If P then Q", which means the same as "Either not-P or Q" and "not both P and not-Q". This meaning gets us into a lot of anomalies with respect to ordinary usage ("paradoxes of strict and material implication"), and a lot of failures in connecting this meaning for "if P then Q" with the use of conditionals in common sense and science.

It does not follow from comments above that there is anything wrong with the intuitive traditional Modus Ponens before mathematical logic interpreted it. That meant, in ordinary language, "Q <u>follows logically from</u> (P and if P then Q)". When we get to Chapter 6, this principle is adopted as an axiom of A-logic but with a meaning for "if…then" which avoids the difficulties mentioned. The conjunction of P<1> with this other meaning of 'if P<1> then Q<2>', logically contains the meaning of Q<2>.

Finally, A-logic does not need TAUT-Det or TF-Modus Ponens in order to derive the theorems of M-logic from axioms of Thomason, Rosser or Quine taken as tautologies of A-logic. All such axioms are derived as theorems in A-logic based on its own axioms and definitions, using only the rules of SynSUB and U-SUB. With these rules there are no steps or theorems which lead to non sequiturs, and the anomalies disappear. Q does not follow logically from P conjoined with (~PvQ), or ~(P&~Q), or (P ⊃ Q), but if P and (~PvQ) are both tautologous, then Q must be tautologous.

5.342 Derivation of Thomason's Axiomatization of M-logic

Thomason's axiomatization consists of six axioms and the primitive rule of infererence—the truth-functional version of "Modus Ponens". If the latter rule is a derived rule in A-logic, and the axioms are all

---

13. Opus Cit., p. 212.
14. See Chapter 7, T7-46 ⊃.[(TP & T(P ⊃ Q)) Cont TQ], and Chapter 8, T8-746a.

derivable in A-logic, then since Thomason proved his system to be complete re: the semantics of M-logic, A-logic will have been proven complete in the same sense.

The proofs of Thomason's six axioms AS1, AS2, AS3, AS4, AS5, and AS6 were given above in the form, respectively, of TAUT-theorems of A-logic T5-03, T5-07, T5-04, T5-438c, T5-437c and T5-333c.

Thomason's Axioms:	A-logic's TAUT-theorems:
AS1. $[P \supset (Q \supset P)])$	T5-03 TAUT$[P \supset (Q \supset P)]$
AS2. $[(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R)]$	T5-07 TAUT$[(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R)]$
AS3. $[(\sim P \supset \sim Q) \supset (Q \supset P)]$	T5-04 TAUT$[(\sim P \supset \sim Q) \supset (Q \supset P)]$
AS4. $[(\forall x)(P \supset Qx) \supset (P \supset (\forall x)Qx)]$	T5-438c TAUT$[(\forall x)(P \supset Qx) \supset (P \supset (\forall x)Qx)]$
AS5. $[(\forall x)(Px \supset Qx) \supset ((\forall x)Px \supset (\forall x)Qx))]$	T5-437c TAUT$[(\forall x)(Px \supset Qx) \supset ((\forall x)Px \supset (\forall x)Qx))]$
AS6. $[(\forall x)Px \supset Pa]$	T5-333c TAUT$[(\forall x)Px \supset Pa]$

Thomason proved the completeness of his system with these axioms and the rule of inference TF-Modus Ponens and U-SUB. Since all of Thomason's axioms and his primitive rule of inference are derivable in A-logic, his proof may also be used to prove that A-logic is complete with reference to the theorems of M-logic in A-logic. That is, for every $\vdash$ [A] in Thomason's logic, there is a $\models$ TAUT[A] in A-logic, and no wff in Thomason's first order logic is TAUT in A-logic unless it is a theorem in Thomason's logic.

The completeness that we have proved is limited to the set of theorems in M-logic. Many derived rules of inference called valid in M-logic, are not valid rules of inference in A-logic. This is because "valid" is defined differently in A-logic. Its definition is A-logic avoids the non- sequiturs sanctioned in M-logic. In A-logic it does not follow from $\models$ TAUT$[P \supset Q]$ that 'If P then Q' is valid or that 'P therefore Q' is valid. Validity in A-logic is very different from tautology. All that a TAUT-theorem (or any theorem of M-logic) establishes is that a certain wff has the logical property of being the negation of an inconsistency.

This proof of completeness, by deriving Thomason's primitive rule of inference and his axioms, together with the earlier proof of soundness, is sufficient to warrant the claim that A-logic contains M-logic so far as M-logic is conceived as the set of its theorems.

How and why Thomason's system, with its individual constants and "individual parameters" is closer to A-logic, than other formulations of M-logic will become apparent from consideration of the ways in which it and A-logic differ from Quine's axiomatization.

5.343 Completeness re: Axiomatizations of Quine and Rosser

We turn to Quine's axiomatization of logic with quantification theory in Chapters 1 and 2 of his *Mathematical Logic*. We will also refer to Rosser's system which is basically like Quine's, but in several respects is more like A-logic, thus partially bridging the differences between the two. We will show how A-logic can account for all of Quine's 56 metatheorems in the two chapters cited, despite the great differences in modes of presentation.

Quine's axiomatization differs from Thomason's system and from A-logic, in several striking ways, some trivial and some not.

1) Quine uses upper case Greek letters, for statements or matrices of quantificational wffs, instead of P_1, P_2, etc. for Predicates (saturated or unsaturated).

2) Quine has only one sentence connective '↓' instead of '~' and '⊃', or '~' and '&' or '~ and 'v'.
3) Quine gives no axioms for sentential logic, relying instead upon truth-tables to distinguish tautologous wffs.
4) Quine presents his principles of logic as "metatheorems", (elsewhere "theorem schemata") rather than "theorems", and uses '⊢ P' to mean "the closure of P is a theorem" instead of merely 'P is provable in M-logic',
5) Quine and Rosser allow vacuous quantifiers in thier wffs and theorems.
6) Quine and Rosser do not include individual constants as logical symbols.
7) Quine and Rosser have no individual parameters.
8) Quine and Rosser treat sentential functions which are neither true nor false, as if they were sentences having truth-tables with T's and F's.[15]
9) For Thomason 'Px' is not a wff; x must be bound to a quantifier.

Some of these differences are trivial. Some represent choices between equally good alternatives. Some represent expedient devices which introduce inessential features. The first difference - Greek letters instead of 'P_1', 'P_2', etc., will be ignored. A-logic, like Rosser and Thomason, uses the latter, and in describing Quine's system we replace his greek letters with them.

There are subtleties to be discussed here. Quine used the first four upper-case greek letters to "refer to" or stand for ordinary statements or sentential functions (i.e., "open sentences" or "matrices") or formulae for such expressions. A-logic, like Thomason, does not treat "open sentences" as formulae. In place of free variables, Thomason's formulae have "individual parameters" and A-logic has argument-position-holders. Despite this difference, theorems of A-logic, in which all variables must be bound, can be correlated with Quine's theorems, in which all variables must be bound. How this is done is shown below.

5.3431 Derivations from Chapter 3 and 4

As a starter we point out that from the CONT-theorems and SYN-theorems in Chapter 3 and 4, it is possible to derive prototypes for 35 of the 56 theorems covered in Quine's Chapters 1 and 2, by using

DR5-5a. If [P CONT Q], then INC[P & ~Q]
DR5-5a′. If [P CONT Q], then TAUT[P ⊃ Q]
DR5-5b. If [P SYN Q], then INC[P & ~Q]
DR5-5b′. If [P SYN Q], then TAUT[P ⊃ Q]
DR5-5d. If [P SYN Q], then INC[(P&~Q)&(Q&~P)]
DR5-5d′. If [P SYN Q], then TAUT[P ≡ Q]

T5-333a. INC[(∀x)Px & ~Pa]
Proof: 1) [(∀x)Px CONT Pa] [T3-33]
2) INC[(∀x)Px & ~Pa] [DR5-5a]

15. Quine calls sentential functions 'matrices' in ML, and 'open sentences' in *Methods of Logic* (cf., 2nd Ed, p 90; 3rd Ed, p 121; 4th Ed, p.134).

T5-333c. TAUT[(∀x)Px ⊃ Pa]

<u>Proof</u>: 1) [(∀x)Px CONT Pa] [T3-33]

2) TAUT[(∀x)Px ⊃ Pa] [DR5-5a′]

The negation-free wffs of Chapter 3 included 22 AL-analogues which convert into TAUT-theorems and with negation added in Chapter 4, 13 more convert into TAUT-theorems including one for Quine's Axiom Schema *101. In addition the proof of TAUT-detachment in Section 5.341, establishes Quine's TF-modus Ponens, *104 "If P and (P ⊃ Q) are theorems, so is Q."

This leaves 20 of Quine's metatheorems not covered. The reason they are not covered lies in the differences 4) through 8) between Quine's system and A-logic, and most of these are rooted in axiom schemata *102 and *103. Before tackling those axioms and showing how they are responsible for the 20 uncovered theorems, let us list the 35 prototypes already at hand. The first 33 prototypical TAUT-theorems are derived by DR5-5a′ and DR5-5d′ as follows:

Quine's "Metatheorems"

*101	T5-437c.	TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)]	[T4-37,DR5-5a′]
*116	T5-510.	TAUT[(∀x)(Px ≡ Qx) ⊃ ((∀x)Px ≡ (∀x)Qx))]	[T5-10,DR5-5a′]
*119	T5-315c	TAUT[(∀x)(∀y)Pxy ⊃ (∀y)(∀x)Pxy]	[T3-15,DR5-5a′]
*130	T5-424c.	TAUT[~(∀x)Px ≡ (∃x)~Px]	[T4-24,DR1,DR5-5d′]
*131	T5-425c.	TAUT[~(∃x)Px ≡ (∀x)~Px]	[T4-25,DR1,DR5-5d′]
*132	T5-426c.	TAUT[$\sim(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n> \equiv (\exists x_1)\ldots(Ea_n)\sim P<x_1,\ldots,x_n>$]	[T4-26,DR1,DR5-5d′]
*133	T5-427c.	TAUT[$\sim(\exists x_1)\ldots(\exists x_n)P<x_1,\ldots,x_n> \equiv (\forall x_1)\ldots(\forall x_n)\sim P<x_1,\ldots,x_n>$]	[T4-27,DR1,DR5-5d′]
*136	T5-336c.	TAUT[(∀x)Px ⊃ (∃x)Px]	[T3-36,DR5-5a′]
*138	T5-316c.	TAUT[(∃x)(∃y)Pxy ≡ (∃y)(∃x)Pxy]	[T3-16,DR5-5d′]
*139	T5-337c.	TAUT[(∃y)(∀x)Pxy ⊃ (∀x)(∃y)Pxy]	[T3-37,DR5-5a′]
*140	T5-313c.	TAUT[(∀x)(Px & Qx) ≡ ((∀x)Px & (∀x)Qx]	[T3-13,DR5-5d′]
*141	T5-314c.	TAUT[(∃x)(Px v Qx) ≡ ((∃x)Px v (∃x)Qx)]	[T3-14,DR5-5d′]
*142	T5-433c.	TAUT[(∃x)(Px ⊃ Qx) ≡ ((∀x)Px ⊃ (∃x)Qx)]	[T4-33,DR5-5d′]
*143	T5-338c.	TAUT[((∀x)Px v (∀x)Qx) ⊃ (∀x)(Px v Qx)]	[T3-38,DR5-5a′]
*144	T5-339c.	TAUT[(∀x)(Px v Qx) ⊃ ((∃x)Px v (∀x)Qx))]	[T3-39,DR5-5a′]
*145	T5-340c.	TAUT[(∀x)(Px v Qx) ⊃ ((∀x)Px v (∃x)Qx))]	[T3-40,DR5-5a′]
*146	T5-341c.	TAUT[((∀x)Px v (∃x)Qx) ⊃ (∃x)(Px v Qx)]	[T3-41,DR5-5a′]
*147	T5-342c.	TAUT[((∃x)Px v (∀x)Qx) ⊃ (∃x)(Px v Qx)]	[T3-42,DR5-5a′]
*148	T5-436c.	TAUT[((∃x)Px ⊃ (∀x)Qx) ⊃ (∀x)(Px ⊃ Qx)]	[T4-36,DR5-5a′]
*149	T5-439c.	TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∃x)Px ⊃ (∃x)Qx)]	[T4-39,DR5-5a′]
*150	T5-440c.	TAUT[((∃x)Px ⊃ (∃x)Qx) ⊃ (∃x)(Px ⊃ Qx)]	[T4-40,DR5-5a′]
*151	T5-441c.	TAUT[((∀x)Px ⊃ (∀x)Qx) ⊃ (∃x)(Px ⊃ Qx)]	[T4-41,DR5-5a′]
*152	T5-343c.	TAUT[(∀x)(Px & Qx) ⊃ ((∃x)Px & (∀x)Qx)]	[T3-43,DR5-5a′]
*153	T5-344c.	TAUT[(∀x)(Px & Qx) ⊃ ((∀x)Px & (∃x)Qx)]	[T3-44,DR5-5a′]
*154	T5-345c.	TAUT[((∀x)Px & (∃x)Qx) ⊃ (∃x)(Px & Qx)]	[T3-45,DR5-5a′]
*155	T5-346c.	TAUT[((∃x)Px & (∀x)Qx) ⊃ (∃x)(Px & Qx)]	[T3-46,DR5-5a′]
*156	T5-347c.	TAUT[(∃x)(Px & Qx) ⊃ ((∃x)Px & (∃x)Qx)]	[T3-47,DR5-5a′]
*157	T5-317c.	TAUT[(∀x)(P&Qx) ≡ (P&(∀x)Qx)]	[T3-17,DR5-5d′]
*158	T5-319c.	TAUT[(∃x)(P&Qx) ≡ (P&(∃x)Qx)]	[T3-19,DR5-5d′]
*159	T5-320c.	TAUT[(∀x)(PvQx) ≡ (Pv(∀x)Qx)]	[T3-20,DR5-5d′]

*160	T5-318c.	TAUT[(∃x)(PvQx) ≡ (Pv(∃x)Qx)]	[T3-18,DR5-5d′]
*161	T5-435c.	TAUT[(∀x)(Px ⊃ Q) ≡ ((∃x)Px ⊃ Q)]	[T4-35,DR5-5d′]
*162	T5-434c.	TAUT[(∃x)(Px ⊃ Q) ≡ ((∀x)Px ⊃ Q)]	[T4-34,DR5-5d′]

The Rule of Alphabetic Variance covers the territory covered by Quine's *170 and *171. If P and Q are alphabetic Variants, then [P SYN Q] and thus TAUT[P ≡ Q] by DR5-5d′.

All essential theorems that Quine can get with the correlated metatheorems, can be gotten in A-logic from the 33 TAUT-theorems above, together with alphabetic variance, relettering rules, and Rule DR5-20 for uniform substitution. This statement pre-supposes several results below, including (i) that the presence of vacuous quantifiers in Quine's theorems is inessential, (ii) that every essential theorem Quine covers with the phrase 'The closure of … is a theorem', can be introduced in A-logic by DR5-20, U-SUB, and (iii) that conditional phrases, like 'If x is not free in P,…' for T5-317 to T5-426 are unnecessary in A-logic due to constraints in substitution rules.

Thus with the 33 TAUT-theorems above, and the Rule of Alphabetic variance, etc., we have provided prototypes for 35 of Quine's 56 metatheorems. In addition the proof of TAUT-Det above, accounts for *104, leaving the following 20 of Quine's metatheorems to be accounted for:

9 Rules of Inference: *100,*111,*112,*113,*115,*117,*123,*124,*163
3 Theorem Schemata: *110,*114,*135,
8 Cond.Theorem Schemata: *102,*103,*118,*120,*121,*122,*134,*137,

The difficulties in translating these into A-logic's language is basically due to three of the five Axioms of Quantification—*100, *102 and *103—which are not yet covered. *101 and *104 have been dealt with. The following are prototypes for Quine's axiom schemata.

*100 If P is tautologous, ⊢ P.
*101 ⊢ [(∀x)(P ⊃ Q) ⊃ ((∀x)P ⊃ (∀x)Q))]
*102 If x is not free in P, ⊢ (Vv_1)P.
*103 If Q is like P except for containing free occurrences of
y where P contains free occurrences of x, ⊢ [(∀x)P ⊃ Q].
*104 If [P ⊃ Q] and P are theorems, so is Q.

The second and third differences listed above between A-logic and Quine, namely, 2) Quine has only one sentence connective, '↓', instead of '∼' and '⊃', or '∼' and '&', and 3) Quine gives no axioms for sentential logic, only tautologousness of wffs, are quickly resolved by considering Rosser's System. Rosser lists his six axioms as :

Ax.1. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)$[P ⊃ (P&P)]
Ax.2. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)$[(P&Q) ⊃ P)]
Ax.3. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)$[(P ⊃ Q) ⊃ (∼(Q&R) ⊃ ∼(R&P))]
Ax.4. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)$[(∀x)(P ⊃ Q) ⊃ (∀x)P ⊃ (∀x)Q)]
Ax.5. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)$[P ⊃ (∀x)P], if x does not occur free in P.
Ax 6. $(\forall x_1)(\forall x_2)\ldots(\forall x_n)$[(∀x)R<x,y> ⊃ R<y,y>]

Instead of letting '⊢ (…)' stand for 'The closure (…) Is a theorem', Rosser lets '⊢ (…)' stand for '(…) Is a theorem', and prefixes '$(\forall x_1)(\forall x_2) \ldots (\forall x_n)$' to each of his axioms on the understanding the 'P', 'Q' ad

'R' may represent "open sentences". For sentences which have no 'free variables'—i.e., for the sentential calculus—the quantifiers may be removed, yielding the three axioms,

Ax.1. $[P \supset (P\&P)]$
Ax.2. $[(P\&Q) \supset P)]$
Ax.3. $[(P \supset Q) \supset (\sim(Q\&R) \supset \sim(R\&P))]$

for the complete sentential calculus. Rosser, like A-logic, has '&' and '∼' as his primitive sentence connectives, and he provides a set of axioms which yield just the set of wffs which are the tautologous wffs in M-logic, referred to in Quine's *100. These, with TF-modus ponens, yield all and only those theorems which Quine calls 'tautologies' and which, for Quine may be composed of either open sentences or statements. (As was mentioned earlier Quine's uses 'tautology' in a narrower sense than A-logic, confining it to what can be determined the truth-table method. In A-logic, it covers all quantificational theorems of M-logic, since it means the denial of a logically inconsistent wff.)

5.3432 Quine's Axiom Schema *100

Quine's metatheorem *100 says "If ϕ is tautologous, the closure of ϕ is a theorem." Also,Quine chose the symbol '↓' for joint denial as his sole primitive sentence connective, and had no axioms for what is generally called sentential logic, using truth-table tests for "tautology" instead.[16]

Quine's choice of primitives and his use of tautology to characterize theorems of sentential logic, are easily reconciled with A-logic which, like Thomason and Rosser, uses other primitives and provides axioms for sentential logic. The expression 'the closure of ϕ' presents other problems for reconciliation.

• 5.34321 Primitives and Tautologies

a) Regarding the choice of primitive constants. As is well known, sentential logic can be axiomatized on bases with different sets of logical constants: 'not' and 'if…then' (in Frege, Church, Thomason), 'and' and 'or' (in Russell and Whitehead, PM, first Ed.), the not-both stroke (Nicod, Russell & Whitehead PM, 1925 Ed), the neither-nor stroke (Peirce, Sheffer, Quine), and 'and' and 'not' (Rosser, A-logic). With different axioms in each of these systems, the definitions of the non-primitive constants end up bestowing the same truth-functional meaning in every system on each logical constant.

In the present version of A-logic, as in Rosser's system, 'and' and 'not' are taken as the initial primitive constants and the other logical constants are introduced by definition. This choice was made because the truth-functional meanings of 'and' and 'or' in M-logic are more closely connected to ordinary usage than the M-logic interpretations of 'or' and 'if…then'.[17]

b) Regarding Quine's reliance on tautology, vs. an axiom set for tautologous wffs; among systems of M-logic it does not matter which of these alternatives are chosen, provided the axioms and rules of inference yield as theorems just that set of wffs which have only T's in the final columns of their truth-tables. With TF-"modus Ponens", Rosser's axioms (like Thomason's) meets this requirement. Quine, in *Mathematical Logic*, simply skips the axiomatization of sentence logic and goes directly to truth-tables to

16. Section §10, *Mathematical Logic*.
17. A-logic, like M-logic, could start off with 'or' and 'not', as primitive notions, with a stroke function, or with '⊃' and 'not' (as Thomason does). With the usual definitions these would yield the same formal results, but would begin with departures from ordinary usage, rather than leading to such departures from more familiar usages. Starting with '⊃' as primitive is especially misleading, as it confuses the familiar "if…then" with a distinctly different concept.

establish tautologies. So Rosser's set of axioms with TF- "modus ponens", is complete with respect to Quine's set of tautologous wffs.

In *Mathematical Logic* Quine uses the upper case Greek letters phi, psi and chi with their accented and SUBscripted variants, and Rosser uses 'P_1', 'P_2', 'P^3',... etc., more or less as "propositional variables", "sentential variables", "sentence letters" are used in other systems. Both Quine and Rosser use such symbols to stand not only for statements, but also for expressions containing unbound individual variables, e.g., 'x is red' or '$(\forall x)x > y$' (variously called "matrices", "propositional functions", "sentential functions", and by Quine, in *Methods of Logic*, "open sentences").

In Rosser's system one can derive all theorems that Quine could derive from *100. Quine's *100 reads, "If P_1 is tautologous, the closure of P_1 is a theorem". All theorems derivable from this are derivable from (i) Rosser's first 3 axioms with TF-Modus Ponens (these axioms with TAUT-Det, have been proven to be complete with respect to the set of all wffs which are tautologous on Quine's definition) together with (ii) Rosser's "Generalization Principle", VI.4.1 "If P_1 is proved, one may infer $(x)P_1$" which, by successive steps with different variables, will yield the closure of P as a theorem.

Rosser's three axioms for sentential calculus were proved above, using only the rules and axioms of A-logic, as follows:[18]

T5-501c.	TAUT[$P \supset (P \& P)$]	(Rosser's Axiom 1)
T5-136c.	TAUT[$((P\&Q) \supset P)$	(Rosser's Axiom 2)
T5-05.	TAUT[$(P \supset Q) \supset ({\sim}(Q\&R) \supset {\sim}(R\&P))$]	(Rosser's Axiom 3)

The method common to those proofs is simple. To prove TAUT[P], 1) one begins with SYN-theorems, from which 2) one derives the appropriate CONTAINMENT theorem by Df 'CONT', from which 3) one derives an INCONSISTENCY theorem by Df'INC' clause (i), within which 4) one substitutes synonyms for synonyms until 5) one reaches INC[$\sim$P] at which point the final step yields TAUT[P] by Df 'TAUT'.

Since proofs can be given that all and only wffs derivable from Rosser's three axioms above, plus TAUT-Det, are tautologous by the truth-table method (or the POS-NEG-table method), this suffices to satisfy the antecedent of *100, i.e., "P is tautologous...".

The consequent in *100 "...then the closure of P is a theorem", is gotten by a prototype for Quine's *100, which Rosser calls the "Generalization Principle", i.e., and is expressed somewhat loosely as, "VI.4.1 If P_1 is proved, one may infer $(\forall x)P_1$". By prefixing '$(\forall x_1)(\forall x_2)\ldots(\forall x_n)$' to each of his six axioms, Rosser says in effect that the result of using the generalization principle on all 'open sentences'—i.e., the closure of the matrix—will be a theorem.

With Alphabetic Variance and A-logic's rules of U-SUB we can derive from $(\forall x)P_1$ all formulae gotten by U-SUB in "the closure of P_1 is a theorem". So we argue below. If this approach is established in A-logic, then every theorem that can be gotten by substitution in Quine's *100 can be gotten in A-logic using its versions of Rosser's axioms and A-logic's rules of inference alone. Mathematical Logic's version of "modus ponens", TAUT-Det, may be employed as an expedient, but neither it, nor any rule which presupposes it, is necessary to these derivations.

This is how Quine's *100 is satisfied in A-logic. By repetitive applications of the principle,

DR5-32'. If TAUT[$P\langle 1\rangle$] then TAUT[$(\forall x_1)Px_1$], followed by

$\models$ If TAUT[$(\forall x_1)P\langle x_1,2\rangle$] then TAUT[$\forall x_2)(\forall x_1)Px_1x_2$], etc., one can reach the more general principle, $\models$ If TAUT[$P\langle 1,\ldots,n\rangle$] then TAUT[$Vx_1),\ldots.(\forall x_n)Px_1\ldots x_n$] for any n.

18. See J. B. Rosser, *Logic for Mathematicians,* McGraw-Hill, 1953, p 55.

5.34322 Re: "the closure of P is a theorem"

Quine presents his principles of logic as "metatheorems" (elsewhere "theorem schemata") rather than "theorems", and uses '⊢ P' to mean "the closure of P is a theorem". Thomason, uses '⊢ P' for 'P is provable in M-logic', and 'X ⊢ P' for "P is deducible from X in M-logic". This is the usual practice, and '⊨ P' in A-logic means P is a provable theorem of A-logic in the usual sense.

Part of the problem in translating Quine's version to A-logic, has to do with his use of the symbols '⊢ P' for the concept, "the closure of P is a theorem." Under this concept two kinds of wffs can be substituted for 'P' in '⊢ P':

1) a wff with no free variables; i.e., a closed wff. E.g. '(∀x)Px'
or 2) a potential matrix or "open sentence" with variables, $v_1, \ldots v_k$ free in P, E.g., 'Qx' or '(∀y)(Pwy ⊃ Rxz)'.

In the first case, what is substituted for P is a closed wff. In the second case the closure of P refers to the wff formed by prefixing $(\forall v_1), \ldots (\forall v_k)$ to what is substituted for P. Thus, if 'P' is replaced by '(∀y)(**w** loves y ⊃ **x** hates **z**))', '⊢ P' is '**(∀z)(∀x)(∀w)**(∀y)(**w** loves y ⊃ **x** hates **z**))'.

In A-logic, and Thomason's M-logic, the class of wffs does not include any expressions of the second sort with free (unbound) variables. Thus such expressions cannot be substituted, by U-SUB, for P. Only the closed wffs in Quine's system are wffs in A-logic and Thomason. In place of Quine's "open sentences", A-logic has predicate schemata having argument-position-holders in place of free variables and Thomason has wffs which have "individual parameters" where Quine has free variables. Thus Quine's move to closure

from '(∀y)(**w** loves y ⊃ **x** hates **z**))' to '**(∀z)(∀x)(∀w)**(∀y)(**w** loves y ⊃ **x** hates **z**))'

becomes in Thomason, a move from a wff with individual parameters 'u_1', 'u_2', 'u_3' to a closed wff, e.g.,

from '(∀y)($\mathbf{u_1}$ loves y ⊃ $\mathbf{u_2}$ hates $\mathbf{u_3}$)') to '**(∀z)(∀x)(∀w)**(∀y)(**w** loves y ⊃ **x** hates **z**))'

and in A-logic, it becomes the universal generalization of a polyadic predicate over a domain:

from '(∀y)(**<1>** loves y ⊃ **<2>** hates **<3>**))' to '**(∀z)(∀x)(∀w)**(∀y)(**w** loves y ⊃ **x** hates **z**))'

Given any tautologous predicate with n distinct argument positions, P<1,2,…,n>, the conjunctive quantification of that predicate is gotten by n successive applications of DR5-32':

If TAUT[P<1>] then TAUT[(∀x)Px].

Thus the 33 prototypes listed above say considerably less than the theorem schemata in Quine which are analogous to them. For the latter are presented in the form '⊢ [P]', which means "the closure of [P] is a theorem" rather than "P is a theorem". Nevertheless, every particular theorem covered in one of Quine's metatheorems can be gotten as a TAUT-theorem in A-logic by a certain routine starting with the prototype. Let us take one example: Quine's *136, '⊢ [(∀x)ϕ ⊃ (∃x)ϕ]'. This is not strictly translatable as T5-336 "[(∀x)Px ⊃ (∃x)Px] is TAUT". A strict translation would be 'The closure of [(∀y)P ⊃ (∃y)P] is TAUT'. This might be conveyed in A-logic by (the device Rosser uses): TAUT $[(\forall x_1)\ldots(\forall x_n)((\forall y)(P(y,x,x_1\ldots x_n) \supset (\exists z)P(z,x_1\ldots x_n)$

However, from any prototypical theorem, P, of A-logic, such as

T5-333c TAUT[(∀x)Px ⊃ Pa], or T5-438c TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)], or
T5-437c TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))], or T5-336c TAUT[(∀x)Px ⊃ (∃x)Px],

there is a routine, using the rule of U-SUB$_{INC}$ (DR5-20) and the derived rules,

DR5-32'. If TAUT[P<1>] then TAUT[(∀z)Pz]
and DR5-34'. If TAUT[P<1>] then TAUT[(∃z)Pz],

which will prove any theorem with n variables that is covered by "the closure of P is a theorem". Consider how T5-336, TAUT[(∀x)Px ⊃ (∃x)Px], yields the tautologousness of its closure from

"If TAUT[~P<1>v P<1>] then TAUT[(∀w)(∀z)...(∀x_n)((∀x)P<x,z,...,x_n>⊃(∃y)P<y,z,...,x_n>)]":

1) TAUT[~P<1>vP<1>]		
2) TAUT[(∃x)(~Px v Px)]		[1),DR5-34']
3) TAUT[(∃x)~Px v (∃x)Px)]		[2),T3-14, SynSUB]
4) TAUT[~(∀x)Px v (∃x)Px)]		[3), Q-Exch]
5) TAUT[(∀x)Px ⊃ (∃x)Px)]	(T5-336)	[4),T4-31, SynSUB]
6) TAUT[(∀x)Px ⊃ (∃y)Py)]		[5),Alph.Var.]
7) TAUT[(∀x)P<x,2> ⊃ (∃y)P<y,2>)]		[5),U-SUB(P<1>/P<1,2>)]
8) TAUT[(∀z)((∀x)Pxz ⊃ (∃y)Pyz)]		[6),DR5-32']
......		
2n+4) TAUT[(∀z)((∀x_1)P<x_1,...,x_n> ⊃ (∃y)P<y,z,...,n>)]		[2n+3),DR5-32']
2n+5) TAUT[(∀z)...(∀x_n)((∀x)P<x,z,...,x_n,**n+1**> ⊃ (∃y)P<y,z,...,x_n,**n+1**>]		[2n+4),U-SUB: 'P<1,2,...,n,**n+1**>' for 'P<1,2,...,n>']
2n+6) **TAUT[(∀w)(∀z)...(∀x_n)((∀x)P<x,z,...,x_n> ⊃ (∃y)P<y,z,...,x_n>]**		[2n+5),DR5-32']

To get to step 8) from step 6), U-SUB allows replacement of 'P<1>' in Step 6) by 'P<1,2>' to yield Step 7), Then by DR5-32', which is "If TAUT[P<2>] then TAUT[(∀z)Pz]", we prefix '(∀z)' to the whole and replace '2' by 'z' to yield 8). Actually DR5-32' is used implicitly in the form,

If TAUT[(∀x)P<x,2> ⊃ (∃y)P<y,2>)] then TAUT[(∀z)((∀x)Pxz ⊃ (∃y)Pyz)] and thence, with 7), by MP to 8) TAUT[(∀z)((∀x)Pxz ⊃ (∃y)Pyz)] and so on, to get the nth closure in step 2n+5.

Using this routine, any prototype of a theorem numbered *100 or higher can be used to prove for any number of variables, and for any predication, what is covered by Quine's "The closure of P is a theorem". Thus A-logic covers all finitely stateable theorems that Quine refers to.

Quine's metatheorems include eight conditional rules of inference with closure clauses in both the antecedent and the consequent (*111, *112, *113, *115, *117, *123, *124, *163). The most widely used is, *111 If ⊦ P and ⊦ (P ⊃ Q) then ⊦ Q, which differs from *104 (TAUT-Det). It reads as,

"If the universal closure of P is a theorem and the universal closure of (P ⊃ Q) is a theorem
then the universal closure of Q is a theorem."

Replacing '(P ⊃ Q)' by its synonym '(~P v Q)', and 'universal closure' by 'conjunctive closure' (which means prefixing conjunctive quantifiers and replacing distinct argument-position-holders with their variables) and 'is a theorem, by 'is TAUT', this translates into,

*111a. If the conjunctive closure of P<t_1,...,t_j> is TAUT
and the conjunctive closure of (~P<t_1,...,t_j> v Q<t_{j+1},...,t_{j+m}>) is TAUT
then the conjunctive closure of Q<t_{j+1},...,t_{j+m}> is TAUT.

The corresponding principle, for inconsistency is,

*111b. If the disjunctive closure of $P<t_1,...,t_j>$ is INC
and the disjunctive closure of $(\sim P<t_1,...,t_j>$ & $Q<t_{j+1},...,t_{j+m}>)$ is INC
then the disjunctive closure of $Q<t_{j+1},...,t_{j+m}>$ is INC.

Given the tautologousness of a closure of $(P \supset Q)$ with k subject-terms, and the tautologousness of P (with $i \leq k$ argument-position-holders) we can always prove the conjunctive closure of Q is TAUT by mathematical induction (with a parallel proof for INC):

To prove that *111a holds, no matter how many argument-position-holders (in lieu of free variables) there are in $(\sim P<t_1,...,t_j>$ v $Q<t_{i+1},...,t_n>)$ we prove

1) Base Step:
If there are no argument-position-holders(i.e. $n = 0$), then *111a holds by TAUT-Det

2) Inductive Step:
If *111a holds for k argument position-holders, it will hold for k+1 argument position-holders. This relies on DR5-20 and DR5-32'.

3) Hence, by principle of mathematical induction,
*111 will hold for any number of argument-position holders. (in lieu of free variables)

Proofs of this general sort can be set up for the other Rules of Inference (*112, *113, *115, *117, *123, *124, *163). Thus a correlated TAUT- or INC-theorem for every actual theorem derivable by these rules in Quine's version of M-logic, will be derivable in A-logic.

Rules of Inference are conditional statements which say the if one type of wff is a theorem (TAUT-theorem) then another type of wff is a theorem (TAUT-theorem). The one primitive rule of inference used by Quine, Rosser and Thomason, is TTAUT-Det, Quine's *104. Besides *100, which is replaced by Rosser's three axioms, the eight other derived rules of inference which remain among Quine's metatheorems (*111,*112,*113,*115,*117,*123,*124,*163) are presumably accounted for.

5.3433 Axiom Schema *102; vacuous quantifiers. 239

What remains are 11 metatheorems: 3 Theorem Schemata: *110,*114,*135, and 8 Cond. Theorem Schemata: *102,*103,*118,*120,*121,*122,*134,*137,
All of these rely for their derivation on one or both of the two conditioned axiom schemata,

*102 If v_1 is not free in P, then $\vdash (\forall v_1)P$.
*103 If Q is like P except for containing free occurrences of v_1
where P contains free occurrences of v_1, then $\vdash [(\forall v_1)P \supset Q]$.

The latter are conditioned metatheorems. That is, they assert that under certain conditions involving free occurrences of variables in a wff, statements of a certain form will be a theorem. They are not principles of inference—i.e., principles for moving from one theorem to another. They are statements pertaining to internal features of a wff which will make it, or some construction from it, a theorem. The statement of such conditions can become very complicated and difficult to grasp, as in Quine's *120:

*120 If Q_2 is like Q_1 except for containing P_2 at a place where Q_1 contains P_1, and $x_1,\ldots x_n$ exhaust the variables with respect to which these occurrences of P_1 and P_2 are bound in Q_1 and Q_2,
then the closure of $[(\forall x_1)\ldots(\forall x_n)(P \equiv Q) \supset (R \equiv S)]$ is a theorem.

Thomason's primary motive for introducing "individual parameters" was to avoid this complicated kind of conditional theorem. He wrote,

> Perhaps the most common usage is to allow formulas in which individual variables need not be bound by quantifiers and to speak of *free* and *bound* occurrences of individual variables. In this case individual parameters are not needed at all: free occurrences of individual variables correspond to individual parameters and bound occurrences of individual variables to [bound] individual variables in our sense. This alternative is simpler, in that it eliminates the need for a separate category of parameters. But the approach that we have chosen is to be preferred in several respects: it makes it unnecessary to distinguish between free and bound occurrences of variables and, as we remarked above, it greatly simplifies rules for introducing and eliminating quantifiers in derivations.[19]

This simplicity in rules and derivations is also a major advantage enjoyed by A-logic. In A-logic the introduction of "argument-position-holders" serves this purpose as well. Buy they are also introduced for a deeper reason: to shift the focus of logic from propositions to predicates, and from truths of statements to meanings of predicates.

To affect the elimination of conditional clauses about free variables, we begin by discussing Quine's *102, which is the same as Rosser's Axiom 5. Let '*102' stand for both of them. We will rule out *102 and two other of Quine's metatheorems altogether, as inessential and extraneous. They unnecessarily mandate that expressions with vacuous quantifiers will be theorems. Any quantifier prefixed to an expression which has no free occurrence of its variable is vacuous by definition. No "theorem" which has a vacuous quantifier is essential to M-logic. The presence or absence of vacuous quantifiers has no effect whatever on truth or falsity, tautology or inconsistency, POS or NEG, CONT or SYN. Vacuous quantifiers add nothing relevant to the meaning of a wff. Vacuous quantifiers are meaningless symbols.

The wffs and statements which satisfy the antecedent of *102, "If x is not free in P", include all of the theorems in Quine's version of M-logic. This clause is satisfied by a wff P if x does not occur in P, or if x occurs but is bound to some quantifier in P. To substitute such a wff at both occurrences of 'P' is to create a theorem in Quine's system in which the '$(\forall x)$' following the '$\supset$' is vacuous and has no meaning at all, since no occurrence of 'x' is bound to it.

The same remarks apply to *118. If x is not free in P, $\vdash [P \equiv (\forall x)P]$ and to *137. If x is not free in P, $\vdash [P \equiv (\exists x)P]$. Both of them mandate that wffs with a vacuous quantifier will be a theorem, but at the same time, the biconditional indicates that P without a vacuous quantifier is logically equivalent to P with a vacuous quantifier. With the substitutivity of the biconditional (Quine's *123) *118 and *137 prove the meaningless of vacuous quantifiers. For every theorem with any vacuous quantifier $(\forall x)$ or $(\exists x)$ can be proved equivalent to a wff which is exactly the same except that it lacks that quantifier. Every theorem which contains one or more vacuous quantifiers is equivalent to and replaceable by a theorem exactly like it except for the absence of all vacuous quantifiers. By the same principles, in Quine's and Rosser's system endless vacuous quantifiers can be introduced and prefixed to every component, atomic or compound, in every wff, theorem or not, without changing its logical status in any way.

---

19. Thomason, Richmond H., *Symbolic Logic*, 1970, p 152.

Thus no "theorems" gotten from Quine's metatheorem's *102, *118 and *137 need be, or should be, TAUT-theorems of A-logic. None of them are essential to M-logic, and, taken by themselves, all they do is to introduce meaningless symbols.

In A-logic all such wffs are ill-formed (i.e., not wffs) and thus can not be theorems. Quantifiers are introduced only if there are variables bound to them. '(∀x)Px' abbreviates '(Pa_1 & Pa_2 &...& Pa_n)' and '(∃x)Px' abbreviates '(Pa_1 v Pa_2 v...v Pa_n)'. To introduce one part of an abbreviation without the other part would violate rules for the formation of well-formed, meaningful wffs. In Thomason also, such wffs are ill-formed and variables cannot be introduced without a quantifier.

The preceding discussion does not entail that anything is wrong with the phrase 'If x is not free in P' by itself. It is only when it is used to mandate, or to allow, expressions with vacuous quantifiers as wffs or theorems, that it should be expunged. "Rules of passage" are essential theorems which allow quantifiers to jump into or out of a conjunction or disjunction. They only apply when the variable in the quantifier is not free in the component being jumped over. For example, in Quine, *157 to *162 are properly expressed as follows:

*157. If x is not free in P, ⊢ [(∀x)(P & Qx)≡(P & (∀x)Qx)]
*158. If x is not free in P, ⊢ [(∃x)(P & Qx) ≡ (P & (∃x)Qx)]
*159. If x is not free in P, ⊢ [(∀x)(P v Qx)≡(P v (∀x)Qx)]
*160. If x is not free in P, ⊢ [(∃x)(P v Qx)≡(P v (∃x)Qx)]
*161. If x is not free in Q, ⊢ [(∀x)(Px ⊃ Q)≡((∃x)Px ⊃ Q)]
*162. If x is not free in Q, ⊢ [(∃x)(Px ⊃ Q)≡((∀x)Px ⊃ Q)]

In A-logic, as in Thomason's system, the antecedent clauses are dropped, because rules of formation and substitution in those systems guarantee that if any predicate letter P or Q, occurs without being prefixed to a certain variable, then neither it, nor anything substitutable for it, will have any free occurrence of that variable. Thus the theorems ranging from T5-317 to T5-434c below are the same as *157 to *162 but without the prefixed clause. The rules of passage are essential theorems for M-logic and for the set of TAUT-theorems in A-logic.

T5-317c. TAUT[(∀x)(P&Qx) ≡ (P&(∀x)Qx)]	Quine's ML*157	[T3-17,DR5-5d']
T5-319c. TAUT[(∃x)(P&Qx) ≡ (P&(∃x)Qx)]	Quine's ML*158	[T3-19,DR5-5d']
T5-320c. TAUT[(∀x)(PvQx) ≡ (Pv(∀x)Qx)]	Quine's ML*159	[T3-20,DR5-5d']
T5-318c. TAUT[(∃x)(PvQx) ≡ (Pv(∃x)Qx)]	Quine's ML*160	[T3-18,DR5-5d']
T5-435c. TAUT[(∀x)(Px ⊃ Q) ≡ ((∃x)Px ⊃ Q)]	Quine's ML*161	[T4-35,DR5-5d']
T5-434c. TAUT[(∃x)(Px ⊃ Q) ≡ ((∀x)Px ⊃ Q)]	Quine's ML*162	[T4-34,DR5-5d']

In Quine's system, *102, *118 and *137, though inessential in themselves, nevertheless play a role in proofs of essential theorems. They are effectively employed in proving the Rules of Passage and many other theorems. In this sense they are expedient, useful devices in Quine's system. They can not be eliminated unless something else is put in their place. In other systems of M-logic, this is accomplished by taking one of the rules of passage (see *157 to *162) as an axiom. Thomason's fourth axiom, for example, is a rule of passage from which the others can be derived: T5-438c TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)]

Among the other metatheorems, some need to be revised to prevent the introduction of vacuous quantifiers. Theorems derived from them do not require vacuous quantifiers, but they allow them. The following three rules of inference, allow instantiations with vacuous quantifiers which would be "theorems" for Quine, but not for A-logic. All three depend on *102 for proof.

*115. If $\vdash$ P then $\vdash$ $(\forall x)P]$
*117. If $\vdash$ $[P \supset Q]$ and none of $x_1,\ldots,x_n$, are free in P, then $\vdash$ $[P \supset (\forall x_1),\ldots,(\forall x_n)Q)]$
*163. If $\vdash$ $[P \supset Q]$ and none of $x_1,\ldots,x_n$, are free in Q, then $\vdash$ $[(\exists x_1),\ldots,(\exists x_n)P \supset Q)]$

These "theorems", of course, can be "cleaned up" by Quine using the method just mentioned to eliminate all vacuous quantifiers. In A-logic, this is not enough. Though they need not be completely eliminated; they must be revised. The essential job they do can be accomplished, while preventing vacuous quantifiers, by the following rephrasing, with '$\vdash$' replaced by 'TAUT'.

*115 may be rephrased as,
115′ "If the closure of $[P<x_1,\ldots,i,\ldots,x_n]$ is TAUT, (Where 'i' is an APH and $1 \leq i \leq n$) then the closure of $[(\forall x_i)Px_1\ldots x_n]$ is TAUT."

This restricts quantifiers to those with variables $x_1,\ldots,x_n$. Every particular wff which satisfies the antecedent, "the closure of $[Px_1\ldots x_n]$", will be the same as the wff which satisfies the consequent except for the order in which the quantifiers occur. The two expressions are then proven logically equivalent and logically synonymous in A logic by one or more uses of T3-15 which interchanges positions of conjunctive quantifiers in a prefix.

The theorem schema *115, If $\vdash$ P then $\vdash$ $(\forall x)P$, either results in a vacuous quantifier, or it is accounted for by T3-15. For either P contains a free occurrence of 'x' or it does not.

If P contains a free occurrence of x, then such occurrences are bound to a universal quantifier in the closure of P. Thus to say the closure of P is a theorem, is to say that $[(\forall x)\ldots)P<\ldots x\ldots>]$ is a theorem, and since this is TF-equivalent and synonymous to the closure of $[(\forall x)P]$, the one closure reduces to the other. In other words, in such a case *115 may change the sequence of effective conjunctive quantifiers in the prefix, and thus the linear ordering in any expansion. But any such changes can be produced in AL by T3-15 which interchanges positions of conjunctive quantifiers in a prefix.

If P does not contain a free occurrence of 'x', then '$(\forall x)$' is vacuous in the closure $[(\forall x)P]$; either it has no occurrence of x, which makes '$(\forall x)$' vacuous in $[(\forall x)P]$, or all occurences of x are already bound in P, in which case the quantifier '$(\forall x)$' prefixed to P, in '$\vdash$ $(\forall x)P$' is redundant, and thus vacuous. Thus if P does not contain a free occurrence of x, a vacuous quantifier is introduced making the inferred wff ill-formed in AL, and the result is logically equivalent in M-logic to a wff exactly the same except for the absence of all vacuous quantifiers.

Quine's *117 and *163, as mentioned, are also rules of inference which, as they stand, allow vacuous quantifiers to be inserted though they don't require them. They could be restated to avoid vacuous quantification in a way acceptable to A-logic as follows:

117'. If the closure of $[P \supset Qx_1\ldots x_n]$ is TAUT
and none of $x_1,\ldots,x_n$, are free in P,
then the closure of $[P \supset (\forall x_1),\ldots,(\forall x_n)Qx_1\ldots x_n)$ is TAUT]

163'. If the closure of $[Px_1\ldots x_n \supset Q]$ is TAUT
and none of $x_1,\ldots,x_n$, are free in Q,
then the closure of $[(\exists x_1),\ldots,(\exists x_n)Px_1\ldots x_n \supset Q)$ is TAUT]

This phrasing avoids vacuous quantifiers, but seems unnecessarily complex. Whatever theorems can be derived from these principles can also be derived -- as we said earlier—from a much simpler set of principles without needing the phrase, 'the closure of P is TAUT'. It is sufficient to point out that all cases derivable from 117' can be gotten using the easily derived rule,

$\models$ If TAUT[$P \supset Q{<}1{>}$] then TAUT[$P \supset (\forall x_1)Qx_1$],

repeatedly applied with U-SUB to introduce additional variables with their quantifiers. Similarly all cases covered by 163' can be gotten from the derived rule,

$\models$ If TAUT[$P{<}1{>} \supset Q$] then TAUT[$(\forall x_1)Px_1 \supset Q$].

One final remark: in Thomason vacuous quantifiers are allowed by rules of formation, but are not required in the derivation of any theorem, and the rules for moving from unquantified to quantified theorems do not open the possibility of inserting vacuous quantifiers. In A-logic rules of formation are stricter. All quantified wffs are abbreviations of generalized conjunctions or disjunctions in which the predicate of the quantification is applied to every member of the domain in turn. Introducing a part of an abbreviating expression,'(∀x)' or 'Px', without the other part can never result in a well-formed wff.

The elimination of *102, *118 and *137 reduces the number of Quine's metatheorems from 56 to 53, and revisions of *115, *117 and *163 leaves the number of Quine's metatheorems not yet accounted for at eight: *103, *110, *114, *135, *120, *121, *122, and *134.

5.3434 Axiom Schema *103; the Problem of Captured Variables

The eight remaining metatheorems all depend directly or indirectly on Quine's *103 which may be represented by the prototype,

*103 If Q is like P, except for containing free occurrences of y
wherever P contains free occurrences of x,
then the closure of [$(\forall x)P \supset Q$] is a theorem.[20]

If P is 'Px' then, this says '$(\forall y)((\forall x)Px \supset Py)$' is a theorem. Thus it might be thought that in A-logic we could use 'TAUT[$(\forall y)((\forall x)Px \supset Py)$; as a prototype, from which we could get all theorems covered by *103, using U-SUB and generalization rules. But it is not that simple. Unless some constraint is placed on what is substituted for 'P' we may get wffs which have false instances and thus can't be theorems. This is what the antecedent of *103 is designed to avoid. In A-logic the same undesired consequences are avoided by clause (v) of DR5-20, which requires that any variables introduced by U-SUB be new ones. The desired results are then derivable from other TAUT-theorems of A-logic with DR5-20.

In Rosser's System, certain conventions are introduced which allow him to simplify Quine's *103 by using instead Rosser's Axiom 6, $\vdash [(\forall x_1)(\forall x_2)\ldots(\forall x_n)((\forall x)Pxy \supset Pyy]$, which does not require any antecedent clause about free variables. The prototype for this axiom is the theorem, T5-09. TAUT[$(\forall y)((\forall x)Rxy \supset Ryy$] which was proven from A-logic's axioms and rules by way of T5-08 in Section 5.24. Many variations are possible, including all those covered by Quine's 'The closure of … is a theorem'. The following examples indicate the possibilities:

---

20. <u>5 Cond.Theorem Schemata</u>:
*103
*120 (depends on *114),
*121 (depends on *120),
*122 (depends on *121),
*134 (depends on *103),

<u>3 Theorem Schemata</u>:
*110 (depends on *103),
*114 (depends on *110),
*135 (depends on *134),

<u>Note</u>: *110, *114 and *135 all use *103 letting y = x (i.e, without a new variable). The captured-variable problem discussed below pertains only to *103, *120, *121, *122 and *134.

1) TAUT[(∀y)((∀x)Rxy ⊃ Ryy)] [T5-09]
2) TAUT[(∀y)((∀x)Ryxy ⊃ Ryyy)] [1),U-SUB:'R<2,1,2>' for 'R<1,2>']
3) TAUT[(∀y)((∀x)R<x,y,3> ⊃ R<y,y,3>)] [1),U-SUB:'R<1,2,3>' for 'R<1,2>']
4) TAUT[(∀z)(∀y)((∀x)Rxyz ⊃ Ryyz)] [3), DR5-32']
5) TAUT[(∀z)(∀y)((∀x)Rzxy ⊃ Rzyy)] [4),U-SUB:'R<3,1,2>' for 'R<1,2,3>']
6) TAUT[(∀y)((∀x)Rxzxy ⊃ Ryzyy)] [5),U-SUB:'R<2,1,2,3>' for 'R<1,2,3>']

Rosser's Axiom 6, can also be proven by mathematical induction in a domain of n as follows :

T5-09 TAUT[(∀y)((∀x)Pyx ⊃ Pyy]
<u>Proof</u>: 1) INC[((($\mathbf{Pa_1a_1}$ & Pa_1a_2 &…& Pa_1a_n) **&** ~$\mathbf{Pa_1a_1}$)
v ((Pa_2a_1 & $\mathbf{Pa_2a_2}$ &…& Pa_2a_n) **&** ~$\mathbf{Pa_2a_2}$))
v….
v ((Pa_na_1 & Pa_na_2 &…& $\mathbf{Pa_na_n}$) **&** ~$\mathbf{Pa_na_n}$))) [Df 'INC'(i),(ii),(iii)]
2) [(∃y)((∀x)Pyx & ~Pyy)) SYN
((($\mathbf{Pa_1a_1}$ & Pa_1a_2 &…& Pa_1a_n) **&** ~$\mathbf{Pa_1a_1}$)
v ((Pa_2a_1 & $\mathbf{Pa_2a_2}$ &…& Pa_2a_n) **&** ~$\mathbf{Pa_2a_2}$))
v….
v ((Pa_na_1 & Pa_na_2 &…& $\mathbf{Pa_na_n}$) **&** ~$\mathbf{Pa_na_n}$))) [Df '(∃y)']
3) [(∃y)((∀x)Pyx & ~Pyy)) SYN
(((∀x)P$\mathbf{a_1}$x & ~$\mathbf{Pa_1a_1}$)
v ((∀x)P$\mathbf{a_2}$x & ~$\mathbf{Pa_2a_2}$))
v….
v ((∀x)P$\mathbf{a_n}$x & ~$\mathbf{Pa_na_n}$))) [2),Df '(∀x)']
4) INC[(∃y)((∀x)Pyx & ~Pyy))] [2),1),SynSUB]
5) INC[~(∀y)~((∀x)Pyx & ~Pyy)] [4), Q-Exch]
6) INC[~(∀y)((∀x)(Pyx ⊃ Pyy)] [5),Df '⊃']
6) TAUT[(∀y)((∀x)(Pyx ⊃ Pyy)] [6),Df 'TAUT']

How *103 and Clause (v) of DR5-20 preserve Tautology. Unless some constraint is placed on what is substitutable for 'P<1>' in the theorem 'TAUT[(∀y)((∀x)Px ⊃ Py]', the wffs which result may have false instances thus not be tautologies or theorems of M-logic. In A-logic, clause (v) in the rule of U-SUB, DR5-20 provides the necessary constraint. It says the expression to be substituted must not contain any individual variable which has occurrences in the expression into which the substitution will be made. In Quine's logic the antecedent clause in *103 is used to avoid these cases. Let us look at what Quine is trying to avoid, then compare these two ways of accomplishing that end.

Quine describes two kinds of substitution procedures Quine which would not preserve theoremhood (or tautologousness). In the first kind, a quantifier in the substituted predicate captures a variable in the initial predicate; in the second kind a free variable in the expression to be substituted is captured by a quantifier in the original expression.[21]

The first kind of unwanted substitution may be illustrated by the following example, which also shows, on the right, why A-logic rules it out:

a) TAUT[(∀y)((∀x)Px ⊃ Py)] [T5-08]
Substituting '(∃y)R<1,y)' for 'P<1>' (Violates clause (v) of DR5-20)
b) TAUT[(∀y)((∀x)(∃y)Rxy ⊃ (∃y)Ryy)] ('(∀y)' is vacuous)

We can ignore ‘(∀y)’ in b) since it has been rendered vacuous. But if “ <1>is less than <2> ” is put for R<1,2> in b), we get: “TAUT[(∀x)(∃y)(**x** is less than y) ⊃ (∃y)(**y** is less than y)]” i.e., “‘If every number is less than some number, then some number is less than itself’ is tautologous”. This is clearly false. Thus b) can not be a theorem; since the antecedent is true and the consequent is false, it can be truthfully denied, and thus it is not tautologous. Although a) is a tautology, b) is not. Thus this substitution can not be allowed.

In Quine’s system the restriction in *103 prevents the introduction of ‘(∃y)R<1,y>’ for ‘P<1>’. It is prevented because ‘(∃y)R**y**y’ is <u>not</u> “like ‘(∃y)R**x**y’ except for containing free occurrences of y where ‘(∃y)R**x**y’ contains free occurrences of x”. For ‘x’ is free in the first position of ‘(∃y)Rxy’ but ‘y’ is not free in the first position of ‘(∃y)Ryy’.

In A-logic, the move from a) to b) is invalid because substituting ‘(∃y)R<1,y>, in a) violates clause (v) of DR5-20, that says all variables in the substituted term must be new to the formula in which it is substituted. It is also invalid because b) is not well-formed and thus not a theorem, since ‘(∀y)’ has been rendered vacuous. Quine allows vacuous quantifiers, so the wff in b) is well-formed in M-logic, but In A-logic, it is not well-formed because ‘(∀y)’ is vacuous and this rules out this substitution.

If clause (v) of U-SUB is satisfied, we have two alternatives, also allowed under Quine’s *103, which preserve theoremhood.

a)	TAUT[(∀y)((∀x)Px ⊃ Py)]	a)	TAUT[(∀y)((∀x)Px ⊃ Py)]
	Substituting ‘(∃z)R<1,z)’ for ‘P<1>’ or		‘(∃z)R<z,1)’ for ‘P<1>’
b’)	TAUT[(∀y)((∀x)(∃z)Rxz ⊃ Ryz)]	b’)	TAUT[(∀y)((∀x)(∃z)Rzx ⊃ (∃y)Rzy)]

The second kind of mistake Quine avoids by the restriction in *103 can be caused if a free variable in the predicate-to-be-substituted, becomes bound to a quantifier in the predicate within which it is substituted, i.e., a quantifier in the original predicate captures a variable in the substituted predicate. For example:

a) TAUT[(∀y)((∀x)Px ⊃ Py)]
Substituting ‘R<x,1,>’ for ‘P<1>’ (Violates DR5-20, clause (v))
b) TAUT[(∀y)((∀**x**)R**x**x ⊃ Rxy)] Also ‘Rxy’ has an unbound ‘x’, hence is not a wff.

If we put ‘<1>equals<2>, for ‘P<1,2>’ this has a false instantiationt: ‘(∀y)((∀x) x=x ⊃ x=y)’. This says, “No matter what y may be, if everything is equal to itself, then it is equal to y”. This is not allowed by *103, since the introduction of ‘P<1,x>’ for ‘P<1>’ violates the restriction in *103: ‘R**x**y’ (Q) is <u>not</u> “like ‘Rxx’ (P) except for containing free occurrences of y where ‘P’ contains free occurrences of x”. For ‘R**x**x’ (P) has a free occurrence of x in the first position, while ‘R**x**y’ does not have a free occurrence of y in the first position.

In A-logic the substitution of ‘R<1,x>’ for ‘P<1>’ in ‘[(∀y)((∀x)Px ⊃ Py)]’ is invalid— again because it violates clause (v) of DR5-20, that all variables in the substituted term must be new to the formula in which it is substituted. This substitution is also invalid in A-logic because ‘Rxy’ is not well-

---

21. In the several editions of *Methods of Logic*, Quine states the two kinds of restrictions. In the 2nd and 3rd editions, they are expressed in terms of substituting predicates for predicate letters or “open sentences”. His first restriction: “variables entering the predicate…must not be such as to be captured by quantifiers within the predicate”. (*Methods of Logic*, 3rd Ed, p. 147). His second restriction: “variables free in the predicate must not be…such as to be captured by quantifiers within the schema into which the predicate is substituted.” (*Methods of Logic*, 3rd Ed, p. 148). In the fourth edition Quine uses class abstracts instead of predicates, but the nature of the restrictions needed remains the same.

formed since the variable ‘x’ occurs in it without being bound to a quantifier. This violates the clause in DR5-20 which says what is substituted must be a wff.

If clauses (i) to (v) of DR5-20 are satisfied, we have many alternatives, also allowed by Quine, which preserve theoremhood.

a)	TAUT[(∀y)(∀x)Px ⊃ Py)]		a)	TAUT[(∀y)(∀x)Px ⊃ Py)]
	Substitute ‘R<1,1>’ for ‘P<1>’	or,		‘R<1,a>’ for ‘P<1>’
b’)	TAUT[(∀y)((∀x)Rxx ⊃ Ryy)]		b’)	TAUT[(∀y)((∀x)Rxa ⊃ Rya)]
	Substitute ‘(∃z)R<1,1,z>’ for ‘P<1>’	or,		‘(∀z)R<a,z,1>’ for ‘P<1>’
b”)	TAUT[(∀y)((∀x)(∃z)Rxxz ⊃ Ryyz)]		b”)	TAUT[(∀y)((∀x)(∃z)Razx ⊃ Razy)]

Quine avoided the undesirable cases by inserting a proviso which requires an investigation in each case of whether a variable will be captured in that case, or a quantifier will capture a variable in that case. A-logic incorporates its restriction in the rule of universal substitution. All one need do is use only new variables in any wff to be substituted for a predicate. The context in which it will be placed need not be examined for scopes of its quantifiers.

103* is an effective device, but not logically significant. The constraint in the antecedent clause of *103 is clearly effective. As a device, it rules out the undesired cases and, taken with Quine’s other four axiom schemata, yields a logic which is complete with respect to the semantical theory of M-logic. However, the property which is required under this constraint is not a significant *logical* property. The restriction is too strong and not logically necessary. The clause serves a limited purpose but does not seize upon a logical defect which must be avoided. For there are cases in which the proviso in *103 is not satisfied, yet the result is a theorem, i.e., y replaces x in a position in which x is free in P but y is not free in P’, yet the result is a theorem. Consider two theorems derivable in both M-logic and A-logic,

	M-logic	A-logic
1)	⊦ [(∀y)**(** (∀x)R**x**y ⊃ R**y**y **)**],	TAUT[(∀y)**(** (∀x)R**x**y ⊃ R**y**y **)**]
2)	⊦ [(∀x)(∀y)Ry**x** ⊃ (∀y)Ry**y**],	TAUT[(∀x)(∀y)Ry**x** ⊃ (∀y)Ry**y**]

The first theorem, 1), is gotten by *103 in Quine. The term “wherever” in the antecedent of *103 means “at all argument positions in which…”. Thus *103 can be read as:

> “If P′ is like P except for containing free occurrences of **y** *at all argument positions in which* P contains free occurrences of **x**, then the closure of [(∀x)P ⊃ P′] is a theorem.”

The first theorem follows because the antecedent of *103 is satisfied in ‘(∀y)((∀x)R**x**y ⊃ R**y**y)’; ‘y’ occurs free in the first argument position of ‘Ryy’ (P′) and ‘x’ occurs free in the first argument position of ‘Rxy’(P). In a domain of 2 this is: ((Raa&Rab)&(Rba&Rbb)) ⊃ (Raa&Rbb)). However, the second theorem cannot be gotten by *103, because its antecedent is <u>not</u> satisfied in ‘((∀x)(∀y)Ry**x** ⊃ (∀y)Ry**y**))’; ‘y’ <u>is not</u> free in the second argument position of ‘(∀y)Ry**y**’ (P′), although ‘x’ <u>is</u> free in the second argument position of ‘(∀y)Ry**x**’ (P). This is strange since 2) is derivable from 1) in M-logic as well as A-logic:

2) TAUT[(∀x)(∀y)Ry**x** ⊃ (∀y)Ry**y**]

Proof: 1) TAUT[(∀y)((∀x)Rxy ⊃ Ryy)] [Theorem 1) gotten by *103]

2) TAUT[(∀y)((∀x)Rxy ⊃ Ryy) ⊃ ((∀y)(∀x)Rxy ⊃ (∀y)Ryy)] [*101]

3) TAUT[(∀y)(∀x)Rxy ⊃ (∀y)Ryy] [1),2),*104]

4) TAUT[(∀x)(∀y)Ry**x** ⊃ (∀y)Ry**y**] [3),*171(Alphabetic variance)]

Thus if the antecedent condition of *103 were required to be imposed on all occurrences of the same predicate with different variables, it would exclude a lot of theorems from being proved by *103.

But doesn't Clause (v) of DR5-20 also exclude cases which are valid? Yes. In some cases there is no need to require a variable new to whole schema. If a variable occurs in the schema where it could neither capture nor be captured by any quantifiers or variables being substituted, the same variable could be used twice. Indeed, after DR5-20 has been employed, quantifiers can be re-lettered by Rules for Alphabetic variance, putting the same variable in quantifications whose scopes don't overlap, and ending up with whatever wffs Quine would end up with.

In A-logic, we might restrict the requirement for a new variable in clause (v) to those cases in which some variable in the predicate would be captured, or some quantifier in the predicate would capture some variable. This, like Quine's clause, would require inspection of each new case, and is not necessary. DR5-20 is better than Quine's constraint on *103 in the following sense. Clause (v) of DR5-20 could be viewed as part of a general rule requiring that all quantifiers in any wff have different variables. Such a general principle would not be a bad one; it would make no difference logically, though it would require more different variables in many wffs. For example, *101 would have to be written as, (∀x)(Px ⊃ Qx) ⊃ ((∀y)Py ⊃ (∀z)Qz)). This might be good - readers would look for logical connections in the predicates, or the kind of quantifier, rather than thinking that the similarity of variables in different quantifiers was logically significant. Clause (v) would be unnecessary, since the general principle would require a new variable when creating any new wff by substitution.

In contrast, a general requirement that different variables could occur in the same predicate only if both were free in that predicate, would make many logical tautologies unprovable, e.g., [(∀x)(∀y)(Rxy ⊃ (∀y)Ryy]. Thus using Clause (v) instead of restrictions on the occurrences of free variables, A-logic avoids a constraint which is effective but of questionable logical merit.

It can be shown that A-logic , like M-logic, will not include non-theorems like
TAUT[(∀y)(∃x)Rx**y** ⊃ (∃x)Rx**x**)]: (in a domain of 2, ((Raa v Rab)&(Rba v Rbb)) ⊃ (Raa v Rbb))
f t t **t** t t f **F** f **f** f

5.35 Completeness re M-logic: Theorems, not Rules of Inference

The completeness we have proved relates only to the theorems, not the rules of inference, of M-logic. For example, in A-logic [Q ⊃ (~PvP)] and [(P&~P) ⊃ Q] are tautologies, but it is not the case that (~PvP) follows logically from Q, or that Q follows logically from [P&~P]. The M-logic rules of inference, "From a [P&~P] infer Q", and "From Q infer (Pv~P)" are not valid rules in A-logic, though they are valid rules of inference for M-logic. The predicates "<1>is valid" is and "<1>is a theorem" are defined differently in M-logic and in A-logic. In A-logic it is agreed that "TAUT[(P&~P)⊃ Q]" and "TAUT[Q ⊃ (Pv~P)]" are properly included among the theorems of A-logic. Further it is agreed that no instances of a tautologous wff can be false. But it does not follow that, If TAUT[P ⊃ Q] is true, then [P, therefore Q] is a valid argument in A-logic, or that if [P ⊃ Q] is a tautology, then [P ⊃ Q] is valid and a theorem of A-logic.

In A-logic the theorems of quantification theory include the following:

a) TAUT[(∀x)∼Gx ⊃ (∀x)(Gx ⊃ Hx))]
and b) TAUT[(∃x)Gx ⊃ ((∀x)Bx ⊃ (∃x)Gx)][22]

These theorems, according to M-logic, with '⊃' read as "if..then", make the following arguments respectively, "logically valid":

a) There are no ghosts.
Therefore, all ghosts are horses

b) Some people are good
Therefore, if everyone is bad then some people are good"

But in A-logic these are provably not valid arguments by its definition of validity. That M-logic should think them valid is due in part to the definition of 'valid' in M-logic, and in part to the account of 'if…then' in M-logic—two aspects of M-logic which are not unconnected.

The concept of validity will be defined and discussed more carefully in the next chapter, after we have added a new primitive sentence connective '=>' for 'if…then' and provided an approperiate axiom. Once again, A-logic is complete with respect to the theorems of M-logic. It can prove tautologous (and incapable of being false), all instances of just those wffs which M-logic lists as its theorems. But A-logic rejects many of the derived rules of inference in M-logic. It rejects the rule, presupposed in M-logic, that if TAUT[P ⊃ Q] then [If P then Q] is Valid and [P therefore Q] is valid. These problems and their solutions are covered in Chapters which follow.

5.4 M-logic as a System of Inconsistencies

The concept that mathematic logic deals only with universal truths, is belied by the fact that the formal system of M-logic can just as easilly be interpreted as a system for enumerating inconsistencies, or universal falsehoods. This can be shown in several ways.

First, by the mere semantic change of interpreting '&' as meaning "or" and 'v' as meaning "and", the system of M-logic is converted without any other change, into a system for producing inconsistencies. For then the truth tables for '&' and 'v' become as shown on the right, and all cases that have *all* T's in their truth-tables, will now have *all* F's in their truth-tables. On this interpretation all axioms and theorems of Sentential Logic (e.g., of Rosser or Thomason) become inconsistencies, and the rule of detachment (TF-Modus Ponens) will be changed from meaning "If P is a theorem and (∼P or Q) is a theorem, then Q is a theorem", to meaning "If P is a theorem and (∼P and Q) is a theorem, then Q is a theorem". Hence all derivations in M-logic lead from inconsistency to inconsistency. In quantification theory the disjunctive quantifier and the conjunctive quantifier also exchange meanings, and *its* axioms also are changed from tautologies to inconsistencies. Thus M-logic becomes a system for producing inconsistencies rather than tautologies. The result is a consistent system in the sense that there is no formula such that both it and its negation are provable, and it is provably complete as a system for enumerating "logical falsehoods" in a sense which is parallel to the sense in which M-logic has been proven semantically complete with respect to the "logical truths".

(P & Q)	(P v Q)
T	T
T	F
T	T
F	F

22. To see that these are tautologies consider :

TAUT[(∀x)∼Fx ⊃ (∀x)(Fx ⊃ Gx))]	Syn INC [∼((∀x)∼Fx ⊃ (∀x)(Fx ⊃ Gx))]	[Df 'TAUT']
	Syn INC [∼∼((∀x)∼Fx & ∼∼(∀x)(Fx & ∼Gx))]	[Df'⊃']
	Syn INC [((∀x)∼Fx & (∀x)(Fx & ∼Gx))]	[DN]
(Expanded in a domain of 2):	Syn INC [((∼**Fa** & ∼**Fb**) & ((**Fa** & ∼Ga) & (**Fb** & ∼Gb))]	
TAUT[(∃x)Gx ⊃ ((∀x)Fx ⊃ (∃x)Gx)]	Syn INC[(∃x)Gx & (∀x)Fx & ∼(∃x)Gx]	

Alternatively, we can keep the semantical theory the same, negate every axiom and change the Rule of Detachment to, "If ∼P is a theorem and ∼(P ⊃ Q) is theorem then ∼Q is a theorem", then proceed exactly as before, step by step. The result will be that all theorems of M-logic are transformed into inconsistencies or "logically false" statements or schemata. If we replace '∼P is a theorem' with '⊢ ∼P', and then replace '⊢' with 'INC', the new Rule of Detachment ,INC-Det, becomes "If INC[∼P] and INC[∼(P ⊃ Q)] then INC[∼Q]", which in A-logic is synonymous with "If TAUT[P] and TAUT[P ⊃ Q] then TAUT[Q]", i.e., with TAUT-Det, and all theorems instead of being read "⊢ ∼P" are interpreted as 'INC[∼P]' which is synonymous in A-logic with "TAUT[P]".

A little more elegantly, using the synonymies of A-logic (including definitions) we can present the system of logic explicitly as a theory of inconsistencies rather than one of tautologies. Such a system is in fact a fragment of Analytic logic, on a par with that fragment consisting of the tautologies which correspond to the theorems of M-logic. Such a system can be gotten from any correct axiomatization of M-logic. Translating from Thomason's system, for example, using primitives '&' and '∼', we get the following system:[23]

Ax.1. INC[P & (Q & ∼P)]
(**Syn** TAUT[P ⊃ (Q ⊃ P)]. = T5-03, Thomason's AS1)
Ax 2. INC[∼(P&(Q&∼R)) & (∼(P&∼Q) & (P &∼R))]
(**Syn** TAUT[(P ⊃ (Q ⊃ R)) ⊃ ((P ⊃ Q) ⊃ (P ⊃ R)] = T5-07,Thomason's AS2)
Ax.3. INC[∼(∼P & Q) & (Q & ∼P)]
(**Syn** TAUT[(∼P ⊃ ∼Q) ⊃ (Q ⊃ P)]; =T5-04, Thomason's AS3)
Ax 4. INC[∼(∃x)(P &∼ Qx) & (P & ∼(∀x)Qx))]
(**Syn** TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)] = T5-438f, Thomason's Axiom 4)
Ax 5. INC[∼(∃x)(Px & ∼Qx) & (∀x)Px & ∼(∀x)Qx)) [1),DR5-5a′]
(**Syn** TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))]; = T5-437c,Thomason's Axiom 5)
Ax 6. INC[(∀x)Px & ∼Pa]
(**Syn** TAUT[(∀x)(Px ⊃ Pa)]; = T5-333c, Thomason's Axiom 6)

From these axioms, with U-SUB, and the dual of TAUT-Det, namely, INC-Det. If INC[∼P&Q] and INC[P], then INC[Q] we can derive all inconsistencies, following essentially the same sequence of steps used in deriving tautologies. We use unrestricted U-SUB, standard definitions and SYN-theorems. The system is consistent and complete, and a mirror image, step-by-step and proof-by-proof of M-logic as developed by Thomason.

In practice a logic of inconsistencies can be just as useful as one of tautologies. The theorems will make clear and label what we should avoid (inconsistencies), including the acceptance of conclusions which are inconsistent with our premisses. And it is less likely that the logic of inconsistencies will lead into the non sequiturs which pass as valid when theorems of M-logic are taken as valid conditionals.

---

23. Another simpler alternative for sentential logic, using Rosser's primitives, is:

Rosser' 1: INC[P & ∼(P&P)]	Syn Rosser 1: TAUT[P ⊃ (P&P)]
Rosser' 2: INC[(P & Q) & ∼P]	Syn Rosser 2: TAUT[(P & Q) ⊃ P)]
Rosser' 3: INC[(∼(P&∼Q)&∼(Q&R)) & R&P)]	Syn Rosser 3: TAUT[(P ⊃ Q) ⊃ (∼(Q&R) ⊃ ∼(R&P)))]]

with U-SUB and INC-Detachment: If INC[P] & INC[∼P & Q)] thenr INC[Q].

5.5 A-validity and M-logic

"The most conspicuous purpose of Logic, in its applications to science and everyday discourse, is the justification and criticism of inference. Logic is largely concerned with devising techniques for showing that a given statement does, or does not, "follow logically" from another."

—W. V. Quine[24]

Although A-logic is complete with respect to the Theorems of M-logic (understood as a set of tautologies), tautologies and inconsistencies have a very different, secondary role in A-logic.

Tautologous expressions are never A-valid and are of little or no importance in A-logic. Inconsistencies are identified only for the purpose of avoiding or eliminating them. Instead, the emphasis is on consistency and validity. The lack of inconsistency in the conjunction of premiss and conclusion is a necessary condition of validity in A-logic. The object is to guide reasoning about contingent facts by concentrating on relationships of containment and synonymy between consistent, non-tautologous ideas and concepts. The primary role of A-logic is to provide conditional principles which can help people make valid inferences from one contingent expression to another.

We represent arguments and inferences of the form '(P_1, P_2, … P_n, therefore Q)' or 'P, hence Q' by the symbols '(P, **.:** Q)', where 'P' represents a string, or a conjunction, of expressions as the ground or premisses of an inference, and Q represents the purported logical consequence or conclusion.

Inferences and arguments may be logically valid or they may not be valid. The primary goal of Formal Logic is the formulation of rigorous methods for deciding when inferences from expressions of one form to expressions of a second form are *valid*; or, conversely, when an expression of the second form *follows logically* from an expression of the first form.

We will symbolize a claim that an inference from some expression P to another expression Q *is valid*, i.e., that Q *follows logically* from P, by the expression 'Valid[P, **.:**Q]'. This may be read "'P therefore Q' is valid" or "'P, hence Q' is valid".

When 'P' and 'Q' are are replaced by wffs according to the axioms and rules of some system of logic, the result is a "valdity theorem" of that logic. Thus "Valid[(P & Q), **.:** Q]" is a validity theorem of both A-logic and M-logic. (Logicians usually call this one "Simplification".)

Both A-logic and M-logic subscribe to the VC\VI principle: a conditional [if P then Q] is valid if and only if the inference from P to Q is valid. Thus there should be two kinds of Validity theorems which are interchangeable with the forms 'Valid[P, **therefore** Q]' and 'Valid[If P then Q]'.

The trouble with M-logic is that its definition of "if…then" together with its defintion of "validity" and the VC\VI principle leads M-logic to call certain arguments, which for intelligent language users are clearly *non sequiturs*, valid arguments—for example, arguments of the forms 'it is not the case that P therefore, if P then Q', and '(P and not-P), therefore Q', among many others.

A-logic has both a different account of conditionals and a different definition of 'validity' designed to preserve the VC\VI principle and avoid calling the *non sequiturs* valid.The different account of conditionals will be presented in Chapter 6. But independently of how conditionals are interpreted, it can be shown how, in this chapter, the different defintion of 'validity' in A-logic will avoid the *non sequiturs* among M-logic's "valid inferences".

24. Quine W.V., *Methods of Logic* (4th Ed.), 1982, Harvard Univerity Press; in Chapter 7, entitled "Implication", page 45.

5.51 A-valid and M-valid Inferences Compared

Since the concept of validity in A-logic and the concept of validity used in M-logic are different, let 'A-valid' mean the sense of 'valid' used in A-logic, and let 'M-valid' mean the sense given the word 'valid' in M-logic. This will avoid confusion and fighting over words.

In both logics validity is attributed both to wffs and to inferences from one wff to another.

In the case of *wffs in M-logic*, M-validity is a monadic predicate, equivalent to the property of being "logically true", i.e.,, being "true under all interpretations of its variables" by virtue of its logical form.[25] But in the semantic theory of M-logic, the validity of an *inference* from one wff to another, requires a relation between the two, which Quine equates with "follows logically from", and calls "implication". Quine wrote,

> From the point of view of logical theory, the fact that the statement 'Cassius is not both lean and hungry' follows from 'Cassius is hungry' is conveniently analyzed into these two circumstances:
> (a) the statements have the respective logical forms '∼(p & q)' and '∼q'…
> (b) there are no two statements which, put respectively for 'p' and 'q', make '∼q' true and '∼(p & q)' false.
> Circumstance (b) will hereafter be phrased in this way: '*(∼q implies ∼(p&q)*'. In general, one truth- functional schema is said to imply another if there is no way of so interpreting the letters as to make the first schema true and the second false … According to our definition, S1 implies S2 if and only if no interpretation makes S1 true and S2 false, hence if and only if no interpretation falsifies the material conditional whose antecedent is S1 and whose consequent is S2. In a word implication is the validity of the conditional.[26]

Note that this analysis employs the VC\VI principle, identifying the validity of the inference with the validity of the material (truth-functional) conditional.[27]

Thus in M-logic the M-validity of an inference is defined as follows:

"M-valid[P, **.:** Q]" Syn_{df}
"By virtue of their logical form, it is never the case that P is true and Q is false."

The definition of validity in A-logic is very different. It does not use the concepts of truth and falsehood. Instead it is based on containment, and on inconsistency (or rather the lack of inconsistency):

Df 5-6 "A-valid[P, **.:** Q]" Syn_{df} "(i) [(P Cont Q) and (ii) not-Inc(P & Q)]" [Df 'Valid']

When validity is based solely on the meanings of logical constants, '&', 'v', '⊃', '∼', '(∀x)', etc., then we are dealing with purely formal validity, signified by capital letters, and

"A-VALID[P, **.:** Q]" Syn_{df} "[(P CONT Q) and not-INC(P & Q)]".

---

25. "The definition of validity [for quantificational schemata] is as before: truth under all interpretations in all non-empty universes." Quine W.V., *Opus cit*., pp 172-3.
26. Quine W.V., Opus cit.., pp 45 and 46. (Underlining is mine; italics are Quine's).
27. Incidently, the example he uses, is—as we shall see—one which is not a valid inference in purely formal A-logic though a) the truth-functional conditional is clearly tautologous, and b) in analytic **truth**- logic, [Q is false, .: (P&Q) is false] is A-valid as Ti7-781a.

This kind of validity can be established on the basis of the SYN- and CONT-theorems in Chapters 1 through 4, where, due to the negation-free axioms in Chapters 1-3 and restrictions of U-SUBab in Chapter 4, all such theorems are inconsistency free.

The first definition of A-validity is more general, and makes allowance for synonymies in the definitions of extra-logical, inconsistency-free predicates. From such definitions many additional validity-theorems can be derived provided inconsistency is avoided by observing the restrictions in U-SUBab and SynSUB is based on with the definitions of the extra-logical terms.

We will also add the following abbreviation, or definition, which is very useful later on**.:**

Df.'Valid-&'. 'Valid(P & Q)' Syn_{df} '(Valid P & Valid Q)'

From this it follows, among other things, that

|= [If Valid (P, **.:**Q) and Valid (Q, **.:** P), then Valid ((P, **.:**Q) & (Q, **.:** P))][28]

There is no way that logic can prevent the introduction of inconsistencies when ordinary English expressions replace the predicate-letters; this is true in both A-logic and M-logic. But the *prima facie* lack of inconsistencies among the wffs proves that, provided further substitutions of wffs or ordinary language are not intrinsically inconsistent, the result of susbstitutions will not be inconsistent, and provided the containment condition is met, they will be valid. In general, actual ordinary language inferences are usually inconsistency-free. When they are not, either the consistencies are obvious or easily identified by logical analysis, or they are subtle inconsistencies which must be established by providing definitions of the extra-logical terms which are used. The lack-of-inconsistency requirement is discussed in more detail in Section 6.341.

The containment requirement in A-validity appeals to the oft-expressed view that if a conclusion follows logically from certain premisses, there must be some connection between the premisses and the conclusion. This requirement is notably lacking in M-logic and in systems of entailment based on "strict implication".

The principle of M-validity stated negatively is: "if the premisses can be true and the conclusion false at the same time, then the inference from the premisses to the conclusion can not be valid". This principle holds for A-validity as well as M-validity. But it is only a negative necessary condition for a A-validity not, as M-logic holds, a positive sufficient condition for establishing validity in general.

The requirement for A-validity that the conjunction of premisses and conclusion be free of inconsistencies is not present in the concept of M-validity. I know of no explicit reference to such a requirement for validity in traditional logic. Nevertheless it is a plausible and desirable requirement. If the most conspicuous purpose of logic lies in the justification and criticism of inference in science and everyday discourse as Quine has said, it seems unjustifiable to hold that an inference from one state of affairs to

---

28. Also |= [If Valid (P, .:Q) and Valid (R, .: S), then Valid ((P, .:Q) & (R, .: S))]
|= [If Valid (P, .:Q) and Valid (Q, .: R), then Valid ((P, .:Q) & (Q, .: R))]
|= [If Valid (P, .:Q) and Valid (Q, .: R), then Valid (P, .: R)]
|= [If Valid (P, .: Q) and Valid (R, .: ~P), then Valid ((P, .: Q) & (R, .: ~P))]
|= [If Valid (P, .:Q) and Valid (~P, .: R), then Valid ((P, .:Q) & (~P, .: R))]
But NOT, If Valid ((P, .:Q) & (R, .: S)) then VALID((P&Q), .: (R&S))]
Since '[If Valid ((P, .:Q) & (~P, .: R) then Valid((P&~P), .: (Q&R))]' would violate the consistency requirement for A-validity.

another could be valid when the two states of affairs could not co-exist because they are inconsistenct. A more extended discussion of this consistency requirement is presented in Section 6.341.

5.52 A-valid Inferences of M-logic

Although no theorems (i.e., tautologies) of M-logic are valid in A-logic because none are C-conditionals, many M-valid argument forms and principles of inference applied in M-logic are A-valid. That M-logic calls these argument forms "valid" is what gives M-logic its plausibility.

Given the definition of A-validity above, and the fact that all theorems of Chapters 1, 2, 3, and 4 are inconsistency-free, we can derive the following 110 theorems about valid inferences from the SYN- and CONT-theorems of those chapters without worrying about the consistency requirement. The TF-conditionals and TF-biconditionals derivable from these theorems are all tautologous and thus theorems of M-logic, but they are not theorems or valid statements of A-logic. However, the inference-forms and argument-forms derivable from them in A-logic are valid, and assertions that they are valid are validity-theorems of A-logic.

Obviously, since all of the A-logic theorems referred to are based solely on the meanings of the syncategorematic "logical constants", the inferences are similarly based, so the validities established are validities of pure formal logic; i.e., they are VALID (with capital letters). When they are based on a CONT-theorem we place an 'a' after the number of the theorem. When they are based on a SYN-theorem, we place a 'b' after the number of the theorem; in the latter case obviously the converse inferences will also be VALID.

From the definition of 'Valid in A-logic, three convenient derived rules follow:

DR5-6a. [If (P CONT Q) & not-Inc (P & Q)) then Valid(P, .:Q)]

Proof:	1) (P CONT Q) & not-Inc(P & Q)	[Premiss]
	2) Valid[P, .:Q]	[Df 'Valid']
	3) [If (P CONT Q) & not-Inc (P & Q)) then Valid(P, .:Q)]	[1) to 2), Cond. Proof.]

DR5-6b.[If (P SYN Q) and not-Inc (P&Q), then Valid (P, .:Q)]

Proof:	1) (P SYN Q) and not-Inc(P&Q)	[Premiss]
	2) P SYN Q	[1) SIMP]
	3) (P CONT Q)	[2), DR1-011, MP]
	4) not-Inc(P&Q)	[1) SIMP]
	5) ((P CONT Q) & not-Inc(P&Q))	[3),4),ADJ]
	6) Valid (P, .:Q)	[5),DR5-6a]
	7) [If (P SYN Q) and not-Inc (P&Q), then Valid (P, .:Q)]	[1) to 6),Cond. Proof]

DR5-6c.[If (P SYN Q) and not-Inc (P & Q), then Valid (Q, .:P)]

Proof:	1) (P SYN Q) and not-Inc(P & Q)	[Premiss]
	2) P SYN Q	[1) SIMP]
	3) (Q CONT P)	[2),DR1-12,MP]
	4) not-Inc(P & Q)	[1) SIMP]
	5) not-Inc(Q & P)	[4), Ax.1-03,SynSUB]
	6) (Q CONT P) & not-Inc(Q & P)	[3), 5) ADJ]
	7) If (Q CONT P) & not-Inc (Q & P)) then Valid(Q, .:P)	[DR5-6a(Re-lettered)]
	8) Valid(Q, .:P)	[6),7),MP]
	9) [If (P SYN Q) and not-Inc (P&Q), then Valid (Q, .:P)]	[1) to 8),Cond. Proof]

DR5-6d.[If ((P SYN Q) and not-Inc (P&Q) then Valid((P, .:Q) & (Q, .: P))]

<u>Proof</u>: 1) ((P SYN Q) and not-Inc (P&Q)) [Premiss]
2) Valid (P, .:Q) [1),DR5-6b,MP]
3) Valid (Q, .:P) [2),DR5-6c,MP]
4) (Valid(P, .:Q) & Valid(Q, .: P)) [2),3),ADJ]
5) Valid((P, .:Q) & (Q, .: P)) [4),Df 'Valid-&']
6) [If ((P SYN Q) and not-Inc (P&Q) then Valid((P, .:Q) & (Q, .: P))] [1) to5),Cond. Proof]

Since all of CONT- and SYN theorems of Chapters 1 through 3 are negation-free and thus incapable of bing *prima facie* inconsistent, every CONT- and SYN-theorem in those chapters will satisfy the requirement that the conjunction of antecedent and consequent not be inconsistent. And since all of the CONT- and SYN-theorems of Chapter 4 used only the restricted rule, U-SUBab, which does not allow any atomic wff to occur both POS and NEG in a theorem, all of them may be assumed to satisfy the same requirement. Thus we can apply DR5-6a,, DR5-6b or DR5-6c directly to the SYN- and CONT-theorems in Chaptes 1 through 4, to derive schemata of valid inferences.as follows:

<u>From SYN- and CONT-theorems in Chapter 1</u>:	<u>From:</u>	
T5-101b. VALID[P, .: (P&P)]	[&-IDEM]	[Ax.1-01,DR5-6b]
T5-102b. VALID[P, .: (PvP)]	[v-IDEM]	[Ax.1-02,DR5-6b]
T5-103b. VALID[(P&Q), .: (Q&P)]	[&-COMM]	[Ax.1-03,DR5-6b]
T5-104b. VALID[(PvQ), .: (QvP)]	[v-COMM]	[Ax.1-04,DR5-6b]
T5-105b. VALID[(P&(Q&R)), .: ((P&Q)&R)]	[&-ASSOC]	[Ax.1-05,DR5-6b]
T5-106b. VALID[(Pv(QvR)), .: ((PvQ)vR)]	[v-ASSOC]	[Ax.1-06,DR5-6b]
T5-107b. VALID[(Pv(Q&R)), .: ((PvQ)&(PvR))]	[v&-DIST-1]	[Ax.1-07,DR5-6b]
T5-108b. VALID[(P&(QvR)), .: ((P&Q)v(P&R))]	[&v-DIST-1]	[Ax.1-08,DR5-6b]
T5-111b. VALID[P, .: P]		[T1-11,DR5-6b]
T5-112b. VALID[((P&Q) & (R&S)), .: ((P&R) & (Q&S))]	[T1-12,DR5-6b]	
T5-113b. VALID[((PvQ) v (RvS)), .: ((PvR) v (QvS))]		[T1-13,DR5-6b]
T5-114b. VALID[(P & (Q&R)), .: ((P&Q) & (P&R))]		[T1-14,DR5-6b]
T5-115b. VALID[(P v (QvR)), .: ((PvQ) v (PvR))]		[T1-15,DR5-6b]
T5-116b. VALID[(Pv(P&Q)), .: (P&(PvQ))]		[T1-16,DR5-6b]
T5-117b. VALID[(P&(PvQ)), .: (Pv(P&Q))]		[T1-17,DR5-6b]
T5-118b. VALID[(P&(Q&(PvQ))), .: (P&Q)]		[T1-18,DR5-6b]
T5-119b. VALID[(Pv(Qv(P&Q))), .: (PvQ)]		[T1-19,DR5-6b]
T5-120b. VALID[(P&(Q&R)), .: (P&(Q&(R&(Pv(QvR)))))]		[T1-20,DR5-6b]
T5-121b. VALID[(Pv(QvR)), .: (Pv(Qv(Rv(P&(Q&R)))))]		[T1-21,DR5-6b]
T5-122b. VALID[(Pv(P&(Q&R))), .: (P&((PvQ)&((PvR)&(Pv(QvR)))))]		[T1-22,DR5-6b]
T5-123b. VALID[(P&(Pv(QvR))), .: (Pv((P&Q)v((P&R)v(P&(Q&R)))))]		[T1-23,DR5-6b]
T5-124b. VALID[(Pv(P&(Q&R))), .: (P&(Pv(QvR)))]		[T1-24,DR5-6b]
T5-125b. VALID[(P&(Pv(QvR))), .: (Pv(P&(Q&R)))]		[T1-25,DR5-6b]
T5-126b. VALID[(P&(PvQ)&(PvR)&(Pv(QvR))), .: (P&(Pv(QvR)))]		[T1-26,DR5-6b]
T5-127b. VALID[(Pv(P&Q)v(P&R)v(P&(Q&R))), .: (Pv(P&(Q&R)))]		[T1-27,DR5-6b]
T5-128b. VALID[((P&Q)v(R&S)), .: (((P&Q)v(R&S)) & (PvR))]		[T1-28,DR5-6b]
T5-129b. VALID[((PvQ)&(RvS)), .: (((PvQ)&(RvS)) v (P&R))]		[T1-29,DR5-6b]
T5-130b. VALID[((P&Q)&(RvS)), .: ((P&Q) & ((P&R)v(Q&S)))]		[T1-30,DR5-6b]
T5-131b. VALID[((PvQ)v(R&S)), .: ((PvQ) v ((PvR)&(QvS)))]		[T1-31,DR5-6b]
T5-132b. VALID[((PvQ)&(RvS)), .: (((PvQ)&(RvS)) & (PvRv(Q&S))] "Praeclarum"		[T1-32,DR5-6b]

T5-133b. VALID[((P&Q)v(R&S)), .: (((P&Q)v(R&S)) v (P&R&(QvS))] [T1-33,DR5-6b]
T5-134b. VALID[((P&Q)v(R&S)), .: (((P&Q)v(R&S)) & (PvR) & (QvS))] [T1-34,DR5-6b]
T5-135b. VALID[((PvQ)&(RvS)), .: (((PvQ)&(RvS)) v (P&R) v (Q&S))] [T1-35,DR5-6b]
T5-136a. VALID [(P&Q), .: P] From CONT-theorems: [T1-36,DR5-6a]
T5-137a. VALID [(P&Q), .: Q] [T1-37,DR5-6a]
T5-138a. VALID [(P&Q), .: (PvQ)] [T1-38),DR5-6a]
T5-122a(1) VALID[(Pv(P&(Q&R))), .: P] [T1-22c(1),DR5-6a]
T5-122a(1) VALID[(Pv(P&(Q&R))), .: (PvQ)] [T1-22c(2),DR5-6a]
T5-122a(2) VALID[(Pv(P&(Q&R))), .: (PvR)] [T1-22c(3),DR5-6a]
T5-122a(4) VALID[(Pv(P&(Q&R))), .: (Pv(QvR))] [T1-22c(4),DR5-6a]
T5-122a(1,2) VALID[(Pv(P&(Q&R))), .: (P&(PvQ))] [T1-22c(1,2),DR5-6a]
T5-122a(1,3) VALID[(Pv(P&(Q&R))), .: (P&(PvR))] [T1-22c(1,3),DR5-6a]
T5-122a(2,3) VALID[(Pv(P&(Q&R))), .: ((PvQ)&(PvR))] [T1-22c(2,3),DR5-6a]
T5-122a(2,4) VALID[(Pv(P&(Q&R))), .: ((PvQ)&(Pv(QvR)))] [T1-22c(2,4),DR5-6a]
T5-122a(3,4) VALID[(Pv(P&(Q&R))), .: ((PvR)&(Pv(QvR)))] [T1-22c(3,4),DR5-6a]

From CHAPTER 3 SYN and CONT theorems with Quantifiers

From SYN-theorems in Chapter 3:	Quine's Metatheorems	
T5-311b. VALID[($(\forall x)_n$ Px), .: (Pa_1 & P_2 &…& P_n)]		[T3-11,DR5-6b]
T5-312b. VALID[($(\exists x)_n$ Px), .: (Pa_1 v P_2 v…v P_n)]	|	[T3-12,DR5-6b]
T5-313b. VALID[(∀x)(Px & Qx), .: ((∀x)Px & (∀x)Qx)]	ML*140	[T3-13,DR5-6b]
T5-314b. VALID[(∃x)(Px v Qx), .: ((∃x)Px v (∃x)Qx)]	ML*141	[T3-14,DR5-6b]
T5-315b. VALID[(∀x)(∀y)Rxy, .: (∀x)(∀y) Rxy]	ML*119	[T3-15,DR5-6b]
T5-316b. VALID[(∃x)(∃y)Rxy, .: (∃y)(∃x)Rxy]	ML*138	[T3-16,DR5-6b]
T5-317b. VALID[(∀x)(P & Qx), .: (P & (∀x)Qx)] } "Rules of Passage"	ML*157	[T3-17,DR5-6b]
T5-318b. VALID[(∃x)(P v Qx), .: (P v (∃x)Qx)] }	ML*160	[T3-18,DR5-6b]
T5-319b. VALID[(∃x)(P & Qx), .: (P & (∃x)Qx)] }	ML*158	[T3-19,DR5-6b]
T5-320b. VALID[(∀x)(P v Qx), .: (P v (∀x)Qx)] }	ML*159	[T3-20,DR5-6b]
T5-321b. VALID[(∀x)Px, .: ((∀x)Px & (∃x)Px)]		[T3-21,DR5-6b]
T5-322b. VALID[(∃x)Px, .: ((∃x)Px v (∀x)Px)]		[T3-22,DR5-6b]
T5-323b. VALID[(∃x)(Px & Qx), .: ((∃x)(Px & Qx) & (∃x)Px)]		[T3-23,DR5-6b]
T5-324b. VALID[(∀x)(Px v Qx), .: ((∀x)(Px v Qx) v (∀x)Px)]		[T3-24,DR5-6b]
T5-325b. VALID[((∀x)Px & (∃x)Qx), .: ((∀x)Px & (∃x)(Px & Qx))]		[T3-25,DR5-6b]
T5-326b. VALID[((∃x)Px v (∀x)Qx), .: ((∃x)Px v (∀x)(Px v Qx))]		[T3-26,DR5-6b]
T5-327b. VALID[(∃y)(∀x)Rxy, .: ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]		[T3-27,DR5-6b]
T5-328b. VALID[(∀y)(∃x)Rxy, .: ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]		[T3-28,DR5-6b]
T5-329b. VALID[(∀x)(Px v Qx), .: ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx))]		[T3-29,DR5-6b]
T5-330b. VALID[(∃x)(Px & Qx), .: ((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx))]		[T3-30,DR5-6b]
T5-331b. VALID[(∀x)(∀y)Rxy, .: ((∀x)(∀y)Rxy & (∀x)Rxx)]		[T3-31,DR5-6b]
T5-332b. VALID[(∃x)(∀y)Rxy, .: ((∃x)(∀y)Rxy & (∃x)Rxx)]		[T3-32,DR5-6b]
From CONT-theorems in Chapter 3:		
T5-333a. VALID [(∀x)Px, .: Pa]		[T3-33,DR5-6a]
T5-334a. VALID [(∀x)(∀y)Rxy, .: (∀x)Rxx]		[T3-34,DR5-6a]
T5-335a. VALID [(∃x)(∀y)Rxy, .: (∃x)Rxx]		[T3-35,DR5-6a]
T5-336a. VALID [(∀x)Px, .: (∃x)Px]	ML*136	[T3-36,DR5-6a]
T5-337a. VALID [(∃y)(∀x)Rxy, .: (∀x)(∃y)Rxy]	ML*139	[T3-37,DR5-6a]

T5-338a. VALID [((∀x)Px v (∀x)Qx), .: (∀x)(Px v Qx)]		ML*143	[T3-38,DR5-6a]
T5-339a. VALID [(∀x)(Px v Qx), .: ((∃x)Px v (∀x)Qx))]		ML*144	[T3-39,DR5-6a]
T5-340a. VALID [(∀x)(Px v Qx), .: ((∀x)Px v (∃x)Qx))]		ML*145	[T3-40,DR5-6a]
T5-341a. VALID [((∀x)Px v (∃x)Qx), .: (∃x)(Px v Qx)]		ML*146	[T3-41,DR5-6a]
T5-342a. VALID [((∃x)Px v (∀x)Qx), .: (∃x)(Px v Qx)]		ML*147	[T3-42,DR5-6a]
T5-343a. VALID [(∀x)(Px & Qx), .: ((∃x)Px & (∀x)Qx)]		ML*152	[T3-43,DR5-6a]
T5-344a. VALID [(∀x)(Px & Qx), .: ((∀x)Px & (∃x)Qx)]		ML*153	[T3-44,DR5-6a]
T5-345a. VALID [((∀x)Px & (∃x)Qx), .: (∃x)(Px & Qx)]		ML*154	[T3-45,DR5-6a]
T5-346a. VALID [((∃x)Px & (∀x)Qx), .: (∃x)(Px & Qx)]		ML*155	[T3-46,DR5-6a]
T5-347a. VALID [(∃x)(Px & Qx), .: ((∃x)Px & (∃x)Qx)]		ML*156	[T3-47,DR5-6a]

<u>From SYN-theorems in Chapter 4</u>:

T5-411b. VALID [(P&Q), .: ~(~Pv~Q)]		[DeM2]	[T4-11,DR5-6b]
T5-412b. VALID [(PvQ), .: ~(~P&~Q)]	[Df ‘v’]	[DeM1]	[T4-12,DR5-6b]
T5-413b. VALID [(P&~Q), .: ~(~PvQ)]		[DeM3]	[T4-13,DR5-6b]
T5-414b. VALID [(Pv~Q), .: ~(~P&Q)]		[DeM4]	[T4-14,DR5-6b]
T5-415b. VALID [(~P&Q), .: ~(Pv~Q)]		[DeM5]	[T4-15,DR5-6b]
T5-416b. VALID [(~PvQ), .: ~(P&~Q)]		[DeM6]	[T4-16,DR5-6b]
T5-417b. VALID [(~P&~Q), .: ~(PvQ)]		[DeM7]	[T4-17,DR5-6b]
T5-418b. VALID [(~Pv~Q), .: ~(P&Q)]		[DeM8]	[T4-18,DR5-6b]
T5-419b. VALID [P, .: (PvP)]		[v-IDEM]	[T4-19,DR5-6b]
T5-420b. VALID [(PvQ), .: (QvP)]		[v-COMM]	[T4-20,DR5-6b]
T5-421b. VALID [(Pv(QvR)), .: ((PvQ)vR)		[v-ASSOC]	[T4-21,DR5-6b]
T5-422b. VALID [(P&(QvR)), .: ((P&Q)v(P&R))]		[v&-DIST]	[T4-22,DR5-6b]
T5-424b. VALID [(∃x)~Px, .: ~(∀x)Px]	[Q-Exch2]	ML*130	[T4-24,DR5-6b]
T5-425b. VALID [(∀x)~Px, .: ~(∃x)Px]	[Q-Exch3]	ML*131	[T4-25,DR5-6b]
T5-426b. VALID [(∃x_1)…(Ea_n)~$P<x_1,\ldots,x_n>$, .: ~(∀x_1)…(∀x_n)$P<x_1,\ldots,x_n>$]	[Q-Exch4]	ML*132	[T4-26,DR5-6b]
T5-427b. VALID [(∀x_1)…(∀x_n)~$P<x_1,\ldots,x_n>$, .: ~(∃x_1)…(∃x_n)$P<x_1,\ldots,x_n>$]	[Q-Exch5]	ML*133	[T4-27,DR5-6b]

<u>From CHAPTER 4, SYN- and CONT-theorems (with TF-conditionals)a</u>.

T5-430b. VALID [(P ⊃ Q), .: ~(P&~Q)]	[Df ‘⊃’]		[T4-30,DR5-6b]
T5-431b. VALID [(~PvQ), .: (P ⊃ Q)]			[T4-31,DR5-6b]
T5-432b. VALID [(P ⊃ Q), .: (~Q ⊃ ~ P)]			[T4-32,DR5-6b]
T5-433b. VALID [(∃x)(Px ⊃ Qx), .: ((∀x)Px ⊃ (∃x)Qx)]		ML*142	[T4-33,DR5-6b]
T5-434b. VALID [(∀x)(Px ⊃ Q), .: ((∃x)Px ⊃ Q)]	“Rules of	ML*162	[T4-34,DR5-6b]
T5-435b. VALID [(∀x)(Px ⊃ Q), .: ((∃x)Px ⊃ Q)]	Passage”	ML*161	[T4-35,DR5-6b]
T5-436a. VALID [((∃x)Px ⊃ (∀x)Qx), .: (∀x)(Px ⊃ Qx)]		ML*148	[T4-36,DR5-6a]
T5-437a. VALID [(∀x)(Px ⊃ Qx), .: ((∀x)Px ⊃ (∀x)Qx)]		ML*101	[T4-37,DR5-6a]
T5-438b. VALID [(∀x)(P ⊃ Qx), .: (P ⊃ (∀x)Qx)))]		(Thomason’s Axiom 4)	[T4-38,DR5-6b]
T5-439a. VALID [(∀x)(Px ⊃ Qx), .: ((∃x)Px ⊃ (∃x)Qx)]		ML*149	[T4-39,DR5-6a]
T5-440a. VALID [((∃x)Px ⊃ (∃x)Qx), .: (∃x)(Px ⊃ Qx)]		ML*150	[T4-40,DR5-6a]
T5-441a. VALID [((∀x)Px ⊃ (∀x)Qx), .: (∃x)(Px ⊃ Qx)]		ML*151	[T4-41,DR5-6a]

5.53 M-valid Inferences Which are Not A-valid

There are vast classes of inferences which are M-VALID but not A-VALID.

First, are the pairs, P and Q, in which P is the denial of some tautology, and Q is any wff whatever. In these cases M-VALID[P, **.:** Q]. The class of wffs which are denials of tautologies (hence inconsistent) is infinite, and for each of these there is the greater infinity of all wffs, each one of which is M-validly inferable from one of the former. None of these inferences are A-valid, because the conjunction of premiss and conclusion is invariably inconsistent due to the inconsistency in the premiss.

Second, there is the different set of pairs of wffs, P and Q, in which P is any member of the infinite set of wffs, and Q is any of the infinite number of tautologous wffs. For all of these pairs, M-VALID[P, **.:** Q]. Some of these are A-valid, e.g., A-VALID[(Pv ~ P), **.:** ((P&P) ⊃ P)]; but, infinitely many are not, including all in which Q has occurrences of atomic formulae which do not occur in P.

Finally, there are the infinities of pairs of wffs, P and Q in which both P and Q are contingent, but

1) [P & ~Q] is INC, i.e., [P ⊃ Q] is TAUT, and
2) Q has some atomic wffs which do not occur in P, i.e. [P CONT Q] false.

For all of these, M-VALID[P, **.:** Q] and for none of these A-VALID[P, **.:** Q]

In A-logic, [P ⊃ Q] is a Tautology in all of the three cases above, while [P, **.:** Q] is not A-valid in any of them, except as noted. Tautologous truth-functional conditionals are not generally translatable into A-valid inferences. From the point of view of intelligent ordinary language and A-logic, the VC\VI principle fails when the conditional is a truth-functional conditional. To save the compatibility of M-logic with the VC\VI principle, non sequiturs had to be called valid inferences. A-validity eliminates these non sequiturs and with them the anomalies of "material and strict implication" as they pertain to inference and argument.

Given the meaning of the truth-functional conditional as expressed in conjunction, disjunction and denial, there are no anomalies at all in declaring the tautologousness of [P ⊃ Q] in the case above. But on the account of C-conditionals in the next chapter, which is closer to ordinary usage of "if…then", the anomalies would be very real if [P => Q] were valid in these cases. However, this is not the case. The anomalous "valid" conditionals are eliminated in the same way the related non sequiturs are are eliminated from A-valid inferences.

Discussion of the similarities and differences between the concepts of validity in M-logic and A-logic will be continued in Sections 6.33, 6.4, 10.1 and elsewhere in chapters that follow.

Section C

Analytic Logic

Chapter 6

"If…then" and Validity

6.1 Introduction

In this chapter a non-truthfunctional meaning of "[If P then Q]" is introduced along with the symbol '[P => Q]' to represent that concept. All possible purely formal logical structures are adequately represented by predicate schemata with placeholders for designations of individual entities, connected or operated on by the syncategorematic expressions 'and', 'or', 'not', 'if…then' and 'all' or 'some', arranged in syntactical groupings and ordered sequences. Thus the system of purely formal A-logic is brought to completion by adding to the material in preceding chapters, its version of conditional expressions together with an appropriate axiom and rules.

Although arguments and inferences which contain no conditionals can be valid in A-logic (as we saw in Section 5.5), the only *statements* that are valid in A-logic are conditional statements. A-logic's concepts of conditionality and validity allow adherence to the VC\VI principle without having to call any inferences or statements that are patently non-sequiturs, "valid".

6.11 A Broader Concept of Conditionality

In the previous chapters there has been no need to introduce a primitive symbol for the connective 'If…then…' in order to construct meaningful well-formed formulae. Wherever '⊃' occurs it may be read as an abbreviation for 'Either not-P or Q' or for 'Not both P and not Q'; no wff used thus far, including '⊃', *had* to be construed as representing a conditional expression.

However, conditional statements are essential components of logic and ordinary reasoning and M-logic could not have claimed to be a general Logic, if it had not interpreted some of its symbols as conditionals. In deference to this, we treat '⊃' as standing for a "truth-functional conditional".

In A-logic, when talking *about* wffs—in particular, in presenting rules of inference—we have used 'if…then' in ways which are essential for logic. In this context, and in the wider context of ordinary language and scientific reasoning, the issue is how best to define or interpret the ordinary words "If…then…" and its dictionary synonyms for purposes of logic.

The concept of conditionality described and formalized in this chapter is broader in certain respects, and narrower in other respects, than that of the truth-functional conditional of M-logic. To distinguish the two concepts involved, while avoiding fruitless discussion about the true or correct meaning of the word ‘conditional’, we use the term ‘TF-conditionals’ for the truth-functional expressions called conditionals in mathematical logic and ‘C-conditionals’ for the different expressions we call “correlational conditionals” in analytic logic. We hold that the C-conditional is the more useful of the two for purposes of logic and rational reasoning, as well as being closer in many ways to ordinary and traditional usage of the words ‘if…then’.

The two concepts differ in several respects. In the first three respects the C-conditional is broader than the TF-conditional; in the last respect it is narrower.

First, C-conditionals are conceived as occurring primarily in predicates, and secondarily in sentences; TF-conditionals are conceived as occurring primarily in sentences.

Second, C-conditionals occur with predicates or sentences in the imperative, interrogative, subjunctive, etc., as well as the indicative mood; TF-conditionals are indicative sentences only.

Third, the concept of C-conditionals in A-logic is based on the many different ways that conditionals are used, including its uses in reasoning about what to do; it is not based solely on the capacity of a conditional to be true or false.

Although determinations of the truth or falsehood of a conditional are often very important, the basic analysis of ‘if…then’ in A-logic is independent of truth or falsehood (e.g., conditional predicates are neither true nor false). The underlying concept of a C-conditional, common to all its uses, is more like that of an admission ticket than a description which is true or false.[1] A ticket allows passage from one place to another. It is not true or false, but it may or may not be valid. The problem is to account for both this basic characteristic, and for truth conditions of conditional statements. (In A-logic, valid statements cannot be false, but may or may not be true.) The assignment of ‘if…then’ as the meaning of ‘⊃’ in M-logic is based on the assumption that a conditional is fundamentally a truth-assertion. Thus the truth and/or falsehood of its components are considered the exclusive determinants of the truth, the meaning and the validity, of “if…then” statements.

Finally, according to A-logic, expressions called “valid” TF-conditionals in M-logic (its M-valid theorems), are neither logically valid (i.e., valid in A-logic) nor conditionals—they are merely tautologies. Further, among the many ‘ => ’-for-‘⊃’ analogues of M-logic theorems, only a restricted sub-class are valid in A-logic. In this sense A-logic is not as broad as M-logic; it counts many less arguments and statements as valid than M-logic. On the other hand, there are infinitely more inconsistencies and tautologies involving C-conditionals, than for their ‘⊃’-for-’ => ’ analogues among TF-conditionals of M-logic. So the class of tautologies is larger in A-logic.

6.12 Role of Chapter 6 in this Book and in A-logic

The system of purely formal A-logic is developed apart from concepts of truth or falsehood, and thereby provides the base for extensions into the logics of various non-formal operators, including those for truth-logic, modal logic, a logic of directives, a logic of value judgments, deontic logic, a logic of probability, and as well as logics of many substantive predicates including ‘is identical with’ and ‘is a member of’. Logics of directives, questions, ought-statements, etc., will be touched on only briefly in passing in this book. Previously they have presupposed M-logic as a base. Their development from A-logic will require additional studies. Only the logic of truth-assertions will be developed fully in Chapters 7 through 10.

---

1. Gilbert Ryle characterized hypothetical, law-like statements as “inference tickets” in *The Concept of Mind*, 1949, p 121.

It is important to see this Chapter 6 as the formal foundation upon which future extensions of logic, including truth-logic, can be built; it is the first part of a job which can be continued in Chapter 7 to 10, and in other future investigations. But the major objective in this book is to provide satisfactory solutions to the problems of truth and conditionality which arose in conjunction with efforts to apply mathematical logic to science and ordinary usage. Both mathematics and the natural sciences are essentially related to truth-determinations, as is much of common sense, history and logic itself.

The achievement of the other tasks is facilitated by focusing first on conditional expressions which do not involve truth-claims, thus separating properties and relations based on the meaning of 'It is true that…' from those based solely on the meanings of syntax and syncategorematic locutions, including 'if…then', which is the business of formal logic. In this chapter for the first time we apply the concept of 'Validity' as defined in formal A-logic to C-conditionals along with arguments and inferences.

Still the major objective of this book can not reach full fruition until it deals with the logical relations between conditionals and the concept of truth. This is done in Chapters 8 and 9, which concentrates on the truth-logic of conditional expressions. By "the truth-logic of conditionals" we mean both (i) the logic of assertions that conditional statements are true or false and (ii) the logic of conditional relationships between statements conceived as true or not-true.

In Chapter 7, we add axioms and definitions relating to the predicate '<1>is true' to the system developed through Chapter 5 without C-conditionals. A trivalent truth-functional model for a semantics of M-logic is provided in that chapter, in effect giving a formal axiomatic system, based on A-logic, for a somewhat revised version of M-logic's "semantic theory".

In Chapter 8, the truth-operator is both applied to components of C-conditionals, and to C-conditionals themselves. A sharp distinction is maintained in between 1) the concept of the truth or falsity of a conditional as a whole and 2) the concepts of validity or invalidity of a conditional which asserts, for example, that *if* P is true, *then* Q is true. Chapter 8 adds two axioms relating to truth and falsehood of C-conditionals; from these many valid theorems of truth-logic follow.

In the rest of Chapter 8 and particularly in Chapter 9, it is shown how A-logic solves or avoids the problems of omission or commission which became apparent as attempts were made to apply M-logic to certain types of reasoning in the natural sciences and common sense.

6.13 Outline of Chapter 6: A-logic

To put the abstract concept of C-conditionality in perspective, Section 6.2 begins by discussing uses of conditional expressions which do not involve truth-claims. This includes the uses of conditional predicates and of conditional sentences or predicates in the imperative, interrogatory, indicative and subjunctive moods and their uses in fiction or humor. Then it lays out four generic, logically indeterminate, characteristics of contingent conditional expressions. These are features of conditional expressions which are independent of the logic of T-operator or other operators.

In Section 6.3 the formal logic of C-conditionals is presented. We first present rules of formation, definitions and a single axiom which are added to those of Chapters 1 through 5. The concept of inconsistent conditionals is defined, and the principle of VI\CI—that if an inference is valid, then the conditional with its conclusion as the consequent, and the conjunction of its premisses as its antecedent is valid—is introduced as rule of inference. Following this SYN- and CONT- theorems, TAUT- and INC-theorems, and VALIDITY-theorems with C-conditionals are proven. With C-conditionals added to the system, it is possible to construe all principles of inference stated earlier the book as conditional statements in the sense of C-conditionals. However, the proofs of the validity of these principles must await Chapter 8, for there is an important sense in which they all lay claim to being true.

The system of this chapter constitutes the system of Formal Analytic Logic, in the same sense in which the propositional calculus and first order predicate logic constitutes Formal Mathematical Logic. The substantive differences and similarities between M-logic and A-logic are found by comparing the M-valid theorems with TF-conditionals in M-logic with the validity- theorems with C-conditionals in A-logic.

6.14 On the Choice of Terms in Logic

In developing analytic logic there has been a recurring problem of choosing names for concepts. The terms "synonymy", "containment", "validity" and "implication" especially when prefaced by the adjective "logical", have been chosen because they are related to important previous uses of those terms. But definitions and uses of those terms in A-logic can not be said to represent *exactly* any previously existing concepts named in logic or in ordinary language.

Although these words (without the prefix) and other terms in A-logic occur frequently in ordinary (English) discourse, it can not be claimed that there are fixed, objective, meanings or concepts pre-existing in the minds of all ordinary users which are exactly captured by them. It is sufficient that the ways these words are used in A-logic are similar in their results to a large and important class of ways in which they are used in ordinary language and logical reasoning. The terms 'implication' and 'valid' are widely used by (English speaking) logicians. Different logicians have used them in different ways. To distinguish the unique meanings of these words in A-logic we frequently prefix 'A-' to them, as in 'A-validity' and 'A-implication'. But 'A-validity' and 'A-implication' do not, to my knowledge, correspond exactly to any previously formalized terms in logic, though they may be closer than some of the latter to ordinary usages of 'valid' and 'implication'.

However, the meaning A-logic gives to 'valid' is compatible with the usage in M-logic and common usage in at least the following respects:

(i) if [P, **.:**Q] and [P $\Rightarrow$ Q] are A-valid, then [If P then Q] is logically unfalsifiable, [P $\supset$ Q] is M-valid (i.e., tautologous) and [P & $\sim$Q] is inconsistent.
(ii) if [P, **.:**Q] and [P $\Rightarrow$ Q] are A-valid, then it is not (logically) possible for both P to be true and Q to be false; in this sense P Strictly implies Q.

On the other hand, the concepts in A-logic of an A-valid inference and an A-valid conditional differ from all previous concepts of validity in a) requiring containment and b) requiring that the conjunction of antecedent and consequent be consistent.

(Comparisons of A-implication and A-entailment to other uses of 'implication' and 'entailment' in logic or ordinary usage, is presented in Chapter 7 and elaborated in Chapter 8. These concepts are introduced in the context of truth-logic and its presuppositions.)

The inconsistencies and tautologies of A-logic are more numerous than in M-logic, primarily because the C-conditional is inconsistent (and its denial tautologous) when the conjunction of antecedent and consequent is inconsistent. These inconsistencies are added to those of conjunction and disjunction.

The appropriate question for logicians is whether the stronger uses of 'A-valid' and the other terms preclude lines of reasoning which could reasonably be considered non-problematic and valid. Is A-logic "complete" in the ill-defined but important sense that it can include all patterns of logical deductions which intelligent reasoners would want to recognize as valid?

6.2 Generic Features of C-conditional Expressions

Conditional expressions include conditional predicates, conditional predicate-schemata, conditional sentences and conditional sentence-schemata (conditional sentences are "saturated" conditional predicates). "Saturated" conditional predicates are predicates in which all argument positions are occupied by individual constants or bound variables which range over the designata of singular terms, i.e.,sentences. Such sentences have the logical properties and stand in the logical relations of their conditional predicates.

Conditional predicates and sentences occur in the indicative mood, the subjunctive mood, the imperative mood and the interrogatory mood, among others. Conditional expressions are used in discourse and inquiries about what is, was, or will be true in fact, what ought to be (or ought to have been) the case morally or theoretically, and in imaginative explorations of the logically possible in art and metaphysics as well as other ways. They are used in mathematics and logic, in all of the natural sciences and social science, in literature and music, humor and theology.

Formal logic should be conceived and developed to operate on all kinds and moods of conditionals, substituting synonyms for synonyms, identifying containments and inconsistent or tautologous expressions, and determining all kinds of implications and logically valid entailments available when dealing with each particular kind of predicate or operator or mood of expression.

6.21 Uses of Conditionals Which Do Not Involve Truth-claims

Mathematical logicians and most philosophers have concentrated on the use of conditionals in the search for truth. There is a good reason for this interest, but to treat this use as the fundamental use and one to which all others must be reduced is to ignore certain essential features of the concept of conditionality as it is employed in practice.

To offset the exclusive preoccupation with the truth or falsity of conditionals, consider the wide implicit and explicit use of hypothetical or conditional imperatives. Every human being carries around countless conditional predicates (implicit or explicit) that are extremely useful, though as predicates they are neither true nor false. Hypothetical imperative sentences, though neither true nor false, are the best mode of expressing directions, instructions and commands. Conditional predicates and conditioned imperatives or instructions which are used to guide action, are like instruments—like rakes in the garden house ready to be used when the leaves fall. They don't apply to everything at all times. They only come into use when conditions described by the antecedent obtain and certain purposes are involved. Sometimes such predicates are congered up and used before we are sure how useful they will be.

6.211 The Ubiquity of Implicit and Explicit Conditionals

We are engulfed, mentally and externally, by sets of directives which are tied, explicitly or implicitly, to conditional predicates or sentences in the imperative mood—expressions which can take the form "If you wish such-and-such as result, do so-and-so!". Examples can be found everywhere: from toilet training to cookbook recipes, to rules for driving the car, manuals of instructions on how to operate a computer, or directions for opening a cereal box; instructions for building entertainment centers from do-it-yourself sets, instructions for drawing up a will or filling out income tax forms, standard operating procedures for requisitioning supplies, how to find your way around the town, etc. etc.

By "an explicit conditional" I mean a conditional predicate or sentence of the form 'If…then…' which is actually uttered or written, and synonyms of such expressions like 'Q, if P', 'Q provided P', 'Whenever P, Q", etc..

The term 'implicit conditionals' is used to refer to conditional statements which people would agree represented the intent or thought behind certain explicit written or spoken language. For example, consider recipes in a cookbook. They usually appear in the form indicated in Figure 1 below. The connective 'if…then…' does not occur explicity anywhere in the recipe, but in general if a recipe is found in a cook book, the following kind of conditional, based on that recipe, would be understood and accepted by the reader and assumed to be in accord with the intent of author and publisher:

(1) **If** you wish to make a (e.g., gingerbread),
then assemble the ingredients listed and process them according to the instructions!

This statement is a conditional imperative. It is neither true nor false. Nor is it really a command. It is a suggestion (which may or not be followed) of what the reader might do if they want to actualize a certain imaginable state of affairs (e.g., having actual gingerbread before one ready to be eaten). To deny that such conditionals express either an implicit intention of the author or a rule of use for the user, is to fail to understand what a cook book is all about.

Figure 1

FORMAT:	**Example:**	
I. NAME OF FOOD	**Ginger bread**	
II. LIST OF INGREDIENTS WITH AMOUNTS	1 cup molasses	½ tspn nutmeg
	½ cup brown sugar	1 tspn ginger
	½ cup oil	1 cup boiling water
	1 tspn cinnamon	2 ½ cups flour
	½ tspn cloves	1 tspn baking soda
III. INSTRUCTIONS ON WHAT TO DO WITH INGREDIENTS	Blend together the first 7 ingredients and stir in boilng water. Mix in flour. Disolve baking soda in hot water. Add batter. Pour into greased pan. Bake 30 minutes at 350°F.	

The concept of the food (e.g.,gingerbread) conveyed in the antecedent is of a kind of physical entity with a distinct taste, consistency and color quite different from the taste, consistency and color of any of the ingredients listed in the antecedent, or of any other edible substance mentioned elsewhere in the cook book. The correlation of the specific ingredients and instructions with the concept of gingerbread is unique: if you interchange any words in the list of ingredients (e.g., if you interchange 'flour' and 'oil' to get '2½ cups oil' and '½ cup flour') or replace any ingredient by another (e.g., '½ cup salt' in place of '½ cup brown sugar') we have a very different conditional, one which does not occur and is not intended by anyone.

Judgments concerning the reliability of a conditional predicate in the imperative mood of the form,

'If you wish <1> to occur, then do <2>!'

are usually related to the frequency of true instantiations of the indicative conditional predicate,

'If <2> is done, then <1> occurs.'

In so far as we rely on the cook book (unreliable cook books are rarely published) the reader assumes that the author's directives are "backed up" in the sense that the author would endorse the following conditional and expect the reader to believe it.

(2) **If** the list of ingredients are assembled and processed in accordance with the instructions, **then** the result will be a (e.g., gingerbread).

This is a conditional in the indicative mood. If and when the antecedent obtains, its says, the consequent will be will be realized. If this is false, the recipe is a bad one. The reliability of the conditional imperative depends on a relation of the indicative conditional to truth. Syntactically, the two conditionals differ:

(1) If you wish to obtain P, then do Q! (imperative)
(2) If Q is done, then P will be obtained. (indicative)

But the concepts of the states of affairs described by P and Q are the same in both.

The significance of the truth-claim is clear if ones tries to imagine the set of constructible conditional statements with the same antecedent or with the same consequent which would be false. For example take any other list of ingredients and instructions in the rest of the cookbook; the statement that they would result in gingerbread would be false. Or, put the name of any other food except gingerbread as title of the recipe keeping the list of ingredients and instructions the same, and the result would be false. Add to these falsehoods, the falsehoods gotten by keeping the same ingredients but in wildly different relative amounts, or by replacing the ingredients with sets of ingredients not found elsewhere in the cook book including components which are non-edible or poisonous. The number of possible falsehoods outrun what we can conceive. The value of the conditional lies in correlating specific ingredients and processes with a specific kind of result.

In speaking of 1) and 2) as "implicit" conditionals relative to recipes in a cook book, I do not mean that they are necessarily present in the reader's or author's unconscious (though a case can be made for this). Rather, I mean that a) these two conditionals would not be rejected by anyone, as not being in accord with the intent of the author and the presuppositions of the user, while b) the infinity of other false conditionals would be rejected outright as contrary to or not in accord with the intent of the authors or the understanding of the user.

What has just been said about recipes in a cook book applies, with some refinements, to chemical formulae in chemistry, to formulae in algebra, to manuals for computer software.

Consider the following excerpt from a computer manual:

3) DISKCOPY (Copy Diskette) Command

Purpose: Copies the content of the diskette in the source drive to the diskette in the target drive.
Format: DISKCOPY [d:][d:]
Remarks: The first parameter you specify is the source drive. The second parameter is the target drive.

Such an entry is readily used to formulate a conditional predicate in the imperative mood such as

4) If <1> is a diskette & <2> is a diskette & you wish to copy the contents of <1> onto <2>, then put <1> in Drive A & put <2> in drive B & type 'DISKCOPY a:b:' at the prompt & press the key marked 'ENTER"!

On first learning to use the computer we, or our teacher, may very explicitly translate (or transform) the entry in the manual into just such an expression. As operations with the computer become more routine, we forget the statement but automatically act in accordance with it when we want the stated result. It may even be difficult to re-capture the rule after it becomes habit; but we are implicitly guided by that rule. If any one were to offer any of a thousand other different instructions that might be put in the consequent (almost none of which would accomplish the desired result mentioned in the antecedent), we would quickly recognize that the latter were wrong (unreliable) rules, and that the initial conditional imperative was the 'right one'.

It is hard to classify grammatically the parts of 3), the original entry in the manual, as being in imperative or indicative moods because not all parts are sentences; and we can not pick out parts which would be judged, prior to translation, as true or false. But we can learn how to translate these entries to create appropriate sentences, as in 4).

As a conditional imperative, 4) is neither true nor false. It is a rule that we can use to guide actions, but (theoretically) it may or may not be reliable or useful. The reliability of that conditional imperative depends on whether instantiations of the following indicative conditional predicate are true.

5) **If** <1>is a diskette & <2>is a diskette & <1>is put in Drive A & <2>is put in drive B & 'DISKCOPY a:b:' is typed at the prompt & and the key marked 'ENTER" is pressed,
then the content of <1> will be copied onto <2>.[2]

This statement is in the indicative mood and is (theoretically) capable of having true or false instantiations. It omits the reference to an agent's wishes in 4), puts the descriptive content of the imperative in the consequent of 4) into the antecedent, puts a description of what might be wished for in the antecedent, into the consequent, and changes the consequent from imperative to indicative mood. It describes one possible state of affairs in the antecedent and another state of affairs in the consequent and says that if and when the first state of affairs is realized, the second will follow. The antecedent of 5) describes states of affairs that I can bring about on my computer. Thus by realizing one state of affairs through willing physical acts by my fingers, I (by means of the machine) bring about a different state of affairs which I desire and could not have had otherwise.

Computer manuals are filled, page after page, with instructions which are interpretable without loss of significant meaning into two quite different types of conditionals:

a) conditional predicates in the indicative mood, of the form
"If <2> takes place, then <1> results,"
b) useful rules in the form of a conditional imperative,
"If you wish the result <1>, then bring about <2>!"

The reliability of the directions in b) depends on the extent to which instantiations of the related indicative conditional in a) happen to be true.

Every key on the keyboard of my computer has a implicit conditional imperative and a conditional indicative associated with it: "If the key with 'T' on it is pressed, the letter 't' will appear on the screen.", "If you wish a capitalized letter to appear on the screen, press the shift key and the key with that

---

2. This is an over-simplified and slightly inadequate account of the standard DOS instruction for copying one diskette to another, but perhaps it serves to indicate how all instructions in a computer manual are in principle equivalent to conditional imperatives.

letter on it!". There are 45 keys plus twelve "function keys" with the marks 'F1', 'F2', etc., on them. If I press the shift key and the key marked 'F7', then a list of seven things related to a print-out apppears on the screen and it is made clear what will come next if I press the key with '1' on it, or the key with '2' on on it, etc. All of these "if then's" presuppose that I have followed certain prior "if…then"s like "if you wish the computer to go on, turn the switch on the right side", "if you wish to used Word Perfect, type 'WP' after the DOS prompt"., etc.

The significance of the specific conditionals which can be drawn from entries in the computer manual, is apparent if we consider the unreliable or false conditionals that theoretically could be (but seldom are) presented. For each of the 45 keys on the keyboard there are 44 possible unreliable or misleading conditional imperatives prescribing the wrong key to achieve the same result. These wrong instructions parallel wrong predictions in the indicative mood. Sometimes the right result may not appear on the screen because the machine was not turned on, or because the machine is operating under the DOS software rather than Word Perfect. But given these other necessary conditions, the indicatives are often true and never false and the rules are reliable. Thus sometimes (indeed often) but not always, when I pressed the key with 'T' marked on it, the letter 't' appeared by the cursor on the screen. But never under any conditions, when I press the key with 'T' marked on it, does the letter 'x' or 'r' or any other appear next to the cursor. The conditional 'If the key marked 'T' is pressed, then 'x' (or 'r' or 'w', etc.) appears on the screen' though never mentioned and never thought of, has no true instances, and would be proven false whenever the key marked 'T' was pressed. The negative conditional 'If the key marked 'T' is not pressed, 't' will not appear on the screen, is true all of those moments when I was not using the computer as well as the moments when I was using the computer but not pressing the key marked 'T'. The correlations described in the reliable and true conditionals are unique, selected from an enormous variety of other conditionals that would be false, less reliable, or totally unreliable.

Learning to use the computer, like learning to use the typewriter, to keep grandfather's clock running, to drive a car, to cook, to communicate in language, requires learning reliable conditional rules—assimilating and accepting for guidance ideas which are fully expressible only in conditional form. Though we do not long retain in memory, or verbalize, such specific conditionals—our actions becoming matters of habit—we can tell whether an explicitly stated rule is in accord with our modes of acting or not.

The same approach applies to any algorithm of mathematics, e.g., the rule "To multiply a^n by a^m, add the exponents of the factors to obtain the exponent of the product!" may be read as, "*if* you wish to name the product of $a^{<1>}$ and $a^{<2>}$, *then* add $<1>$ and $<2>$ and put this sum as the exponent of a !". For example, suppose I wish to know the product of 2^5 and 2^9; if I follow the imperative in the consequent of the conditional, I will say this product is $2^{(5+9)}$ i.e., 2^{14}. Using such a conditional directive correctly has nothing to do with its being true or false. (I could just as correctly follow a rule similar to the one given above except that 'add' was replaced by 'subtract', though the result would not be true).

Whether such a rule is useful, reliable rule or not—in this case, whether it will lead to a true result—is very much tied to questions of truth or falsity. Its usefulness, or validity, is tied to whether the equation '$(a^n \text{ X } a^m) = a^{n+m}$' is true for every a and every n and m. In the particular case above we can show this is true: $2^5 = 32$, $2^9 = 512$, 32 X 512 = 16,384 and $2^{14} = 16{,}384$. But the larger questions is whether the following conditional predicate is true for all instantiations:

> If $<1>$ is a number & $<2>$ is a number & a is a number & b is the sum of $<1>$ and $<2>$, then $(a^{<1>} \text{ X } a^{<2>}) = a^b$,

This must be proven from the foundations of mathematics.

Rules of inference, including those of pure logic, have this same character. As rules, they may be viewed as conditional predicates in the imperative mood. They are neither true nor false, being predicates and non-indicative. In general, they are reliable rules for passing from synonymies or containments to other synonymies or containments, from a tautologous predicate to another tautologous predicates, from inconsistencies to other inconsistences, etc. This permits inferences of imperatives from other imperatives, questions from other questions, statements from other statements. In truth-logic, the rules of inference are used when one wishes to pass from the possible truth of one sentence to the possible truth of others which would "follows logically". Proofs of the reliability of a rule of inference in truth-logic consists of demonstrations that it is possible for them to lead from a true premiss to a true conclusion and that the rule could not lead from a truth to a non-truth. The usefulness of rules of logical inference depends not on their being true, but on their being valid. What we mean by 'valid' is spelled out in Section 5.5 and 6.34.

The concept of conditionality I have tried to convey is partly related, I believe, to ideas expressed by Gilbert Ryle in 1949, when he spoke of lawlike statements (universally generalized conditionals) as "an inference ticket (a season ticket) which licenses its possessors to move from asserting factual statements to asserting other factual statements."[3] A season ticket to a series of concerts is not used for all purposes, but is brought forth and displayed when one wishes to be admitted to a concert. It satisfies a necessary condition for getting admitted to the concert. Indicative conditionals are rarely universal truths, true of all things. If true, they usually signify unique correlations between different kinds of states or affairs, events or ideas. Conditional imperatives suggest ways (of more or less reliablity) to bring about states of affairs if we wish to do so.

6.212 Logical Schemata of Conditional Predicates

Mathematical logic, being concerned with truth and extensions, conceives universality as expressed primarily by "universal quantifications" of the form '(∀x)Px' or '(∀x)(If Px then Qx)'. There is another sense of universality. Some unextended predicates, of the forms 'P<1>', or '(P<1> ⇒ Q<1>)', can be intensionally universal in the sense that we can know **a priori** that they can have true instances but can not have false instances. We hold the predicate in mind and know it to be "universal" though we neither enumerate nor define by description the infinite variety of its possible instances. All tautologous predicates are of this type. Such are the predicates '(~P<1> v P<1>)' and '(T(P<1> & Q<2>) ⇒ T(Q<2>)'. If a predicate has universal intension, its conjunctive quantification will have universal extension.

Thus we consider some structures of conditional predicates which have various degrees of universality. Conditional predicates in general are of the form '$(A<t_1,\ldots,t_n> \Rightarrow C<t_{n+1},\ldots,t_{n+m}>)$' where each t_i is an argument position holder, not necessarily different from others in the predicate. Here 'A' is used to stand for the predicate's antecedent and 'C' is used to stand for the predicate's consequent. Either or both A and C may be a quantificational predicate. The antecedent and consequent may be simple or compound.

In ordinary language, some of the simpler conditional predicates in the indicative mood might be expressed thus:

3. Ryle, Gilbert, *The Concept of Mind*, 1949, p 121.

1) "…is bald, if old and bald"
2) "…is male, if a nephew",
3) "…is in Europe, if in France"
4) "…has a black bib, if a chicadee"
5) "…boils at 212° F., if water at sea level"
6) "…dies in two years, if has AIDS"
7) "…lights, if a match which is struck"
8) "…causes death, if swallowed by a human"

In M-logic, the logical structures of such predicates are sometimes displayed in the form of conditional propositional functions with free variables (e.g., '(If Px then Qx)', which Quine calls an "open sentences"). We use conditional predicates, of the form 'If P<1> then Q<1>', with argument position holders.[4] Such predicates convey ideas which are useful. They are schemata which are brought into play when the predicate in the antecedent finds a subject to which it applies. They have a kind of generality—they are ready to be used whenever the antecedent applies, no matter what it may apply to.

Using argument position holders instead of free variables and the non-truth-functional '=>' for "if…then" instead of the truth-functional '⊃', the predicates above are expressed as,

1) "(<1>is old & <1>is bald) => <1>is bald"
2) "<1>is a nephew => <1>is male"
3) "<1>is in France => <1>is in Europe"
4) "<1>is a chicadee => <1>has a black bib"
5) "(<1>is water & <1>is at sea level & <1>is heated to 212° F) => <1>boils".
6) "(<1>has AIDS at time<2> & <3>is a time two years after <2>)
   => <1>dies before<3>."
7) "(<1>is a match & <1>is struck) => <1>lights"
8) "<1>is swallowed by<2> => <1>causes<2>to die"

There is no limit to the size and complexity of conditional predicates, just as there is no limit to the size and complexity of statements. Every statement may be conceived as having a single over- all predicate; in complicated cases, it is always a logical structure of many elementary predicates. The predicates above are all relatively simple ones.

The syntactical structure of predicates is displayed in logical schemata. The following schemata represent the logical structures of each of the predicates above, taken singly:

1) ((P<1> & Q<1>) => Q<1>)
2) (P<1> => Q<1>)
3) (P<1,a> => P<1,b>)
4) (P<1> => Q<1>)
5) ((P<1> & Q<1> & R<1>) => S<1>)
6) ((P<1,2> & R<3,2>) => Q<1,3>)
7) ((P<1> & Q<1>) => R<1>)
8) (R<1,2> => Q<1,2>)

---

4. As mentioned in previous chapters we avoid free variables on the ground that variables have clear meanings only when bound; free variables are ambiguous.

Only the first schema displays a predicate structure which is logically determinate; the antecedent contains the consequent. The other schemata are logically indeterminate. The results of putting different singular terms in the argument positions and various predicates for the predicate letters, covers the whole range of logical properties possible for conditional predicates.

Conditional predicates may be logically, or semantically, or theoretically unfalsifiable; or they may be logically, semantically or theoretically inconsistent.

6.213 Conditional Imperatives and Questions, vs. Indicative Conditionals

Conditional predicates, as well as conditional sentences, may be indicative, imperative or interrogative.

While the antecedent of a conditional statement is always in the indicative mood, its consequent may be in the indicative, imperative or interrogative mood. The mood of a conditional statement as a whole is determined by the mood of the consequent. For example,

If it is hot, the door is closed. (Indicative conditional)
If it is hot, close the door! (Conditional imperative)
If it is hot, is the door closed? (Conditional interrogative),

The logic of conditionals which are imperatives, directives or questions will be a special sub-logic based in part on what is meant by an expression's being in the imperative, interrogatory, etc., mood. Such principles are not reducible to principles in the logic of indicatives or to truth- logic. This makes the discipline of logic more complex than mathematical logic would have it. But rational and logical thinking is not necessarily simple; it deserves an organon suited to its degree of complexity.

Like conditional statements, conditional predicates and predicate schemata have moods—indicative, imperative, interrogative, etc; but the antecedent must be descriptive or indicative.

a) "drive slowly, if icy!":
"If <1>is a road which is icy, then drive slowly on<1>!"
[P<1> => Q<1> !] [Imperative Mood]
b) "Does<2>drive slowly on<1> if <1>is a road which is icy?"
[P<1> => Q<2,1> ?] [Interrogative mood]
c) "<2>drives slowly if it is icy"
"If <1>is an icy road, then <2>drives slowly on<1>."
[P<1> => Q<2,1>]" [Indicative Mood]

Logic can operate on all such predicates and any sentences which instantiate them, substituting synonyms for synonyms, identifying containments and inconsistent or tautologous expressions, and sometimes determining what follows logically from what depends on the meanings of the component expressions.

Although an indicative antecedent can not logically entail a predicate in the imperative or interrogatory mood, one imperative predicate may logically contain or entail another, and one question may logically contain or entail another question. For example, it could be argued that since 'is<1>a mother?' semantically entails 'does<1>have a child?', therefore "If <1>asks 'is Jean a mother?' then <1>asks 'Does Jean have a child?'" is a semantically valid conditional (but not a logically valid one). On the other hand, expressions of the form '(P? => Q?)' (e.g., "If is Jean a mother? then is Jean a female?") are not grammatically acceptable.

Imperative or interrogatory predicates may also include sentential operators involving truth, obligation, possiblitity, value,etc.

"Is it true that<1>is a Catholic, if it is true that<1>is Irish?" (T(Irish<1>) => T(Catholic<1>?)

This predicate can easily be found false—but can be useful as a schema for investigating the frequency of being Catholic, if Irish.

"<1>ought to keep a promise, if <1>made the promise"
"If <1> promised to do A, => <1> ought to do A" (T[Prom(do A)] => O[do A]

"If T(P<1> & Q<2>) is possible, then T(R<1,2>) is possible.
M[T(P<1> & Q<2>)] => M[T(R<1,2>)]

The point is that the discipline of logic should be conceived and developed to operate on all kinds or predicates and sentences substituting synonyms for synonyms, identifying containments and inconsistent or tautologous expressions, and determining what kinds of logically valid inferences, conditionals, entailments and implications are available in dealing with that particular kind of predicate or operator or mood of expression.

6.214 Merely Descriptive Indicatives: Fiction and Myth

Even in the indicative mood, not all statements (including conditionals with both antecedent and consequent in the indicative mood) are presented as truth-claims. Indicative conditionals in jokes, fiction and myth, like other indicative sentences in those contexts, are not required or expected to be true of the actual world. They are, so to speak, directives for the imagination with no implicit claims to truth. Even more than other kinds of indicative sentences, an indicative [P => Q] by itself is not necessarily intended to be taken as '[P=>Q] is true '.

Explicit conditional statements are the exception rather the rule in fiction, though they may occur more prominently in some genres (detective stories) than others. When used they presuppose a field of reference, but it is a field constructed by the novelist to advance the story, not a field treated as a reality external to the book, and not necessarily one that obeys the law of non-contradiction.

Consider the conditional in the following passage about a "romantic young lady" in Fyodor Dostoyevshy's *The Brothers Karamazxov*, (p.2), who:

> …ended by throwing herself one stormy night into a deep river from a high bank, almost a precipice. And so she died, entirely to satisfy her own whim, and to be like Shakespeare's Ophelia. **If** this precipice, a chosen and favorite spot of hers, had been less picturesque, **if** there had been a prosaic flat bank in its place, most likely the suicide would never have taken place. This is a fact, and there probably have been a few similar cases in the last two or three generations.

The conditional here is used to stress the romantic motive in the suicide, by picturing conditions without which it would not have happened. Or consider, in the same book (Ch 2):

> …a faithful servant of the family, Gregory, took the three-year old Dmitri into his care. **If** this servant hadn't looked after him there would have been no one even to change the child's little shirt.

This conditional helps portray Dmitri's father, Fyodor Karamazov, as uncaring for his own child.

Both examples hypothesize a state of affairs differing from the field of reference otherwise painted by the novelist, and say what would have happened in that case to show something of the motives of its characters. They are in the subjunctive mood; the antecedent is contrary to facts in the field of reference the novelist has sketched. But what they describe does not conflict with the novel's field of reference because the antecedent is understood as hypothesizing a state of affairs which is not part of the field of reference. There is no question of truth involved; the novelist could have chosen a contrary conditional had he wished to paint a different picture. But the coherence of the novel may depend on consistency of motives pictured in this way with actions or motives elsewhere pictured as part of the story.

An intelligent reader of fiction frequently makes logical deductions and inferences based on the meanings of terms in order to understand the story, usually trying to keep the picture consistent. The statement from Greek mythology,

Chiron was a centaur and a son of Cronos.

combined with our understanding of the meaning of 'centaur',

'<1> is a centaur 'syn '<1> is an animal with a man's torso on a horse's body'

yields an obvious inference from the first statement:

Chiron was a creature with a man's torso on a horse's body.

This process does not necessarily involve or imply any truth-claims; it relies only a logical inference, based on a meaning postulate, to fill in the picture being painted.

In religion many theologians hold that the laws of truth-logic do not obtain in all matters. Kierkegaard held that "the knight of faith" was he who could believe a contradiction despite its violating "laws of reason"—e.g., that Jesus was both a finite man and the non-finite God. Thomas Aquinas, though subjecting natural science to laws of logic, held that there are theological truths which can not be "grasped by reason" including the apparent contradiction that God is both one person and three distinct persons. Such positions do not deny the presence of a logical inconsistency; but they attribute it to limitations of human faculties. It is not clear how they hold that it also reflects a truth, and such positions conflict with traditional logic. Theologicans differ on whether religious statements and theologies should be taken as truth-claims and be subject to the same principles of truth-logic as apply to natural science, natural history, and common sense.

The point of this and preceding sections is to free the study of the logic of conditionals from exclusive preoccupation with logical principles based on the concept of the operator 'It is true that', or the predicate '<1>is true', or other special operators. Formal analytic logic is based on the meanings of syncategorematic words and syntactical stuctures. Expressions containing truth-claims are but one—though an important one—of several major sub-classes of expressions in logic's field of reference.

The way is opened for a wide variety of new investigations which are not confined to truth-logic; the logics of imperatives, questions, directives, ought-statements (deontic logic), modal logic epistemic logic, logic of causal statements etc. Efforts to construct logics in these areas as extensions of M-logic, have not acquired general acceptance. Though alternative approaches will not be developed in this book, we will suggest some benefits for those logics if they start with A-logic and C-conditionals instead of M-logic and TF-conditionals.

6.22 General (Logically Indeterminate) Properties of Contingent Conditionals

Any attempt to list properties or features common to all conditional predicates or sentences must apply to conditionals which are synthetic, non-analytic, and contingent as well as conditionals with logical properties. The common features, according to the meaning we ascribe here to "if…then", may be characterized as those of a "correlational" conditional (abbreviation: 'C-conditional'), vs. a "truth-functional" conditional (abbr: 'TF-conditional').

This general meaning is proposed for all conditional expressions whether predicates or sentences, whether contingent, inconsistent, tautologous or valid, and regardless of grammatical mood or mode of acceptance (as true, as right, as correct).

Let [A => C] abbreviate any expression [If A then C], provided "if A then C" is taken as a C-conditional with 'A' representing the antecedent, 'C' representing the consequent, 'A' and 'C' being place-holders for either predicates or sentences. [A => C] has five general features:

1) While the consequent, C, may be in the declarative, interrogatory, or imperative, mood, the antecedent must be in a declarative or descriptive mood.
2) The consequent, C, and thus the conditional as a whole, is intended to be *applied or used* only on occasions or in contexts in which the antecedent is imagined, believed, or known to be fulfilled.
3) C-conditionals convey the concept of correlating the occurrence of the antecedent with the co-occurrence in some context of antecedent and consequent.
4) There is a sense in which conditionality expressed by 'if…then' is an ordering relation; if the antecedent and consequent exchange positions, the meaning is changed.
5) The use of the connective 'if…then' is inappropriate if no connection can be conceived between subject(s) in the consequent and subject(s) in the antecedent, or between the predicates in the consequent and the predicates in the antecedent.

These features are discussed in Sections 6.221 to 6.225 below.

6.221 The Antecedent is Always Descriptive

While the consequent, C, may be in the declarative, interrogatory, or imperative, mood, the antecedent must be in a declarative or descriptive mood.

Expressions of the forms 'If P? then Q' and 'If P! then Q' are ungrammatical and not well-formed. But we include in the logical treatment of conditionals, both indicative conditionals, subjunctive conditionals, hypothetical imperatives and conditional questions, while pointing out that the antecedent must always be declarative sentences or descriptive predicates.

The following represent four different moods of conditionals.

1) If it is cold, then open the door!	Imperative mood
2) If it is cold, is the door open?	Interrogatory mood
3) If it is cold, the door is open.	Indicative mood
4) If it were cold, the door would be open.	Subjunctive mood

The mood of the conditional as a whole is determined by the mood in the consequent as indicated for 1) to 4). All are meaningful and usable in human affairs. All are potentially subject to rules and principles of logic, though the logic may differ for conditionals in different moods.

A hypothetical imperative, e.g., 'If it is cold, open the door!' has an indicative or declarative antecedent and a consequent in the imperative mood. The activation of the consequent is contingent upon having antecedent fulfilled. A conditional question, "If it is cold, is the door open?"—again has an declarative antecedent, and an answer to the question is relevant only on the assumption that the antecedent is satisfied.

To see that the antecedent is always indicative consider whether imperative or interrogative antecedents make grammatical sense. I know of no conditionals with forms like

> If open the door! then…
> If is the door closed?, then…

which occur in ordinary discourse. It makes no sense to put non-descriptive components in the antecedent, as in, 1') If open the door! then it is cold, or 2') If is the door open? then it is cold.

Interrogatory and imperative expressions should not be confused with indicative sentences which *mention* them. 'If the king commanded, 'open the door!', then the door ought to be opened' is a perfectly grammatical conditional ought-statement, and a legitimate object of logical scrutiny. But here the antecedent is not the imperative, 'Open the door!' but the indicative sentence, 'the king commanded, 'open the door!'', which satisfies the requirement that the antecedent be in the declarative mood.[5]

The general rule, then, is that the antecedent must be declarative or descriptive; i.e., in the indicative or subjunctive mood.

If it is argued that *not all* conditional sentences in ordinary discourse fit this requirement, the counter-examples should be examined and considered in terms of their uses. But in any case, there is a wide usage of conditionals in which all antecedents are descriptive (including subjunctive descriptions), though the consequent may not be descriptive, and we shall define C-conditionals as conditionals that meet this requirement, believing this covers all useful kinds.

By including conditional imperatives (or directives or suggestives) and conditional questions among the sentences subject to logical analysis and rule-making, we extend the reach of logic beyond the indicative conditionals which are true or false. Thus the scope of the logic of C-conditionals is broader than that of TF-conditionals, though with constraints.

6.222 The Consequent Applies Only When Antecedent Obtains

Given a C-conditional, [A=>C], the consequent, C, is brought into play (activated?) only when the antecedent is satisfied or considered to obtain. This may be called a principle of storage and use. Conditionals are like rakes and snow shovels in a garden house. They are not used all the time, and don't apply to everything, but it is advantageous to have them stored and ready for use, so that when the appropriate antecedent condition is met, the consequent can be brought into play and activated.

The conditional as a whole is intended to be evaluated for acceptance or rejection only with respect to cases in which the antecedent is assumed, imagined, believed, or known to be satisfied. If the antecedent does not apply, the conditional does not apply though it is not discarded. No determinations of the applicability, acceptability, truth, etc., of the consequent are relevant to a judgment of the acceptability of the conditional as a whole unless the antecedent is assumed to obtain.

---

5. A logic of imperatives will presumably include, $\models$ [Do(A v B)! CONT (Do A! v do B!)] and we may want to say that "Do(A v B)!" *logically contains* "(Do A! v do B!)". But "**If** Do(A v B)**! then** (Do A**!** v do B**!**)]" remains ungrammatical.

This principle of storage and use holds—whether [A => C] is a conditionalized predicate proposed as instrumental, a conditional statement relating two truth-claims, a conditioned moral imperative, or a conditional question relevant to some issue.

The principle of storage and use can be represented in a a 3X3 matrix. Let '0' refer to cases in which the antecedent or the consequent are inapplicable or irrelevant; let '1' stand for cases in which the antecedent or consequent are accepted and '2' for cases in which they are rejected. As the matrix suggests, the conditional as a whole is only accepted or rejected in those cases in which the antecedent is both relevant and accepted and the consequent can be either accepted or rejected. In other cases the conditional as a whole is inapplicable and neither accepted nor rejected (represented by '0'). A trivalent matrix of this form is introduced in Chapter 8 when dealing with the truth-logic of conditionals.[6] The principle of storage and use is more advantageous in its logical consequences, and is closer to ordinary usage than the alternatives which are embedded in the TF-conditional. Trivalent truth-tables for C-conditionals with truth-claims as components are discussed in specific detail in Chapter 8.

		C		
	(A => C)	0	1	2
	0	0	0	0
A	1	0	1	2
	2	0	0	0

6.223 Conditional Expresses a Correlation

An essential component of the meaning of the C-conditional is the concept of a correlation of the state of affairs suggested by by the antecedent with a state of affairs suggested by the consequent. This concept is independent of the truth or falsity of those ideas—i.e., it is not truth- functional.

The idea is that a conditional predicate "If P<1> then Q<2>" conveys the concept of correlating the applicability of [P<1>] with the applicability of [P<1> & Q<2>]. "If <1>is married, then <1>has a child", rightly or wrongly, correlates the concept of being married with that of having a child. Derivatively, indicative expressions of the form 'If S1 then S2' convey the idea of a correlation between states of affairs described by S1 and what is expressed by the conjunction of S1 and S2. In the presupposed field of reference, P is conceived as occurring *if and only if* P occurs with Q in some sense. This, I think, captures roughly what we mean by "a correlation" of one state affairs with another, or of two kinds of events.

This approach is reminiscent of Hume's analysis of causal propositions as involving constant conjunctions in contexts in which the two events involved are contiguous in time and/or space. For Hume the concept of correlations was logically prior to the concept of causality. (Not all conditional statements are causal statements). In a similar vein, I suggest that correlations are the kind of relation expressed in an indicative conditional, advocated in an imperative conditional, suggested or commanded in a directive or imperative conditional and inquired about in a conditioned question. What is correlated is 1) the concept in the antecedent (or what the concept applies to) with 2) the concept of conjoining the designata of the antecedent with that of the consequent.

A mutual correlation—two entities bearing a mutual or reciprocal relation, or invariably accompanying one another—is expressed most precisely in a biconditional. But a C-conditional [P => Q] is not the same as the biconditional 'P if and only if Q'; for the latter implies that Q will not occur without P or P without Q, while the former says only that P will not occur without Q. To preserve the asymmetry of the conditional then, we must say that P is correlated with the conjunction (P & Q), not with Q alone.

6. The truth-operator will yield the semantic Synonymy, T(P =>Q) Syn F(P => ~Q). This is not a pure formal Logical SYNONYMY due solely to syncategorematic operators, but a Synonymy due to the substantive concept of truth. See the discussion following T8-18 and T8-19 in Chapter 8, on why this Syn-theorem holds, but '(P => Q)' is *not* Syn to '~(P => Q)'].

When P occurs, both P and Q occur—(P => Q) or (P <=> (P & Q)). This is not the same as, "when Q occurs, both P and Q occur—(Q => P) or (Q <=> (P&Q))". (P & Q) is interchangable with P though is it not interchangeable with Q; [P => Q] means [P <=> (P & Q)]. This account fits the truth-tables of both TF-conditionals and indicative C-conditionals. In M-logic the truth-tables of (P ⊃ Q) and (P ≡ (P & Q)) are the same and in analytic logic (P => Q) and (P <=> (P&Q)) have the same trivalent truth-tables.[7]

This equivalence is not so clear in conditioned questions and hypothetical imperatives. It may seem awkward to say that a conditional (P => Q!), e.g., "If it is cold, close the door!" is synonymous with a biconditional, (P <=> (P & Q!)), e.g., "It is cold if and only if it is cold and close the door!"? Nevertheless, the C-conditional can connect the concept of being cold with the concept of both its being cold and a command to close the door. In hypothetical imperatives as well as in conditioned questions, there is still the concept of a one-way correlation—a state of affairs indicated by the antecedent is correlated with conjoining it to a proposed action or question; even though with the antecedent fullfilled, conditional imperatives and questions are neither true nor false.

This feature of the C-conditional, combined with the feature discussed in section 6.222, has a particularly neat relationship to the concept of conditional probability in standard probability theory. For what is more natural than to say that the probability of a correlation of an event-type a_1 with an event-type a_2 is found by giving the ratio of occurrences of a_1-with-a_2 to the occurrences of a_1 (with or without a_2)**?**

6.224 The Ordering Relation Conveyed by "If…then"

The inferential conditional, 'If P then Q' is non-commutative. This means that the order of presentation counts. The order of presentation is the left-to-right order (in most natural languages) on the printed page, and it is a temporal order in spoken language. In the case of '(P & Q)' and '(P v Q)' the referential meaning is independent of the order of presentation. Changes in the order of presentation do not change the referential meaning. Thus '(P & Q)' and '(P v Q)' are both commutative. This is is what Ax.103 [(P & Q) SYN (Q & P)] and Ax.104 [(P v Q) SYN (Q v P)] assert in Chapter 1.

In contrast, the order of presentation in '(P => Q)' carries a meaning. '(P => Q)' and '(Q => P)' do not mean the same thing; they are not SYN. 'P' and 'Q' remain the same; only the order of presentation differs, but this difference represents a difference in meaning. We shall say the order of presentation in 'If P then Q' **represents** a primal order in the field of reference; that the concept of such a primal order is part of its meaning.

M-logic proper has no expressions which convey the concept of a primal order. The connectives '&' and 'v' are commutative, and the negation operator, '∼', changes the meaning.

It may appear that '(P ⊃ Q)' and '(Q ⊃ P)' are non-commutative in the same way that '(P => Q)' and '(Q => P)' are, since they are neither synonymous nor truth-functionally equivalent. But this is misleading. Stripped of the abbreviation, '⊃', the non-equivalence of '(P ⊃ Q)' and '(Q ⊃ P)' is the non-equivalence of '(∼P v Q)' and '(∼Q v P)'.[8] Here the antecedent and consequent do not remain the same while changing position; rather, 'P' is replaced by '∼P' and '∼Q' is replaced by 'Q '. The elementary components of the two disjunctive expressions have different meanings. What makes them non-equivalent and non-synonymous is that they have different component expressions (each original component is negated), it is not a mere change in the order or presentation of two components with the same meaning.

---

7. See Chapter 8.
8. '∼(P & ∼Q)' and '∼(Q & ∼P)'.

The concept of a primal order in a field of reference, represented by the order of presentation of symbols, is very abstract, allowing much variation in its instantiations. Among its instantiations are such things as the temporal order of events which we directly apprehend, and the temporal order represented in calendars and historical chronology. Without any 'if' an expression, "P, then Q, then R, then S", suggests a temporal, or perhaps, hierarchical, or protocol order. The more basic concept of primal order is suggested by the concept of necessary or sufficient conditions. To say that P is a necessary condition of Q, is to say "if no P then no Q"; i.e., Q depends upon, and requires, P. To say P is a sufficient condition is to say "if P is the case, then Q applies", or "if P is the case, do Q!".

The basic primal order conveyed when we use 'If…then' is certainly an order in the structure of our ideas—and of the way we conceive the field of reference. But it can also be viewed as an order in an objective field of reference independent of our ideas. The concepts of physical causes, or physical conditions, or physical laws which account for observed events, are of sufficient or necessary conditions for unobserved events. This implicitly involves primal ordering relations, which can only be expressed with conditional statements. There are also concepts of moral order, and of what we might call means-end order: "if P occurs, then you ought to do Q", or "If you wish P, then do Q".

In a sense, P is prior to Q; Q is the dependent variable. Whether P and Q will co-relate is dependent on P's coming. P is the agent for bringing about the correlation of P and Q.

The simplest way to express it is that 'P => Q' means a one-way correlation; that given P, (P & Q) occur together in some way. This is not the same as saying, (Q => P), i.e., that given Q, (P & Q) occur together. In the first P is responsible for the correlation of P and Q, in the second Q is responsible for the correlation of P and Q. The order is different when that correlation is conditioned on P rather than Q, and vice versa.

Mathematical logicians represent various kinds of ordering relations extensionally, using set theory and M-logic. Relations are construed extensionally as sets of ordered-tuples, and ordered n-tuples are definable from the definition of an ordered pair of entities. Quine, following Frege, Peano and Weiner, defines the ordered pair $<a_1,a_2>$ of two distinct entities in a field of reference, as a two-member class whose sole members are (i) the class whose sole member is the class whose sole member is a_1 and (ii) the class whose sole members are the class whose sole member is a_1 and the class whose sole member is a_2.[9]

Representing 'the class whose sole member(s) are …' by the Greek letter '**ι** …', Quine's definition of an ordered pair would put '(**ιι**a_1 ∪ **ι**(**ι**a_1 ∪ **ι**a_2))' for our '$<a_1,a_2>$'. This has certain formal similarities with the ordering relation in (P => Q). For

|= [(P => Q) SYN (P <=> (P & Q)] But not: [(P=>Q) SYN (Q=>P)]
'$<a_1, a_2>$' for '(**ιι**a_1 ∪ **ι**(**ι**a_1 ∪ **ι**a_2))' but not: ['$<a_1,a_2>$' for '$< a_2, a_1 >$']
|= (P<=> (P & Q)) SYN ((P & Q) <=> P)] and not: [(P <=> (P&Q)) SYN (Q <=> (P&Q))]
'(**ιι**a_1 ∪ **ι**(**ι**a_1 ∪ **ι**a_2))' Syn '(**ι**(**ι**a_1 ∪ **ι**a_2) ∪ **ιι**a_1))' but not: '(**ιι**a_1 ∪ **ι**(**ι**a_1 ∪ **ι**a_2))' for '(**ιι**a_2 ∪ (**ι**a_1 ∪ **ι**a_2))'

The left to right order is not what gives the primal order, for in the correct Syn-statements, the left-to-right order of components changes. Rather, it is the internal relations between to two distinct entities. If one of them is primary (in the antecedent), then the other one is not primary in that conditional. To correlate (P & Q) are with P, is not the same as correlating (P & Q) with Q. The C-conditional "If P, then Q" means the same as "Q, if P" but not the same as "If Q then P". In A-logic, primal order is conveyed by '=>' .

9. See W.V.Quine, *Mathematical Logic*, 1981, Section 36, "Pairs and relations".

In M-logic genuine primal order is not addressed until set theory with relations. The set- theoretical definition above has the required formal structure and provides an effective syntactical criterion, but it is an artificial construct and appears to be an expedient notation rather than an explanation of the meaning of primal order.

The kind of order conveyed by a conditional is more abstract than the kinds definable in set theory. It is not even a quasi-ordering. Where all components are in the indicative mood, it might be said that (P => Q) represents a quasi-ordering, I.e., '=>' is reflexive and transitive, since '[P => P]' and '[((P=>Q) & (Q=>R)) => (P=>R)]' seems to hold. But if P is a directive or imperative, both of these schemata are ill-formed; "If do P! then do P!" makes no sense. If P is descriptive and Q and R are deontic, a case can be made for transitivity but not reflexivity: "If P is the case, Q ought to be, and if Q ought to be then R ought to be" seems to entail, if P is the case the R ought to be. But "if P is the case, then P ought to be" is not logically necessary. Where all components are in the descriptive mode, as in truth-logic, conditionality seems to represent a quasi-ordering relation—both reflexive and transitive. But it clearly does not represent a relation with the properties of being symmetric, asymmetric, antisymmetric or connected; it does not represent a strict simple ordering relation like "$<1>$ is greater than $<2>$", or a function like "$<1>$ is the sucessor of $<2>$". It represents a minimal and very abstract ordering, but one capable of expressing ordered relations of much more precise and varied sorts depending on what is put for P and what is put for Q. For example, the concept that R is a strict simple ordering (abbr. SSO), is expressed by '$\text{SSO[R] Syn}_{df} \models [((R<1,2> \,\&\, R<2,3>) \Rightarrow R<1,3>) \,\&\, {\sim}R<1,1>]$; any predicate, R, which contains this as a conjunct of its definiens, is a strict simple ordering.

6.225 Connection Between Consequent and Antecedent

It has been suggested many times that meaningful conditional statements assert or presuppose a "connection" of some sort between the antecedent and the consequent.[10] A-logic accepts this as a feature that the C-conditional can meet. There are different kinds of connections. The kind of connection needed to make a conditional logically valid is clear and is based on logical containment; the meaning of the consequent must be contained in the meaning (including presuppositions) of the antecedent. The kinds of connections which make a contingent conditional meaningful and "empirically valid"are looser and more varied. All of them share the common characteristic of asserting that **if** the antecedent obtains, **then** the consequent must obtain.

A general kind of connectness that includes, but goes far beyond logical containment, is the following:

> Requirement 5) In a fully expressed conditional either the primary subject term(s) in the consequent is (are) also subject term(s) in the antecedent, or the predicates in the consequent also occur in the antecedent.

This is a requirement which M-logic does not meet since, if any two statements are each either true or false (regardless of what they talk about) one may infer a TF-conditional which is either true or false depending only on the structures and truth-values of the two statements. If P is true and Q is false, then

---

10. See Storr's McCall, "Connexive Implication", *Journal of Symbolic Logic*, Vol 31 (1966), pp 415-433. McCall cited Aristotle, Chrysippus and Boethius as ancient advocates of a connection between antecedent and consequent, at least in the form that the denial of the consequent must be incompatible with the antecedent. This feature is incorporated in the truth-logic of conditionals in Chapter 8; but it is not quite the same notion of connection as is presented in this section.

$(Q \supset P)$, $(\sim P \supset \sim Q)$, $(Q \supset \sim P)$, $(P \supset \sim Q)$, $(\sim P \supset Q)$, and $(\sim Q \supset P)$, are true TF-conditionals regardless of whether P and Q have anything in common. The "paradoxes of strict implication", $[(P\&\sim P) \supset Q]$ and $[P \supset (Qv\sim Q)]$, are theorems of M-logic which sanction this disconnectedness.

In contrast, no C-conditionals can be logically A-valid unless they satisfy this Requirement of Connectedness since analytic containment, on which logical validity depends, requires that the subjects (as well as what is predicated of them) in the consequent must occur also in the antecedent.

But Requirement 5) is intended to be applied also to synthetic conditionals. It is satisfied by identity of subjects but with different things predicated of them. For example, all instantiations of 'If $<1>$is a human then $<1>$has twelve toes' will have the same subject in antecedent and consequent, but since the predicates are different and are not analytically connected, the conditional as a whole may be either true or false; it is contingent but meaningful. The subject in the antecedent and that in the predicate are connected by the identity relation; '$<1>$' occurs in both antecedent and consequent. A conditional of the form ((Pa & Rac) => Fc), e.g., "If a is a Parisian and a read the book c, then c was written in French" includes an co-existence connection between a person and a book, which (however dubious) satisfies requirement 5).

Requirement 5) would also be satisfied by a synthetic conditional in which different subjects occur in consequent and antecedent, but with the same predicate, e.g., "If a is a Samoan then b is a Samoan", or "If a was punished, b ought to be punished" though the justification of such a conditionals seems to require some further connection based on identity of subject (they were born in the same place, or they did the same bad deed). Again, the antecedent and consequent are connected by having a common term, but can be either true or false. It is contingent and meaningful.

Sometimes the antecedent and consequent are connected by reference to the same time frame "If sunspots flare, there will be a period of prosperity in the next year" says 'If at time t_1 the sun has sunspots, then within $t_{1\ +\ 1\ year}$ human society on earth will prosper'. The antecedent and consequent describe events connected by both being related to the same moment in time. It has the form '(If $P(a,t_1)$ then $Q(b,t_1)$' where 'a' denotes the sun, 'b' denotes human society on earth, '$P<1,t_1>$' is '$<1>$has sun spots at t_1' and '$Q<2,t_1>$' is '$<2>$ will prosper within a year after t_1'. The existence of the subjects within a space-time region beginning with t_1 is presupposed.

When conditionals are about facts in the actual world, the antecedent and consequent are connected by the relation of co-existing in in the actual world. An indicative C-conditional, taken as a whole, is shown to be true if and only if the conjunction of its antecedent and its consequent is shown true. Can we therefore form a true conditional from any two true statements, by making one the antecedent and the other the consequent? For example 'The Empire State Building is 1250 feet tall' and 'Caesar died' are both truein the actual world. At first it seems absurd to conclude that this establishes the truth in the actual world of 'If it is true that the Empire State Building is 1250 feet tall, then it is true that Caesar died'. There is no *logical* connection between what is talked about in the antecedent and what is talked about in the consequent. Yet, in discussing the logic of *truth-assertions*, we shall find that this kind of inference—from the premiss that P is true and Q is true, to the conclusion that (If P is true then Q is true) is true—is essential, if we are to account for the way in which true empirical generalizations can be logically based on particular facts. The silly, trivial cases, of inference to a conditional from facts are distinguished from useful, important cases, by their usefulness in making future judgements for predication, retrodiction and control of events.

In empirical sciences and common sense a conditional is often thought to be meaningful because it involves the concept of a causal connection between what is described by the antecedent and what is described by the consequent. Although, as Hume said, causal connections can not be directly observed (although subjects, times, places, and many other properties and relations can be) the evidence for postulating causal connections clearly depends on the evidence of occurrences and non-occurrences of pairs of events at particular times and places. We will talk about the concepts of empirical generalizations and causal connections in Chapters 9 nd 10, contrasting the inadequacy of M-logic, with A-logic and its C-conditionals.

Requirement 5) is far from a precise or complete formulation of what distinguishes a meaningful conditional from a mere conjunction of two statements, but it rules out many conditional expressions as lacking in meaning, and provides something to look for in assertaining the significance of a conditional.

Often synthetic conditionals are ambiguous because of failure to indicate the connection clearly enough. Such ambiguities have been used to argue that subjunctive conditionals are intrinsically incapable of being expressed in a formal notation analogous to the formal notation for TF-conditionals.[11] But this misplaces the source of the difficulty.

In any case, in illustrating both analytic and non-analytic cases of C-conditionals we shall assume requirement 5) holds. All logically valid C-conditionals satisfy this requirement, though not all tautologous TF-conditionals do. If there are significant contingent conditionals which do not meet this requirement, perhaps a more careful definition is needed. But we do not believe any important cases are excluded.

6.3 The Formal System of Analytic Logic

Formal logic is about the properties of, and relationships between, the meanings of compound expressions just so far as they depend on the order, repetition, and grouping of components and the meanings of the syncategorematic words, 'and', 'or', 'not', 'all', 'some', and 'if…then'. This holds for traditional logic, M-logic and A-logic.

Chapter 4 gives a putatively complete account of the relationships of logical synonymy and logical containment so far as they are determined by occurrences of the words 'and', 'not', 'or', 'all' and 'some' in expressions.

Chapter 5 gives a complete account of that fragment of A-logic which deals with the properties of inconsistency and tautology so far as they applies to expressions in which the words 'and', 'not', 'or', 'all' and 'some' occur. The meanings assigned to those words, as well as the meaning of 'inconsistency' and 'tautology' are in accord with meanings assigned in M-logic.

Although the symbol '⊃' was used extensively in Chapters 4 and 5 and its interpretation as 'if…then' in M-logic was noted, in A-logic, [P ⊃ Q] is interpreted simply as an abbreviation of [~(P&~Q] or its synonym [~PvQ], and is held to be inadequate as a symbol for 'if…then' in logical discourse. In A-logic "if…then" is associated with the symbol '=>', and its meaning differs radically from that of '⊃', as explained in the preceding sections of this chapter.

The formal system of A-logic is thus only completed with the introduction of a symbol for "if…then" with the characteristics and meaning explained in Sections 6.2 of this chapter.

---

11. Such is the conclusion suggested by Quine in response to the problem of choosing between 'If Bizet and Verdi had been compatriots, Bizet would have been Italian' and 'If Bizet and Verdi had been compatriots, Verdi would have been French.' [Quine, *Methods of Logic*, 1982, p 23, for further discussion see Quine, *Word and Object*, par 46, pp222-226.] But the reason for being unable to choose is that the antecedent does not make clear what we are talking about—what connections are presupposed. The predicate "If <1>is a compatriot of<2> then <2> is a citizen of<3>" is an incomplete, poorly-formed conditional schema. What is needed is something in the antecedent connecting <3> with <1> or with <2>. E.g. if either 'Bizet was a citizen of France' or 'Verdi was a citizen of Italy' (but not both) is added in the antecedent, then the appropriate consequent follows from the structure and the meanings of 'is a compatriot' and 'is a citizen of' .

6.31 The Base of Formal A-logic

To complete the axiomatization of A-logic as a purely formal system with only syncategorematic words we add '=>' as a primitive symbol as well as three definitions and one axiom in which '=>' occurs. In addition we add the VC\VI principle as a rule of inference. The result, which incorporates all results of the preceding chapters, is the axiomatic base of purely Formal A-logic, given below.

Theorems and principles of inference in preceding chapters are covered by the first five axioms of this base. All SYN- and CONT-theorems of Chapter 4, which incorporate the proofs and theorems of Chapter 1 to 3, and all INC- and TAUT- theorems from Chapter 5, where the definition of inconsistency is introduced are derivable. There is no need to derive them anew. Except for names and numbering nothing need be changed in either theorems or proofs.[12]

This base includes six clauses in addition to those which cover the cumulative results of the previous five chapters:

Primitive Symbols: '=>' is a predicate operator;
Rule of Formation: [P => Q] is a wff schema
Definitions: Df'<=>'. [(P <=> Q) SYN ((P => Q) & (Q => P))]
Df 'Inc =>'. [INC(P) Syn_{df} (P Syn (Q => R) & Inc(Q & R)]
Df 'Valid'. [Valid(P, .:Q) Syn_{df} (P CONT Q) & not-INC(P & Q)]
Axiom Ax.6: [(P & (P => Q)) SYN (P & (P => Q)) & Q)]
Rule R6-6. Valid[P, .:Q] if and only if Valid[P => Q] "VC\VI"

These added clauses encapsulate the logical properties and relations peculiar to C-conditionals. The theorems and principles in the Logic of C-conditionals are derived from these additions together with definitions, axioms and rules drawn from earlier chapters. In the next sections we concentrate on the new theorems and principles derivable from those portions of the base in which C-conditionals appear which have no analogues in M-logic.

The full system of Purely Formal Analytic Logic includes the base which follows and any theorems and rules derivable from it, including all theorems and rules in the previous chapters.

1. Primitives:

Grouping devices	), (,], [, >, <		
Predicate letters:	$P_1, P_2, \ldots, P_n$.	[Abbr.'P','Q','R']	PL
Argument Place holders:	1, 2, 3,…		APH
Individual Constants:	$a_1, a_2, \ldots a_n$,	[Abbr.'a', 'b', 'c']	IC
Individual variables:	$x_1, x_2, \ldots x_n$,	[Abbr.'x', 'y', 'z']	IV
Predicate operators:	&, ~, =>		
Quantifier:	$(\forall x_i)$		

Primitive (2nd level) Predicate of Logic: Syn

12. We could incorporate them within the whole system of A-logic, simply by changing the name or numbers of the theorems, to a numbering system unique to this chapter. E.g., (i) Each theorem given a name or number in this book, will be given a number preceded by 'A', meaning it is a theorem of A-logic.

2. Formation Rules

FR1. If P_i is a predicate letter, $[P_i]$ is a wff
FR2. If P and Q are wffs, [(P&Q)] is a wff.
FR3. If P is a wff, [~P] is a wff.
FR4. If P and Q are wffs, $[(P \Rightarrow Q)]$ is a wff. **Added**
FR5. If P_i is a wff and each t_j ($1 \leq j \leq k$) is an APH or a IC, then $P_i\langle t_1,\ldots,t_k\rangle$ is a wff
FR6. If $P_i\langle 1\rangle$ is a wff, then $[(\forall x_j)P_ix_j]$ is a wff.

3. Abbreviations, definitions

Predicate Operators:

Df6-1. $[(P \,\&\, Q \,\&\, R)\ SYN_{df}\ (P \,\&\, (Q \,\&\, R))]$
Df6-2. $[(\forall_k x)Px\ SYN_{df}\ (Pa_1 \,\&\, Pa_2 \,\&\, \ldots \,\&\, Pa_k)]$ [Df '∀']
Df6-3. $[(P \vee Q)\ SYN_{df} \sim(\sim P \,\&\, \sim Q)]$ [Df 'v',DeM1]
Df6-4. $[(P \supset Q)\ SYN_{df} \sim(P \,\&\, \sim Q)]$ [Df '⊃']
Df6-5. $[(P \equiv Q)\ SYN_{df}\ ((P \supset Q)\&(Q \supset P))]$ [Df '≡']
Df6-6. $[(\exists x)Px\ SYN_{df} \sim(\forall x)\sim Px]$ [Df '(∃x)']
Df6-7. $[(P \Leftrightarrow Q)\ SYN_{df}\ ((P \Rightarrow Q)\&(Q \Rightarrow P))]$ [Df '⇔'] **Added**

Logical Predicates

Df 'Cont'. $[(P\ Cont\ Q)\ Syn_{df}\ (P\ Syn\ (P \,\&\, Q))]$
Df 'Inc'. $[Inc(P)\ Syn_{df}\ ($ $(P\ Syn\ (Q \,\&\, \sim R)) \,\&\, (Q\ Cont\ R))$
$\vee\ (P\ Syn\ (Q \,\&\, R)) \,\&\, Inc(R))$
$\vee\ (P\ Syn\ (Q \vee R)) \,\&\, Inc(Q) \,\&\, Inc(R))$
$\vee\ (P\ Syn\ (Q \Rightarrow R) \,\&\, Inc(Q \,\&\, R))]$ **Added**
Df 'Taut'. $[Taut(P)\ Syn_{df}\ Inc(\sim P)]$
Df 'Valid'. [Valid(P, **.:**Q) Syn_{df} ((P Cont Q) & not-Inc(P & Q))]

4. Axioms

Ax.6-1. [P Syn (P & P)] [&-IDEM1]
Ax.6-2. [(P & Q) Syn (Q & P)] [&-COMM]
Ax.6-3. [(P &(Q & R)) Syn ((P & Q)& R)] [&-ASSOC1]
Ax.6-4. [(P &(Q v R)) Syn ((P & Q)v(P & R))] [&v-DIST1]
Ax.6-5. [P Syn ~~P] [DN]
Ax 6-6. $[(P \,\&\, (P \Rightarrow Q))\ Syn\ (P \,\&\, (P \Rightarrow Q) \,\&\, Q)]$ [MP] **Added**

5. Principles of Inference

R6-1. If $\models$ P, and Q is a component of P, and $\models$ [Q Syn R]
then $\models$ P(Q//R) [SynSUB]
R6-2. If $\models$ R and $P_i\langle t_1,\ldots,t_n\rangle$ occurs in R,
and Q is a suitable h-adic wff, where $h \geq n$,
and Q has an occurrence of each numeral 1 to n,
and no individual variable in Q occurs in R,
then $\models [R(P_i\langle t_1,\ldots,t_n\rangle/Q)$ [U-SUB]
R6-3. If $\models P\langle t_1,\ldots,t_n\rangle$ then $\models P\langle t_1,\ldots,t_n\rangle(t_i/a_j)$ [INST]
R6-6. Valid$[P \Rightarrow Q]$ if and only if Valid[P, **.:**Q] [VC\VI] **Added**

R6-6, the VC\VI principle, states that a conditional is valid if and only if its antecedent is the premiss (or the conjunction of premisses) of a valid inference and its consequent is the conclusion of that Valid inference. As a principle for deriving theorems of formal A-logic is reads:

R6-6. VALID[P=> Q] if and only if VALID[P, .:Q] “VC\VI Principle”

This principle stands in sharp contrast to its ‘⊃’-for ‘=>’ analogue using the M-logic definition of ‘Valid’. In M-logic this is expressed by “|- (P ⊃ Q) if and only if P, |- Q”. The “only if” half is called “the Deduction Theorem”. There it is intended to be understood in ordinary language as “[P ⊃ Q] is valid if and only if [P, therefore Q] is a valid argument”. On our analysis it means, more precisely, “[P, therefore Q] is a valid argument if and only if [~P v Q] is Tautologous”. The “if” part of this biconditional sells the ordinary notion of a valid argument short. According to this principle it follows (among many other things) that since [(P&~P) ⊃ Q] is TAUT, [(P & ~P) .: Q] is M-valid, and from any argument which has inconsistent premisses, all the wisdom and all the nonsense in the world follows logically. To common sense and A-logic these would be *non sequiturs* and they are avoided if the VC\VI principle is formulated as R6-6.

The enumeration of theorems which assert the validity of specific inferential conditionals is the main business of formal logic. In A-logic, validity is a property only of C-conditionals or of arguments (inferences). But a preliminary requirement is the establishing of SYN- and CONT-theorems, and an ancillary task is the development of INC-theorems. Thus we will consider in turn, (i) SYN-and CONT-theorems, (ii) INC and TAUT-theorems, but most importantly (iii) VALIDITY-theorems in which C-conditionals occur.

The wffs in which ‘=>’ occurs together with the new definitions and axioms, produce theorems of five new kinds, none of which are reducible to theorems of M-logic.

1) Since expressions of the form ‘(P=>Q)’ are wffs, these wffs may be substituted for predicate letters in accordance with the clauses of U-SUB. The logical relations SYN and CONT, and the properties INC and TAUT, are preserved under such substitutions. Thus all SYN-theorems, CONT-theorems, INC-theorems and TAUT-theorems of the previous chapters may be augmented by putting C-conditional predicates for predicate letters using the unrestricted rule of U-SUB. None of these expressions or theorems are expressible in M-logic because M-logic lacks ‘=>’.

2) Axiom 6, being a SYN-theorem, yields new SYN- and CONT-theorems which can not be gotten by U-SUB from those established in earlier chapters. These are presented in Section 6.32..

3) From the clause, Df ‘INC=>’, which defines the inconsistency of a C-conditional as the inconsistency of the conjunction of its components, many new INC-theorems and TAUT-theorems, including INC[P => ~P] and TAUT[~(P=> ~P)], are derived. These new theorems are presented in Section 6.33. None of them are expressible in M-logic and only a very few have ‘⊃’-for-‘=>’ analogues.

4) From Df ‘Valid’ with previously established SYN-theorems and CONT-theorems, many ‘=>’-for-‘⊃’ analogues of M-logic theorems are proven to be VALID. But also many ‘=>’-for-‘⊃’ analogues of M-logic theorems can not be proven VALID. The set of VALID C-conditionals does not include any ‘=>’-for-‘⊃’ analogues of any of the “paradoxes of material and strict implication”, among others.

In deriving SYN- and CONT-theorems, or INC- and TAUT-theorems we can use an unrestricted rule of U-SUB in our derivations. But since Validity in A-logic requires mutual compatibility of premiss and conclusion, derivations of new VALIDITY- theorems by U-SUB require constraints to avoid inconsistencies, so we will use U-SUB_{ab} from Chapter 4.

5) Finally, principles of inference in previous chapters expressed in “if…then” form, may now be interpreted as C-conditionals, expressed with ‘=>’, and presented as conditional theorems of A-logic.

However, proofs of these principles must await Chapter 8; they require a truth-logic for conditionals, since the theorems and principles of A-logic include implicit assertions of second-order truths.

6.32 SYN- and CONT-theorems With C-conditionals

There are several respects in which CONT- and SYN-theorems with C-conditionals are like or unlike CONT- or SYN-theorems between expressions with '&' or 'v' as their main connective.

As in all other cases of SYN and CONT, if [(P=>Q) SYN R] or [(P=>Q) CONT R], R must not have any elementary component which is not CONTained in [P=>Q]. Thus P can not Contain (Q => P) or or be Synonymous with (P & (~Q => P)). Compare this with P's implying (Q ⊃ P) and being TF-equivalnt to (P & (~Q ⊃ P)) in M-logic.

Unlike both '(P&Q)' and '(PvQ)', '(P=>Q)' is not idempotent: P is not SYN with (P=>P). Unlike '(P&Q)' and '(PvQ)', '(P=>Q)' is not commutative: (P=>Q) is not SYN with (Q=>P).[13]

Also, unlike '(P&Q)' and '(PvQ)', '(P=>Q)' is not associative; [P=>(Q=>R)] is not SYN with [(P=>Q)=>R].

Like '(PvQ)', and unlike '(P&Q)', (P => Q) does not CONT its components P and Q; neither P nor Q, nor (P&Q) nor (PvQ) are contained in (P=>Q).

Axiom 6-6 says that the conjunction [(if P then Q) and P] is logically synonymous with [(if P then Q) and P and Q].

Ax.6-6. [(P & (P=>Q)) SYN (P & (P=>Q) & Q)]

Axiom 6-6 satisfies the requirements for SYN that (1) both sides contain the same elementary expressions, (2) what occurs POS (NEG) on one side must occur POS (NEG) on the other, and (3) is compatible with the third requirement, that whatever is contained in a wff is contained in its synonyms. Ax.6-6 will sometimes be referred to as Ax.6-06 and as T6-06.

From Ax.6-6 with the other axioms and definitions, we immediately derive

T6-11. [(P & (P=>Q)) SYN ((P&Q) & (P=>Q))]	[Ax.6, &-ORD]
T6-12. [(P & (P=>Q)) CONT (P&Q)]	[T6-11,Df 'CONT']
T6-13. [((P=>Q) & P) CONT Q]	[Ax.6-6,Df 'CONT']

Ax. 6-6 is more easily understood through the CONT-theorem, T6-13. This says in effect that the very notion of a conditional (If P then Q) being conjoined with its antecedent P, by logic, implicitly Contains the notion of the consequent standing alone. T6-13 is a necessary condition for the validity of the *Modus Ponens*, the most fundamental principle of valid inference without which we could not detach a conclusion from its premisses.

Ax.6-6 and T6-13 hold for all expressions—regardless of the mode of discourse or sentential operators employed:

Instances in truth-logic: [(TP & (TP=>TQ)) CONT TQ)],
[(FP & (FP=>FQ)) CONT FQ)]

---

13. In Chapter 8, their trivalent truth-tables are different:

P	(P=>P)	(P =>Q)	(Q => P)
0	0	0 0 0	0 0 0
T	T	0 T F	0 T 0
F	0	0 0 0	0 F 0

Instances in the logic of imperatives: [(TP & (TP=> Q!)) CONT Q!)]

Instances in the logic of questions: [(TP & (TP=> Q?)) CONT Q?)]

Instances in deontic logic: (with 'OQ' for 'It ought be that Q')
[(P & (P=> OQ)) SYN (P & OQ)]

The first definition in which '=>' occurs, Df '<=>', [(P <=> Q) SYN$_{df}$ ((P=> Q) & (Q=> P))] is not novel in any way. It is the usual definition of a biconditional, found in every effort to axiomatize a logic of conditionals. Well-formed biconditionals occur only in the logic of expressions in the indicative mood, since '((P=> Q!) & (**Q!**=> P))' or '((P=> Q?) & (**Q?**=> P))' are ill-formed grammatically. In purely formal A-logic all component expressions are in the indicative mood. Therefore well-formed biconditionals can only contain indicative conditionals.

T6-14. [(P <=> Q) CONT (Q => P)]
<u>Proof</u>: 1) [(P <=> Q) SYN (P <=> Q)] [T1-11,U-SUB]
2) [(P <=> Q) SYN ((P => Q) & (Q => P))] [1),Df '<=>',R1]
3) [(P <=> Q) CONT (Q => P)] [2),Df 'CONT',R1]

T6-15. [(P => Q) CONT (P => Q)] [T1-11,U-SUB]
T6-16. [(P <=> Q) CONT (P => Q) [T1-11,Df '<=>',Df 'CONT',R1]

Note that (P => Q) does <u>not</u> CONT (Q => P). The order, P-followed-by-Q, conveys an essential feature of the meaning of (P => Q).[14] This facet of its meaning is not preserved in the order in (Q => P). A change of order changes the meaning of a conditional; (P => Q) is not commutative.

In Quantification theory, the principle of Modus Ponens yields (the T6- numbers are arbitrary).

T6-20. [((∀x)((Px => Qx) & Px) CONT (∀x) Qx]
<u>Proof</u>: 1) ((∀x)((Px => Qx) & Px) SYN (∀x)(((Px => Qx) & Px) & Qx)
2) ((∀x)((Px => Qx) & Px) SYN (((Pa => Qa) & Pa)&Qx) & (((Pb=> Qb) & Pb)&Qx) &…)
3) (((Pa => Qa) & Pa) & Qa) & (((Pb => Qb) & Pb)&Qb) & ….) CONT (Qa & Qb &…)
4) ((∀x)((Px => Qx) & Px) CONT (Qa & Qb & …)
5) ((∀x)((Px => Qx) & Px) CONT (∀x)Qx

T6-21. [((∀x)((Px => Qx) & (∀x)Px) CONT (∀x)Qx]
<u>Proof</u>: 1) ((∀x)(Px => Qx) & (∀x)Px) SYN ((∀x)((Px=> Qx) & Px) [T3-13,U-SUB]
2) ((∀x)((Px => Qx) & Px) CONT (∀x)Qx [T6-20]
3) ((∀x)((Px => Qx) & (∀x)Px CONT (∀x)Qx) [2),1),SynSUB]

6.33 INC- and TAUT-Theorems With C-conditionals

The definition of inconsistency of a C-conditional, Df 'Inc=>': 'INC[P => Q]' Syn$_{df}$ 'INC[P & Q]' says that to predicate "< 1 > is inconsistent" of a C-conditional means the same as saying that the conjunction of its antecedent and consequent is inconsistent. If 'If P then Q' is to be valid and acceptable, the co-incidence of P and Q must be at least logically possible. This will not be logically possible if (P & Q) is inconsistent. Thus, though inconsistent conditionals are not eliminated, it furthers the work of logic to

14. The difference in order is reflected in the difference in their trivalent truth-tables.

recognize them as inconsistent when the conjunction of their components is inconsistent. This definition is incorporated in Df 'Inc', as clause (iv):

Df 'Inc'. [Inc(P) Syn_{df} (((P Syn (Q & ~R)) & (Q Cont R))
v((P Syn (Q & R)) & Inc(R))
v((P Syn (Q v R)) & Inc(Q) & Inc(R))
v**((P Syn (Q => R) & Inc(Q & R)))**]

If a C-conditional is inconsistent whenever the conjunction of antecedent and consequent is inconsistent, then the C-conditional is inconsistent if either a) the antecedent is inconsistent *with* the consequent though neither is inconsistent in themselves (by clause (i)), or b) the antecedent is inconsistent or c) the consequent is inconsistent, or d) both antecedent and consequent are inconsistent.

From definition of 'Inc(P => Q)' an enormous number of INC-theorems and TAUT-theorems of A-logic can be derived which have no analogues in M-logic. The following derived rules can be used to derive INC and TAUT- theorems with C-condtionals from CONT- or SYN-theorems derived earlier.

DR6-5a. [If (P CONT Q) then INC(P => ~Q)]
DR6-5b. [If (P CONT Q) then TAUT(~(P => ~Q))]

Since Simplification is the paradigm case for all logical containment, each of the CONT and SYN-theorems in Chapters 1 through 4 yield an inconsistent C-conditional with a negated consequent. These same theorems all have '⊃'-for-'CONT' or '≡'-for-'SYN'analogues in M-logic.

DR6-5a [If (P CONT Q) then INC(P => ~Q)]

Proof:	1) (P CONT Q)	[Premiss]
	2) INC(P & ~Q)	[1),Df 'INC(i)',SynSUB]
	3) (INC(P => ~Q) SYN INC(P & ~Q))	[Df 'Inc(iv)']
	4) INC(P => ~Q)	[3),4),SynSUB]
	5) If (P CONT Q) then INC(P=> ~Q)	[1) to 5), Cond.Pr.]

and it follows, by double negation and Df 'TAUT' that

DR6-5b [If (P CONT Q) then TAUT(~(P=> ~Q))]

Proof:	1) [If (P CONT Q) => INC(P => ~Q)	[1),DR6-5a]
	2) If (P CONT Q) => INC(~~(P => ~Q))	[1),DN,SynSUB]
	3) If (P CONT Q) then TAUT(~(P=> ~Q))	[2), Df'TAUT']

If (P SYN Q) then (P CONT Q) and (Q CONT P). Therefore from every SYN-theorem, as well as every CONT-theorem, an INC-Theorem of the form INC(P => ~Q)and a TAUT- Theorem of the form 'TAUT(~(P => ~Q))' is derivable by DR6-5a and by DR6-5b. For example,

T6-30. INC[(P => ~P)]

Proof:	1) [P CONT P]	[Ax.6-1, Df 'CONT']
	2) INC[(P => ~P)]	[1),DR6-5a,MP]

T6-31. TAUT[∼(P ⇒ ∼P)]
Proof: 1) [P CONT P] [Ax.6-1, Df 'CONT']
2) INC[(P ⇒ ∼P)] [1),DR6-5b,MP]

T6-32. INC[(P&Q) ⇒ ∼P]
Proof: 1) (P&Q) CONT P [T1-36c]
2) INC((P&Q) ⇒ ∼P) [1),DR6-5a,MP]

T6-33 . TAUT[∼((P&Q) ⇒ ∼P)]
Proof: 1) (P&Q) CONT P [T1-36c]
2) TAUT(∼((P&Q) ⇒ ∼P) [1),DR6-5b,MP]

Because TAUT-theorems with a TF-conditionals in Chapter 5 correspond one-on-one to theorems in M-logic, an inconsistent C-conditional can be derived from each theorem of M-logic, using the principle, DR6-5c. If TAUT[P ⊃ Q] then INC[P ⇒ ∼Q]:

DR6-5c [If TAUT(P ⊃ Q) then INC(P⇒∼Q)]
Proof: 1) TAUT(P ⊃ Q) [Premiss]
2) TAUT(P ⊃ Q) Syn INC(∼(P ⊃ Q)) [Df 'TAUT']
3) TAUT(P ⊃ Q) Syn INC(∼∼(P &∼Q)) [2),Df '⊃',SynSUB]
4) TAUT(P ⊃ Q) Syn INC(P &∼Q) [3),DN,SynSUB]
5) TAUT(P ⊃ Q) Syn INC(P=>∼Q) [4),Df 'INC=>',SynSUB]
6) INC(P ⇒ ∼Q) [1),5),SynSUB]
7) If TAUT(P ⊃ Q) then INC(P=>∼Q) [1) to 6), Cond.Pr.]

And since INC(P⇒Q) Syn TAUT(∼(P⇒Q)) by the definition of 'TAUT', from every theorem of mathematical logic we get another kind of tautology with a denied C-conditional by the principle, DR6-5d. If TAUT(P ⊃ Q) then TAUT(∼(P ⇒ ∼Q)):

DR6-5d [If TAUT(P ⊃ Q) then TAUT(∼(P⇒∼Q)]
Proof: 1) If TAUT(P ⊃ Q) then INC(P⇒∼Q) [DR6-5c]
2) INC∼∼(P⇒∼Q)) Syn INC(P⇒∼Q)) [1),Ax.6-5,SynSUB]
3) If TAUT(P ⊃ Q) then INC(∼∼(P⇒∼Q)) [1),2),SynSUB]
4) If TAUT(P ⊃ Q) then TAUT∼(P⇒∼Q)) [4),Df 'Taut',SynSUB]

Thus for every wff of the form '(P ⊃ Q)' which is a theorem of M-logic, A-logic has an INC-theorem of the form INC(P ⇒ ∼Q) and a TAUT-theorem of the form 'TAUT∼(P⇒ ∼Q)' for which there are no analogues in M-logic. All of the 33 prototypes of Quine's metatheorems in Section 5.331 quickly yield such TAUT-theorems in A-logic, and from Thomason's six axioms in Section 5.322, by DR6-5c, we get,

AS1 TAUT[P ⊃ (Q ⊃ P)]	.: INC [P ⇒ ∼(Q ⊃ P)]
AS2 TAUT[(P ⊃ (Q ⊃ R)) ⊃ ((P ⊃ Q) ⊃ (P ⊃ R))]	.: INC [(P ⊃ (Q ⊃ R)) ⇒ ∼((P ⊃ Q) ⊃ (P ⊃ R))]
AS3 TAUT[(∼P ⊃ ∼Q) ⊃ (Q ⊃ P)]	.: INC [(∼P ⊃ ∼Q) ⇒ ∼(Q ⊃ P)]
AS4 TAUT[(∀x)(P ⊃ Qx)) ⊃ (P ⊃ (∀x)Qx)]	.: INC [(∀x)(P ⊃ Qx) ⇒ ∼(P ⊃ (∀x)Qx)]
AS5 TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)]	.: INC [(∀x)(Px ⊃ Qx) ⇒ ∼((∀x)Px ⊃ (∀x)Qx)]
AS6 TAUT[(∀x)(Px ⊃ Pa)]	.: INC [(∀x)Px ⇒ ∼Pa)]

and from DR6-5d we get,

AS1 TAUT$[P \supset (Q \supset P)]$.: TAUT$[\sim(P \Rightarrow \sim(Q \supset P))]$
AS2 TAUT$[(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R))]$.: TAUT$[\sim((P \supset (Q \supset R)) \Rightarrow \sim((P \supset Q) \supset (P \supset R))]$
AS3 TAUT$[(\sim P \supset \sim Q) \supset (Q \supset P)]$.: TAUT$[\sim((\sim P \supset \sim Q) \Rightarrow \sim(Q \supset P)]$
AS4 TAUT$[(\forall x)(P \supset Qx) \supset (P \supset (\forall x)Qx)]$.: TAUT$[\sim((\forall x)(P \supset Qx) \Rightarrow \sim(P \supset (\forall x)Qx)]$
AS5 TAUT$[(\forall x)(Px \supset Qx) \supset ((\forall x)Px \supset (\forall x)Qx)]$.: TAUT$[\sim((\forall x)(Px \supset Qx) \Rightarrow \sim((\forall x)Px \supset (\forall x)Qx)]$
AS6 TAUT$[(\forall x)(Px \supset Pa)]$.: TAUT$[\sim((\forall x)Px \Rightarrow \sim Pa)]$

In short, from every SYN- or CONT-theorem derivable in the preceding chapters, as well as from every TAUT-theorem in Chapter 5, we can derive the inconsistency of a C-conditional, and a TAUT-theorem of A-logic which negates a C-conditional. These results have no analogues in M-logic.

Since validity entails containment, and inconsistency is defined by containment, an INC-theorem and a TAUT-theorem can be derived from every Validity-theorem. The following principles could (but won't) be used as derived rules.[15]

$\models$ [If Valid$(P \Rightarrow Q)$ then Inc$(P \& \sim Q)$].
$\models$ [If Valid$(P \Rightarrow Q)$ then Taut$(P \supset Q)$].
$\models$ [If Valid$(P \Rightarrow Q)$ then Inc$(P \Rightarrow \sim Q)$].
$\models$ [If Valid$(P \Rightarrow Q)$ then Taut$(\sim(P \Rightarrow \sim Q))$].

Intuitive support for these principles is suggested by the argument that since 'If (P&Q) then Q' is a <u>logically valid</u> conditional, 'If (P&Q) then not-Q' ought to be inconsistent (as it certainly appears to be). Thus the denial of 'If (P&Q) then not-Q)' should be tautologous and logically indefeasible. This intuition is expressed formally in the derivable rule, If Valid$[P \Rightarrow Q]$ then Taut$[\sim(P \Rightarrow \sim Q)]$. The root concept in A-logic of an inconsistent conditional is not expressible in M-logic, although its cogency seems clear. The analogue with a TF-conditional does not hold either in M-logic or in A-logic; $((P\&Q) \supset Q)$ does not entail $\sim((P\&Q) \supset \sim Q)$ or the inconsistency of $((P\&Q) \supset \sim Q)$.

The converses of these last four principles fail of course. For example, Taut $(P \supset Q)$ includes all the anomalies with respect to common usage which are attributed to "material and strict implication" due to construing '$\supset$' as 'If…then'. Valid$[P \Rightarrow Q]$ has no such anomalies.

In looking at the similarities and differences between A-logic and M-logic, the contrasts with respect to inconsistency and tautology of the "conditionals" in the two logics (due to Df 'Inc$\Rightarrow$' in A-logic), is striking. The inconsistency of C-conditionals differs markedly from inconsistencies of the TF-conditional, $(P \supset Q)$, or its synonyms $(\sim PvQ)$ or $\sim(P \& \sim Q)$. In M-logic the only TF-conditionals which can be logically inconsistent are those with a tautologous antecedent and an inconsistent consequent. For if the antecedent is false in any truth-table row, the TF-conditional is true in that row, hence not inconsistent; and if the consequent is true in any truth-table row, the TF-conditional as a whole is true in that row, hence not inconsistent. Hence, the only TF-conditionals which can be inconsistent (and whose denial can be tautologous) are those with a tautologous antecedent which has no false cases, and an inconsistent consequent has no true cases. This case would be SYN to negating a conjunction of two tautologies.[16]

15. We will not use these principles in derivations since we have already shown more direct ways to derive the same inconsistencies and tautologies.
16. $[((\sim P \text{ v } P) \supset (P \& \sim P))$ SYN $\sim((\sim P \text{ v } P) \& \sim(P \& \sim P))]$

More simply put: by clause (iii) of the definition of INC, [~P v Q], is inconsistent if and only if both [~P] and [Q] are inconsistent, i.e., if and only if P is tautologous and Q is inconsistent.

The simplest TAUT-theorems with C-conditionals begin with '(~P => P)'. This is inconsistent, as Aristotle observed, so that '~(~P => P)' is logically tautologous and unfalsifiable.

The radical difference between '(P => Q)' and '(P ⊃ Q)' with respect to inconsistency and tautology, is indicated by the following examples:

A-logic	M-logic	
INC[P => ~P]	not-INC[P ⊃ ~P],	(since (P ⊃ ~ P) SYN ~P)
INC[~P => P]	not-INC[~P ⊃ P],	(since (~P ⊃ P) SYN P)
TAUT[~(~P => P)]	not-TAUT[~(~P ⊃ P)],	(since ~(~P ⊃ P) SYN ~P)
TAUT[~(P => ~P)]	not-TAUT[~(P ⊃ ~P)],	(since ~(P ⊃ ~P) SYN P)

In M-logic the tautology [(P & ~P) ⊃ Q] is called a logical "implication". It belongs to a huge sub-class of the set of so-called "paradoxes of strict implication", namely, the set of non-sequiturs in which the antecedent is inconsistent and part or all of the consequent is not contained in the antecedent. The '=>'-for-'⊃' analogue is not valid in A-logic. Replacing the TF-conditional by a C-conditional, the denial of the result is tautologous:

T6-34 TAUT[~((P & ~P) => Q]

Proof:	1) [(P & Q) CONT P]	[T1-36]
	2) INC[(P & Q) & ~P]	[From 1), Df 'INC']
	3) [((P & Q) & ~P) SYN [((P & ~P) & Q]	[&-ORD]
	4) INC[(P & ~P) & Q]	[2),3),SynSUB]
	5) **INC[(P & ~P) => Q]**	[4,Df 'INC=>']
	6) INC[~~ ((P & ~P) => Q)]	[5), DN]
	7) **TAUT[~ ((P & ~P) => Q)]**	[6),Df 'TAUT']

Contrast the opposing logical properties of analogous C-conditionals and TF-conditionals in the following pairs of theorems from A-logic and M-logic:

A-logic	M-logic
INC[((P & ~P) => Q)],	TAUT[((P & ~P) ⊃ Q)]
TAUT[~((P & ~P) => Q)],	INC[~((P & ~P) ⊃ Q)]
not-INC[~((P & ~P) => Q)]	INC [~((P & ~P) ⊃ Q)]
not-TAUT[((P & ~P) => Q)],	TAUT[((P & ~P) ⊃ Q)]

Those on the right-hand side belong to that sub-set of wffs in A-logic which contains all and only the theorems (M-valid wffs) of M-logic. The top two in the left-hand column are new theorems of A-logic, which have no analogues in M-logic. The bottom two in the left-hand column show contrasting applications of INC and TAUT with respect to C-conditionals and TF-conditionals.

For more contrasts, consider: if (P ⊃ Q) is TAUT, then (P => Q) is INC and thus not- TAUT, and if (P => Q) is INC, then (P ⊃ Q) is TAUT and hence not-INC:

With C-conditionals		With TF-conditionals
INC[(P & ~P) => P]	vs.	TAUT[(P & ~P) ⊃ P]
TAUT[~((P & ~P) => P)]	vs.	INC[~((P & ~P) ⊃ P)]
INC[(P & Q) => ~P]	vs.	not-INC[(P & Q) ⊃ ~P]
TAUT[~((P & Q) => ~P)]	vs.	not-TAUT[~((P & Q) ⊃ ~P]
INC[(∀x)(∀y)(Ryx => ~Ryy)]	vs.	not-INC[(∀x)(∀y)(Ryx ⊃ ~Ryy)]
TAUT[~ (∀x)(∀y)(Ryx => ~Ryy)]	vs.	not-TAUT[~(∀x)(∀y)(Ryx ⊃ ~Ryy)]
TAUT[(∃x)(∃y) ~(Ryx => ~Ryy)]	vs.	not-TAUT[(∃x)(∃y)~(Ryx ⊃ ~Ryy)]

All of these INC- and TAUT-theorems are correct. The fact that what is INC with a C-conditional is TAUT with a TF-conditional and vice-versa, does not show that one is correct and the other wrong. It simply show that they are very different logically.

There are no instances at all in which [P => Q] and [P ⊃ Q] are both TAUT. [P ⊃ Q] is TAUT if and only if [P => Q] is not-TAUT. For, unlike TF-conditionals, C-conditionals are never tautologous. C-conditionals can be inconsistent, and the negation of an inconsistent wff is a tautologous wff. Thus only denials of C-conditionals can be tautologous. And,just as denials of conjunctions are not conjunctions, denials of conditionals are not conditionals. Thus no C-conditionals are TAUT.[17] In contrast, every wff which is a theorem of M-logic, is either a tautologous TF-conditional or is synonymous with a TAUT TF-conditional, and no wff in M-logic which is not a theorem is synonymous with a TAUT TF-conditional.

But enough. The central idea is in Df ‘Inc=>’, the definition of inconsisteny in a condtional. References to tautologies are really unnecessary. They are thrown in only because tautology (denied inconsistency) is the defining characteristic—the necessary and sufficient condition—of being a theorem in M-logic. The important notion is that of inconsistency. In logical reasoning inconsistencies should be identified and avoided. The point is that if (P & Q) is inconsistent, then (P => Q) is inconsistent and is of no use in applied logical reasoning. Enumerating TAUT-theorems of this or any other sort, including the theorems of M-logic in general, is of little interest to A-logic. Inconsistencies are more relevant. But the primary purpose of logic is not to enumerate inconsistencies either. It is to enumerate and clarify valid principles of inference.

6.34 VALIDITY-Theorems with C-conditionals

In A-logic ‘<1>is valid’ applies only to inferences and arguments, or to inferential C-conditionals, or to conjunctions or disjunctions of such C-conditionals. It never applies to wffs or statements compounded solely from ‘and’, ‘or’, ‘not’, ‘all’ and ‘some’, as in M-logic. Thus no wffs counted as theorems in M-logic can satisfy the criterion of “valid” used in A-logic; no M-valid wffs are A-valid.

---

17. Since (P =>Q) is POS, it is not Syn to ~(P => ~Q), which is NEG, altho their trivalent truth-tables are the same. In Ch. 8 T(P => Q) Syn F(P => ~Q) because both are POS, due to ‘T’, (not to the synonymy of (P => Q) and ‘~(P => ~Q)’. But INC is not a truth-function in A-logic; it depends on CONT. If ‘(P => Q)’ were Syn to ‘~(P => ~Q)’ then we woud get INC(P => Q) Syn INC(~(P => ~Q) by SynSUB, and hence, TAUT~(P => Q) Syn TAUT(P => ~Q), .: TAUT~((P & Q)=> ~P) Syn TAUT((P & Q)=> P). The principle that no C-conditionals are Taut would be violated, as would the principle that if P is POS and Q is NEG, P can not be Syn to Q.

As we said in the last chapter, the criterion of validity in A-logic has two parts. [P, .:Q] is VALID, i.e., logically valid by virtue of its "logical words" (including 'if...then') and the definitions of any extra-logical words, only if

(1) P logically contains Q
and (2) [P & Q] is not-inconsistent.

If an inference is valid, then by the VC\VI principle any conditional statement with the premisses of that inference conjoined in its antecedent and the conclusion of that inference as the consequent, is a Valid conditional. This is the meaning of the "only if" clause in VC\VI. In purely formal A-logic, the validity of a conditional or an argument, is independent of the meanings of 'true' and 'false' or any other extra-logical terms.

In Section 6.341 we resume discussion of the consistency requirement for validity and the problem of expanding the set of VALIDITY-theorems by the restricted rule of U-SUB_{ab}. Next in Section 6.342, we present some derived rules, based on the definition of Validity and the VC\VI principle, for converting SYN- and CONT-theorems from this and previous chapters into theorems about logically valid conditionals. Thirdly, we derive Validity-theorems from the SYN- and CONT-theorems of previous chapters and sections Section 6.343, and finally we look at previous Principles of Inference as Valid C-conditionals of A-logic in Section 6.344.

The set of VALIDITY-theorems derivable in this Chapter omits some traditional principles that it might be expected to include. This is because traditionally and currently, logicians have focused on the logic of truth-claims, or propositions. Though these principles are not developed in universal A-logic, they will be established or otherwise accounted for in the analytic truth-logic of Chapters 7 and 8. In this way our omissions and shown to agree with or account for the relevant principles of traditional and current logic.

6.341 The Consistency Requirement for Validity

In A-logic arguments (and related C-conditionals) are invalid if the conjunction of premisses and conclusion (antecedent and consequent) is inconsistent. What is the justification for making this a requirement for logical validity? M-logic does not require it for M-validity and so far as I know traditional logic does not explicitly either allow or disallow inconsistency in a conjunction of the components of a valid argument.

The main point is that if a rule of inference is to be valid, it should be possible for both the antecedent and the consequent (or premisses and conclusion) to co-exist. Rules of inference should be applicable and usable in all possible worlds. Allowing inference rules that could only be applied in impossible worlds is pointless. Other positive arguments in favor of the consistency requirement for validity, are that it eliminates a major sub-class of "Paradoxes of Strict Implication" and contributes to our solution of the problem of counterfactual conditionals and other problems. (See Chapter 10)

In M-logic, an argument is called "valid" if and only if there is no case in which the premisses are true and the conclusion is false. This permits M-valid arguments to include arguments in which (i) one or more of its premisses are individually inconsistent, though the conclusion is not (this situation makes the argument M-valid and is the source of non-sequiturs), or (ii) the premisses, though individually consistent, are inconsistent taken together, and the conclusion is not inconsistent (this is also sufficient to make the argument both M-valid), or (iii) both the conclusion and the premisses are inconsistent (the latter either collectively or by some individually inconsistent premiss). These three kinds of arguments and the related C-conditionals are not A-valid, by the definition of 'Valid' in A-logic.

In M-logic the only arguments (and related TF-conditionals) which are ruled out as invalid because logically inconsistent, are those in which all of the premisses are tautologous and the conclusion is inconsistent.

In A-logic the premisses may logically contain the conclusion even if one of the premisses is inconsistent. But logical containment, though necessary, is not sufficient for A-validity. If an argument meets the containment requirement but has an inconsistent premiss, the inconsistency must be removed, to make what remains A-valid. The premisses may also logically contain a conclusion which is inconsistent provided that that specific inconsistency is in the premisses, but again the argument will not be A-valid.

It may be objected that a consistency requirement for validity is not necessary because the traditional distinction between a *sound* argument and a *valid* argument is quite sufficient. Let us agree that (i) a truth-argument is *sound* if its premisses are true and it is valid, and (ii) whether an argument is valid or not is independent of the truth or falsity of the premisses and (iii) an argument can be valid but unsound if its premisses are not true and (iv) if the premisses are inconsistent, they can not be true, and this will make the argument unsound. It does not follow that the requirement for soundness coincides with the requirement that premisses and conclusion be jointly consistent. Soundness overrides the distinction between falsehood and inconsistency. If premisses are false as a matter of fact, this does not exclude the possibility that they might be true. But if they are inconsistent, they cannot possibly be true. In excluding all inconsistencies from valid arguments, A-logic leaves the distinction between sound and valid arguments intact; an argument may still be A-valid (and thus free of inconsistency) but unsound because of an untrue premiss.

There are two problems which must be dealt with if we make consistency a necessary condition of A-validity: 1) Preserving consistency when instantiating with extra-logical expressions, and 2) preserving consistency when substituting wffs for predicate or sentence letters.

The first problem is one that faces every system of formal logic including traditional logic and M-logic. Although the abstract forms of arguments, conditional statements, and other expressions, may be free of *prima facie* inconsistency (e.g., if no predicate letter occurs both POS and NEG), there is no way that formal logic can prevent inconsistencies from being introduced when sentence or predicate letters are replaced by extra-logical expressions from other disciplines or ordinary language. In other words, an argument which is not INC (not inconsistent by virtue of ite logical structure), may be Inc (inconsistent by virtue of inconsistencies among the extra-logical words). Dealing only with the meanings of wffs based on 'and', 'not', 'if…then' etc., logic can not cover all inconsistencies which can be derived from the meanings assigned to extra-logical words. For example, the valid theorem "VALID[(Pa & Qa & Ra) => Ra)]" has as an instance, that "If a is a triangle which is both scalene or equilateral, then a is equalateral". But, describing any triangle as both scalene (Syn_{df} "all sides are unequal") and equilateral (Syn_{df} "all sides are equal") is inconsistent; so this instantiation would be not merely be false, but inconsistent and therefore not A-valid. No rule of formal logic can or should prevent inconsistencies among extra-logical words.

Thus all that a VALIDITY-theorem says in formal A-logic, is that so far as the logical structure of its wffs goes, it is VALID; nothing in its logical structure mandates that the conjunction of premisses and conclusion be inconsistent. A VALIDITY-theorem displays an inference schema (or conditional) with a logical structure which has many valid, consistent instantiations. But instantiations of this form are logically valid only if what is substituted for instantiation does not make introduce a new inconsistency. Vigilance to avoid inconsistency is required in applications of formal logic.

The second problem has to do with how we can generate new VALIDITY-theorems with U-SUB if it is required that the result of substitution be *prima facie* not-INC. U-SUB is an indispensable means of increasing the variety of logical laws, since it alone allows the introduction of new predicates with more complicated structures within structures. The problem has two parts: 1) the formulation of restrictions upon U-SUB which would prevent substitutions that would produce INC wffs from wffs that are not-INC; 2) the problem of proving that all VALID inference schemas could be established from the axioms and rules (including U-SUBab) of A-logic; this would establish the completeness of the set of provable VALIDITY- theorems with respect to A-logic's criterion of a VALID wff.

To accomplish the first part we need criteria for the consistency of a wff, and a restricted rule of substitution which will preserve consistency when the initial wff substituted is *prima facie* not inconsistent. For unquantified wffs the test for consistency is simple: every such wff is reducible to a synonymous disjunctive normal form, and this disjunctive normal form is a consistent wff if and only if it has at least one disjunct in which no atomic component that occurs both POS and NEG. For quantified wffs the determination of consistency is more difficult in some cases. But there are many easy tests which guarantee that a Q-wff will not be inconsistent in any domain. Among such tests of consistency is that if a wff is inconsistent, at least one atomic component must occur both POS and NEG. If this is not the case the wff is *prima facie* not-INC.

All of the axioms and all of the Syn- and Cont-theorems in the first three chapters were completely negation-free, and thus satisfy the consistency requirement for A-validity. In addition, the axioms of Chapter 4, which introduced negation signs, all satisfied that requirement and.by restricting substitution to U-SUBab, no SYN- or CONT-theorem in Chapter 4 was inconsistent. The axioms and rules of inference of Chapter 6 will satisfy the same criterion and all theorems gotten from them by restricted substitution,U-SUBab, will also preserve consistency.

U-SUBab does not prevent the derivation of many valid inferences which have inconsistent parts in their components. If any wff has a synonymous disjunctive normal form in which at least one disjunct has no atomic wff occurring both negated and unnegated, then that wff is consistent even though one or more of the other disjuncts are, in themselves, inconsistent. If any substitution by U-SUBab is applied to that wff the resulting wff and all its synonyms will preserve consistency. Thus the set of A-valid arguments and conditionals may be further expanded.[18]

So we have a large array of SYN- and CONT-theorems which are *prima facie* not-inconsistent, and from which, with df 'Valid' and U-SUBab, we can derive many more valid inference schemata. The results are completely unlike any axiom system or set of theorems of M-logic, for every axiom and every theorem in M-logic must have at least one atomic component occurring both POS and NEG in **every** conjunct of its basic conjunctive normal form, in order to be tautologous.

---

18. For example, the SYN-theorem, 1) below has Q occurring both negated and unnegated on both sides of 'SYN', but the conjunction of the two is not INC. Thus 1) yields seven CONT theorems which are also not-INC, from which seven valid inference schemata are derived. U-SUBab can then be applied to these VALIDITY-Schemata, and consistency and VALIDITY will be preserved in all the resulting schemata.

1) |= [((P&~Q & R)v(P&~Q & Q)) SYN (P & ~Q & (RvQ))] [&v-DIST]
Not-INC[((P&~Q & R)v(P&~Q & Q)) & (P & ~Q & (RvQ))]
TT TF **T** T **T** TT TF **F** F **T** TT TF **T** TTF [Inspection]
2) Valid[((P&~Q & R)v(P&~Q & Q)) .: (P & ~Q & (RvQ))] [1),DR5-6b]
3) Valid[((P&~Q & R)v(P&~Q & Q)) .: (P & ~Q)] [1),Df.'Cont', DR5-6a]
4) Valid[((P&~Q & R)v(P&~Q & Q)) .: (P & (RvQ))] [1),Df.'Cont', DR5-6a]
5) Valid[((P&~Q & R)v(P&~Q & Q)) .: (~Q & (RvQ))] [1),Df.'Cont', DR5-6a]
6) Valid[((P&~Q & R)v(P&~Q & Q)) .: P] [1),Df.'Cont', DR5-6a]
7) Valid[((P&~Q & R)v(P&~Q & Q)) .: ~Q] [1),Df.'Cont', DR5-6a]
8) Valid[((P&~Q & R)v(P&~Q & Q)) .: (RvQ)] [1),Df.'Cont', DR5-6a]

By the VC\VI principle, everything that has been said about the consistency requirement for valid inferences and arguments applies to C-conditionals. The focus shifts from conjunction of premisses and conclusion, to the conjunction of antecedent and consequent. The techniques and problems of preserving *prima facie* consistency are the same for inferences and conditionals.

With respect to the second part of our problem, it has not been shown whetherU-SUBab with the definitions, axioms and other rules of inference will yield a set of Validity-theorems which is complete with respect to the semantic theory of A-logic. This would require a proof that all consistent quantified wffs can be distinguished from inconsistent quantified wffs; a difficult task in view of Church's theorem that the class of consistent (satisfiable) quantification theorems in M-logic is undecidable. Whether or not this is shown later, it is clear that the consistency requirement permits a vast array of Validity-theorems in A-logic with unquantified and quantifed wffs which provably satisfy the consistency requirement. Whether this is only a proper subclass of the class of all inference-schemata and C-conditional schemata which would fit the criterion of A-validity remains a question for further research.

6.342 Derived Rules from Df 'Valid(P, .:Q)' and the VC\VI Principle

Derivations of Validity-theorems will be facilitated by six derived rules which are based on the definition of a valid inference in Chapter 5, and the VC\VI Principle, i.e., by

Df 5-6. Valid(P, **.:** Q) Syn_{df} ((P Cont Q) & not-Inc(P&Q)] [Df'Valid[P, **.:** R]
and R6-6. Valid[P => Q] if and only if Valid[P, **.:**Q] [VC\VI]

The following Derived Rules provide short-cuts for proofs of validity theorems:

DR6-6a. If [P Cont Q] & not-Inc[P & Q], then Valid[P => Q].
DR6-6b. If [P SYN Q and not-Inc (P&Q)] then Valid [P => Q]
DR6-6c. If [P SYN Q and not-Inc (P&Q)] then Valid [Q => P]
DR6-6d. If [P Syn Q] & not-Inc[P & Q], then Valid[P <=> Q].
DR6-6e. If Valid[P => Q] then Valid[P, **.:** Q]
DR6-6f. If Valid[P, **.:**Q] then Valid[P => Q]

We start with the last two principles which come directly from R6-6, the VC\VI principle.

DR6-6e. If Valid[P => Q] then Valid[P, **.:** Q]
Proof: 1) (If Valid[P => Q] then Valid[P, **.:** Q] and If Valid[P, **.:**Q] then Valid[P => Q] [VC\VI, Df 'if & only if'']
2) If Valid[P => Q] then Valid[P, **.:** Q] [1), SIMP]

DR6-6f. If Valid[P, **.:**Q] then Valid[P => Q] [Similar to proof for DR6-6e]

A proof that a particular C-conditional is logically valid, consists in showing that its components possess the properties required by the definition of 'Valid'.

DR6-6a. If [P Cont Q] & not-Inc[P & Q], then Valid[P => Q].
Proof: 1) If [P Cont Q] & not-Inc[P & Q], then Valid[P, **.:** Q] [DR5-6a].
2) If Valid[P, **.:**Q] then Valid[P => Q] [DR6-6f]
3) If [P Cont Q] & not-Inc[P & Q], then Valid[P => Q] [1),2) HypSYLL]

DR6-6b. If [P SYN Q and not-Inc (P&Q)] then Valid [P => Q]

Proof: 1) [If (P SYN Q) and not-Inc (P&Q), then Valid (P, .:Q)] [DR5-6b]
2) If Valid[P, .:Q] then Valid[P => Q] [DR6-6f]
3) If [P SYN Q] & not-Inc[P & Q], then Valid[P => Q] [1),2) HypSYLL]

DR6-6c. If [P SYN Q and not-Inc (P&Q)] then Valid [Q => P]

Proof: 1) [If (P SYN Q) and not-Inc (P&Q), then Valid (Q, .:P)] [DR6-6c]
2) If Valid[Q, .: P] then Valid[Q => P] [DR6-6f (re-lettered)]
3) If [P SYN Q] & not-Inc[P & Q], then Valid[Q => P] [1),2) HypSYLL]

DR6-6d shortens the proof from Df 'Valid' to that a biconditional, [P <=> Q], is Valid.

DR6-6d. If [P Syn Q] & not-Inc[P & Q], then Valid[P <=> Q].

1) P Syn Q & not-Inc(P & Q) [Premiss]
2) P Syn Q [1) SIMP]
3) P Cont Q & Q Cont P [2),DR1-14,MP]
4) [P Cont Q] [3),SIMP]
5) [P Cont Q] & not-Inc[P&Q] [1),2),ADJ]
6) Valid[P, .: Q] [5),DR5-6a,MP]
7) not-Inc[Q&P] [1),Ax.2,SynSUB]
8) [Q Cont P] & not-Inc[Q&P] [3),2),ADJ]
9) Valid[Q, .:P] [7),DR5-6a,MP]
10) [Valid(P, .:Q) & Valid(Q, .:P)] [6),9),ADJ]
11) [(Valid(P, .:Q) & Valid(Q, .:P)) Syn Valid(P, .:Q) & (Q, .:P)] [Df,Valid-&,U-SUB]
12) [(Valid(P, .:Q) & Valid(Q, .:P)) Syn Valid((P => Q) & (Q => P)] [11),VI/VC(R6-6)]
13) Valid[(P => Q) & (Q => P)] [10),12),MP]
14) Valid[(P <=> Q)] [13),Df'<=>']
15) If P Syn Q and not-Inc(P&Q) then Valid[P <=> Q] [1) to 13),C.P]

The Hypothetical Syllogism may be expressed as: "If [P => Q] is valid and [Q => R] is valid, then [P => R] is valid. "This principle is based on DR1-19 and its proof in A-logic is:

DR6-119. [(VALID (P => Q) & VALID (Q => R)) => VALID (P => R)]

Proof: 1) [(VALID (P => Q) & VALID (Q => R)) [Premiss]
2) [(VALID (P => Q)] [1),SIMP]
3) [(P CONT Q) & not-INC(P&Q)] [2),Df 'VALID']
4) [(VALID (Q => R)] [1),SIMP]
5) [(Q CONT R) & not-INC(Q&R)] [4),Df 'VALID']
6) [(P CONT Q) & (Q CONT R)] [3),5),Simp(twice),Adj]
7) Not-INC (P&R) [3),5)][19]
8) [P CONT R] [6),DR1-19,MP]
9) VALID[P => R] [7),8),Df 'VALID']
10) [(VALID (P => Q) & VALID (Q => R)) => VALID (P => R)] [1) to 9),Cond.Pr.]

---

19. Step 7 follows from Steps 3) and 5), because if P CONT Q and Q CONT R, and P&Q is not inconsistent, and Q&R is not inconsistent, then since Q is a conjunct in P and has nothing which is inconsistent with any conjunct or set of conjuncts in P, and R is a conjunct of Q and has nothing that is inconsistent with any conjunct or set of conjuncts in Q, then R can have nothing that is inconsistent with a conjunct or set of conjuncts in P.

Another version of the hypothetical syllogism, which will be proved in Chapter 8, is

T8-815. Valid [(T(P => Q) & T(Q=>R)) => T(P => R)]

where 'T' is read, "It is true that…". But this and others like it will be shown to be somewhat trivial theorems in A-logic, since they reduce to the assertion that [T(P&Q&R) => T(P&R)] is valid.[20]

By steps similar to those in the proof of DR6-119 derived rules of inference for arriving at a validity theorem from prior validity theorems can be established from each of the Derived Rules DR1-11 to DR1-28 in Chapter 1. But derivations from these to other principles must abide by the constraints of U-SUBab. For example, among those that are useful are:

DR6-121. If Valid[P => Q] then Valid[(R&P) => (R&Q)]
from DR1-21. If [P CONT Q] then [(R&P) CONT (R&Q)]

But if ∼P were substituted for R, contrary to U-SUBab, we would get

If Valid[P => Q] then Valid[(∼P&P) => (∼P&Q)]

which yields an inconsistent, hence *not*-valid, conditional in the consequent.

Better and more detailed analysis of the proofs of theorems and inference principles will be given in Chapter 8, after we have developed the logic of the predicate '<1>is true'. For truth- predication has been implicitly employed in every principle of inference presented so far. Implictly we mean by '[If (P Cont Q) then Valid(P **.:** Q)], "If the result of substituting some wffs for P and Q in '(P CONT Q)' is true then the result of making the same substitutions in 'Valid (P **.:** Q)' is true. In this chapter we do not include the extra-logical predicate '<1>is true', so the more detailed analysis of proofs must wait until Chapter 8.

Proofs of a particular validity theorem,—i.e., that some particular conditional or biconditional wff or predicate or statement is valid—may be presented according to the following proof schemas, using the derived inference rules. For example, using DR6-6a and DR6-6d :

|= Valid[P=>Q]

Proof:		
1)	**[P Cont Q]**	[T…]
2)	[not-Inc(P&Q)]	[Inspection]
3)	[(P Cont Q) & not-Inc (P&Q)]	[1),2),ADJ]
4)	[([P Cont Q] & not-Inc(P&Q) => Valid[P => Q]	[DR6-6a]
5)	**Valid[P => Q]**	[3),4),MP]

|= Valid[P<=>Q]

Proof:		
1)	**[P Syn Q]**	[T…]
2)	**[not-Inc(P&Q)]**	[Inspection]
3)	[(P Syn Q) & not-Inc (P&Q)]	[1),2),ADJ]
4)	[([P Syn Q] & not-Inc (P&Q) => Valid[P <=> Q]	[DR6-6d]
5)	**Valid[P <=> Q]**	[3),4),MP]

20. See Section 8.221, especially the proof of T8-15 and the paragraph which follows.

By "Inspection" for step 2), I mean either a) inspecting it visually to determine whether any predicate letter occurs both POS and NEG in [P&Q] or b) reducing (P&Q) to its basic Disjunction Normal Form and check to see if its has any disjunct in which no atomic wff occurs both POS and NEG, or 3) assigning POS/NEG values to show that it might have case in which the whole is NEG and thus the whole is not-inconsistent (since Inc is always POS).

In some cases, where the absence of inconsistency is clear step 2) can be omitted. The proof of validity is established by DR6-6d merely by citing a SYN-theorem, or by DR6-6a and a CONT-theorem. All of the Cont-and Syn-theorems of Chapters 1 through 4, are of this sort, since none of them have atomics wffs which occur both POS and NEG. Further all theorems derived from any of those theorems by U-SUBab have the same property, since U-SUBab prevents the assigment, of any additional letters which would introduce inconsistencies into componenets which are initially free of inconsistency. Finally, one can construct Syn-theorems and Cont-theorems in which the disjunctive normal forms of the antecedent contain inconsistent disjunctions, but have at least one disjunct which is free of inconsistencies; if U-SUB is restricted to U-SUBab on such wffs, the freedom from inconsistencies will be guaranteed.

6.343 VALIDITY Theorems

The proofs of all theorems derived from SYN- and CONT-theorems in Chapter 1 through 4 can be specified in one line by citing DR6-6a or DR6-6d, as in:

T6-101. VALID[P <=> (P&P)]	&-IDEM	[Ax.1-01,DR6-6d]
T6-437. VALID [(∀x)(Px ⊃ Qx) => ((∀x)Px ⊃ (∀x)Qx)]	ML*101	[T4-37,DR6-6a]

The lack of any predicate variables which occur both POS and NEG, and therefore the lack of inconsistency, is clear by inspection. In Chapters 1 to 3, the Rules of Formation did not permit the negation sign in wffs, so all wffs are negation-free and thus can't be inconsistent. In Chapter 4 no predicate letters occurred both POS and NEG in any axiom or definition, and theorems were deliberately limited to those that could be gotten by only rules which prevent the derivation of inconsistent wffs from consistent ones, including the restricted rule U-SUBab. Proofs of Validity based on SYN and CONT theorems in Chapter 6 initially appeal to inspection to establish consistency and therefore validity, and thereafter must derive other validity theorems under the constraints of U-SUBab. Thus the validity theorems derived from SYN- and CONT-theorems of Chapters 1 through 4, and Chapter 6, will all meet the consistency requirement for validity.

Further validitity theorems can be derived from any one of them by using U-SUBab to preserve consistency. Still more validity-theorems can be added in which components have inconsistent sub-components, provided they do not produce inconsistencies with components which are initially inconsistency-free. The validity-theorems of Chapter 6 can also be extended by U-SUBab, and by disjoining components with unrelated wffs. But theorems derived in this section will include only the first level of possible derivations. We do not derive from any one of them the many additional Validity-theorems that could be derived using U-SUBab or other devices.

6.3431 From SYN- and CONT-theorems in Chapters 1 to 3

The following is a complete list of a validity-theorems with C-conditionals that are theorems derivable from the SYN and CONT-theorems in Chapters 1 through 3:

From SYN-theorems in Chapter 1:

T6-101. VALID[P <=> (P&P)]	[&-IDEM]	[Ax.1-01,DR6-6d]
T6-102. VALID[P <=> (PvP)]	[v-IDEM]	[Ax.1-02,DR6-6d]
T6-103. VALID[(P&Q) <=> (Q&P)]	[&-COMM]	[Ax.1-03,DR6-6d]
T6-104. VALID[(PvQ) <=> (QvP)]	[v-COMM]	[Ax.1-04,DR6-6d]
T6-105. VALID[(P&(Q&R)) <=> ((P&Q)&R)]	[&-ASSOC]	[Ax.1-05,DR6-6d]
T6-106. VALID[(Pv(QvR)) <=> ((PvQ)vR)]	[v-ASSOC]	[Ax.1-06,DR6-6d]
T6-107. VALID[(Pv(Q&R)) <=> ((PvQ)&(PvR))]	[v&-DIST-1]	[Ax.1-07,DR6-6d]
T6-108. VALID[(P&(QvR)) <=> ((P&Q)v(P&R))]	[&v-DIST-1]	[Ax.1-08,DR6-6d]
T6-111. VALID[P <=> P]		[T1-11,DR6-6d]
T6-112. VALID[((P&Q) & (R&S)) <=> ((P&R) & (Q&S))]		[T1-12,DR6-6d]
T6-113. VALID[((PvQ) v (RvS)) <=> ((PvR) v (QvS))]		[T1-13,DR6-6d]
T6-114. VALID[(P & (Q&R)) <=> ((P&Q) & (P&R))]		[T1-14,DR6-6d]
T6-115. VALID[(P v (QvR)) <=> ((PvQ) v (PvR))]		[T1-15,DR6-6d]
T6-116. VALID[(Pv(P&Q)) <=> (P&(PvQ))]		[T1-16,DR6-6d]
T6-117. VALID[(P&(PvQ)) <=> (Pv(P&Q))]		[T1-17,DR6-6d]
T6-118. VALID[(P&(Q&(PvQ))) <=> (P&Q)]		[T1-18,DR6-6d]
T6-119. VALID[(Pv(Qv(P&Q))) <=> (PvQ)]		[T1-19,DR6-6d]
T6-120. VALID[(P&(Q&R)) <=> (P&(Q&(R&(Pv(QvR)))))]		[T1-20,DR6-6d]
T6-121. VALID[(Pv(QvR)) <=> (Pv(Qv(Rv(P&(Q&R)))))]		[T1-21,DR6-6d]
T6-122. VALID[(Pv(P&(Q&R))) <=> (P&((PvQ)&((PvR)&(Pv(QvR)))]		[T1-22,DR6-6d]
T6-123. VALID[(P&(Pv(QvR))) <=> (Pv((P&Q)v((P&R)v(P&(Q&R)))]		[T1-23,DR6-6d]
T6-124. VALID[(Pv(P&(Q&R))) <=> (P&(Pv(QvR)))]		[T1-24,DR6-6d]
T6-125. VALID[(P&(Pv(QvR))) <=> (Pv(P&(Q&R)))]		[T1-25,DR6-6d]
T6-126. VALID[(P&(PvQ)&(PvR)&(Pv(QvR))) <=> (P&(Pv(QvR)))]		[T1-26,DR6-6d]
T6-127. VALID[(Pv(P&Q)v(P&R)v(P&(Q&R))) <=> (Pv(P&(Q&R)))]		[T1-27,DR6-6d]
T6-128. VALID[((P&Q)v(R&S)) <=> (((P&Q)v(R&S)) & (PvR))]		[T1-28,DR6-6d]
T6-129. VALID[((PvQ)&(RvS)) <=> (((PvQ)&(RvS)) v (P&R))]		[T1-29,DR6-6d]
T6-130. VALID[((P&Q)&(RvS)) <=> ((P&Q) & ((P&R)v(Q&S)))]		[T1-30,DR6-6d]
T6-131. VALID[((PvQ)v(R&S)) <=> ((PvQ) v ((PvR)&(QvS)))]		[T1-31,DR6-6d]
T6-132. VALID[((PvQ)&(RvS)) <=> (((PvQ)&(RvS)) & (PvRv(Q&S)] "Praeclarum"		[T1-32,DR6-6d]
T6-133. VALID[((P&Q)v(R&S)) <=> (((P&Q)v(R&S)) v (P&R&(QvS))		[T1-33,DR6-6d]
T6-134. VALID[((P&Q)v(R&S)) <=> (((P&Q)v(R&S)) & (PvR) & (QvS))]		[T1-34,DR6-6d]
T6-135. VALID[((PvQ)&(RvS)) <=> (((PvQ)&(RvS)) v (P&R) v (Q&S))		[T1-35,DR6-6d]

From CONT-theorems in Chapter 1:

T6-136. VALID[(P&Q) => P]	[T1-36,DR6-6a]
T6-137. VALID[(P&Q) => P)]	[T1-37,DR6-6a]
T6-138. VALID[(P&Q) => (PvQ)]	[T1-38,DR6-6a]
T6-139. VALID[(P& (QvR)) => ((P&Q) v R)]	[T1-39,DR6-6a]

<u>From SYN and CONT theorems with Quantifiers in Chapter 3</u>

From SYN-theorems in Chapter 3:

		<u>Quine’s Metatheorems</u>	
T6-311. VALID[((∀x)$_n$ Px) <=> (Pa$_1$ & P$_2$ &…& P$_n$)]			[T3-11,DR6-6d]
T6-312. VALID[((∃x)$_n$ Px) <=> (Pa$_1$ v P$_2$ v…v P$_n$)]		|	[T3-12,DR6-6d]
T6-313. VALID[(∀x)(Px & Qx) <=> ((∀x)Px & (∀x)Qx)]		ML*140	[T3-13,DR6-6d]
T6-314. VALID[(∃x)(Px v Qx) <=> ((∃x)Px v (∃x)Qx)]		ML*141	[T3-14,DR6-6d]
T6-315. VALID[(∀x)(∀y)Rxy <=> (∀y)(∀x)Rxy]		ML*119	[T3-15,DR6-6d]
T6-316. VALID[(∃x)(∃y)Rxy <=> (∃y)(∃x)Rxy]		ML*138	[T3-16,DR6-6d]
T6-317. VALID[(∀x)(P & Qx) <=> (P & (∀x)Qx)	}	ML*157	[T3-17,DR6-6d]
T6-318. VALID[(∃x)(P v Qx) <=> (P v (∃x)Qx)]	“Rules of	ML*160	[T3-18,DR6-6d]
T6-319. VALID[(∃x)(P & Qx) <=> (P & (∃x)Qx)]	Passage”	ML*158	[T3-19,DR6-6d]
T6-320. VALID[(∀x)(P v Qx) <=> (P v (∀x)Qx)]	}	ML*159	[T3-20,DR6-6d]
T6-321. VALID[(∀x)Px <=> ((∀x)Px & (∃x)Px)]			[T3-21,DR6-6d]
T6-322. VALID[(∃x)Px <=> ((∃x)Px v (∀x)Px)]			[T3-22,DR6-6d]
T6-323. VALID[(∃x)(Px & Qx) <=> ((∃x)(Px & Qx) & (∃x)Px)]			[T3-23,DR6-6d]
T6-324. VALID[(∀x)(Px v Qx) <=> ((∀x)(Px v Qx) v (∀x)Px)]			[T3-24,DR6-6d]
T6-325. VALID[((∀x)Px & (∃x)Qx) <=> ((∀x)Px & (∃x)(Px & Qx)]			[T3-25,DR6-6d]
T6-326. VALID[((∃x)Px v (∀x)Qx) <=> ((∃x)Px v (∀x)(Px v Qx))]			[T3-26,DR6-6d]
T6-327. VALID[(∃y)(∀x)Rxy <=> ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]			[T3-27,DR6-6d]
T6-328. VALID[(∀y)(∃x)Rxy <=> ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]			[T3-28,DR6-6d]
T6-329. VALID[(∀x)(Px v Qx) <=> ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx))]			[T3-29,DR6-6d]
T6-330. VALID[(∃x)(Px & Qx) <=> ((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx))]			[T3-30,DR6-6d]
T6-331. VALID[(∀x)(∀y)Rxy <=> ((∀x)(∀y)Rxy & (∀x)Rxx)]			[T3-31,DR6-6d]
T6-332. VALID[(∃x)(∀y)Rxy <=> ((∃x)(∀y)Rxy & (∃x)Rxx)]			[T3-32,DR6-6d]

<u>From CONT-theorems in Chapter 3</u>:

T6-333. VALID [(∀x)Px => Pa		[T3-33,DR6-6a]
T6-334. VALID [(∀x)(∀y)Rxy => (∀x)Rxx		[T3-34,DR6-6a]
T6-335. VALID [(∃x)(∀y)Rxy => (∃x)Rxx		[T3-35,DR6-6a]
T6-336. VALID [(∀x)Px => (∃x)Px]	ML*136	[T3-36,DR6-6a]
T6-337. VALID [(∃y)(∀x)Rxy => (∀x)(∃y)Rxy]	ML*139	[T3-37,DR6-6a]
T6-338. VALID [((∀x)Px v (∀x)Qx) => (∀x)(Px v Qx)]	ML*143	[T3-38,DR6-6a]
T6-339. VALID [(∀x)(Px v Qx) => ((∃x)Px v (∀x)Qx)]	ML*144	[T3-39,DR6-6a]
T6-340. VALID [(∀x)(Px v Qx) => ((∀x)Px v (∃x)Qx)]	ML*145	[T3-40,DR6-6a]
T6-341. VALID [((∀x)Px v (∃x)Qx) => (∃x)(Px v Qx)]	ML*146	[T3-41,DR6-6a]
T6-342. VALID [((∃x)Px v (∀x)Qx) => (∃x)(Px v Qx)]	ML*147	[T3-42,DR6-6a]
T6-343. VALID [(∀x)(Px & Qx) => ((∃x)Px & (∀x)Qx)]	ML*152	[T3-43,DR6-6a]
T6-344. VALID [(∀x)(Px & Qx) => ((∀x)Px & (∃x)Qx)]	ML*153	[T3-44,DR6-6a]
T6-345. VALID [((∀x)Px & (∃x)Qx) => (∃x)(Px & Qx)]	ML*154	[T3-45,DR6-6a]
T6-346. VALID [((∃x)Px & (∀x)Qx) => (∃x)(Px & Qx)]	ML*155	[T3-46,DR6-6a]
T6-347. VALID [(∃x)(Px & Qx) => ((∃x)Px & (∃x)Qx)]	ML*156	[T3-47,DR6-6a]

In the broad sense of logic, pure positive logical theorems can be greatly expanded using definitions of extra-logical terms which are negation-free in both definiendum and definiens.

6.3432 From SYN- and CONT-theorems in Chapter 4, with Negation

In Chapter 4, negation is introduced, but its SYN-theorems and CONT-theorems are such that no conjunction of components in these theorems is inconsistent because we carefully confined our derivations to SYN- or CONT-theorem in which no predicate letter occurred both POS and NEG by following restrictions in U-SUBab . Thus VALID conditionals can be derived using

DR6-6a If [P Cont Q] & not-Inc[P & Q], then Valid[P => Q], and
DR6-6d If [P Syn Q] & not-Inc[P & Q], then Valid[P <=> Q].

For example, we get an analogue of T5-437. TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))], a common axiom of quantification theory (Quine's *101 in *Mathematical Logic*) in the line:

T6-437. VALID [(∀x)(Px ⊃ Qx) => ((∀x)Px ⊃ (∀x)Qx)]   ML*101   [T4-37,DR6-6a]

Behind this one-line "proof" are solid proofs like the following one, which shows step by step how the Validity theorem is derived from the CONT-theorem and its lack of inconsistency**:.**

T6-437. VALID[(∀x)(Px ⊃ Qx) => ((∀x)Px ⊃ (∀x)Qx)]
Proof: 1) [(∀x)(Px ⊃ Qx) CONT ((∀x) Px ⊃ (∀x)Qx)]   [T4-37]
2) not-INC[(∀x)(~Px v Qx) & ((∃x)~Px v (∀x)Qx)]   [Inspection]
3) not-INC[(∀x)(~Px v Qx) & (~(∀x)Px v (∀x)Qx)]   [2),Q-exch,SynSUB]
4) not-INC[(∀x)(Px ⊃ Qx) & ((∀x)Px ⊃ (∀x)Qx)]   [3),T4-31,SynSUB(twice)]
5) VALID[(∀x)(Px ⊃ Qx) **.:** ((∀x)Px ⊃ (∀x)Qx)]   [1),4),Df 'VALID']
6) VALID[(∀x)(Px ⊃ Qx) => ((∀x)Px ⊃ (∀x)Qx)]   [5),VC\VI]

The complete list of validity theorems from SYN- and CONT-theorems in Chapter 4:

T6-411. VALID [(P&Q) <=> ~(~Pv~Q)]   [DeM2]   [T4-11,DR6-6d]
T6-412. VALID [(PvQ) <=> ~(~P&~Q)]   [Df 'v']   [DeM1]   [T4-12,DR6-6d]
T6-413. VALID [(P&~Q) <=> ~(PvQ)]   [DeM3]   T4-13,DR6-6d]
T6-414. VALID [(Pv~Q) <=> ~(~P&Q)]   [DeM4]   [T4-14,DR6-6d]
T6-415. VALID [(~P&Q) <=> ~(Pv~Q)]   [DeM5]   [T4-15,DR6-6d]
T6-416 VALID [(~PvQ) <=> ~(P&~Q)]   [DeM6]   [T4-16,DR6-6d]
T6-417. VALID [(~P&-Q) <=> ~(PvQ)]   [DeM7]   [T4-17,DR6-6d]
T6-418. VALID [(~Pv-Q) <=> ~(P&Q)]   [DeM8]   [T4-18,DR6-6d]
T6-419. VALID [P <=> (PvP)]   [v-IDEM]   [T4-19,DR6-6d]
T6-420. VALID [(PvQ) <=> (QvP)]   [v-COMM]   [T4-20,DR6-6d]
T6-421. VALID [(Pv(QvR)) <=> ((PvQ)vR)   [v-ASSOC]   [T4-21,DR6-6d]
T6-422. VALID [(P&(QvR)) <=> ((P&Q)v(P&R))]   [v&-DIST]   [T4-22,DR6-6d]
T6-424. VALID [(∃x)~Px <=> ~(∀x)Px]   [Q-Exch2]   ML*130   [T4-24,DR6-6d]
T6-425. VALID [(∀x)~Px <=> ~(∃x)Px]   [Q-Exch3]   ML*131   [T4-25,DR6-6d]
T6-426. VALID [$(\exists x_1)\ldots(Ea_n)\sim P<x_1,\ldots,x_n>$ <=> $\sim(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n>$]
[Q-Exch4]   ML*132   [T4-26,DR6-6d]
T6-427. VALID [$(\forall x_1)\ldots(\forall x_n)\sim P<x_1,\ldots,x_n>$ <=> $\sim(\exists x_1)\ldots(\exists x_n)P<x_1,\ldots,x_n>$]
[Q-Exch5]   ML*133   [T4-27,DR6-6d]

From CHAPTER 4, SYN- and CONT-theorems (with TF-conditionals).

T6-430. VALID [(P ⊃ Q) <=> ~(P&~Q)]	[Df '⊃']		[T4-30,DR6-6d]
T6-431. VALID [(~PvQ) <=> (P ⊃ Q)]			[T4-31,DR6-6d]
T6-432. VALID [(P ⊃ Q) <=> (~Q ⊃ ~ P)]			[T4-32,DR6-6d]
T6-433. VALID [(∃x)(Px ⊃ Qx) <=> ((∀x)Px ⊃ (∃x)Qx)]		ML*142	[T4-33,DR6-6d]
T6-434. VALID [(∃x)(Px ⊃ Q) <=> ((∀x)Px ⊃ Q)]	} "Rules of	ML*162	[T4-34,DR6-6d]
T6-435. VALID [(∀x)(Px ⊃ Q) <=> ((∃x)Px ⊃ Q)]	} Passage"	ML*161	[T4-35,DR6-6d]
T6-436. VALID [((∃x)Px ⊃ (∀x)Qx) => (∀x)(Px ⊃ Qx)]		ML*148	[T4-36,DR6-6a]
T6-437. VALID [(∀x)(Px ⊃ Qx) => ((∀x)Px ⊃ (∀x)Qx)]		ML*101	[T4-37,DR6-6a]
T6-438. VALID [(∀x)(P ⊃ Qx) => (P ⊃ (∀x)Qx)))]	(Thomason's Axiom 4)		[T4-38,DR6-6d]
T6-439. VALID [(∀x)(Px ⊃ Qx) => ((∃x)Px ⊃ (∃x)Qx)]		ML*149	[T4-39,DR6-6a]
T6-440. VALID [((∃x)Px ⊃ (∃x)Qx) => (∃x)(Px ⊃ Qx)]		ML*150	[T4-40,DR6-6a]
T6-441. VALID [((∀x)Px ⊃ (∀x)Qx) => (∃x)(Px ⊃ Qx)]		ML*151	[T4-41,DR6-6a]

6.3433 From SYN- and CONT-theorems based on Axiom 6.06, Chapter 6

So far, Validity-theorems have had '=>' as the main connective, but none have had conditionals in either the antecedent or the consequent. For example, we have proved,

T6-429. VALID[(∀x)(Px ⊃ Qx) => ((∀x)Px ⊃ (∀x)Qx)]

But which, if any, among the following '=>'-for-'⊃' analogues are VALID?

[(∀x)(Px => Qx) => ((∀x)Px => (∀x)Qx)]?
[(∀x)(Px => Qx) => ((∀x)Px ⊃ (∀x)Qx)]?
[(∀x)(Px ⊃ Qx) => ((∀x)Px => (∀x)Qx)]?
[(∀x)((Px => Qx) & (∀x)Px) => (∀x)Qx)]?

Let us call C-conditionals with occurrences of '=>' in antecedent or consequent "2nd-level conditionals".

By U-SUB, all theorems gotten by uniform substitution of conditional wffs for predicate letters in previously established VALIDITY-theorems are valid provided (i) the conditional substituted is not inconsistent and (ii) there is at least one disjunct of the MODNF of the conjunction of antecedent and consequent, which is not-INC.

A few VALIDITY-theorems with '=>' in the antecedent of the conditional (2nd-level conditionals) follow from Axioms and Definitions in the base. The most important, is "modus ponens", T6-613 and T6-621:

T6-613. VALID[(P & (P=>Q)) => Q] "Modus Ponens"

Proof: 1) (P & (P=>Q)) CONT Q] [T6-13]
2) not-INC(P & (P=>Q) & Q) [Inspection]
3) VALID[(P & (P=>Q)) => Q] [1),2),Df 'VALID']

T6-613 holds for all consistent expressions—regardless of the mode of discourse or sentential operators employed.[21]

But the theorems which depend on Axiom 6 for their derivation are quite a limited group, and this may at first seem inadequate. Our interest here is in those VALIDITY-Theorems which depend on Axiom 6 and the new definitions for their derivation.

We discuss first the VALIDITY-theorems derivable from Axiom 06 which have conditionals as components of conditionals. Then we will mention some Principles of Valid Inference based on Axiom 6.

To get VALIDITY-theorems we must begin with CONT-theorems derivable from Axiom 6, since the VALIDITY-theorems will be based on them.

<u>From SYN- and CONT-theorems in Chapter 6</u>.

T6-611. VALID[(P&(P=>Q)) <=> ((P&Q)&(P=>Q))]		[T6-11,DR6-6d]
T6-612. VALID[(P & (P=>Q)) => (P&Q)]		[T6-12,DR6-6a]
T6-613. VALID[((P=>Q) & P) => Q]	‘MP’,”Modus Ponens”, ML*104	[T6-13,DR6-6a]
T6-614. VALID[(P<=>Q) => (Q=>P)]		[T6-14,DR6-6a]
T6-615. VALID[(P<=>Q) => (P=>Q)]		[T6-15,DR6-6a]
T6-620. VALID[((∀x)((Px => Qx) & Px) => (∀x)Qx)]		[T6-20,DR6-6a]
T6-621. VALID[((∀x)(Px => Qx) & (∀x)Px) => (∀x)Qx]		[T6-21,DR6-6a]

All theorems gotten by one or more applications of U-SUBab in any of the theorems above, will preserve both the containment and consistency requirements for Validity. By U-SUB, all theorems gotten by uniform substitution of conditional wffs for predicate letters in previously established VALIDITY-theorems are valid provided (i) the conditional substituted is not inconsistent and (ii) there is at least one disjunct of the MODNF of the conjunction of antecedent and consequent, which is not-INC.

6.344 Principles of Inference as Valid Conditionals in Applied A-logic

Up to this point all rules of inference presented in this book, including derived rules, have been expressed in English as conditional statements of the form ‘(If P, then Q)’. Further, it has been presupposed that the reader would recognize that we were presenting them as <u>valid</u> principles of logic. Presumably the reader has recognized this. If he or she accepted them they were accepted as valid. If they were questioned, it was their validity which was questioned.

In this chapter we have a symbol, ‘=>’, for the concept of “if…then” which we wish to use in A-logic and we have defined the predicate ‘< 1 > is valid’ as in terms of the concepts of containment and consistency which are the central semantic concepts of A-logic. We can now express our principles of inference in the symbolism of A-logic, making the claim of logical validity explicit in a precise, defined sense. To take the simple examples in the preceding sections,

---

21. T6-613. VALID[(P & (P=>Q)) => Q)] — VALID[[(~P & (~P=>Q)) => Q)]
Instances in truth-logic: VALID[[(TP & (TP=>TQ)) => TQ)]
VALID[(FP & (FP=>FQ)) => FQ)]
Instances from the logic of imperatives: VALID[(TP & (TP=>Q!)) => Q!)]
Instances from the logic of questions: VALID[(TP & (TP=>Q?)) => Q?)]
Instances from deontic logic (‘O’ for ‘ought to be’): VALID[[(OP & (OP=>OQ)) => OQ)]
Instances from mixed truth and deontic: VALID[[(TP & (TP=>OQ)) => OQ)]

From Df 'Inc' and Df 'Taut'

DR6-5a. If [P Cont Q] then Inc(P=> ~Q) I.e., |= Valid [(P Cont Q) => Inc (P=> ~Q)]

DR6-5b. If [P Cont Q] then Taut[~(P=> ~Q)] I.e., |= Valid [(P Cont Q) => Taut (~(P=> ~Q))]

From Df 'Valid=>':

DR6-6a. If [P Cont Q] & not-Inc[P&Q], then Valid[P=>Q].
I.e., |= Valid[((P Cont Q) & ~Inc (P&Q) => Valid (P=>Q)].

DR6-6d. If [P Syn Q] & not-Inc[P&Q], then Valid[P<=>Q].
I.e., |= Valid[((P Syn Q) & ~Inc (P&Q) => Valid (P<=>Q)].

DR6-6e. If Valid[P=>Q] then Valid[P, **.:** Q] I.e., |= Valid [Valid (P=>Q) => Valid (P, **.:** Q)]

DR6-6f. If Valid[P, **.:**Q] then Valid[P=>Q] I.e., |= Valid [Valid (P, **.:**Q) => Valid (P=>Q)]

These are not theorems of pure formal A-logic, since they have substantive predicates in them, like 'Syn', 'Inc' and 'Valid', which are not purely syncategorematic "logical words". The syncategorematic words of formal logic, 'and', 'or', 'if...then,', 'not' are used in all disciplines. The substantive predicates are only the special semantic predicates necessary for logic. They are not formally VALID, because not based only on the meanings of syncategorematic words. But they are 'Valid' theorems of applied A-logic, like theorems in the logic of mathematics, the logic of Physics, the logic of Truth-statements or the logic of value judgements. The VALIDITY-theorems of formal A-logic, apply to all disciplines, including the discipline of logic itself. These are special theorems for logic itself.

All Rules and Derived Principles of inference can be similarly put into the symbolism of A-logic. Though some are more difficult to translate than others, when these principles are translated from English into the logical symbolism of A-logic they are given a more fixed and precise meaning than ordinary English. Their meanings are given in the definitions we have given. The definitions of logic predicates and principles of inference at the base of formal A-logic are expressible in the symbolic language of A-logic.

Translating these principles into the language of A-logic and asserting that they are valid, is not the same as proving them Valid. Most of these principles are best understood as involving implicit predications of truth. In saying "If [P CONT Q] then INC(P => ~Q)]" we mean that no matter what well-formed expressions is substituted for P and Q, "If it is true that [P logically contains Q] then it is true that [If P then ~Q] is logically inconsistent". Therefore detailed analyses of proofs of these inference rules in A-logic will come after Chapter 7 (which prefixes the truth-operator to wffs of M-logic), and Chapter 8 (where logic is extended to cover truth-claims about or within C-conditionals).

A-logic differs from M-logic in allowing the principles of inference, which M-logic would call metatheorems, to become theorems of applied A-logic itself. A-logic can apply to itself without inconsistency. In contrast, if the semantic meta-theorems of M-logic are made theorems of M-logic, paradoxes follow; M-logic cannot consistently be used to prove its own principles of inference.

6.4 Valid Conditionals in A-logic and M-logic Compared

Let us review and compare the concept of validity in A-logic and the concept of validity used in M-logic.

In M-logic "valid" wffs are all synonymous with tautologies—denials of inconsistencies. Tautologies are of little or no interest in A-logic; and inconsistencies are of interest primarily as errant properties to be identified in order to avoid or eliminate them. In formal A-logic no wff which is M-valid (i.e., a tautology) can be A-valid, because none are synonymous with a C-Conditional. The only wffs, predicates or statements which are A-valid are C-conditionals. A C-conditional can be inconsistent but only its denial—never the C-conditional itself—can be TAUT. Since no C-conditionals are denials of inconsistencies, none can be TAUT. No wff that is A-valid is INC or TAUT, and no wff that is TAUT or INC is A-valid. Thus no M-valid wff or statement is Valid in A-logic (though many, but not all, M-valid inferences are A-valid).

Although A-valid C-conditionls and M-valid wffs (TAUT-theorems) are both derivable from SYN and CONT-theorems, they are derived in different ways. Validity-theorems are derived from SYN- and CONT-theorems directly, provided their components are jointly consistent. INC- and TAUT-theorems are derived from CONT- and SYN-theorems by negating at least one conjunct in the antecedent, and at least one atomic component of such theorems must occur both POS and NEG.

There are other differences.

A-validity is a property only of arguments and argument-schemata, or of C-conditional wffs or statements. Disjunctions and conjunctions of disjunctions of elementary wffs can be tautologous, unfalsifiable and true but never A-valid. By contrast, every wff which is M-valid is synonymous with some disjunction or with a conjunction of disjunctions without C-conditionals.

A-validity is attributed only to those arguments or C-conditionals in which the conclusion (or consequent) is contained in the premisses (or antecedent), or in which the denial of the conclusion (consequent) contains the denial of the premisses (antecedent). For M-validity there is no *requirement* of a positive connection betweem premiss and conclusion or antecedent and consequent, even though the "antecedent" and "consequent" of M-valid TF-conditionals often are such that the latter logically contains the former. In such cases the related argument schemata are A-valid, but the TF-conditionals are not.

In A-logic the antecedent (premiss) and the consequent (conclusion) must be capable of being true together—(A&C) must be consistent. No such requirement is made in M-logic.

A-validity is independent of the concept of truth. The concepts of A-validity and logical inconsistenty are prior to the concept of truth. They apply to other modes of discourse than truth-talk. Even in analytic truth-logic, A-valid wffs have many cases which are not true (and not false) but '0'; this is true in particular of C-conditionals. Non-falsifiability, rather than universal truth, is a necessary (though not sufficient) property of A-valid wffs in A-logic. By contrast, according to M-logic's semantics, a wff is M-valid if and only if it is true for all values of its variables.[22] No sameness of content and no logical connection between premiss and conclusion is required. M-valid theorems are viewed as being universally true—true about everything. In A-logic the validity of a C-conditional is not determined by its truth conditions. Although it is a necessary condition of validity that that the consequent (conclusion) not be false if the antecedent (premiss) is true, this not a sufficient condition for either A-validity or for the truth of a conditional.

Many traditional principles of valid inference which M-logic seeks to express are included in Analytic *truth*-logic, rather than in the universal formal A-logic of this chapter. In purely formal A-logic, using only the words 'and'. 'or', 'not', 'if...then' and 'all', we can not derive the traditional principles of Hypothetical Syllogism, of Transposition, of Addition, or Modus Tollens as VALIDITY-theorems. They rely on the concept of 'truth' which is handled in the truth-logic of Chapter 7 and 8.

Many principles which are extremely useful in reasoning logically, require something more than the basic formal A-logic based solely on the meanings of syncategorematic words. Though formal A-logic provides the rich variety of Validity-theorems listed above and these hold in every field of knowlege, the most useful principles of logic in specific fields, require supplements to Formal A-logic based on mean-

22. Cf. Quine W.V., *Methods of Logic* (4th Ed.), p. 60, p172. The latter says, "The definition of validity is as before: truth under all interpretations in all non-empty universes".

ings of syntactical operators and mood indicators. Traditional Validity-schemata which are not derivable in formal A-logic, are derivable (some in altered form) with the introduction of the Truth-operater and Truth-logic (Chapters 7 and 8).

The universal, formal A-logic of this chapter, also differs from M-logic in a more elementary sense. It does not include, or aspire to include, all of the traditional principles associated with the logic of truth-claims. For example, [((~P&P) v Q) => Q)] is not valid in purely formal A-logic, since the antecedent does not logically contain the consequent. However Chapters 7 and 8 include Validity-theorems in which the inconsistency of a component in a truth-claim, entails the truth of a contingent component. For example if [Either (P & ~P) or Q] is true, then Q must be true, since (P&~P) can not be true. Thus [T((~P & P) v Q) => TQ)] is A-valid, though [((~P&P) v Q) => Q)] is not.

The traditional principles of Hypothetical Syllogism, Transposition, Addition and Modus Tollens are not derivable as VALIDITY-theorems in formal purely A-logic for various reasons, but they emerge in the truth-logic of Chapter 7 and 8.

The hypothetical syllogism, '[((P=>Q) & (Q=>R)) => (P=>R)]' would require the consequent of the first C-conditional to be the antecedent of the next one. If the first C-conditional is in the imperative or interrogative mood, the consequent is not descriptive but a prescriptive or interrogative expression. Putting a prescriptive or interrogative expression in the antecedent of the second conditional would make it ill-formed. Thus "((TP => Q!) & (Q! => R)) => TP => R)" is ill-formed.[23] Thus if formal A-logic is to be a universal logic which allows directives,as well as descriptive predicates and sentences, for its variables, '[((P=>Q) & (Q=>R)) => (P=>R)]' will be an inappropriate way to symbolize a valid hypothetical syllogism. Instead of this simple symbolic representation, valid hypothetical syllogisms need specific operators to render them valid, as in,

DR6-119. [(VALID (P =>Q) & VALID (Q => R)) => VALID (P => R)]
and T8-815 Valid[(T(P => Q) & T(Q=>R)) => T(P => R)]
(for: " 'If (P =>Q) is true and (Q=>R)is true, then (P =>R) is true' is Valid")

which were mentioned earlier..

Addition, in the form [P => (P v Q)] is not A-valid, since the consequent has components which do not occur in the antecedent. But [TP => T(P v Q)] will be established as an "implication theorem" in truth-logic, based on the meanings of 'either or' amd 'is true'.

If some predicate letters are negated in the consequent and not in the antecedent, or vice versa, the C-conditional can not be A-valid. This rules out all '=>'-for-'⊃' analogues of the Transposition TAUTologies in M-logic: [(P=>Q) => (~Q => ~P)], [(~P=>Q) => (~Q => P)], etc., But in Chapter 8 a modified version of transposition with the C-conditionals and the truth- operator is established as T8-824. Valid [T(P => Q) => ~F(~Q => ~P)], This theorem says that "If it (P =>Q) is true then (~Q => ~P) is not false" is Valid. But for good reasons to be discussed, certain other formulations of the traditional transposition principle are not provable.

Modus Tollens [((P=>Q) & ~Q) => ~P] also fails, though in the Truth-logic of Chapter 8 we can prove the A-implication theorem, Ti8-844. Valid$_I$ [((TP =>TQ) & ~TQ) => ~TP].

---

23. The prescriptive, "Do X!", must not be confused with a description of X's being done, "X is done". "If there is a rattlesnake in the woodpile, kill it! If the rattlesnake is killed, rejoice!" does not entail "If there is a rattlesnake in the woodpile, rejoice!" because 'Kill it!' Is not the same as 'It is killed', and 'If kill it! then rejoice!" would not be a well-formed conditional.

In general nested conditionals in which the consequent contains components not in the antecedent, e.g. [(P=>Q) => ((Q=>R) => (P=>R))], are never A-valid, by the meaning of containment.

If A-logic is to be open to including the logic of directives and the logic of questions, as well the logic of expressions in the declarative mood, it must recognize that principles which work for expressions in one mood, may not work for other moods.

Thus there are very real differences between Formal A-logic and M-logic. But looking past the differences in terminology and syntax, there is a very substantial common area of agreement in which many ordinary conditionals, whether construed as C-conditionals or as TF-conditionals, will be called "valid" both in M-logic and in A-logic. Though '⊃' and '=>' have different meanings, both are efforts to capture an ordinary meaning of 'If…then'. And despite the different meanings given to both 'valid' and to 'If…then' both A-logic and M-logic seek to capture central features of the meaning of '<1>is valid' as used in traditional logic and much rational discourse. There is a large overlap—areas of agreement in the sense that for the same P and Q, (P ⊃ Q) is M-valid and (P=>Q) is A-valid and 'If P then Q' is valid in ordinary language.

These areas of agreement are important. The continuity of discussion about alternative logics, is grounded in common objectives and common agreements. The changes we propose—(i) to use the concept of A-validity in application to arguments and C-conditionals, rather than using M-validity and M-valid TF-conditionals, and (ii) to include among logical inconsistencies inconsistent C-conditionals (as defined in Df.6-2), thus adding many new categories of logical inconsistency and tautology—are offered as better means to reach common objectives in a sea of principles that contain a variety of problems and anomalies, but many more principles on which all logicians agree.

The main task is to formulate reliable rules of valid inference for use in reasoning and problem solving in areas beyond logic itself. In A-logic this can be done in the absence of inconsistency and tautology. M-logic, as a system for the enumeration of inconsistencies or denials of inconsistencies (tautologies), is a side show to this task. We need to be able to test individual predicates and propositions for inconsistency; we need to recognize inconsistency in order to avoid it. But the main task of logic is not accomplished by creating systems for generating either inconsistencies or tautologies.

6.5 A-logic Applied to Disciplines Other Than Logic

Beyond the VALID wffs of formal logic, there are Valid wffs of other disciplines than logic. There the synonymies and containments are relations of substantive meanings, intended or defined. Reasoning based on synonymies or containment of extra-logical words falls under the rubric of Applied (vs Formal) logic. When principles of formal A-logic are applied to postulates and definitions of extra-logical disciplines, we speak of this as "A-logic in the broad sense".

When A-logic is applied to itself (see Section 6.344) its principles are not based solely on the meanings of 'and', 'not' and 'if…then'. The logical predicates 'Syn', 'Cont', 'Inc', 'Taut' and 'Valid' have substantive meanings which we have tried to define clearly. The definiens of their definitions referred, for the most part, to expressions with different sorts of purely logical structures. But these terms are not syncategorematic; they are not operators like 'and' and 'not' and 'if…then' which presuppose nothing substantive. They are particular predicates which apply to particular kinds of entities. These definitions are used to establish the logical validity (in the broad sense) of the rules of inference used in Logic. In this sense, our meta-logic consists of results based on definitions of substantive concepts used in logic, just as the logic of mathematics, or the logic of physics, is developed from definitions of their most important predicates.

Among disciplines other than Formal Logic, some extra-logical synonymies and containments are due to the meanings of specific non-syncategorematic operators, including the truth-operator, which we will consider in Chapter 7. But there are also a vast number of synonymies and semantic containments intended or defined with ordinary extra-logical noun phrases, verbs and adjectives. These, like the synonymies of formal logic, are semantic properties and relationships; but unlike formal logic they are not limited to meanings or properties of logical structures involving only parentheses, order, and syncategorematic locutions. These are predicates which apply to some things and not to others. Often they are only relevant in certain fields of inquiry, i.e., to certain classes of entities. Logic in its broader sense relies on this much larger class of synonymy relations.

Formal logic provides a ground for logic in the broad sense, because if the latter is to be rigorous, it must refer to meanings or definitions. If a definition is to be used in rigorous deduction and reasoning, the definiens must be expressible with a formal logic structure. Provided extra-logical definitions are constructed under the guidance of appropriate principles of formal logic, logical reasoning in all applied Logics proceeds within the framework of Formal logic.

In many cases, ordinary language has customary uses which are the basis of agreement between speaker and listener on the meanings of non-logical words as used in particular segments of discourse. There need be no assumption that such words have fixed, eternally unchangeable ordinary meanings. A word or phrase may be defined by the speaker and accepted by listeners, as having a novel and unusual meaning for the purposes of particular discussion, or a specific line of inquiry or action, or for the application of a particular civil law. The logical consequences are just as binding within a local or temporary contexts, as in cases where more universal or permanent meanings are envisaged. What is important is: when in doubt see what meaning the parties intend should be given to their terms.

In this book we give no detailed studies of the way the conditional operates in most of these extensions. But there is one exception—that special extension of logic which deals with the predicate "<1>is true", i.e, with the truth-operator. This is developed in Chapter 8. In Chapter 7 the truth-operator is examined without the C-conditional, to show how it operates in M-logic.

Part II

Truth-logic

Chapter 7

"Truth" and Mathematical Logic

7.1 Introduction

The purpose of this chapter is to introduce 'truth' into the object language of A-logic without paradox, and to show how resulting theorems agree with or differ from the semantics of M-logic. To make these differences clear, wffs and theorems of Chapter 6 with C-conditionals ('=>') are omitted, leaving only the wffs of M-logic (reducible to negations, conjunctions and quantifiers) with 'T' prefixed to them.

Thus to the formal A-logic of Chapters 1 through 5, we add the primitive, 'T', a rule of formation, definitions, axioms and two rules of inference in which 'T' occurs as follows:

Rules of Formation:	RF 7-1. If A is wff, T(A) is wff
Abbreviations:	Df 'F'. F[P] Syn_{df} T~P
	Df '0'. 0[P] Syn_{df} (~TP & ~FP)
Axioms:	Ax.7-1. TP Syn (TP & ~FP)
	Ax.7-2. FTP Syn ~TP
	Ax.7-3. T(P & Q) Syn (TP & TQ)
	Ax.7-4. T(P v Q) Syn (TP v TQ)
	Ax.7-5. ((TP & ~TP)v TQ)) Cont TQ
Rules of Inference:	R7-1. If (P Syn Q) then (TP Syn TQ)
	R7-2. If Inc(P) then $\models$ ~T(P)

In Section 7.1, the central differences between the treatment of "truth" in mathematical logic and in A-logic are outlined.

Section 7.2 presents a set-theoretical account of the correspondence theory of truth presupposed in A-logic. This explication of the meaning of the predicate '<1>is true' is contrasted with Tarski's "semantic conception of truth" and some of its offspring.

Section 7.3 presents trivalent truth-tables for the three basic truth-operators, 'T', 'F' and '0', and their occurrences with the syncategorematic operators '~', '&', 'v' and '⊃'. These will be used as checks on various theorems in the next section.

Section 7.4 presents the base of an axiomatic system for analytic truth-logic theorems about logical properties and relations of T-wffs. This section includes theorems of synonymy and containment, inconsistency and tautology, and theorems asserting logical unsatisfiability and logical unfalsifiability, logical truth and falsehood. It also defines ‘analytic implication’ and proves a set of Impl-theorems.

Section 7.5 uses truth-tables to help establish consistency and the completeness of this system with respect to the tautologies and inconsistencies of M-logic.

The logic of this chapter, like mathematical logic, covers the logic of certain sentences that are either true or false exclusively. But there are basic differences which we will now discuss.

7.11 Truth-logic as a Special Logic

The first difference has to do with the fundamental nature of basic logic, and the relation of that logic to the logics of special operators and special fields. M-logic assumes that it, and all extensions and special logics, are concerned with determinations of truth and falsity. A-logic does not.

Although M-logic and its extensions prohibit the inclusion of statements about the truth of their own statements, in the semantic theories of M-logic it is made clear that M-logic deals with indicative sentences and assumes that these are always implicit truth assertions and therefore that every such truth-assertion is either true or false exclusively. It further assumes that every indicative sentence is acceptable only if it true, and that the only reason for non-acceptance of an well-formed indicative sentence is the judgment that it is false. M-logic is presented as a truth- logic. Its theorems are presented as universal **truths**. Each of its logical constants, and the properties of inconsistency and tautology, are explicated as **trut**h-functions. Its semantics is based on truth-values, its variables and wffs and are place-holders for propositions, i.e, expressions which must either be **true or be false** (never neither). To take M-logic as the basis of all logic is to assume other branches of logic will be developed as extensions of this truth-logic.

In formal A-logic, the properties and relations of truth and falsity are not its central concern. The basic properties and relations that it studies are validity, inconsistency, and logical synonymy and containment. Its theorems, of course, implicitly assert that it is true that the wffs or predicates in subject positions have the logical properties and relations attributed to them. But its task is to make true statements about the logical properties and relations of various forms of expressions, not to make true statements about whether expressions are true or false. (Just as astronomers in seeking truths about properties and relations of stars make true statements about stars, not true statements about the truth or falsehood of statements about stars).

Truth-logic is one of many special kinds of logic which depend on the basic A- logic of predicates which contain no words other than syncategorematic words—the A-logic of Chapter 6. Truth-logic is an extension of formal analytic logic, which studies the logical groupings and structures built up with ‘and’, ‘not’, ‘if-then’, ‘all’ and the special predicate, ‘<1>is true’ or the operator, ‘It is true that..’. (Though in this chapter we do not include ‘If…then’).

The expression “<1>is true” is not syncategorematic. It has a semantic content, conveying the idea of a thing or things in an objective field of reference which has no inconsistencies and of relations of correspondence that may or may not obtain between the ideas or meanings of sentences and actual properties or relations of things in the field of reference. The “theory of truth” in Section 7.2, is an explication of this meaning of ‘<1>is true’.

The logical properties and relations of any truth-logic are contained in, or follow from, the meaning given to the predicate ‘<1>is true’ and to the sentential operator ‘It is true that…”. These two ways of using the word ‘true’ may be described, respectively, as (i) its use as a predicate and (ii) its use as a sentential operator. Although they are not quite synonymous, ‘<1> is true’ is correctly predicated of a term denoting a sentence if and only if ‘It is true that P’ is true.

Other comparable sentential operators, which are also not syncategorematic, may neither entail nor imply truth. 'It is morally right that P' does not contain or imply 'It is true that P'; nor does the converse relation hold.[1] 'It is possible that P' does not entail 'It is true that P', but the converse entailment does hold. If general logic is to be useable in all kinds of rational problems, truth-logic must be separated from, but connectable with, the logics of other modes of discourse.

By taking predicates rather than sentences as its basic subject matter, formal A-logic frees itself from the assumption that it is concerned only with expressions that are either true or false.

But even if we confine ourselves to sentences, not all sentences are judged by whether they are true or not. Questions and directives or suggestions for action are neither true nor false. Yet there are logical relations and logical properties of predicates in interrogatives and directives. It is the proper business of Logic to identify relations of logical synonymy, containment and implication between questions and between directives, and to examine the logical roles of tautologous or inconsistent predicates in questions and directives.

Even if we confine ourselves to indicative sentences, many meaningful indicative sentences are not intended as truth-assertions. Jokes, imaginative essays, fictions and poetry contain indicative sentences which are understood, absorbed, and accepted for the interesting situations or emotive effects they evoke without any thought of whether they are true in the important, strict sense discussed below. Thus not all indicative sentences are implicitly truth-claims. Some are and some are not.[2] There are logical properties and logical relationships between the ideas and sentences in jokes, poetry and fiction which have nothing to do with their truth or falsehood but may have something to do with whether jokes are funny, poetry is incisive, fiction is authentic.

The point is that there is a clear distinction between the utterance of an indicative sentence and the claim that that sentence is true. This distinction is not just ignored in M-logic; in its semantic theories it is denied by conflating [P] with ['P'is true']. In A-logic, by contrast, the distinction is essential and formally recognized. It becomes particularly important in the logic of conditionals, which may be valid when the antecedent is false, but is neither true nor false about states of affairs in which the antecedent is not fulfilled.

In analytic logic, the logic of truth-statements is one of the two or three most important branches of logic, all of which rest on a foundation of the logical properties and relations of syntactical forms and the meanings of the purely "logical" words. Formal A-logic is intended to serve as the base for all branches in this wide scope.

The enormous importance of the concepts of truth and falsity is not diminished by demoting it from its central role in M-logic. Of the many branches of logic, the logic of truth- statements, the logic of value-judgments, the logic of directives and the logic of questions, are the most important because of the wide scope of each. Many other special logics—a logic for physics, a logic for mathematics, a logic for

---

1. Whether moral judgments are truth-assertions is a philosophical issue. But a logic of moral predicates or sentential operators can be developed without confusing the principles of entailment for moral operators with those for the truth-operator.

2. If one is uncertain, and wants to know if an ordinary sentence is implicitly intended as a truth- assertion, one can generally get an explicit first-level response to "Is that true?" But in general, if any sentence is written in a news report, a corporation's financial report, history book, geography book, astronomy book, minutes of a meeting, etc., it is implicitly understood that it is intended to be taken as a true statement and that the author can be held accountable for its truth. If a sentence is uttered in a joke, or imaginary day-dreaming, or a novel, or a play, or in describing a non-documentary movie one does not assume this; one can assume they are intended to convey merely the ideas they evoke.

biology … down to logics of particular local terms—can be developed. A-logic is intended to provide a foundation for the whole wide scope of actual or possible logical investigations in specific fields.

The central importance of truth-logic remains. It is implicitly understood with respect to news reports, history books and geography books that their statements purport to be true (in the correspondence sense). Science and mathematics texts have mixtures of implicit or explicit truth- assertions, together with directives (on how to solve, or what to do), and questions. Many sentences in ordinary conversation, business and practical affairs, are offered and received as statements which the speaker intends the listener to regard as true, though such sentences are rarely explicitly asserted to be true. Sentences in a newspaper rarely begin "It is true that…", and are almost never presented in quotes followed by "…is true." But we know by the context that they are intended to be taken as true sentences.

Every search for truth in every area of inquiry, must be subject to the critical examination of truth claims and for this it must presuppose and require an adequate logic of truth-statements.

7.12 A Formal Definition of Analytic Truth-logic

"Truth-logic" is the logic of the predicate '<1>is true' and/or the sentential operator, "It is true that…". Its subject matter covers explicit and implicit assertions of truth, non-truth, falsehood and non-Falsehood. I use the letter 'T' prefixed to a wff, to stand for the phrase, "It is true that…". This phrase, and thus 'T', is called a truth-operator. A sustained search for truth in any area of human inquiry requires the use of principles of logic for moving from premisses asserting the truth or non-truth, falsity or non-falsity of certain expressions to the truth, falsity, non-truth or non-falsity of a conclusion.

By a T-wff, I mean any wff such that every one of its atomic wffs lies in the scope of an occurrence of 'T'. By T-logic (or "truth-logic") I mean the study of the logical properties and relationships of T-wffs, i.e., the study of the logic of expressions which are explicitly or implicitly related to truth. In rational searches to arrive at truths, truths follow logically only from truths—not from value judgements, directives, or questions.

Prefixing 'It is true that…' is not necessary in most contexts where the intention is to convey or arrive at truth; it is understood by the context, hence omitted. Challenges, such as "Is that true?" concerning its truth or falsity discloses its implicit presence.

To make every truth-claim explicit by prefixing 'It is true that…' would make language impossibly cumbersome. However, in such contexts any statement, simple or complex, intended as a truth-assertion may be thought of as implicitly prefixed by 'It is true that…'. Were this prefix actually applied no one would hold that the authors' intention or meaning was changed. Implicit truth-assertions are thus equivalent to statements which make them explicit by prefixing a single 'T' (for 'It is true that…'). For purposes of rigorous logical analysis it is helpful to recognize that in discourse of this type, the intended meaning would not be changed by prefixing 'It is true that..' to any or all indicative statements.[3]

The primitive symbol 'T' is not the only truth-operator. By a T-operator in general, I mean any ordered sequence of signs prefixed to a well-formed formula of M-logic or A-logic, which contains one or more occurrences of 'T' with or without occurrences of '$\sim$'. As we shall see, there are just four basic

3. To say that *usually* if a person says "It snowed today in Detroit" they intend the listener to take this as "It is true that it snowed today in Detroit" is not the same as saying (as Tarski does in his Semantic Theory of Truth) that these two sentences are equivalent. What we are saying is that if we want a logic of just those linguistic expressions which purport, implicitly or explicitly, to be true or false, then, since we have restricted our investigation to that kind of discourse, we may prefix 'T' to any expression large or small, when it helps to make the logic clearer, without changing the meaning. The difference between this and Tarski is made clear in the truth-tables in Section 7.24, and in theorems of A-logic.

T-operators: ‘TP_i’, ‘$\sim TP_i$’, ‘$T\sim P_i$’, ‘$\sim T\sim P_i$’ to which all others can be reduced. Abbreviations of expressions defined solely in terms of ‘T’ and ‘$\sim$’, are also, of course, T-operators. We use just two such abbreviations:

$$\text{Df ‘F’: } [FP_i \ Syn_{df} \ T\sim P_i] \qquad \text{Df ‘0’: } [0P_i \ Syn_{df} \ (\sim TP_i \ \& \ \sim FP_i)]$$

Thus ‘FP_i’ and ‘$\sim FP_i$’ are abbreviaitions of ‘$T\sim P_i$’ and ‘$\sim T\sim P_i$’.

Every complex prefix formed by prefixing one or more occurrences of ‘$\sim$’, ‘T’, ‘F’ or ‘0’ (e.g.,’$\sim T\sim F0\sim T\sim\sim(\ldots)$’) is also called a T-operator. Every wff with a complex T-operator prefixed to it is reducible to a synonymous “normal form” T-wff in which ‘T’ occurs only in one of the four basic T-operators, T, F, $\sim$T or $\sim$F, and these are prefixed to all and only the atomic wffs.[4] Thus, though rules of formation allow unlimited varieties of complex truth-operators, all wffs are synonymous with wffs in which ‘T’ only applies to ‘P’ or ‘$\sim$P’, i.e., in which all and only its elementary wffs occur as explicitly asserting their truth or falsehood.

This definition of Truth-logic is not acceptable for M-logic. For no branch or extension of M-logic can allow “<1>is true” to be applied to its own sentences or to have ‘It is true that …’ prefixed to its own sentences. In M-logic this would lead to a paradox as explained in Section 10.11.

7.13 The Status of ‘Truth’ in A-logic

A-logic differs from M-logic on the role and status of the predicate ‘<1>is true’ and the truth-operator, ‘It is true that…’.

In Analytic truth-logic “<1>is true” can be applied to the name of any sentence and “It is true that…” can be prefixed to any sentence or sentence schema in the object-language. The result may be true or false, but it does not lead to paradox. That is, there are no valid arguments of A-logic which yield a contradiction from a premiss that is consistent and asserted to be true.

In M-logic this is not permitted. M-logic requires a hierarchy of languages, and ‘<1>is true’ can only occur at the first or a higher levels above the object-language and at such levels it can only apply to sentences of lower levels. The base level of the object-langauge in M-logic cannot have ‘T’ in it at all. For according to M-logic, this leads to a logical paradox.

In translating M-logic into ordinary language, it is customary to speak of “it is false that P” as a sentential operator; ‘$\sim$P’ has been frequently interpreted as ‘It is false that P’. In ordinary usage “It is true that P” is just as frequently used as a sentential operator. But M-logic refuses to recognize this in its formal language by introducing ‘It is true that …’ as a sentential operator. As we said, its reason for this refusal is a technical one, due to paradoxes of its own making.

A logical paradox is a statement which is inconsistent, but follows by logically valid steps from premisses which are not logically inconsistent. Tarski showed how the introduction of ‘T’ into the object language of M-logic leads to a paradox. Given the concept of validity in M-logic, there are grammatically well-formed sentences which assert that a certain sentence is not true and lead by M-logic’s rules of valid inference to the conclusion the same sentence is both true and not true. To avoid this paradox, we must either (i) accept a logic with a self-contradictory theorem from which everything can follow according to M-logic (over-complete), or (ii) banish the truth-predicate from the language, or (iii) alter M-logic’s concept of a “valid argument”. Tarski chose option (ii). We choose (iii).

---

4. This is proved in Section 7.42122.

Because A-logic has different metalogical assumptions and rules than M-logic and defines "validity" differently the use of T-operators in the object language of A-logic does not lead to a logical paradox. Section 10.11 explains why Tarski's paradox is not a paradox in A-logic.

7.14 'Not-false' Differs From 'True'; 'Not-true' Differs From 'False'

Another basic difference between analytic-truth-logic and the semantics of M-logic is that the latter only allows that an indicative sentence may be True, or False. A-logic allows that an expression or sentence may be True, False or Neither. A better way to state it is: M-logic treats 'is false' and 'is not-true' as meaning the same, and 'is true' and 'is not-false' as meaning the same thing; it recognizes only two distinct, consistent properties. A-logic treats the predicates 'is True', 'is False', 'is not-True' and 'is not-False', as having four distinct meanings. It turns out that these four concepts can be reduced to three mutually exclusive, consistent alternatives for any linguistic expression: T, F, and 0 (for neither true nor false).

In M-logic since [P is false] is logically equivalent to [P is not-true], and [P is True] is logically equivalent to [P is not false], [~TP&~FP] means the same as [FP & TP] and [FP & TP] is inconsistent, i.e., no expression can be both true and false. This leaves only two *consistent* alternatives for an indicative sentence in M-logic; it is True or it is False exclusively.

But in A-logic, [P is false] does not mean the same thing as [P is not-true], though the former entails the latter. [P is True] does not mean the same thing as [P is not false], though the former entails the latter. Further, according to A-logic some linguistic entities, including some indicative sentences, are both not true and not false; expressions of the form (~TP&~FP) are not inconsistent. This leaves the three consistent, mutually exclusive, alternatives [TP&~FP], [FP&~TP] and [~TP&~FP], which is abbreviated as [0P]. The other three possibilities, [TP&~TP], [FP&~FP] and [TP&FP] are all inconsistent.

Whether a particular expression is, in fact, both not true and not false, is either true or false. It is true of unsaturated predicates, questions, commands, and some indicative sentences, but false of vast numbers of significant indicative sentences which are useful in human affairs.

The position taken in M-logic not only applies to a narrower class of linguistic expressions; it introduces inconsistency. For the first step in expounding M-logic to students is to state that the subject matter is propositions; propositions are expressed in sentences which are either true or false (never neither), as contrasted with questions, commands, predicates-without-subjects, single nouns or noun phrases, verbs and verb phrases, etc., all of which are neither true nor false. If "P is false" means the same as as "P is not true", and/or, "P is true" means the same as "P is not false", then proponents and teachers of M-logic have been presenting certain contradictions as truths. No matter what predicates P and Q may be, the concept that some thing, a_i, is neither P nor Q, is expressed in M-logic and in A-logic as [$\sim Pa_i$ & $\sim Qa_i$]. Thus a statement "a_i is neither true nor false" means 'a_i is not-true & $a_i$is not false', i.e., [$\sim Ta_i$ & $\sim Fa_i$]. But if 'true' means the same as 'not-false' then, substituting 'T' for '$\sim F$', we have the contradiction [$\sim Ta_i$ & Ta_i]. Thus if a_i is a question, command, unsaturated predicate, etc., to say a_i is neither true nor false, is to say a_i is both true and not true. This is a contradiction in both A-logic and M-logic. Hence, to avoid contradiction, we must agree that 'P is false' means something different than 'P is not true'.

Similarly, to say that 'P is false' means the same as 'P is not-true', makes any statement that a question, command, predicate, etc., is neither true nor false, into a statement of the form, '(Fa_i & $\sim Fa_i$)', which again is a contradiction. Hence, to avoid contradiction, we must reject the idea that 'P is true' means the same as 'P is not false'.

One of the three or four most important things to grasp, if one is to recognize A-logic as a viable alternative to M-logic, is the differences in the meanings of the four T-operators; especially the difference between '[P] is false' which is defined as '[∼P] is true' (i.e., FP Syn_{df} T∼P), and 'it is not true that [P]' which is symbolized '∼TP'. In analytic logic the predicate '<1> is false' entails '<1> is not true. But '<1> is not true' does not entail '<1> is false' since some statements which are not false may also not be true. Consequences of these differences are far-reaching.

Mathematical Logic Holds:	Analytic Logic Holds:
∼F(P) implies T(P)	∼F(P) does not imply or entail T(P)
∼T(P) implies F(P)	∼T(P) does not imply or entail F(P)
Inc(∼T(P) & ∼F(P))	(∼T(P) & ∼F(P)) is not inconsistent and can be true
Taut(T(P) v F(P)	(T(P) v F(P) is not tautologous and can be false.

7.15 Four Presuppositions of M-logic Rejected

There are four assumptions of Mathematical logic that A-logic rejects. All are closely interrelated and mutually support each other in M-logic. They are:

M-1) Every indicative sentence is either true or false, and none are neither true nor false. Though often called the "Law of Excluded Middle", we will call it the Principle of Bivalence—that there are just two truth-values, True and False.

M-2) No distinction is made between 'It is not the case that' and 'It is false that'. Indeed the negation sign is often read as 'it is false that'. I.e., in M-logic '∼P' entails 'FP' and 'FP' entails '∼P'.

M-3) Every indicative sentence P is equivalent to '[P] is true' or 'It is true that P'. (Call this "Tarski's Rule") I.e., 'P' entails 'TP' and 'TP' entails 'P' in M-logic.[5]

M-4) '[P] is false' is logically equivalent to '[P] is not true'.

If one accepts M-1), that all expressions studied in logic are either true or false and none are neither, and also M-2) that '[P] is false' means the same as '[∼P]' (M-2), then (M-3 follows: 'P' is equivalent to '[P] is true'.

If one accepts both M-3) and M-4), there is no point in prefixing T's or 'F's, for (P v ∼P) means the same as (TP v FP), and since the former is tautologous, the latter must be also (the former is, but the latter is not tautologous in A-logic). And so on.

These assumptions and their consequences are rejected in analytic logic. Collectively they create problems and block solutions to problems logicians want to solve. They also depart from ordinary language unnecessarily.

The alternatives, A-1 to A-3 below, are useful in solving these problems, and are marginally closer to ordinary usage, though ordinary usage varies so much no logical formulation can capture it exactly. The formal development of these alternatives in analytic logic is made possible by the introduction of the truth-operator, T, and the distinction between 1) negation and 2) it is not true that… and 3) it is false that… .

---

5. Tarski's "basic conception of truth" was that 'X is true' is equivalent to a sentence p, when X is a name of the sentence p. [Alfred Tarski, "The Semantic Conception of Truth", 1944, See Feigl & Sellars, *Readings in Philosophical Analysis*, p 55]

A-1) Every linguistic expression is either true, false, or neither-true-nor-false—this is the Principle of Trivalence for truth-logic. |= [TP v FP v 0P]

A-2) The following two concepts are distinct: "it is not the case that P"(symbol '~P') and "It is false that P" (in symbols 'FP'). Hence Not: '~P' Ent 'FP', Not: '~~P' Ent '~FP', Not: 'FP' Ent '~P'.

A-3) 'P' and 'It is true that P' have distinct meanings; they predicate different things of different subjects and neither entails the other. Not: 'P' Ent 'TP' ; Not: 'TP' Ent 'P' '; Not: '~P' Ent '~TP .

A-4) The following two concepts are formally distinct: "It is false that P" (in symbols 'F(P)') and "it is not-true that P" ('~T(P)').

In ordinary usage descriptive sentences which are true and are understood by the reader to be implicitly asserted as true (e.g., in newspapers and history books), vary rarely have the explicit form, 'S is true'. But it is simply not the case that in ordinary usage a sentence S *is* said to *mean* the same thing as '[S] is true', or that acceptance of S is the same as accepting S as true. To predicate '<1>is true' of S can say both more and less than what S by itself says. It is more if the sentence S is intended only as describing something for imaginative enjoyment; it asserts a correspondence with facts. 'T(S)' says less than 'S' if, among other things, S is a conditional statement (See Section 8.11).

7.16 *De Re* Entailment and *De Dicto* Implication

In the broad sense, logic includes not only the formal logic based on the meanings of syncategorematic terms, but also many special logics based on the meanings given to non-syncategorematic terms with substantive meaning. In A-logic '<1>is true' and 'It is true that...' are treated as substantive, non-logical terms.

In developing the logic of a substantive term we need a distinction between logical containment, which is based on the meanings of the "logical words" only, and entailments and implications which depend in addition on the meaning of one or more substantive terms. This distinction is needed whether the term is a sentential operator (like 'it is true that' or 'it is necessary that' or 'it ought to be that') or is an ordinary predicate like '<1>is a number' or '<1>equals<2>', or '<1>is a member of<2>' or 'the specific gravity of<1>is<2>', or '<1>is a father'. In logic broadly construed, synonymies and containments are not based only on logical structures with only syncategorematic words, but also on substantive meanings of truth-operators or predicates. In formulating a logic for extra-logical words we include axioms, e.g., Ax.7-01 [TP Syn (TP&~FP)], which are based on the meanings of words we wish to study logically. These axioms permit derivations based on the general meaning of 'logically contains', but the containment is not rooted in formal logic but in the meaning of the extra-logical term.

To recognize this difference we use 'CONT' and 'SYN' for cases of logical containment and synonymy based only on the meanings of the logical words and logical structure, and we use 'Ent' (for "logically entails") for cases of containment or sameness of meanings based on the meanings of extra-logical sentential operators or predicates. If [P Cont Q] is true, then either [P Ent Q] or [P CONT]. Containment in the broad sense is either due to "logical words and structure alone", or it is due in part, at least, to meanings given to extra-logical words. 'Truth' and 'Falsity' are extra-logical words. They depend, for their intended, correct application, on something more than the meanings of logical structures with syncategorematic words only.

The distinction between entailment and containment is illustrated by the truth-operator. Axiom 7-01 says [TP Syn (TP & ~FP)]; by the definition of 'F' this yields [TP Syn (TP & ~T~P)] from which, by definition of 'containment' we derive [TP Cont ~T~P]. If the occurrences of 'T' are eliminated, we have left 'P Cont ∼∼P' and of course by Ax.5, 'P Syn ∼∼P'. But TP is not Syn to ∼T∼P in A-logic. Something is added to the meaning by 'T' which is accounts for this difference. Again 'TP' has a special meaning, different from other operators. If 'T' (for 'it is true that') is replaced by 'M' (for 'it is possible that') the 'M'-for-'T' analogue of Ax.7-01 fails. MP does not contain ∼M∼P; i .e., "P is possible" does not contain "It is not possible that P is not the case" i.e., "P is necessary". Thus the meaning of [T(P)] is distinct from the meaning of [M(P)] in A- logic.

Neither the synonymy nor the containment from Axiom 7-01, are formal SYN- and CONT-theorems. They can not be derived from the meanings of the syncategorematic "logical words". The expression '∼T∼P' is not logically synonymous with 'T(P)' though the latter contains the former. The difference is based on the substantive meaning of 'It is true that'.

All theorems and rules of A-logic are *de dicto* statements. The relations of 'Syn' and 'Cont' are relations between meanings of linguistic expressions. The property of being inconsistent is a property that belongs to the meanings of sentences or predicates, and to the ideas they express.

But among the logical structures and theorems and principles of logic, it is important to distinguish between those theorems which (i) are only *de dicto* as far as logic is concerned, and those that (ii) can also be considered *de re*.

Inc-theorems and TAUT-theorems are only *de dicto* since inconsistency is excluded from being a property of the world or of objective reality. Tautology is a property only of linguistic expressions by virtue of being negations of inconsistencies. The concept of an objective field of reference is of a domain in which no inconsistencies exist, and thus a tautology—the denial of an inconsistency—can not convey any information about states of affairs in an objective field of reference.

In contrast Cont-theorems, if consistent, are capable of mirroring, or corresponding, to objective facts. Theorems and principles of Entailments and Containment can be viewd as *de re* statements, provided the components are not inconsistent. The paradigm for Containment- theorems is "Simplification": [(P&Q) Cont Q]. Every Cont-theorem is synonymous with a theorem in which the consequent is a conjunct in a conjunctive normal form of the antecedent. In truth-logic, '[T(P&Q) Cont TQ]' says the idea that [P&Q] is true contains the idea that [Q] is true; i.e., the meaning of 'T(P&Q)' contains the meaning of 'TP'. Unless (P&Q) is inconsistent, this amounts to a logically unobjectionable assertion. There is a correspondence between the logical *containment* of the meaning of [P] *in* the meaning of [P&Q], and the actual containment in the objective field of reference, of the fact described by P in the more complex fact described by [P&Q]. The logical structure of language may correspond to the structure of facts in an objective field of reference. The concept of an objective field of reference is the concept of a conjunction of many facts great and small—the whole truth about the field would contain all of them. Particular truths describe simple or complex particular facts. Quantified truth-assertions which are not inconsistent may be viewed as *de re* descriptions of some predicate's applying to all or some entities in the field.

Thus Containment-theorems in truth-logic and truth-assertions which are POS and not inconsistent, may be treated as not merely *de dicto*, but also *de re*.

The greatest pragmatic value of truth-logic lies in its applicability to *de re* problems about non-linguistic facts and events. Many solely *de dicto* theorems and principles are pragmatically useful as means to that end. But it is pragmatically very important to distinguish logical statements which should be considered *de dicto* only from those that can also be considered *de re*. Mistakes in reasoning take place when the two are confused (see Section 10.2).

As defined in A-logic, Implications (i.e., 'A-implications') are not entailments; by definition, the antecedent by itself does not logically contain the consequent (though it is an ellipsis for another containment statement). To distinguish 'implication' as defined in A-logic, from the very different concept of 'implication' in M-logic, we use the terms 'A-implication' and 'M-implication' when needed.

The "Principle of Addition" is an A-implication, not an entailment. Although 'TP' does not contain 'T(PvQ)', (since neither Q nor TQ is a conjunct of the normal form of 'TP'), if P is true, this implies (due to the meanings of 'true', and 'either or') that (P or Q) is true. It is a *de dicto* principle which does not make sense as *de re* principle. The permissibility of adding the disjunct 'Q', which may be replaced by any sentence or expression in our lexicon, can not be justified by reference to any facts in an objective field of reference, for 'Q' can just as easily be false as true, and the implication holds regardless of whether 'Q' or '~Q' or any other expression is disjoined with P. Implications have a significant role to play in logical investigations, but it has to do with definitions and linguistic conventions—i.e., with what we *mean*—rather than properties and relations of entities in a objective field of non-linguistics facts and events.

Though A-implications are not containments, they are based on containments. To prove that TP implies T(PvQ) one must prove that the premiss that P is true with the presupposition that (PvQ) is either true, false or neither, logically contains or entails that (PvQ) must be true. Thus the A-logic's requirement of containment in logical inference, is implicitly retained in implications.

The distinction between entailments and implications in A-logic, and the differences and similarities between these concepts in A-logic and M-logic is developed in detail in Section 7.423.

7.2 A Correspondence Theory or Truth

A theory of truth must clarify the notions of truth and non-truth, making clear the relationships of truth, negation and falsehood. In effect it defines meanings for the predicates '$<1>$is true' and '$<1>$is false' and their negations. The theory presented here differs from the theory which operates in M-logic because '$<1>$is false' does not mean the same thing as '$<1>$is not true'.

In the sense used here these predicates apply truthfully or falsely only to certain types of entities. In ordinary discourse we say that certain beliefs, or ideas are true or not true, or that certain statements which express these ideas are true or not. However here we confine the discussion to the truth and falsity of sentences, taking '$<1>$is a sentence' to entail not only that $<1>$ is a certain kind of string of physical words (written or spoken), but also that $<1>$is meaningful—that its syntax is grammatical and its component parts convey meanings or ideas which are at least roughly describable (even if as a whole it may not be coherent[6]). Moreover, in the present discussion 'sentence' means 'descriptive sentence', for it is only descriptive sentences (indicative or subjunctive) that can be true or false. The logic of other kinds of sentences may be discussed elsewhere.

Formal logic tries to establish an isomorphism between properties and relationships of structures of compound ideas or meanings, and the properties and relationships of the physical structures of linguistic signs for predicates and sentences.

7.21 General Theory of Truth

We have rejected the view that P means the same as 'P is true'. Thus the question is, What does the predicate '[P] is true' mean over and above the meaning of '[P]?'.

Roughly, the answer is that the idea expressed in '$[P<a_1,\ldots,a_n>]$ is true' contains the idea that there is an objective field of reference, such that meaning of $[P<a_1,\ldots,a_n>]$ corresponds to, or correctly describes, a state of affairs or fact about the designata of the subject terms 'a_1',...,'a_n' in that field

6. E.g., Russell's 'Quadruplicity drinks procrastination'. Neither this sentence nor its negation can be said to be true or false though it is grammatically well-formed and all components have recognizable meanings.

of reference. The idea of an **objective** field of reference is that of a domain of individual entities which have properties and relationships which are what they are independently of what we might wish, imagine, or say about them, and is absolutely free of inconsistency.

We first define the predicates T, F, and ~T and ~F for elementary sentences of the forms '$P<c_1,\ldots,c_n>$' and '$\sim P<c_1,\ldots,c_n>$'. Then we deal with these predicates as they apply to conjunctions, disjunctions, quantified expressions and their negations. This explication says that both 'S is true' and 'S is false' presuppose

(I) that S is a sentence with meaning,
(ii) that there is an objective field of reference which is independent of S or its meaning,
(iii) that singular terms in S stand for or refer to a unique individuals in that field of reference,
(iv) that the predicate of S is relevant in that field of reference—i.e., the predicate as a whole, and all elementary predicates of which it is composed are such that either that predicate, or some contrary of that predicate could apply to or describe properties or relationships which might obtain of the designated individual(s) in the field of reference, and finally,
(v) that the predicates actually used do so apply.

More formally, we begin with the definition of truth for atomic wffs:

1) '$<1>$ is true' syn_{df} (i) $<1>$ is a declarative sentence $P<c_1,\ldots,c_n>$ with meaning,
and (ii) there is a field of reference, R, which is independent of $<1>$ or its meaning,
and (iii) each c_i in '$<c_1,\ldots,c_n>$' stands for or refers to a unique individual $o_i \in R$, so that '$c_1,\ldots,c_n \in R$' would be true,
and (iv) P is relevant in that field of reference—i.e., the meaning of P, or some contrary of P, could apply to or describe correctly properties or relationships of one or more individuals in R.
and (v) The meaning of '$(c_1,..,c_n) \in R$ & $P<c_1,\ldots,c_n>$)' correctly applies to or describes properties or relationships of the referents of $<c_1,\ldots,c_n>$ in R.

With the introduction of negation we get the general rules:

2) For any wff P and any field of reference, R,
If T(P) then F(~P) and ~T(~P).
If F(P) then T(~P) and ~F(~P).
If 0(P) then 0(~P) and ~T(P) and ~F(P).

From this it follows that the definition of '$<1>$ is false' is similar to that of '$<1>$ is true' in all respects except that 'P' is replaced throughout by '~P'. Thus,

'[P] is false' syn '[~P] is true' [FP Syn T~P]
whence, by U-SUB we get,
'[~P] is false' syn '[~~P] is true' [F~P Syn T~~P]
whence, by double negation, we get,
'[~P] is false' syn '[P] is true'. [F~P Syn TP]

The concept of falsehood, on this definition, is the concept of a kind of truth—the truth of a negated positive sentence. Both truth and falsehood are properties of positive sentences, not of negative ones. Statements of the form 'T(P)' and 'F(P)' are always POS, never NEG. For, by the definitions above,

a) '$<1>$is true' applies to descriptive sentences, which must therefore have individual constants (or variables) in all argument positions,
b) '$<1>$ is true' *means* implicitly, by clause (iii), that the designata of its singular terms are members of the intended field of reference. (That this is part of its *meaning* does not entail that this is true).
c) by clause (v) a sentence,'[$\sim P<c_1,\ldots,c_n>$]is true' implicitly contains conjuncts which assert such membership: $T\sim(P<c_1,\ldots,c_n>)$' entails '$(c_1 \in R \ \&\ldots\&\ c_n \in R\ \&\ \sim P<c_1,\ldots,c_n>)$'.
d) sentences of the form '$(c_1 \in R \ \&\ldots\&\ c_n \in R\ \&\ Q)$' are POS.

Since '[P]is false' syn '[$\sim$P]is true' assertions of falsehod are also POS. The predicate '$<1>$is false' is always POS, since by definition, it is correctly used only when '1' is replaced by the name of a sentence, and that sentence contains (implicitly or explicitly) at least one POS conjunct, namely, the presupposition that each entity referred to in the sentence exists in the domain of reference, as stated in clause (v).

It follows from these definitions that '[P]is false' and '[P]is true' are contraries, not contradictories. More precisely, since $F(P<c_1,\ldots,c_n>)$ Syn $T(\sim P<c_1,\ldots,c_n>)$ and,

$T(P<c_1,\ldots,c_n>)$ Syn $T((c_1,..,c_n)\in R\ \&\ P<c_1,\ldots,c_n>))$

$T(\sim P<c_1,\ldots,c_n>)$ Syn $T(c_1,..,c_n)\in R\ \&\ \sim P<c_1,\ldots,c_n>))$,

the two right-hand wffs have the forms (Q&P) and (Q&$\sim$P) which are contraries, not contradictories. Contraries can not be true together, but can be false together.

It follows that it is not logically possible for [TP&FP] to be true on these definitions. Since T(P) and F(P) are contraries, [TP & FP] is inconsistent and thus by logic, not-true. This conclusion can be expressed in the law $\models\ \sim$T[TP & FP], which says "By logic ($\models$) any expression of the form '(TP & FP)' is not true". It is based on the logic of the idea of "truth", which, again, involves correspondence with facts in some *objective* field of reference which, by definition, contains no inconsistencies .

It does not follow from these definitions that a sentence can not be neither true nor false. The Law of Bivalence, misleadingly called "excluded Middle"—that an indicative sentence must be either true or false and can't be neither—is not derivable from these definitions.

Sentences of the forms $\sim$T(P) and $\sim$F(P), being negations of POS wffs, are always NEG.

How does this relate to the definitions above? What does '$<1>$is not true', symbolized by '$\sim TP_i$', mean? By this predicate A-logic rejects negation of the POS predicate 'TP_i', which is synonymous with the negation of the definiens of 'TP_i'. Since that definiens consists of a conjunction of five clauses, its negation is equivalent to denying that conjunction as a whole or, by DeMorgan laws, to a disjunction of denials of each conjunct. In the latter form we get,

'$\sim(<1>$is true)' Syn_{df}

(i) Not ($<1>$is an indicative sentence $P<c_1,\ldots,c_n>$ with its meaning,

or (ii) Not (there is a field of reference, R, which is independent of $<1>$ or its meaning)

or (iii) Not (each c_i in '$<c_1,\ldots,c_n>$' stands for or refers to a unique individual o_i e R, so that $c_1,\ldots,c_n$ e R is true) or (iv) Not (P is relevant in that field of reference)—i.e., the meaning of P or some contrary of P does not applies to or describes correctly the properties or relationships of one or more individuals in R)

or (v) Not (The meaning of '$(c_1,..,c)eR$ & $P<c_1,\ldots,c_n>$)' correctly applies to or describes properties or relationships of the referents of $<c_1,\ldots,c>$ in R)).

The negated predicate "$<1>$is not false", symbolized by '$\sim$FP', is defined by a similar disjunction, except that '$\sim$P' replaces 'P' throughout since '$\sim$FP' is synonymous with '$\sim$T$\sim$P'.

The negation of the clause (i) permits both 'not-true' and 'not-false' to apply to entities which are not sentences (words, predicates, phrases, dogs and houses), or are not indicative sentences (questions, commands, etc.), or are sentences syntactically OK but without meaning. By clauses (ii) and (v) indicative sentences which are uttered without any claim to describe an objective field of reference (jokes, fiction, poetry) will not be taken as truth-claims or falsehood claims. By clauses (iii) and (iv) both '$1>$is not true' and '$<1>$is not false' apply to simple indicative sentences with singular terms that do not refer to any entities in the field of reference, or simple sentences with predicates such that neither they nor their contraries ever apply to the kind of entities of which they are predicated.

In summary, the concepts of truth and falsehood both presuppose reference to an objective field of reference. The concept of an objective field of reference is the concept of a domain in which there are no contradictions or inconsistencies. The definitions of truth and falsehood entail that if an expression is inconsistent it can not describe a fact in an objective field of reference. If a sentence is inconsistent it can be meaningful, but it can not be true. This is expressed in the rule R7-2. If Inc(P) then $\sim$T(P).

From similar considerations it follows that if P is false, it is not true and if P is true it is not false. I.e., "If T(P) then $\sim$F(P)" and "If F(P) then $\sim$T(P)" will be valid principles of inference, but their converses will not. Since TP and FP can't be true together, if one is true the other is not true. If F(P) is true, then clause (v) of the definition of 'T(P)' fails and the denial of that clause, which is clause (v) of the the definition of '$\sim$T(P)' is satisfied, thus satisfying the disjunction which is the definiens of '$\sim$T(P)'. A similar argument holds for 'If TP then $\sim$FP'.

And of course 'TP' and '$\sim$TP', as well as 'FP' and '$\sim$FP' are straight out contradictories. They can be neither true together, nor false together.

The rest of the semantic theory for truth-logic, which gives the truth conditions for negations, conjunctions and, derivatively, for any quantified wffs and disjunctions of M-logic wffs is as follows:

2) For any wff P and any field of reference, R,
   If T(P) then F($\sim$P) and $\sim$T($\sim$P).
   If F(P) then T($\sim$P) and $\sim$F($\sim$P).
   If 0(P) then 0($\sim$P) and F(T(P)) and F(F(P).
3) For any wffs P and Q and any field of reference R,
   If T(P) & T(Q), then T(P&Q)
   If (F(P) v F(Q), then F(P&Q)
   Otherwise, 0(P&Q) (i.e, $\sim$T(P&Q) & $\sim$F(P&Q))
4) For any predicate P, any individual variable x_i, and any field of reference R,
   T($\forall x_i$)Px_i iff for every a_i, a_i e R, T(Pa_i)
   F($\forall x_i$)Px_i iff for some a_i, a_i e R, F(Pa_i)
   Otherwise, 0($\forall x_i$)Px_i i.e, ($\sim$T($\forall x_i$)Px_i & $\sim$F($\forall x_i$)Px_i).

5) If T(inc(P)) then ~T(P)
"If any result of putting a proposition for 'P' in '(P)' is inconsistent,
then it is true by logic that that result is not true."[7]

The last principle, 5), in the semantics above does not define 'logical truth' or 'validity' as M-logic does in terms of universal truth, or truth in all possible worlds. M-logic and A-logic have quite different concepts of both 'logical truth' and of 'validity'.[8]

A word about "existential" quantifiers, "truth", and ontological commitment: The practice of calling the disjunctive quantifier the "existential quantifier" inserts a metaphysical bias into the language of logic. It probably came from the assumption (which we reject) that every inquiry in philosophy or science is a search for truth, together with the assumption that every sentence is either true or false exclusively. Apparently it was felt that "Some x is F", meant "There exists an x which is F" which seems close to "It is true that some x if F". Thus mathematical logicians tend, in their philosophical moments, to attach ontological commitment to the existential operator. Quine wrote,

> …the notion of ontological commitment belongs to the theory of reference. For to say that a given existential quantification presupposes objects of a given kind is to say simply that the open sentence which follows the quantifier is true of some objects of that kind and none not of that kind.[9]

But it is the concept of truth which involves ontological commitment, not the disjunctive (miscalled "existential") quantifier.

To say some sentence or idea is true, is to posit or presuppose the existence of a field of reference in which facts are as they are regardless of the speaker. It presupposes objective states of affairs composed of objects which exist in that field and have certain properties and stand in relationships which are independent of what the thinker may think or wish.

To say "Once upon a time there was a witch…" and "Some witches are wicked and some witches are good"' do not mean the same things as "It is true that once upon a time something was a witch" and

---

7. In the customary format of M-logic's semantics, 2) to 4) could be written as follows with 'V… =—' standing for 'The truth-value of … equals—", and "any field of reference, R" could be read as "any possible world in the set of possible worlds".

2) For any wff P and any field of reference, R,
If $V<P,R> = T$, then $V<{\sim}P,R> = F$ and $V<{\sim}P,R> = {\sim}T$
If $V<P,R> = F$, then $V<{\sim}P,R> = T$ and $V<{\sim}P,R> = {\sim}F$
If $V<P,R> = 0$, then $V<{\sim}P,R> = 0$ and $V<TP,R> = F$ and $V<FP,R> = F$

3) For any wffs P and Q and any field of reference R,
If $V<P,R> = T$ & $V<Q,R> = T$, then $V<(P\&Q),R> = T$
If $V<P,R> = F$ v $V<Q,R> = F$, then $V<(P\&Q),R> = F$
Otherwise, $V<(P\&Q),R> = 0$

4) For any predicate P, any individual variable vi and any field of reference R,
$V<(\forall xi)Px_i, R> = T$ iff for every a_i, a_i e R, $V<Pa_i,R> = T$
$V<(\forall xi)Px_i, R> = F$ iff for some a_i, a_i e R, $V<Pa_i,R> = F$
Otherwise, $V<(\forall x_i)Px_i, R> = 0$.

8. In Chapter 8 with the re-introduction of C-conditionals, the semantics is completed to include a sharp distinction between the truth (logical or factual) of universal generalizations and the validity of arguments or of particular or generalized C-conditionals. This distinction is essential to solving the problems of confirmation and of counterfactuals.

9. W.V. O. Quine, *From a Logical Point of View*, Harper, 1961, pp 130-1

"There exist in the actual world witches which are wicked and witches which are good". People who believe there really are witches can indeed utter the latter sentences to express their beliefs though many people will disagree with them. But even those who do not believe that any witches exist in actuality can find the first two sentences meaningful and useful if only to paint a picture for the imagination to enjoy (as in *The Wizard of Oz*). However, the second two sentences mean that the pictures described in the first corresponds to some objective reality such as the actual world of common sense (though not necessarily as in *The Wizared of Oz*). It is the words "It is true that.." which adds ontological commitment, not the word "something".

"There is a witch", or "Some thing is a witch" with the logical form '(∃x)Wx', paints a picture for the imagination. The statement "**It is true that** there is a witch" with the logical form'**T**[(∃x)Wx]', adds the concept of an objective field of reference and asserts that the picture expressed by 'There is a witch' describes a real state of affairs in that field. Without the 'T' we have a field of reference but not one that is necessarily either dependent on, or independent of, our thoughts. No relation of correspondence is asserted. It is neutral with respect to ontological commitment and objectivity. Reading or writing a story we create a field of reference and populate it with individuals that have properties and relationships with others. We may read about this field for fun. But there is no suggestion that we must either accept or reject it as corresponding to some other objective reality.

Asserting "some x is F" simply says that among the things we are talking about at least one is such that the predicate 'F<1>' applies to it. To say "Some x is F and some x is not F and some x is G" is to indicate a variety of properties attributed to things being talked about. These are distinct operations, separate from asserting that what we are saying corresponds to facts in an objective field of reference.

7.22 Semantic Ascent[10]

The theory of truth presented here provides for three types, or "levels", of wffs which can be distinguished in truth-logic. Questions about the relation of A-logic to M-logic, including questions of completeness or inclusion of one relative to the other, must take into account the differences beween these levels.

A **zero-level T-wff** is any wff without any occurrences of a T-operator in it. All wffs of which 'SYN' or 'CONT' or 'INC' or 'TAUT' or 'VALID' are predicated in Chapters 1 through 6, are zero-level wffs.

A **first-level T-wff** is any wff in which all elementary wffs occur in the scope of just one T-operator. First-level T-wffs can come in several formally distinct varieties including,

(i) T-wffs in which a single T-operator is prefixed to an elementary or compound zero-level T-wff;
Examples: 'T~P', 'T~(Pa & ~Pa)', 'T(P v (~Q & R))', 'T(∃x)~(Px & ~Qx)'

(ii) T-wffs in which 'T' is prefixed only to atomic wffs;
Examples: '~TP', '~(TPa & ~TPa)', '(TP v (~TQ & TR))', '(∃x)~(TPx & ~TQx)'

10. The term 'semantic ascent' is Quine's; see W. V. O. Quine, *Word and Object*, 1960, Section 56,p 271. Quine uses this term for the move from an object language from which the truth-predicate is excluded to a metalanguage in which the truth-predicate is used to talk about expressions in the object languagee.

Neither of these classes are reducible to the other and a third class includes both of them. The truth-operator '0' requires both of them since '0P' Syn_{df} '(~ TP & ~ T ~ P)'.

(iii) Compound T-wffs with 'T' prefixed both the atomic wffs and to compound: wffs
Examples: '(~ TP & T ~ P)', ' ~ (TPa & T ~ Pa)', '(TP v T(~ Q & R))', '(∃x) ~ (TPx & ~ Tpx)'

Unsatisfiability- and Unfalsifiability-Theorems can be expressed with first-level T-wffs, such as
|= ~ T[P& ~ P] and |= ~ F[~ PvP], but theorems of Logical Truth and Logical Falsehood can not.
'|= ~ T...' means "by logic alone, it is not-true that ...", i.e., '...' is logically unsatisfiable.

Finally, **a second-level T-wff,** is a wff which has a first or higher level T-wff lying in the scope of one or more T-operators. The Logical Truth of tautologous T-wffs and the logical Falsehood of inconsistent T-wffs can only be expressed in second level T-wffs such as

|= T[~ TP v TP], |= F[TP & FP], and |= T[TP v FP v 0P] and |= F[~ TP & ~ FP & ~ 0P].

In any search for truth, the focus is on the way things are in an objective field of reference. **Zero-leve**l indicative sentences may or may not be intended to be taken as true statements. We think of them as providing descriptive accounts of various non-linguistic entities, with no explicit ontological commitments. They may be descriptions of imaginary things or states of affairs as in fairy stories and novels or jokes, or they may be intended, without explicitly saying so, to be taken as true descriptions of facts in some objective field of reference. If they are so intended, then the intention is made clear and explicit by prefixing truth-operators, i.e., by first-level T-expressions.

First-level T-expressions are explicitly intended to be taken as true statements and as conveying information about an objective field of reference. That they are so intended does not entail that they are true in fact. In asserting the truth of a zero-level sentence, the truth-operator T refers to and says something about both language (the meanings of the sentence it is prefixed to) and reality (the state of affairs in the objective field of reference it presupposes). It asserts that the ideas conveyed by the predicate in the zero-level sentence corresponds to the real properties or relationships of the entities, denoted by the subject terms, in an objective field of reference. Conversely it says that reality in that field corresponds to the ideas expressed in the zero-level sentence. Though a truth-claim may not be true, its reference to an objective field of reference is not thereby altered. First-level T-wffs show the form of explicit *de re* discourse. .

In contrast, **Second- and higher-level T-expressions** talk about language only. A second-level truth-assertion is talking about the truth-operators in the first-level expression. 2nd- level expressions have the form of 2nd-level T-wffs, and are strictly *de dicto* statements. They can add no new information about non-linguistic entities, and consequently, as instruments in the search for truth, they are all eliminable in the sense of being reducible to referentially synonymous 1st-level expressions.

These levels are related to what Quine called "Semantic Ascent". But Quine, following Tarski, conflates our zero-level T-wffs with first-level wffs, holding that [P] and ['P'is true] are equivalent**,** but he does not allow the truth-predicate to occur in the object language (in order to avoid the liar paradox). He "resorts to semantic ascent" introducing the truth-predicate in order to ascribe Logical Truth or Logical Falsehood saying "We need it to restore the effect of objective reference when for the sake of some generalization we have resorted to semantic ascent."[11] His ascent from his zero-level to his first level T-wffs, is replaced in A-logic by the distinction between zero-level and first-level, and then an "ascent"

11. Quine, W. V. , *Philosophy of Logic*, pp 12-13.

from first-level to second-level T-wffs before we reach logical truths and falsehoods. We agree that 'truth' is necessary to introduce the concept of objective reference, but hold that 0-level wffs are not equivalent to truth-assertions. On our view generalization is not the purpose of semantic ascent, and Logical Truth does not entail the *de re* universality that Quine and proponents of M-logic seem to claim for it; it becomes a mundane, dispensable, *de dicto* property.

7.23 'Truth', 'Logical Truth' and 'Validity' in M-logic and A-logic

The relation between A-logic and this theory of truth is very different from the relation between M-logic and it's account of "truth".

First, the theory above is a correspondence theory of truth in the very traditional sense that presupposes meanings or ideas that correspond, or not, to objective facts. Tarski suggested that his "Semantic Conception of Truth" was a "correspondence theory"[12] But Sellars called it the 'disappearance theory of truth' and Quines calls his version of Tarski's theory a "disquotational theory of truth" in opposition to the traditional correspondence theory. Quine rejects the concept that meanings or ideas intervene between the physical signs and the facts they report. He holds, with Tarski, that ''Snow is white' is true' and 'Snow is white' are equivalent since they are made true by the same fact. Therefore, he says we can eliminate expressions predicating truth of a sentence in quotes and just use that sentence without quotes to simply state facts in the world.[13] Tarski's and Quine's theories of truth are a consequence of presuppositions of M-logic. We reject those presuppositions because they 1) treat non-sequiturs as valid and 2) render logic incapable of doing other jobs logic is expected do, and 3) discard the very useful, common sense notion of truth whereby the meaning or idea conveyed by a sentence is true or false if it corresponds to the reality it purports to describe.

Secondly, the concept of "logical truth" as used in M-logic has a special meaning which differs from what A-logic would call "a truth of logic". The phrase "a logical truth" might be thought to mean the same as "a truth of logic". Without laying claim to any distinction recognized in ordinary discourse, the differences in the two concepts may nevertheless be clarified as follows:

In A-logic, when a statement is said be a "truth of logic", it means a) it is statement which belongs to the discipline of logic, applying the predicates which are the special province of logic to the entities that logic talks about, and b) it is true of facts in the objective field of reference which logic deals with.

We would expect "a truth of zoology" to be expressed in a statement which had as its subject terms, words which designated some kind of animal or parts of some kinds of animals, and we would expect the predicate to be a predicate describing properties or relationships which animals or parts of animals might have. Among all possible statements of this sort, the 'truths of zoology" would be members of a sub-set which correspond to facts in the actual world. Similarly for 'truth of chemistry", "truth of physics", etc. A "truth of history" (though normally called "an historical truth") we would expect to be a statement which had subject terms standing for entities that might exist in the actual world, and predicates saying that those entities were in such and such a state, or did such and such a thing, at some specified place during some specified time. Truths of history are distinguished from fiction, as documentary films are distinguished from non-historical fictional movies.

In logic the kinds of things we talk about are ideas—the ideas or meanings associated with words and phrases—and what we predicate of them are certain kinds of relations and properties which are the special business of logic. The predicates 'is valid' and 'is inconsistent' are the most important predicates. These are both based on the more basic relationship conveyed by the predicates "logically con-

12. P. 54, Feigl and Sellars
13. Cf. Quine, W.V. , *Philosophy of Logic*, pp 12, 97

tains" and "is logically synonymous with". The truths of logic are the statements which correctly say that the ideas conveyed by certain expressions and certain forms of expressions have the logical relationships or properties predicated of them.

There are truths and falsehoods of M-logic in this sense of a "truth of logic". Given the definition of 'valid' in M-logic, "[(P&∼P) ⊃ Q] is a valid wff'" is a truth of M-logic, and "[∼(P ⊃ ∼P)]' is not a valid wff'" is a truth of M-logic. So "[∼(P ⊃ ∼P)]' is a valid wff'" is a falsehood of M-logic.

But in M-logic "logical truth" is different than this sense of "a truth of logic". According to Quine, in M-logic "A logically true (or false) sentence is a sentence whose truth is assured (or precluded) by its logical structure."[14] The second reference to truth in this definition seems to mean factual truth. Thus "Either Zeus died or Zeus did not die" is a logical truth, because a) it is factually true and b) it is factually true because of its logical structure.[15] Quine says elsewhere,

> Logical truths are…not only true, but stay true even when we make substitutions upon their component words and phrases as we please, provided merely that the so-called logical words" '=', 'or', 'not', 'if-then', 'everything', 'something', etc., stay undisturbed.[16]

"Thus it is that logical truths are commonly said to be true by virtue of the meanings of the logical words," Quine continues. But this does not make clear how logic can make a statement factually true. Factual truth is determined by facts independent of the language we use. A-logic can agree that the truths of logic are *de dicto* truths determined by the meanings of logical words. But it does not follow that they are *de re* truths about the actual world or any other objective non-linguistic field of reference. That they are is a metaphysical proposition inessential to logic. Everyone agrees that logical truths give no information about the world. We can agree that they cannot be false. But in what sense could they describe the world truthfully? Truths of formal logic are easiest viewed as truths about ideas associated with language.[17]

Our real interest in the logic of truth-assertions is directed at contingent statements—statements which are neither logically true nor logically false. Which of these are, or might be, true and which ones false? Which ones follow from which ones? Which ones can't be true, if another is known to be true? All atomic statements are contingent. The greater truths are based on conjunctions of true atomic statements.

Thirdly, A-logic and M-logic have different concepts of validity. In A-logic, logical validity is defined independently of any theory or truth, whereas in the semantic theory of M-logic the 'validity' of a wff is defined in terms of the truth of all its substitution instances. To avoid ambiguity, we call validity in A-logic 'A-validity' when contrasting it with M-validity (validity in M-logic).

Quine defined 'validity' for M-logic in terms of truth, as follows:

> A truth-functional schema is called *valid* if it comes out true under all interpretations of its letters."[18]
> "The definition of validity [for quantificational schemata] is as before: truth under all interpretations of its variables in all non-empty universes.[19]

---

14. Quine, W.V. , *Philosophy of Logic*, 1970, p 48
15. In contrast, A-logic would say " 'Either Zeus died or Zeus did not die' is tautologous" is a truth of logic. This is not the same as saying "Either Zeus died or Zeus did not die" is a logical truth.
16. *Methods of Logic*, 4th Ed., Harvard University Press, 1982, p. 4
17. "I shall argue against the doctrine that the logical truths are true because of grammar, or because of language." W.V.O Quine, *Philosophy of Logic*, 1970, page ix, Here we differ with Quine .
18. Ibid. p 40.
19. Ibid. p 172.

Using "valid in M-logic" as just defined, wffs and statements which are M-valid all reduce to TF-equivalent disjunctions, or conjunction of disjunctions. Given two statements S_1 and S_2, S_2 is said to "follow logically" from, S_1, and an inference from S_1 to S_2 to be 'M-valid', if there is no interpretation that would make S_1 true and S_2 false.[20] This definition makes everything "follow from" a contradiction and every logical truth "follow from" any sentence whatever.

In A-logic, statements or inferences are valid if and only the component expressions (antecedent and consequent of a C-conditional, or premisses and conclusion of an argument) are such that the first logically contains the second and the conjunction of the components is not inconsistent.[21] A-validity is independent of the truth of its component parts, and even in truth-logic A-valid inferences are not normally said to be **true**. It may be true that a statement or argument is valid, but the statement that is valid, is not necessarily true. The statements and wffs that are A-valid are all C-conditionals or C-biconditionals, as laid out in Chapters 6 and 8.

No disjunctions of conjunctions of M-logic (hence, no statements or wffs of M-logic) are A-valid, because they are not C-conditionals.

C-conditional wffs which are A-valid are not A-valid in M-logic because there are no C-conditionals in M-logic. But certain *inferences* between 1st-level T-wffs of the form [P, **.:** Q] or [P therefore Q] can be proven to be A-valid using only the wffs of this chapter. For if P and Q are lst level T-wffs, by definition of 'Valid', '[P, **.:** Q] is A-valid' Syn '[|= (P Cont Q) & not-Inc(P & Q)]'.

Although A-validity is defined independently of truth**,** in truth-logic A-valid inferences take on some features, due to the meaning of 'true', which differ from validity in M-logic or other branches of A-logic. For example, since |= [TP Cont ~FP], and '(TP & ~FP)' is not Inconsistent, the inference [T(P) therefore, ~F(P)] is A-valid.

That no interpretation that would make S_1 true and S_2 false is a necessary, but not a sufficient, condition of A-validity in truth-logic. Although many *inferences* which are M-valid are also A-valid, many other M-valid inferences are not A-valid because either the containment or consistency requirement, or both, are not satisfied. If [P **.:** Q] is A-valid, then Inc[TP & FQ], and if Inc[TP &FQ] then [TP **.:** FQ] is not A-valid. Hence, If [P **.:** Q] is A-valid then [TP **.:** FQ] is not A-valid.

Returning to the concept of "truths of logic" which is espoused in A-logic, these truths of A-logic are presented as factual truths. They are true statements about a kind of reality, but not the kind of non-lingustic reality which is generally meant by *"de re* propositions".

What is the nature of the objective field of reference, and the facts, that make a statement of A-logic—one that predicates a logical property of some expression—true or false? The two statements 'INC[P&~P]' and 'INC[P&Q]' are statements that belong to logic. Where do we find the facts and the objective field of reference, that makes the first one true and the second one false?

Formal logic is designed to make it possible for a person who does not understand meanings of English words, or a machine, to start with a set of meaningful English sentences, then by manipulating the symbols in the sentences according to the rules for symbolic transformation set down in formal logic, arrive at another set of symbols which is a very different meaningful English sentence which follows logically from the first sentences. What makes formal logic rigorous is just this capability.

This does not mean that logic is merely a method of manipulating symbols. There are many systems for manipulating symbols, and there are many different meanings that can be associated with given symbols. Whether a set of symbols, with rules for their manipulation, is properly called a formal

20. Quine, *Methods of Logic*, (4th Edit). Section 7, Implication.
21. Cf. Ch. 5: Valid[A, therefore B] Syndf [A Cont B & not-Inc(A&B)].
Ch. 6: VALID[P => Q] Syndf [A Cont B & not-Inc(A&B)].

logic, depends on what meanings or ideas are associated with the symbols. In Section 5.5 we saw that if we interchange the meanings, or ideas, associated with '&' and 'v' while leaving purely symbolic system intact, we change the theorems from statements which can't be false (tautologies) to inconsistencies which can't be true. If we change the meaning of 'T' in analytic truth-logic from "is true" to "is inconsistent" then the various axioms and rules of inference become false or invalid. "T(P&∼P) Cont TP" for example, would mean "[P and ∼P] is consistent" contains "P is inconsistent", which is clearly false on our ordinary understanding, and our definition of 'inconsistent' and 'contains'.

Thus whether a formal logical system is acceptable or not depends on what meanings or ideas we associate with which symbols, and whether the properties and relations of these ideas are associated with properties and relations of the symbols in ways which insure that the results of manipulating the symbols will parallel results which we get, or want to get, when we reason logically. It is a set of properties and relations of ideas (most of which we know before we ever hear about logic) regardless of our native language, which is the objective field of reference by which we judge whether a theorem, or principle of inference, or a whole system of formal logic is good, or is true of that field of reference. The ideas expressed in English by 'Both…and—', 'It is not the case that…', 'If…then—', 'All', 'Some' and 'True' are ideas the use of which is indispensable for any human's survival, whether or not associated with precisely those word-signs in their native language. A child, and a dog, know the experience associated with being told to do something and to not do that thing at the same time. They may not say the commands were "inconsistent", but they can distinguish inconsistent commands which they can't carry out, from commands which are clear and distinct, positive and consistent.

The objective field of reference which contains the facts that logic deals with is in the realm of ideas; but formal logic requires, for rigor and intersubjective agreement, that we talk not about ideas but about language. For language consists of physical signs, written or spoken, which are associated with ideas or concepts. What makes formal logic rigorous is that it 1) correlates precisely defined features and relationships of physical signs, with certain properties and relationships of ideas, like 'inconsistency' and 'logical synonymy', and 2) it correlates rules of transformation for moving from one set of linguistic symbols to another, with principles for passing from one or more ideas to others which are different but "follow logically".

All theorems of logic implicitly purport to be truths of logic. Formal logic deals with properties and relations of ideas associated with linguistic signs, i.e., their meanings. It connects statements about the abstract physical forms of linguistic signs, with properties or relationships of the meanings (or ideas) associated with such signs. The truths of logic are statements which purport to be true about the logical properties or relations of the meanings of linguistic expressions: properties such as being inconsistent or tautologous and relations such as one expression's meaning the same thing as, or containing, or implying, the meaning of another.

Facts about properties and relationships of meaning can be just as recalcitrant, once meanings have been fixed, as facts about the actual world. In every rigorous system of formal logic certain ideas about the meanings of "and", "or", "not", and "if…then", "all" are laid down and taken as fixed. Logical thinking, by its definition, takes place only if its users hold fast to meanings and rules initially accepted, and accept the consequences of them. Given what is thus laid down, properties and relationships of compound expressions are-what-they-are independently of what we might wish. Any one who has conscientiously tried to establish a new theory of logic, has run up against the recalcitrance and "objective" nature of this type of facts, which follow from our initial definitions and rules. The set of meanings, and rules associating signs with these meanings, become a set of facts in a fixed, objective field of reference, that the logician constantly refers to.

In M-logic the association of 'If P then Q' with 'not both P and not Q' was established by a choice, albeit in an effort to capture the meaning of a familiar term. If this identity of meaning is accepted, a great many "logical facts" follow objectively from this meaning—whether we want them to or not. It is a mark of the intellectual honesty of the great proponents of M-logic, (Russell, Godel, Tarski, Church, Carnap, Quine, Goodman, Hempel) that they recognized the consequences, often unwanted, which follow objectively from the meaning they accepted for this and related terms.

In A-logic a different meaning for 'If P then Q' is fixed initially—also established by a choice, and also in a effort to capture better an ordinary meaning. Many different initially unknown "logical facts" follow inexorably from fixing the definition this new way.

In deciding which logic or which definition to accept and use—we can do no better than to follow the principle of Russell and Whitehead, who wrote in the Preface to Principia Mathematica, "…the chief reason in favour of any theory on the principles of mathematics must always be inductive, i.e., it must lie in the fact that the theory in question enables us to do ordinary mathematics." The argument in favour of A-logic must be that it provides a set of theorems and principles of inference which comes closer than alternative systems to doing the jobs that we have come to expect logic to do, after years of exposure to many examples of ordinary logical thinking and argument.

In logic in the broader sense—where logic depends upon the meanings assigned to non-logical words—facts of logic are-what-they-are regardless of what might be wished once the meanings of non-logical terms are set down as fixed. The meaning of a term is never fixed in concrete, and it can not be legislated for everyone. It can be abandoned, altered by usage, or revised, accepted by some people and rejected by others. But as long as it is held by anyone, for those who hold it there is an objective field of reference and objective consequences for logic to explore.

7.3 Trivalent Truth-tables for A-logic and M-logic

Truth-tables are a short-hand device for a set of inferential conditionals, one for each row, which tell what the truth-value of a compound expression would be for each possible case in which the components have truth-values.

Truth-tables serve several functions,

1) The set of conditionals represented by each row, when read with their ordinary language meanings, help to explicate the meanings of the "logical constants", 'not', 'and', 'or', 'if…then', and of complex expressions built up by these sentential operators, etc.

2) Truth-tables can be used to determine the truth-value of complicated actual sentences when one knows the truth-values of its elementary components.

3) Truth-tables can help to decide whether a wff or a linguistic expression has certain logical properties, or stands in a certain logical relationship to another expression. In both A-logic and M-logic, the properties of tautologousness and inconsistency, are decidable for all unquantified wffs by constructing and inspecting their truth-tables. In A-logic truth-tables alone cannot establish the A-validity of any expression, or whether one expression is logically contained in, or synonymous with, another, but truth-tables can show that one wff is not contained in another, or is is not synonymous with another. Sameness of (the final columns of) truth-tables is a necessary but not sufficient condition of synonymy.

4) Finally, truth-tables are useful is solving some problems about the consistency of a logical systemn and the completeness of a logical system relative to all or part of another system.

Analytic truth-logic is not, and is not intended to be, complete with respect to truth-tables, since the central relationships in A-logic, logical synonymy and logical containment, are not determined by purely truth-functional relations. But it can be proven using truth-tables that certain fragments of axiomatic A-logic in this chapter are sound and complete with respect to the theorems of M-logic.

M-logic, which confines its study to propositions which are either true or false (never neither), assumes only two truth-values, T and F, and uses bivalent truth-tables. A-logic, studying logical relations of C-conditionals and predicates which may be neither true nor false, needs three truth-values, T, F and 0. Thus trivalent truth-tables are the object of study below.

In M-logic only indicative sentences, sentential functions or other wffs are substituted for its variables, P_1, P_2, etc. Since only indicative sentences can be true or false, it makes no sense to talk about applying truth-predicates to, or making up truth-tables for, other than indicative sentences.

In A-logic it is assumed that some indicative sentences can be neither-true-nor-false. The most important cases are indicative C-conditionals, for which there is no symbol in M-logic. Being neither true nor false—symbolized by '0'—is thus an important third "truth-value" in A-logic. Other indicative sentences may be neither true nor false because, though grammatical, they make no literal sense, e.g., Russell's 'quadruplicity drinks procrastination'. Still other meaningful sentences in jokes, most poetry, fiction, and conjectures, are not asserted to be either true or false relative to the actual world as the field of reference. Others are neither true nor false relative to a given field of reference because the subject referred to is not a member of its domain, or because the predicate is not applicable in that field. However, for purposes of formal analytic truth-logic, we need only one kind of meaningful indicative sentence which can be neither true nor false. C-conditionals fill that bill and the crucial importance for logic of the third value will be established in Chapter 8 and 9 which follow.

Thus, though only wffs of sentential M-logic are used in this chapter, truth-tables for indicative sentences will presuppose three values in anticipation of the complete system which follows.[22]

Elementary wffs will have three possible "truth-values",—true, false, and neither-true-nor-false. Using T for '<1>is true' and 'F' for '<1>is false' we add '0' for '<1>is null', i.e., '<1> is not-true & <1> is not-false'. A wff with two components will have nine rows and an expression with **n** different elementary components will have 3^n rows in its truth-table, one for each possible case. Every truth-table with **n** atomic wffs is short-hand for 3^n conditionals. Each conditional tells what the truth-value of the whole compound sentence will be when each of the n atomic components is assumed to have just one of the three truth-values, T, F and 0. Every row in any truth-table abbreviates a particular kind of subjunctive or contrary-to-fact conditional in which the antecedent describes a possible assignments of truth-values to the atomic components, and the consequent states what the truth-falue of the whole expression would if the components have those values.

The trivalent truth-table for the connective, '&', is a device which represents principles and rules for deriving the truth-value of a complex expression, if we have information about the truth-values of its components. The principles involved, are stated in the usual way on the right.

	P	Q	Truth-table (P&Q)	(The rows may be read as saying:)
Row 1:	0	0	0	If P is Null and Q is Null, then [P&Q] is Null.
Row 2:	T	0	0	If P is True and Q is Null, then [P&Q] is Null.
Row 3:	F	0	F	If P is False and Q is Null, then [P&Q] is False.
Row 4:	0	T	0	If P is Null and Q is True, then [P&Q] is Null.
Row 5:	T	T	T	If P is True and Q is True, then [P&Q] is True.
Row 6:	F	T	F	If P is False and Q is True, then [P&Q] is False.
Row 7:	0	F	F	If P is Null and Q is False, then [P&Q] is False.

22. Unsaturated predicates, noun phrases, questions, directives , etc., automatically take the value '0' but these cases are not of interest for truth-logic. If a wff has only unsaturated predicate schemata as atomic wffs, there is no point to a truth-table; POS-NEG tables will decide whether the predicate schema is Taut or Inc or Contingent.

Row 8:	T	F	F	If P is True and Q is False, then [P&Q] is False.
Row 9:	F	F	F	If P is False and Q is False, then [P&Q] is False.

For each row the information in the index columns is described in the antecedent and the value of the whole conditional under those conditions is described in the consequent. The nine cases exhaust the set of possibiliies. 'T' is assigned to [P&Q] only in the one case of row 5. Since Row 5 is the only row of the nine possible rows which make the conjunction as a whole true, it gives both the necessary and sufficient condition for [P&Q]'s being true. A conjunction is false if any conjunct is false, thus [P&Q] is assigned F in rows 3, 6, 7, 8 and 9 wherever one of the conjuncts is false. In the remaining cases (1, 2 and 4), the conjunction is neither-true-nor-false: if both conjuncts are neither-true-nor- false or if one is true and the other neither-true-nor-false.

The same truth-table may be expressed as a 3X3 matrix in which the value of the left-hand conjunct is given in the column to the left of the truth-table, the value of the right-hand conjunct is given in the top-most 0,T,F row, the value of the left-hand comjunct, and the value of the whole conjunction, given the values of the components, is given where the rows and columns intersect.

			Q	
	(P&Q)	0	T	F
	0	0	0	F
P	T	0	T	F
	F	F	F	F

To read these matrices, one reads "if P is true (2nd *row* named in index column on left) and Q is false (3rd *column* named in the index row on top), then (P&Q) is false (the value at the location where the row and column intersects). The resulting conditional is that of Row 8 in the first truth-table for '&'.

The left-most seven truth-tables below, constitute a complete set of trivalent truth-tables for connectives of sentential M-logic; tables for 'P', 'not', the truth-operator 'T', conjunction, disjunction, the TF-conditional and TF-biconditional. There are other ways to interpret '&' and 'v' in a trivalent system which would be compatible with the theorems of M-logic.[23] But this set is the one which best fits the axiomatic system of A-logic. The right-most table for '=>', the C-conditional, will not be considered until the next chapter, since it is not reducible to any connective in M-logic.[24]

---

23. See Stephen Cole Kleene, in *Introduction to Metamathematics*, 1952 (p.334). Kleene called the set of truth-tables above for M-logic, "strong" 3-valued truth-tables. He called an alternative set with the &' and 'v' tables shown in the box, the "weak" 3-valued truth-tables. Although intuitively it may not appear so, this interpretation will also assign T's but no F's to all and only theorems of sentential M-logic. (Kleene used 'u' for "undefined" instead of '0' for "neither true nor false").

To coordinate our axiom system with these, Ax.7-4 [(T(P v Q) Syn (TP v TQ)] must be replaced by [(0(P v Q) Syn (0P v 0Q)]. Among other results: the "weak" system yields only [(T(P v Q) Cont (TP v TQ)]—not its converse, which follows from Ax.7-4; also, the "weak" system yields only |= [TQ Impl ~F(P v Q)], instead of Ti7-51. [TP Impl T(P v Q)].

~P		(PvQ)	0	T	F	(P&Q)	0	T	F
0	0	0	0	0	0	0	0	0	0
F	T	T	0	T	T	T	0	T	F
T	F	F	0	T	F	F	0	F	F

24. This whole set of truth-tables, including that of the C-conditional, is the same as that presented by DeFinetti to account for conditional probability. The relation to DeFinetti's "logic of probability' is explained in Section 9.312. In Chapter 8 we will provide truth-tables for C-conditionals with T-wffs, and then prove the A-validity of C-conditionals each rule associates with a row in each of the trivalent truth-tables. These proofs are based in large part on theorems, including A-implication-theorems, established in this chapter. In this chapter, lacking the required conditional, we can derive only the relation of A-implication between premiss and conclusion in an argument, on which validity of those C-conditionals is based.

P	~ P	T P	& | 0 T F	v | 0 T F	⊃ | 0 T F	≡ | 0 T F	||	=> | 0 T F
0	0 0	F 0	0 | 0 0 F	0 | 0 T 0	0 | 0 T 0	0 | 0 0 0	||	0 | 0 0 0
T	F T	T T	T | 0 T F	T | T T T	T | 0 T F	T | 0 T F	||	T | 0 T F
F	T F	F F	F | F F F	F | 0 T F	F | T T T	F | 0 F T	||	F | 0 0 0

If there is a particular compound factual sentence, and we want to decide whether this particular sentence as a whole is true or false or neither, we use truth-tables in the following way:

First determine the truth-values which belong to each of its atomic components. Then find the one row in the table which, in the index column, assigns these values to those atomic parts. Then look to the final column of the truth-table (or the point of intersection of rows and columns in the matrix) for the truth-value of the statement as a whole. This process is equivalent to using Modus Ponens with the conditional associated with that row as the major premiss. In short, by truth-tables we can determine the truth-value of a particular compound statement, by locating the relevant single row. For example, if any sentence has the form '(Pa & Qb)', and we find that the component which replaces 'Pa' is true while that which replaces 'Qb' is neither true nor false, then we should go to the second row of the truth-table for '[P & Q]'. The rule for that row says that given these findings, a sentence of the form [Pa & Qb] is neither true nor false. So by modus ponenes, it would be correct to prefix 'It is neither true nor false that' to yield '(Pa & Qb)' in such a case. Of course, determining whether the atomic components of an actual sentence are T, F or 0, depends on knowing what they mean as well as relevant facts in the field of reference. This use of truth-tables is a way to determine the truth-value of particular (especially complex) sentences based on determinations of the truth values of the components.

A second use of truth-tables is as a device—in some cases a short-cut—to determine or confirm whether the abstract logical structures displayed in wffs, have certain logical properties or relationships. In this case we are not looking for a single appropriate row, but for properties of the truth-table as whole—properties and relations that hold of the set of all rows.

With respect to inconsistency and tautologousness, the most we can say is,

1) a) If the truth-table of a wff has a T in its final column,
   then the wff is **satisfiable** (capable of having true instances) and **not inconsistent**.
2) If the truth-table of a wff has a no Ts in its final column
   it is **unsatisfiable—**incapable of having true, instances,
3) If the truth-table of a wff has an F in its final column,
   it is **falsifiable** (capable of having false instances) **and not-tautologous**.
4) If the truth-table of a wff has an F and no Ts in its final column,
   it is **falsifiable and unsatisfiable—**capable of having false, but not true, instances,
5) if the truth-table of a wff has a T and an F in its final column,
   then the wff is neither inconsistent nor tautologous, but **contingent**.
6) If two wffs differ in truth-value at the final column of any row,
   then they are **not logically synonymous**.
7) If P and Q are wffs and and the truth-table of P has the value T in the final column
   of some row in which Q has the value 0 or F,
   then **P does not logically contain Q**.

If two wffs are synonymous their truth-tables must be the same. But, since truth-functional equivalence and implication do not require any containment connection between antecedent and consequent, synonymy and containment, entailment, implication and validity are not determinable by truth-tables alone. At best truth-tables can provide short-cut methods for determining that these relations or properties do not hold of specific expressions.

These trivalent truth-tables will be utilized in Section 7-5 to resolve issues concerning the consistency, soundness and completeness of Analytic Truth-logic relative toM-logic, and the consistency and completeness of the axioms of A-logic relative to its truth-table model.

7.4 A Formal Axiomatic Logic with the T-operator

The formal system in this chapter is a sub-system of analytic truth-logic. It is not complete because we omit the C-conditional and theorems of Chapter 6, leaving only wffs of M-logic. But it is consistent and complete with respect to the theorems of M-logic, though it also includes distinctions and theorems not expressible in M-logic because M-logic does not allow T-operators.

7.41 The Logistic Base

The logic of the truth-operator adds the following to the base of Chapter 5:

RULES OF FORMATION: RF7-1 .If A is wff, T(A) is wff

ABBREVIATIONS: Df 'F'. [FP Syn_{df} T∼P]
Df '0'. [0(P) Syn_{df} (∼TP & ∼FP)]

AXIOMS: Ax.7-1. [TP Syn (TP & ∼FP)]
Ax.7-2. [FTP Syn ∼TP]
Ax.7-3. [T(P & Q) Syn (TP & TQ)]
Ax.7-4. [T(P v Q) Syn (TP v TQ)]
Ax.7-5. [((TP & ∼TP)v TQ)) Cont TQ]

RULES OF INFERENCE: R7-1. [If (P Syn Q) then (TP Syn TQ)]
R7-2. [If Inc(P) then |= ∼T(P)]

This base is added to the base of Chapter 5, in which the wffs were the wffs of M-logic. From this base, with the theorems of Chapters 1 to 5, we derive T-theorems—theorems of truth-logic. By rule R7-1 we derive Syn- and Cont-theorems of related T-wffs from the synonymy theorems and containment theorems in Chapter 1 to 4. A derived rule, DR7-5, [If INC(P) then Inc(TP)], extends the Inc-and TAUT-theorems of Chapter 5 to T-wffs. The rule R7-2 and derived rules DR7-2' to DR7-2b' all derive theorems of logical unverifiability and logical unfalsifiability (with '∼T' or '∼F' prefixed to some wff) from the Inc- and Taut- theorems. T-theorems are statements about the logical properties and relations of any assertions of truth and non-truth represented by T-wffs of M-logic.

The definitions, and axioms Ax.7-1 through Ax.7-5, determine the logical relationships of the operator 'T' with the other operators and with itself:

Df 'F', [FP Syn_{df} T(∼P)], determines the relationship between primitives 'T', '∼', and 'F'.
Df '0'. [0(P) Syn_{df} (∼TP & ∼FP)], determines relationships between 'T', 'F',' ∼', and '0'.
Ax.7-1. [TP Syn (TP & ∼FP)], and Ax.7-2, [FTP Syn ∼TP], together determine all relationships between T and itself, T and F, and T and the negation sign.

Ax.7-3, [T(P & Q) Syn (TP & TQ)], determines the relationships between T and &.
Ax.7-4, [T(P v Q) Syn (TP v TQ)], determines the relationships between T and v.
Ax.7-5, [((TP & ~TP) v TQ)) Cont TQ], determines certain detachment rules, based on a relationship between T and compounds with ~, &, and v.

The relationships between T and quantifiers follow from these axioms and the definitions of quantifiers. The relations between T and the C-conditional are deferred to the next chapter. Every axiom in this set is acceptable to M-logic's semantics. The main difference between analytic truth-logic and M-logic are theorems of A-logic with C-conditionals that can not be derived from this base.

7.42 Theorems and Inference Rules

There are six different sorts of theorems to be considered:

1) Syn- and Cont-theorems of truth-logic
2) Inc- and TAUT-theorems of truth-logic
3) Unsatisfiability- and unfalsifiability-theorems of truth-logic
4) Presuppositions of Truth-logic, Logical-truth and Logical-falsehood
5) Implication theorems of Truth-logic
6) Validity-theorems for inference schemata.

We begin with Syn- and Cont-theorems with T-wffs and inference rules for deriving such theorems in Section 7.421. These are the fundamental theorems. Each of the other kinds depends on these relationships. The second, third and fourth kinds of theorems will be discussed in Section 7.422, with particular attention to theorems which distinguish wffs which are logically true (i.e., true for all values of their variables). The latter provide the presuppositions which underlie theorems of *de dicto* implication. In Section 7.423 we develop implication-theorems of a special sort based on analytic truth-logic. In Section 7.424 theorems about valid inferences from one of M-logics T-wffs to another are developed; these are the grounds of M-logic's claim to plausability.

7.421 Syn- and Cont-theorems for T-wffs

Many Syn- and Cont-theorems of truth-logic are uniquely derivable from the axioms and definitions in the logical base of this chapter. Others follow by rules derived from inference rule R7-1, using Syn- and Cont-theorems of previous chapters. We look first at the latter derived rules.

7.4211 Rules for Deriving Syn- and Cont-theorems of T-logic from Syn- and Cont-theorems in Chapters 1 to 4

The inference rule, R7-1, [If (P SYN Q) then (TP Syn TQ)], says that if two expressions are logically synonymous, then the assertion that one of them *is true* is synonymous with the assertion that the other *is true*. By R7-1 every Syn-theorem derivable from previous chapters, is immediately convertible into a theorem which asserts the synonymy of two T-wffs. For example we get,

|= [T(Pv(Q&R)) Syn T((PvQ)&(PvR))] [by Ax.4-4,R7-1]

The Syn- and Cont-theorems gotten from theorems in Chapters 1 through 4 by R7-1 and other derived rules, will be numbered T7-xyz, where x is the number of the earlier chapter and yz is the two-digit number of the Syn- or Cont-theorem in chapter x from which T7-xyz is derived. For example,

T7-455. [T(~ Pv ~ Q) Syn T ~ (P&Q)]	[T4-15,R7-1]
T7-464. [T(P ⊃ Q) Syn T(~ P v Q)]	[T4-24,R7-1]
T7-119. [T((PvQ)&(RvS)) Syn T(((PvQ)&(RvS)) & (PvRv(Q&S))]	[T1-19,R7-1]
T7-480. [T(∀x)(Px ≡ Qx) Syn T((∀x)Px ≡ (∀x)Qx)))]	[T4-30,R7-1]

But 'T' is not the only T-operator which can be prefixed to T-wffs. From R7-1 we derive additional rules to preserve synonymy upon prefixing F, ~ T or ~ F to synonymous expressions:

DR7-1a. [If (P SYN Q) then (F(P) Syn F(Q))]

Proof: 1) [P SYN Q]	[Premiss]
2) [~ P SYN ~ Q)]	[1),DR4-1,MP]
3) [T(~ P) Syn T(~ Q))]	[2),R7-1,U-SUB]
4) [F(P) Syn F(Q)]	[3),Df 'F']
5) [If (P SYN Q), then (F(P) Syn F(Q))].	[1) to 4),C.P.]

DR7-1b. [If (P SYN Q) then (~ T(P) Syn ~ T(Q))]

Proof: 1) [P SYN Q]	[Premiss]
2) [TP Syn TQ)]	[1),R7-1.MP]
3) [~ TP) Syn ~ TQ))]	[2),DR4-1,MP]
4) [If (P SYN Q), then (~ T(P) Syn ~ T(Q))]	[1) to 3),C.P.]

DR7-1c. [If (P SYN Q) then (~ F(P) Syn ~ F(Q))]

Proof: 1) [P SYN Q]	[Premiss]
2) [FP Syn FQ)]	[1),DR7-1a,MP]
3) [~ FP Syn ~ FQ]	[2),DR4-1,MP]
4) [If (P SYN Q), then (~ F(P) Syn ~ F(Q))]	[1) to 3),C.P.]

Thus using R7-1 and derived rules DR7-1a, DR7-1b, and DR7-1c, every SYN-theorem in previous chapters is converted into a theorem asserting that falsehood, non-truth and non-falsehood are preserved upon SynSUBstitution.

The same also holds for Containment, when derived from SYN-theorems. By DR1-11, If A Syn B, then A Cont B. Thus we can have Containment-theorems for any pair of wffs which are logically synonymous. The following (unnumbered) rules are valid.

|= [If (P SYN Q) then (TP Cont TQ)]

Proof: 1) [P SYN Q]	[Premiss]
2) [TP Syn TQ)]	[1),R7-1,MP]
3) If [TP Syn TQ] then [TP Cont TQ]	[DR1-11]
4) TP Cont TQ	[2),3),MP]
5) [If (P SYN Q), then (TP Cont TQ)]	[1) to 4) Cond.Proof]

|= [If (P SYN Q) then (TQ Cont TP)]

Proof: 1) [P SYN Q]	[Premiss]
2) [Q SYN P]	[1),DR1-01]
3) [If (Q Syn P) then (TQ Cont TP)]	[Preceding theorem]
4) [TQ Cont TP]	[2),3),MP]
5) [If (P SYN Q), then (TQ Cont TP)]	[1) to 4),Cond.Proof]

|= [If (P SYN Q) then (FP Cont FQ)]
|= [If (P SYN Q) then (FQ Cont FP)] [Proof similar, from DR7-1a]
|= [If (P SYN Q) then (~TP Cont ~TQ)]
|= [If (P SYN Q) then (~TQ Cont ~TP)] [Proof similar, from DR7-1b]
|= [If (P SYN Q) then (~FP Cont ~FQ)]
|= [If (P SYN Q) then (~FQ Cont ~FP)] [Proof similar, from DR7-1c]

These rules may also be used to get deduction rules like the following which are <u>used</u> throughout the text and proved in Chapter 8:

|= [If (TP & T(P Syn Q)) then TQ]
|= [If (~TP & T(P Syn Q)) then ~TQ]
|= [If (FP & T(P Syn Q)) then FQ]
|= [If (~FP & T(P Syn Q)) then ~FQ]

However, moves from a CONT-theorem (vs. SYN-theorems) to a Containment between two T-wffs, are more limited than moves from A SYN B. The conditionals "[If (A CONT B) then (FA Cont FB)]" and "[If (A CONT B) then (~TA Cont ~TB)]" are falsifiable and invalid. If A CONT B, it does not follow that 'A is false' Contains 'B is false', or that 'A is not true' Contains 'B is not true'. For example, (P&Q) contains Q, but it does not follow that if (P&Q) is false (or is not true), then Q is false (or is not true). There are no analogues of DR7-1a and DR7-1b for CONT-theorems.

<u>Valid</u>:	<u>Not valid</u>:	<u>Valid</u>:	<u>Not valid</u>:
<u>. P Cont Q</u>	<u>~P Cont ~Q</u>	<u>~P Cont ~Q</u>	<u>P Cont Q</u>
.: TP Cont TQ	.: ~TP Cont ~TQ	.: FP Cont FQ	.: FP Cont FQ

It should also be noted that Cont-theorem are not transposable; if P Cont Q, it does not follow that ~Q Cont ~P. For example, |= [(P & Q) CONT Q] but ~Q does not Contain ~(P & Q), since neither P nor ~P occurs in ~Q.

The two principles which <u>are</u> correct depend on Ax.7-3 and Ax.7-4. They are:

DR7-1d. [If (P CONT Q) then (TP Cont TQ)]
<u>Proof</u>: 1) [P CONT Q] [Premiss]
2) [P SYN (P&Q)] [Df 'CONT']
3) [TP Syn T(P&Q)] [2),R7-1,MP]
4) [TP Syn (TP & TQ)] [3),Ax.7-3,MP]
5) [TP Cont TQ] [4),Df 'CONT']
6) [If (P CONT Q) then (TP Cont TQ)] [1) to 5), C.P.]

DR7-1e. [If (P CONT Q) then (~FP Cont ~FQ)][25]
<u>Proof</u>: 1) [P CONT Q] [Premiss]
2) [P SYN (P&Q)] [1),Df 'CONT']
3) [~FP Syn ~F(P&Q)] [2),DR7-1c]

---

25. Doesn't hold in Kleene's "weak" truth-tables: (P & Q) Cont Q ~F(P & Q) ~FQ
(See footnote 23) 0 0 F F **T**F 0 0 F **F**TF

4) [∼FP Syn ∼T∼(P&Q)] [3),Df 'F']
5) [∼FP Syn ∼T(∼Pv∼Q)] [4),DeM,R1-1]
6) [∼FP Syn ∼(T∼PvT∼Q)] [5),Ax.7-4]
7) [∼FP Syn ∼(FP v FQ)] [6),Df 'F']
8) [∼FP Syn (∼FP & ∼FQ)] [7),DeM]
9) [∼FP Cont ∼FQ] [6),Df 'Cont']
10) [If (P CONT Q) then (∼FP Cont ∼FQ)] [1) to 9), C.P]

The containment theorems derivable by DR7-1d and DR7-1e include a great many which can not be gotten directly from any of the previous rules. These are cases where A CONT B, but not A SYN B. Using rules R7-1 to DR7-1e, all of the SYN-theorems and CONT-theorems of chapters 1 to 4 are convertible into T-theorems which assert the containment relation between T-wffs. To the names of these theorems we will add 'a','b', etc., for the derived rules DR7-1a, DR7-1b, ..., DR7-1e, etc. For example:

T7-176d. [T(P & Q) Cont TP] [T1-36,DR7-1d]
T7-176e. [∼F(P & Q) Cont ∼FP] [T1-36,DR7-1e]
T7-357d. [T(∃y)(∀x)Rxy Cont T(∀x)(∃y)Rxy] [T3-37, DR7-1d]
T7-357e. [∼F(∃x)(∀y)Rxy Cont ∼F(∀y)(∃x)Rxy] [T3-37, DR7-1e]
T7-359d. [T(∀x)(Px v Qx) Cont T((∃x)Px v (∀x)Qx)] [T3-39,DR7-1d]
T7-469d. [T(∀x)(Px ⊃ Qx) Cont T((∀x)Px ⊃ (∀x)Qx] [T4-29, DR7-1d]

Thus for every SYN- or CONT-theorem in Chapters 1 through 4, there are at least six Syn- or Cont-theorems with T-operators prefixed to the two major components.

This plethora of theorems is further increased by Axioms 7-1 to 7-5 of the truth-operator, for there are many ways to distribute truth-operators within components of a T-wff. One way which is very useful, is described in the Normal Form Theorem for T-wffs, i.e, the derived rule DR7-NF which is established below in Section 7.42122.

7.4212 Syn- and Cont-theorems of Chapter 7

Turning to the Synonymy and Containment theorems uniquely derivable from the Chapter 7 Axioms & Definitions which have occurrences of 'T', we look first at those which require only Df 'F' and Axiom 7-1. The novelties are 1) falsehood is not synonymous with non-truth and truth is not synonymous with non-falsehood, 2) there are theorems about expressions which are neither true nor false, and 3) there are constraints placed upon the transposition principles of Containment statements.[26]

7.42121 Syn- and Cont-theorems from Ax.7-1 and Df 'F'

The following theorems are in accord with M-logic's semantics, though the ommission of other Cont-statements as theorems signifies distinctions that M-logic does not recognize.

26. These constraints lead later to constraints on valid transposition principles of C-conditionals.See

T7-06. [FP Syn T ~ P] [Df 'F']
Ax.7-1. [TP Syn (TP & ~FP)]
[FP Syn (FP & ~TP)]
T7-11. [TP Syn TP]
T7-12. [FP Syn FP]
T7-13. [TP Cont ~FP]
T7-14. [FP Syn (FP & ~TP)]
T7-15. [FP Cont ~TP]
T7-16. [TP Syn F ~ P]

T7-11 is the correlate of T1-11 in truth-logic:

T7-11 [TP Syn TP]
Proof: 1) P Syn P [T1-11]
2) TP Syn TP [1),R7-1]

T7-12 follows from T7-11:

T7-12 [FP Syn FP]
Proof: 1) ~P Syn ~P [T1-11,DR4-1]
2) T ~ P Syn T ~ P [1),R7-1]
3) FP Syn FP [2),Df 'F'(twice)]

T7-13. TP Cont ~FP
Proof: 1) TP Syn (TP & ~FP) [Ax.7-1]
2) TP Cont ~FP [1),Df 'Cont']

T7-14. FP Syn (FP & ~TP)
Proof: 1) T ~ P Syn (T ~ P & ~F ~ P) [Ax.7-1,U-SUBb]
2) FP Syn (FP & ~T ~~ P) [1),Df 'F' (thrice)]
3) FP Syn (FP & ~TP) [2),DN]

T7-15. FP Cont ~TP
Proof: 1) T ~ P Cont ~F ~ P [T7-13,U-SUBb]
2) T ~ P Cont ~T ~~ P [1),Df 'F']
3) T ~ P Cont ~TP [2),DN]
4) FP Cont ~TP [3),Df 'F']

T7-16. TP Syn F ~ P
Proof: 1) TP Syn T ~ (~ P) [Ax.5-5,R7-1]
2) TP Syn F ~ P [1)Df 'F',SynSUB]

The two one-way containments in T7-13 and T7-15 follow from Ax.7-1 and they are contrapositives. Their converses, '~FP Cont TP' and '~TP Cont FP', do not hold, though in M-logic semantics the two components are treated as logical equivalents. Neither [FQ Cont FP] nor [~TQ Cont ~TP] follow from [TP Cont TQ], so transposition does not hold for Cont-statements in general (for example, T(P&Q) CONT TQ, but neither FQ nor ~TQ Contains ~T(P&Q) since 'P' does not occur in 'FQ' or '~TQ').

The set of T-theorems of A-logic differs from M-logic then, in the absence of certain theorems due to the absence of the converses of T7-13 and T7-15. If A contains B, then A can't be true and B false. By trivalent truth-tables, the converses of T7-13 and T7-15 fail this test.

T7-15. FP Cont ~TP	~TP Cont FP	T7-13.TP Cont ~FP	~FP Cont TP
~(FP & ~~TP)	~(~TP & ~FP)	~(TP & ~~FP)	~ (~FP & ~TP)
T F0 F FT F0	**F** TF0 T TF0	**T** F0 F FT F0	**F** T F0 T T F0
T FT F TF TT	**T** FTT F TFT	**T** TT F FT FT	**T** T FT F F TT
T TF F FT FF	**T** TFF F FTF	**T** FF F TF TF	**T** F TF F T FF

If we had Convention T or an axiom Ax.Z. [TP Syn ~FP] then the converses of T7-13 and T7-15 would hold and the resulting theorems would conform to M-logic semantics.

Under Convention T. [TP Syn P]

1) T~P Syn ~P	Conv.T,U-SUB,'~P' for 'P']
2) FP Syn ~P	[1),Df 'F']
3) ~TP Syn ~P	[Conv.T,DR4-1]
4) FP Syn ~TP	[3),2),SynSUB]
5) ~~TP Syn ~FP	[4),DR4-1]
6) TP Syn ~FP	[5),DN]

Ax.Z [TP Syn ~FP]

1) T~P Syn ~F~P	[Ax.Z, U-SUB]
2) FP Syn ~T~~P	[1), Df 'F']
3) FP Syn ~TP	[2), DN]

Instead, in A-logic, ~FP does not contain TP, although TP contains ~FP, and TP does not contain ~FP, although FP contains ~TP. That is, instead of "TP Syn ~FP" (Ax.Z above) we have two one-way Cont-theorems, T7-13 and T7-15.

M-logic can not express these differences because it does not include a T-operator and does not distinguish T~P from ~TP, or TP from ~FP. The differences are reflected in the differences between the trivalent truth-tables for 'TP', 'T~P'(i.e. FP), '~TP', and '~T~P'(i.e.,'~FP'), and two-valued truth-tables. If these differences are not recognized M-logic's "Excluded Middle" (TP v FP) and A-logic's Principle of Trivalence (TPvFPv0P), i.e., (TPvFPv(~TP&~FP), are synonymous.

If instead of Ax.7-1 we had followed Tarski's Convention and added T[TP Syn P], then keeping all other rules and definitions, in addition to the theorems we do derive we would derive many theorems not derivable from Ax.7-1, and all occurrences of T and F would be eliminable.

If, in accord with Tarski's "convention T", TP and P are equivalent, every atomic wff should have 'T' prefixed to it thus we would have '~TP' but no 'T~P' (no 'F'). All and only the inconsistencies and tautologies of M-logic would be expressible. Elementary wffs would have only two kinds of truth-tables. It could never be true an atomic wff was both not- true and not-false. If [TP Syn P] were an axiom of A-logic, no difference could be drawn between 'It is false that P' and 'It is not true that P'. The same consequences hold if only 'F' is prefixed to every atomic wff. Only if every atomic wff, P, is not intersubstitutable with 'TP',—i.e., either 'T' or 'F' (i.e., 'T~') can be prefixed to it—can the distinctions in A-logic be expressed in its symbols.

TP	~TP
F 0	**T** F 0
T T	**F** T T
F F	**T** F F

M-logic is always presented as a truth-logic. In its semantical and philosophical interpretation it is described as a logic of propositions; of statements which must be either true or false—never neither and never both. In its semantic theory, its operators are explicated by truth- tables which assume that every component is either true or false (never neither). Validity, and invalidity are defined in terms of truth and falsehood. Yet, anomalously, the predicate '<1>is true' and the operator 'It is true that...' are banned from the object-language on grounds that they would lead to logical paradoxes, i.e., contradictions proven by valid arguments.

If the basic wffs, '∼TP' and 'T∼P', mean the same thing—as they are thought to do in M-logic—then either this chapter's truth-logic collapses into M-logic with meaningless occurrences of 'T', or, if Tarski's arguments are accepted, it self-destructs into inconsistency due to the Liar paradox.

If, on the other hand, '∼TP' for 'It is not true that P' and 'T∼P' for 'It is false that P', are granted to have different meanings, then the theorems of analytic truth-logic can be grouped into classes of those which are in one-to-one correspondence with the theorems of M-logic, and the class of T-theorems of A-logic which are not in 1-1 correspondence with M-logic theorems because they have meanings in A-logic which are not expressible in M-logic or its semantics.

7.42122 The Normal Form Theorem for T-wffs, from Ax.7-1 to Ax.7-4

We define a Normal Form T-wff as a wff which is in normal form and has a single T prefixed to all and only the elementary wffs. I.e., either a 'T' or an 'F' is prefixed to each atomic wff and to no larger components. The Normal Form Theorem states that every T-wff is synonymous with one or more Normal Form T-wffs. More precisely, it the definition is,

DR7-NF. Every wff that lies in the scope of a 'T' (whether or not also in the scope of an '∼'), is reducible to a synonym which has 'T' or 'F' prefixed to all and only its atomic wffs.

To establish DR7-NF, we need a limit on the meanings of iterated truth-operators. This is established in Ax.7-2. [FTP Syn ∼TP], which is synonymous (by Df 'F') with |= [T∼TP Syn ∼TP]. The latter says that prefixing a 'T' to a '∼T' adds nothing to the meaning. Since 'FP' Syn_{df} 'T∼P', prefixing a 'T' to '∼F' also adds nothing to the meaning. Theorems T7-17 to T7-21 are based on Ax.7-2. These, together with T7-16, and the definition of 'F' are sufficient to prove DR7-NF provided we apply them, in their general forms, to quantified wffs.

The singulary operators, '∼', 'T', and 'F' when prefixed to any wff, form another wff. Thus any sequence of occurrences of '∼', 'T', or 'F' is a well-formed prefix. DR7-NF says that any wff with more than two T-operators prefixed is reducible to synonymous wffs with only one occurrence of 'T' (or 'F') prefixed to each of its atomic wffs. Thus, every expression with iterated occurrences of T and F (with or without negation) prefixed to 'P', such as TF(P), FT∼FT(P), ∼FTFTFT(P), etc., is reducible to a synonymous expression with just one of the four operators, T(P), F(P), ∼T(P) or ∼F(P). This is proved as follows:

There are nine possible sequences of two singulary operators:'TT','TF', 'T∼', 'FT', 'FF', 'F∼', '∼T', '∼F' and '∼∼'. Theorems T7-11 toT7-21 prove that each is Syn with either 'T', 'F', '∼T' or '∼F'.

T7-17 FFP Syn ∼FP

Proof: 1) FT(∼P) Syn ∼T(∼P) [Ax.7-2,U-SUB]

2) FF(P) Syn ∼F(P) [1),Df 'F',(twice)]

Axiom 7-2 adds a feature to A-logic which makes the letters T and F contain, as subordinate components, expressions with endless iterations of T and and/or F prefixed to the same expression. The simplest examples are T7-18 and T7-19. Some of these underly truth-table rules.

T7-18 FP Cont FTP [See Row 3 of truth-table for 'TP']
Proof: 1) FP Cont ~TP [T7-15]
2) FTP Syn ~TP [Ax.7-2]
3) FP Cont FTP [1),2),SynSUB]

T7-19 TP Cont FFP [See Row 2 of truth-table for 'FP']
Proof: 1) TP Cont ~FP [T7-13]
2) FFP Syn ~FP [T7-17]
3) TP Cont FFP [1),2),SynSUB]

Axiom 7-2 says that 'FTP' ("It is false that P is true") is synonymous with '~TP' ("It is not the case that P is true)'. From this it follows that, 'it is not false that P is true' means the same as 'P is true', i.e.,. |= [~FTP Syn TP]. This is used as Step 3) in the following proof:

T7-20 TTP Syn TP
Proof: 1) FTP Syn ~TP [Ax.7-2]
2) ~FTP Syn ~~TP [1),DR4-1]
3) ~FTP Syn TP [2),DN]
4) TP Syn ~FTP [3), DR1-01]
5) TTP Syn T~FTP [4),R7-1]
6) TTP Syn FFTP [5),Df 'F',SynSUB]
7) TTP Syn ~FTP [6),T7-17,SynSUB]
8) TTP Syn TP [7),3),SynSUB]

T7-21 TFP Syn FP
Proof: 1) TT(~P) Syn T(~P) [T7-20,wff-SUB]
2) TF(P) Syn F(P) [1),Df 'F']

Thus each of the nine ordered pairs of singulary operators is synonymous with one of the basic four T-operators, 'T(P)', 'F(P)', '~T(P)' and '~F(P)':

TT(P) Syn T(P)	[T7-20]	FT(P) Syn ~T(P)	[Ax.7-2]	~T(P) Syn ~T(P)	[T7-11]
TF(P) Syn F(P)	[T7-21]	FF(P) Syn ~F(P)	[T7-17]	~F(P) Syn ~F(P)	[T7-12]
T(~P) Syn F(P)	[Df 'F']	F(~P) Syn T(P)	[T7-16]	~(~P) Syn (P)	[Ax.4-5]

Any wff with a sequence of *n* T-operators prefixed to it, it is reducible, **salve sens**, to a logically synonymous wff with only one of the four basic operators prefixed to it.

Reduction to normal form T-wffs requires also that a T-operator prefixed to a conjunction or disjunction will always be synonymous with some wff in which the components have T-operators prefixed but the whole does not. If any conjunction or disjunction lies in the scope of a T (or F), the T can be moved inside and applied, perhaps with a Negation operator, to the components of a synonymous wff. This follows from

Ax.7-3. [T(P & Q) Syn (TP & TQ)]
Ax.7-4. [T(P v Q) Syn (TP v TQ)]
T7-22. [F(P & Q) Syn (FP v FQ)]
T7-23. [F(P v Q) Syn (FP & FQ)]

The last two are proved using DeMorgan theorems T4-15, [~ (P&Q) SYN (~ Pv ~ Q)] and T4-14, [~ (PvQ) SYN (~ P& ~ Q)]:

T7-22. [F(P & Q) Syn (FP v FQ)]

<u>Proof</u>: 1)	[T ~ (P & Q) Syn T(~ P v ~ Q)]	[T4-15,R7-1]
2)	[T ~ (P & Q) Syn (T ~ P v T ~ Q)]	[1), Ax.7-4]
3)	[F(P & Q) Syn (FP v FQ)]	[2),Df ‘F’(thrice)]

T7-23. [F(P v Q) Syn (FP & FQ)]

<u>Proof</u>: 1)	[T ~ (P v Q) Syn T(~ P & ~ Q)]	[T4-14,R7-1]
2)	[T ~ (P v Q) Syn (T ~ P & T ~ Q)]	[1),Ax.7-3]
3)	[F(P v Q) Syn (FP & FQ)]	[2),Df ‘F’(thrice)]

T-operators can also have occurrences of ‘0’, ‘ ~ 0’, or ‘0 ~ ’. When this is the case, the wff expands into conjunctions or disjunctions. Two examples:

a) ‘T0 ~ FP’ becomes ‘T(~ T(~ FP) & ~ F(~ FP))’
which is Syn to ‘F(T(~ FP) v F(~ FP))’ and reduces to ‘(FP & ~ FP)’.

b) ‘ ~ T ~ F0 ~ T ~~ P’ reduces to (~ TP v TP), i.e., to “Either P is not true or P is true.”[27]
(This reduction by SynSUB is carried out in the footnote)

Syn- and Cont-theorems involving the 0-operator, have their own interest and will be developed in Section 7.42125. The present point is that by means of the metatheorems above, every wff which lies in the scope of an occurrence of T-operator is reducible to a synonym in which all and only its atomic wffs have either a T or an F prefixed to them.

Thus DR7-NF is proven for all unquantified T-wffs.

A normal form T-wff, or ‘NFT-wff’, may now be defined formally by adding a clause (iii) to the two-clauses of Df ‘NF’, in Section 1.131, so that Df ‘NFT-wff’ reads:

---

27. Section 8.31. In the following chapters this is a necessary element in the solution of major problems which have haunted M-logic. Transposition principles of TF-conditionals remain unchanged throughout. As an example consider the proof that <u>[~ T ~ F0 ~ T ~~ P Syn (~TPvTP)]</u>

<u>Proof</u>: 1)	~T ~F0 ~T ~~ P	Syn ~T ~T ~(~T(~T ~~ P) & ~T ~(~T~~P))	[Df ‘F’, Df,’0']
2)	“	Syn T ~(~T(~T ~~ P) & ~T ~(~T ~~ P))	[1), ~T ~T Syn T]
3)	“	Syn T ~(~T(~~ T P) & ~T ~(~T P))	[2),DN]
4)	“	Syn T ~(~T(~T P) & ~T (T P))	[3),DN]
5)	“	Syn T ~(T P & ~T (T P)	[4), ~T ~T Syn T]
6)	“	Syn T ~(T P & ~T P)	[5),TT Syn T]
7)	“	Syn T (~T P v ~~ T P)	[6),DeM]
8)	“	Syn T (~T P v T P)	[7),DN]
9)	“	Syn (T ~T P v T T P)	[8),Ax.6-4]
10)	“	Syn (F T P v T T P)	[9),Df ‘F’]
11)	“	Syn (F T P v T P)	[10),T6-1]
12)	~T ~F0 ~T ~~ P	Syn (~T P v T P)	[11), Ax.6-1]

Checked by truth-tables:

~T ~F0(~T(~ ~ P))	Syn	~T ~T ~ (~T(~T ~~ P) & ~F(~T~~P))	Syn	(~TP v TP)
TFFTF TF T F 0		**T**FFTT FT TF000 F TF TF000		TF0 **T** F0
TFFTF FT FT T		**T** FFTT TF FTTFT F FT FTTFT		FTT **T** TT
TFFTF TF T F F		**T** FFTT FT TFFTF F TF TFFTF		TFF **T** FF

A wff, A, is a **Normal Form T-wff** (abbr.'TNF') $\underline{Syn}_{df}$
(i) A has no sentential operators other than '&','v' and '∼',
(ii) and negation signs,if any, occur only in elementary wffs,
(iii) 'T' is prefixed to all and only atomic wffs or their negations.[28]

Additional Normal Form Metatheorems for T-wffs follow by putting 'NFT' for 'NF' in their definitions (yielding 'CNFT' from 'CNF', 'DNFT' from 'DNF', 'MOCNFT' from 'MOCNF', 'MinODNFT' from 'MinODNF', etc.). Thus SYN-metatheorems 4 to 7 are converted to Syn-Metatheorems 4_T to 7_T, which read exactly as before except "atomic wff" is replaced by "atomic T-wff" (which is defined as an ordinary atomic wff immediately preceded by 'T' or 'F'); elementary T-wffs are defined as atomic T-wffs and their negations. Proofs of the Syn- Metatheorems 4_T–7_T then follow from DR7-NF. As we shall see, every T-wff, including those in Quantification Theory, can be reduced to a normal form T-wff.

T-operators can be moved inward, but there is no general rule allowing all T-operators to be moved outward from a component to the compound expression. For example, '(TP & ∼TQ)' and 'T(P& ∼TQ)' are synonymous, but there is no way to get the 'T' in '∼TQ' out to the left of the whole conjunction. Only when all conjuncts or disjuncts are prefixed by T or F, is there a synonymous expression in which all truth-operators occur to the left of the whole conjunction or disjunction.

T-operators prefixed to quantifiers can be moved inward until the 'T' is prefixed only to atomic wffs or their negations. If a quantified negation-free wff has a 'T' prefixed to it, by repeated applicatons of Ax.7-3 and/or Ax.7-4 the 'T's can be distributed inward over the conjuncts or disjuncts in any expansion of T(∀x)Px or T(∃x)Px until every expansion is synonymous with the result of prefixing 'T' to all and only its elementary components.

T-operators are moved in or out of the scope of quantifiers by successive uses of Ax.7-3 and Ax.7-4 in the Boolean expansions of the quantifiers. For example, in a domain of 4,

T7-24 [T(∀x)Px Syn (∀x)TPx]
Example: 1) [T(∀x)Px Syn T(Pa_1 & Pa_2 & Pa_3 & Pa_4) (in a domain of 4)
2) [T(∀x)Px Syn ((TPa_1 & TPa_2 & TPa_3 & Tpa_4) [1),Ax.7-3(Gen)]
3) [T(∀x)Px Syn (∀x)TPx [2),Df '∀']

T7-25 [T(∃x)Px Syn (∃x)TPx]
Example: 1) [T(∃x)Px Syn T(Pa_1 v Pa_2 v Pa_3 v Pa_4) (in a domain of 4)
2) [T(∃x)Px Syn ((TPa_1 v TPa_2 v TPa_3 v Tpa_4) [1),Ax.7-4(Gen)]
3) [T(∃x)Px Syn (∃x)TPx [2),Df '∃']

In Quantification theory, for every finite domain, a T-wff which has a T prefixed solely to the whole wff, is synonymous with a similar T-wff with T's prefixed to all disjunctive or conjunctive components, and to a T-wff in which T's are prefixed to all and only its elementary wffs. For example,

1) |= [T(∀x)(∃y)(Rxy & Ryy) Syn (∀x)T(∃y)(Rxy & Ryy)]
2) |= [" Syn (∀x)(∃y)T(Rxy & Ryy)]
3) |= [" Syn (∀x)(∃y)(TRxy & TRyy)]

28. The basic normal form wffs of Chapter 1 to 5 are also converted into normal form wffs of T-logic simply by prefixing 'T' only to atomic wffs, but this would yield only a proper sub-set of A-logic's T-wffs.

In a domain of three, {a,b,c}, the four synonymous wffs in 1), 2) and 3) are expanded as, respectively.

T(((Raa & Raa)v(Rab & Rbb)v(Rac & Rcc)) Syn (T((Raa & Raa)v(Rab & Rbb)v(Rac & Rcc))
&((Rba & Raa)v(Rbb & Rbb)v(Rbc & Rcc)) &T((Rba & Raa)v(Rbb & Rbb)v(Rbc & Rcc))
&((Rca & Raa)v(Rcb & Rbb) v (Rcc & Rcc))) &T((Rca & Raa)v(Rcb & Rbb)v(Rcc & Rcc)))

T(“)Syn ((T(Raa & Raa)vT(Rab & Rbb)vT(Rac & Rcc))
&(T(Rba & Raa)vT(Rbb & Rbb)vT(Rbc & Rcc))
&(T(Rca & Raa)vT(Rcb & Rbb)vT(Rcc & Rcc)))
T(“)Syn (((TRaa & TRaa)v(TRab & TRbb)v(TRac & TRcc))
& ((TRba & TRaa)v(TRbb & TRbb)v(TRbc & TRcc))
& ((TRca & TRaa)v(TRcb & TRbb)v(TRcc & TRcc)))

Without the T-operator there are four basic Quantificational prefixes: (∀x)Px, (∃x)Px, (∀x)~Px, (∃x)~Px, which are synonymous by Q-Interchange with ~(∃x)~Px, ~(∀x)~Px, ~(∃x)Px, and ~(∀x)Px respectively. With T-operators there are eight basic quantificational forms :

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
T(∀x)Px	~F(∀x)Px	T(∃x)Px	~F(∃x)Px	F(∃x)Px	~T(∃x)Px	F(∀x)Px	~T(∀x)Px

These eight forms stand in non-synonymous relationships which are representable in a “Cube of Opposition” (See Section 8.31).

Theorems T7-24 and T7-25 showed how ‘T’ can be moved inside negation-free quantified wffs. Syn-theorems T7-26 to T7-31 show how the remaining basic T-operators (~F, F, and ~T), which require negation signs, can be moved from the left of a quantifier to being prefixed to the matrix, Px, while preserving synonymy.

T7-26 [~F(∀x)Px Syn (∀x)~FPx]
<u>Proof</u>: 1) [~F(∀x)Px Syn ~T~(∀x)Px [T1-11,Df ‘F’]
2) [~F(∀x)Px Syn ~T(∃x)~Px [1),T4-20]
3) [~F(∀x)Px Syn ~(∃x)T~Px [2),T7-25]
4) [~F(∀x)Px Syn (∀x)~T~Px [3),T4-21]
5) [~F(∀x)Px Syn (∀x)~FPx [4),Df ‘F’]

T7-27 [~F(∃x)Px Syn (∃x)~FPx]
<u>Proof</u>: 1) [~F(∃x)Px Syn ~T~(∃x)Px [T1-11,Df ‘F’]
2) [~F(∃x)Px Syn ~T(∀x)~Px [1),T4-21]
3) [~F(∃x)Px Syn ~(∀x)T~Px [2),T7-24]
4) [~F(∃x)Px Syn (∃x)~T~Px [3),T4-20]
5) [~F(∃x)Px Syn (∃x)~FPx [4),Df ‘F’]

T7-28 [F(∃x)Px Syn (∀x)FPx]
<u>Proof</u>: 1) [F(∃x)Px Syn T~(∃x)Px [T1-11,Df ‘F’]
2) [F(∃x)Px Syn T(∀x)~Px [1),T4-21]
3) [F(∃x)Px Syn (∀x)T~Px [2),T7-24]
4) [F(∃x)Px Syn (∀x)FPx [3),Df ‘F’]

T7-30 [~ T(∃x)Px Syn (∀x) ~ TPx]
Proof: 1) [~ T(∃x)Px Syn ~ (∃x)T(Px) [T7-25,DR4-1]
2) [~ T(∃x)Px Syn (∀x) ~ TPx [2),T4-21]

T7-29 [F(∀x)Px Syn (∃x)FPx]
Proof: 1) [F(∀x)Px Syn T ~ (∀x)Px [T1-11,Df 'F']
2) [F(∀x)Px Syn T(∃x) ~ Px [1),T4-20]
3) [F(∀x)Px Syn (∃x)T ~ Px [2),T7-25]
4) [F(∀x)Px Syn (∃x)FPx [3),Df 'F']

T7-31 [~ T(∀x)Px Syn (∃x) ~ TPx]
Proof: 1) [~ T(∀x)Px Syn ~ (∀x)TPx [T7-24,DR4-1]
2) [~ T(∀x)Px Syn (∃x) ~ TPx [1),T4-20]

To summarize:

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
T(∀x)Px	~ F(∀x)Px	T(∃x)Px	~ F(∃x)Px	F(∃x)Px	~ T(∃x)Px	F(∀x)Px	~ T(∀x)Px
Syn	Syn	Syn	Syn	Syn	Syn	Syn	Syn
(∀x)TPx	(∀x) ~ FPx	(∃x)TPx	(∃x) ~ FPx	(∀x)FPx	(∀x) ~ TPx	(∃x)FPx	(∃x) ~ TPx
[T7-24]	[T7-26]	[T 7-25]	[T7-27]	[T7-28]	[T7-30]	[T7-29]	[T7-31]

Note that in (1) to (4) Truth and non-Falsehood move through a quantifier without changing it, while in (5) to (8) Falsehood and non-Truth change the quantifier when they move.

As T7-24 to T7-31 show, any quantified wff with a T-operator prefixed to it can be replaced by a synonymous quantification with no T-operators prefixed to a quantifier, but with a T-operator prefixed to the matrix instead. The matrices in turn can be replaced by synonymous matrices which have T-operator prefixes to all and only the atomic components. The whole wff can then be put in prenex normal form, by the methods laid out in Chapters 4 and 5. The resulting T-wffs are the normal form T-wffs with quantifiers for analytic truth-logic.

As an example, suppose we have 'T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz'. This is proven Syn to the wff, (∀x)(∃y)(∀w)(∃z) ~ TRxywz, a 1st-level wff of type (ii), as follows:

|= T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)(∀w)(∀z) ~ TRxywz
Proof:
1) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)T ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz [T7-24]
2) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)**F**F(∃y)F(∀w) ~ T(∀z) ~ TRxywz [1),Df 'F']
3) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x) **~ F**(∃y)F(∀w) ~ T(∀z) ~ TRxywz [2),T7-17]
4) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)**(∃y) ~ F**F(∀w) ~ T(∀z) ~ TRxywz [3),T7-27]
5) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)- **~ F**(∀w)T(∀z) ~ TRxywz [4),T7-17]
6) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)**F**(∀w)T(∀z) ~ TRxywz [5),DN]
7) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)**(Ew)F**T(∀z) ~ TRxywz [6),T7-29]
8) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)(Ew) **~ T**(∀z) ~ TRxywz [7),Ax.7-2]
9) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)(∀w)**(∃z) ~ T**TRxywz [8),T7-31]
10) T(∀x) ~ F(∃y)F(∀w) ~ T(∀z) ~ TRxywz Syn (∀x)(∃y)(∀w)(∃z) **~ T**Rxywz [9),7-4]

In some cases the matrix may have T-operators that can not be eliminated from a conjunction or disjunction by distributing them outward by Axiom 7-03 and 7-04 (e.g., (~ TP & TP) or (~ TP v TP)).

But T-operators prefixed to a compound wff can always be moved inward and prefixed only to elementary components within the compound.

Besides Syn-theorems for T-wffs, containment theorems can be derived from CONT-theorems in Chapter 3 by DR7-1d, [If P CONT Q, then TP Cont TQ], and the components can be reduced to a form with T prefixed to atomic wffs by the principles just mentioned:

1)	[T(∀x)(Px v Qx) Cont T((∃x)Px v (∀x)Qx)]	[T3-39,R7-1]
2)	[(∀x)T(Px v Qx) Cont T((∃x)(Px v (∀x)Qx)]	[1),T7-24,SynSUB]
3)	[(∀x)T(Px v Qx) Cont (T(∃x)(Px v T(∀x)Qx)]	[2),Ax,7-4,SynSUB]
4)	[(∀x)(TPx v TQx) Cont (T(∃x)Px v T(∀x)Qx)]	[3),Ax.7-4,SynSUB]
5)	[(∀x)(TPx v TQx) Cont ((∃x)TPx v T(∀x)Qx)]	[4),T7-25,SynSUB]
6)	[(∀x)(TPx v TQx) Cont ((∃x)TPx v (∀x)TQx)]	[5),T7-24,SynSUB]

The class of normal form T-wffs will be expanded further to include wffs with C-conditionals after the C-conditional is re-introduced in Chapter 8.

7.42123 Other Syn- and Cont-theorems from Axioms 7-2 to 7-4

Before considering the consequence of Ax.7-5, we present five consequnces of Axioms 7-1 to 7-4, which diverge from M-logic and will be used in subsequent discussions.

First, with regard to the role of the value 0, 'neither true nor false' and its contradictory, 'either true or false' (exclusively) in truth logic.

It is true, and not inconsistent, to say that many zero-level expressions are neither true nor false. But a 1st-level T-expression cannot be neither true nor false. It must be either true or false exclusively. To say a 1st-level truth-claim is not false, is a 2nd-level assertion that is synonymous with a 2nd-level assertion that the truth-claim is true.

T7-32. [∼FTP Syn TTP] .: Inc[∼F(TP) & ∼T(TP)] Syn Inc[0(TP)]

Proof:	1) FTP Syn ∼TP	[Ax.7-02]
	2) ∼FTP Syn ∼∼TP	[1),DR4-1]
	3) ∼FTP Syn TP	[2),DN,SynSUB]
	4) ∼FTP Syn TTP	[3),T7-20,SynSUB]

Similarly, to say that a 1st-level falsity-assertion is not false is a 2nd-level assertion synonymous with the 2nd-level assertion that the 1st-level falsity assertion is true. It follows that it is inconsistent to say that "Either P is true or P is false" is neither true nor false, though it is not inconsistent to say [Either P or not-P] is neither true nor false. (See Section 5.212 on Inc-theorems). These consequences follow from Axiom 7-02.

T7-33. (∼FFP Syn TFP) .: Inc[∼F(FP) & ∼T(FP)] .: Inc[0(FP)]

Proof:	1) FT∼P Syn ∼T∼P	[Ax.7-02,U-SUB]
	2) ∼FT∼P Syn ∼∼T∼P	[1), DR4-1]
	3) ∼FFP Syn ∼∼ FP	[2),Df 'F']
	4) ∼FFP Syn FP	[3),DN]
	5) ∼FFP Syn TFP	[4),T7-21]

However the terms 'T(TP)' and 'T(FP)' are 2nd-level, merely *de dicto*, expressions, and for purpose of genuine truth-searches, they reduce to the 1st-level *de re* expressions 'TP' and 'FP' respectively.

Tarski's claim that P is intersubstitutable with TP, is replaced by T7-20, |= [TP Syn TTP]; "it is true that P is true" says the same as "it is true that P". Similarly, by Ax.7-19, "it is true that P is false" is the synonymous with "P is false". Prefixing 'T' to any T-wff adds nothing to its meaning.

Secondly, although M-logic does not distinquish between ''T(P ⊃ Q)' and '(TP ⊃ TQ)', a distinction is made in A-logic. With the TF-conditional; T(P ⊃ Q) contains (TP ⊃ TQ) but the converse fails. Consequently, [T(P ⊃ Q) ⊃ (TP ⊃ TQ)] is tautologous, but [(TP ⊃ TQ) ⊃ T(P ⊃ Q)] is not. This is shown by T7-34:

T7-34. [T(P ⊃ Q) Cont (TP ⊃ TQ)]

Proof:	1) [(FP v TQ) Syn (FP v TQ)]	[T1-11]
	2) [(FP v TQ) Syn ((FP & ~TP) v TQ)]	[1),T7-14,SynSUB]
	3) [(FP v TQ) Syn ((FP v TQ) & (~TP v TQ))	[2),v&-DIST]
	4) [(FP v TQ) Cont (~TP v TQ))	[3),Df 'Cont']
	5) [(T~P v TQ) Cont (~TP v TQ)]	[4),Df 'F']
	6) [(T(~P v Q) Cont (~TP v TQ)]	[5),Ax.7-4]
	7) [(T(P ⊃ Q) Cont (TP ⊃ TQ)]	[6),T4-31,(twice)]

From T7-34 Taut[T(P ⊃ Q) ⊃ (TP ⊃ TQ)] follows. But interestingly, the converse of T7-34 does not hold, and its '⊃'-for-'Cont' analogue is not tautologous since it can be false: The converse of T7-34 is [(TP ⊃ TQ) Cont T(P ⊃ Q)]; its '⊃'-for-'Cont' analogue (TP ⊃ TQ) ⊃ T(P ⊃ Q)]

F0 **T** FF **F** 0 0 F F0 **T** FF **F** **F** 0 0 F

Thirdly, we saw in T4-32 that [(P ⊃ Q) SYN (~Q ⊃ ~P)] is a theorem of A-logic, yielding TAUT[(P ⊃ Q) ≡ (~Q ⊃ ~P)] as a theorem of M-logic. But the distinction between FP and ~TP requires differentiating several versions of transposition principles; some versions, even with the TF-conditional, are not tautologous.

Two versions of transposition with TF-conditionals, hold:

T7-35. [T(P ⊃ Q) Syn T(~Q ⊃ ~P)]

Proof:	1) [~(P & ~Q) SYN ~(~Q & ~~P)]	[T1-11,&-COMM,DN]
	2) [(P ⊃ Q) SYN (~Q ⊃ ~P)]	[1),Df '⊃' (twice)]
	3) [T(P ⊃ Q) Syn T(~Q ⊃ ~P)]	[2),R7-1]

and,

T7-36. [(TP ⊃ TQ) Syn (~TQ ⊃ ~TP)]

Proof:	1) [~(P & ~Q) SYN ~(~Q & ~~P)]	[T1-11,&-COMM,DN]
	2) [(P ⊃ Q) SYN (~Q ⊃ ~P)]	[1),Df '⊃' (twice)]
	3) [(TP ⊃ TQ) SYN (~TQ ⊃ ~TP)]	[2),U-SUB]

But some formulations of transposition (or contraposition) with the TF-conditional, do not have comparable CONT-theorems. E.g., when P has the value 0 "If P is true then Q is true" does not logically contain "If Q is false then P is false". The converse containment fails when Q has the value 0, as truth-table principles show.

[(TP ⊃ TQ) Cont (FQ ⊃ FP)]	[(FQ ⊃ FP) Cont (TP ⊃ TQ)]
F0 **T** FF X TF **F** F0	F0 **T** FT X TT **F** F0
[(TP ⊃ TQ) ⊃ (FQ ⊃ FP)]	[(FQ ⊃ FP) ⊃ (TP ⊃ TQ)]
F0 **T** FF **F** TF **F** F0	F0 **T** FT **F** TT **F** F0

More will be said about transposition principles in the discussion of C-conditionals in truth-logic. C-cconditionals require even more restrictions on transposition. This is not bad. The failure of certain versions of transposition is a necessary prequisite for a theory of C-conditionals which will solve the problems raised by the TF-conditional (See Section 9.41). The versions of transposition which hold are quite sufficient for those purposes for which transposition is useful.

Fourth, there are various Syn-theorems and Cont-theorems, which are closely related to principles of the truth-tables for '∼', '&' and 'v'.

From Ax.7-3. [T(P & Q) Syn (TP & TQ)] and T7-23. F(P v Q) Syn (FP & FQ) rules for rows in the two- and three-valued truth-tables for conjunctions, disjunctions, and conditionals in M-logic and in A-logic follow. The rule for the fifth row of the truth-table for conjunction is "If P is true and Q is true then (P&Q) is true". This is derivable in the next chapter, from Axiom 7-3. Theorem T7-23, above is the basis for the rule for the ninth row of the disjunction table: "If P is false and Q is false, then (PvQ) is false." Cont-theorems which help to establish rules of the 5th row of the disjunctive truth-table, and the 9th row of the conjunction table are:

T7-37 (FP & FQ) Cont F(P & Q) [Cf. 9th Row, truth-table of '&']

Proof: 1) (FP & FQ) Cont (FP v FQ) [T7-38,U-SUB]
2) (FP & FQ) Cont (T∼P v T∼Q) [1),Df 'F']
3) (FP & FQ) Cont T(∼Pv∼Q) [2),Ax.7-4,SynSUB]
4) (FP & FQ) Cont T∼(P&Q) [3),T4-18,SynSUB]
5) (FP & FQ) Cont F(P&Q) [4),Df 'F']

T7-38. (TP & TQ) Cont T(PvQ) [Cf. 5th Row, truth-table of 'v']

Proof: 1) (TP & TQ) Cont (TP v TQ) [T1-38,U-SUB]
2) (TP & TQ) Cont T(P v Q) [1),Ax.7-4,SynSUB]

T7-39. [(FP & TQ) Cont T(P ⊃ Q)] [Cf.6th Row,table of '⊃']

Proof: 1) (T∼P & TQ) Cont T(∼P v Q) [T7-38,U-SUB]
2) (FP & TQ) Cont T(∼P v Q) [1),Df 'F']
3) (FP & TQ) Cont T(P ⊃ Q) [3),T4-31,SynSUB]

T7-40. [(TP & FQ) Syn F(P ⊃ Q)] [Cf. 8th Row, truth-table of '⊃']

Proof: 1) (TP & T∼Q) Syn T(P & ∼Q) [Ax.7-3,U-SUB]
2) (TP & FQ) Syn T(P & ∼Q) [1),Df 'F']
3) (TP & FQ) Syn T∼∼(P & ∼Q) [2),DN]
4) (TP & FQ) Syn F∼(P & ∼Q) [3),Df 'F']
5) (TP & FQ) Syn F(P ⊃ Q) [4),Df '_']

Some proofs of logical falshood and logical truth in Section 7.4223 are based T7-41 and T7-42:[29]

T7-41. [F(TP & ∼TQ) Syn ∼T(TP & ∼TQ)]

Proof:	1) [F(TP & ∼TQ) Syn F(TP & ∼TQ)]	[T1-11,U-SUB]
	2) [" Syn F(TP & FTQ)]	[1),Ax.7-2,SynSUB]
	3) [" Syn F(TTP & FTQ)]	[2),T7-20,SynSUB]
	4) [" Syn F(TTP & T∼TQ)]	[3),Df 'F',SynSUB]
	5) [" Syn FT(TP & ∼TQ)]	[4),Ax.7-3,SynSUB]
	6) [F(TP & ∼TQ) Syn ∼T(TP & ∼TQ)]	[5),T7-02,SynSUB]

T7-42. [T(∼TP v TQ) Syn ∼F(∼TP v TQ)]

Proof:	1) [F(TP & ∼TQ) Syn ∼T(TP & ∼TQ)]	[T7-41]
	2) [F∼(∼TP v TQ) Syn ∼T∼(∼TP v TQ)]	[1),T4-16,(DeM),SynSUB(twice)]
	3) [F∼(∼TP v TQ) Syn ∼F(∼TP v TQ)]	[2),Df 'F',SynSUB]
	4) [T∼∼(∼TP v TQ) Syn ∼F(∼TP v TQ)]	[3),Df 'F',SynSUB]
	5) [T(∼TP v TQ) Syn ∼F(∼TP v TQ)]	[4),DN,SynSUB]

Many other Syn- and Cont-theorems are derivable which express intuitive truth-functional notions concerning truth-functional dependence including:

|= [F(P&Q) Syn (FP v FQ)]	"(P & Q) is false" means "P is false or Q is false"
|= [∼T(P&Q) Syn (∼TP v ∼TQ)]	"(P&Q) is not true" means "either P is not true or Q is not true"
|= [∼F(P&Q) Syn (∼FP & ∼FQ)]	"(P & Q) is not false" means "P is not false and Q is not false"
|= [∼T(PvQ) Syn (∼TP & ∼TQ)]	"(PvQ) is not true" means "both P is not true and Q is not true"
|= [∼F(PvQ) Syn (∼FP v ∼FQ)]	"(P v Q) is not false" means "P is not false or Q is not false"
|= [T(P ⊃ Q) Syn (FP v TQ)]	"(P ⊃ Q) is true" means "Either P is false or Q is true"

Finally, the theorem T7-43 [T(∀x)(Px ⊃ Qx) Cont (T(∀x)Px ⊃ T(∀x)Qx] is of special interest because of its relation to Quine's *101, with T-wffs. This is frequently used as an axiom of quantification theory. It derivation requires requires Ax.7-1 as well as Ax.7-4.

T7-43. [T(∀x)(Px ⊃ Qx) Cont (T(∀x)Px ⊃ T(∀x)Qx)]

Proof:	1) [T(∀x)(Px ⊃ Qx) Cont (T((∀x)Px ⊃ (∀x)Qx]	[T4-29,DR7-1d]
	2) [((∀x)Px ⊃ (∀x)Qx) Syn (∼(∀x)Px v (∀x)Qx]	[T4-24,U-SUB]
	3) [T(∀x)(Px ⊃ Qx) Cont (T(∼(∀x)Px v (∀x)Qx]	[1),2),SynSUB]
	4) [T(∀x)(Px ⊃ Qx) Cont (T∼(∀x)Px v T(∀x)Qx]	[3),Ax.7-04,SynSUB]
	5) [T(∀x)(Px ⊃ Qx) Cont (F(∀x)Px v T(∀x)Qx]	[4),Df 'F']
	6) [F(∀x)Px) Syn (F(∀x)Px & ∼T(∀x)Px)]	[T7-14,U-SUB]
	7) [(T∼(∀x)Px) Syn (F(∀x)Px & ∼T(∀x)Px)]	[6),Df 'F',SynSUB]

---

29. From T7-41 and T7-42 many others can be derived including,

|= [T(∼FP v FQ) Syn ∼F(∼FP v FQ)]	[T7-42,U-SUB'∼P' for'P','∼Q'for'Q',Df 'F']
|= [T(FP & ∼FQ) Syn ∼F(FP & ∼FQ)]	[T7-41,U-SUB'∼P for'P','∼Q'for'Q',Df 'F']
|= [T(TP & ∼TQ) Syn ∼F(TP & ∼TQ)]	[T7-41,DR1-01,R4-3,DN]
|= [F(∼FP v FQ) Syn ∼T(∼FP v FQ)]	[T7-42,DR1-01,R4-3,DN]
|= [F(∼TP v TQ) Syn ∼T(∼TP v TQ)]	[T7-42,DR1-01,R4-3,'∼P for'P','∼Q'for'Q',DN]
|= [F(FP & ∼FQ) Syn ∼T(FP & ∼FQ)]	[T7-41,DR1-01,R4-3,'∼P for'P','∼Q'for'Q',DN]

8) [T(∀x)(Px ⊃ Qx) Syn ((F(∀x)Px & ~T(∀x)Px) v T(∀x)Qx] [3),7),SynSUB]
9) [T(∀x)(Px ⊃ Qx) Syn ((F(∀x)Px v T(∀x)Qx) & (~T(∀x)Px v T(∀x)Qx)] [8),v&-DIST]
10) [T(∀x)(Px ⊃ Qx) Cont (~T(∀x)Px v T(∀x)Qx)] [9),Df 'Cont']
11) [T(∀x))Px ⊃ Qx) Cont (T(∀x)Px ⊃ T(∀x)Qx)] [10),T4-31,SynSUB]

Using T7-24 three times with SynSUB we get,

|= [(∀x)T(Px ⊃ Qx) Cont ((∀x)TPx ⊃ (∀x)TQx)] [13),T4-24,SynSUB]

which is closedly related to Quine's *101.

7.42124 Cont-theorems for Detachment from Ax.7-5

In M-logic (P & (~PvQ)) M-implies Q. This is because '(P& (~PvQ) & ~Q)' is inconsistent and thus '(P& (~PvQ) ⊃ Q)' is a tautology and therefore M-valid. But in A-logic '(P& (~PvQ))' does not logically Contain Q; thus neither [(P& (~PvQ)) **.:** Q] nor [(P& (~PvQ)) => Q] are A-valid.

However, if extra-logical predicate operators such as T or F, or INC or TAUT are prefixed, related principles are can be justified in A-logic from the meanings of the operators.

The primary rule of inference for M-logic is Taut-detachment (and/or Inc-Detachment):

TAUT-Det. If TAUT[P] and TAUT[P ⊃ Q], then TAUT[Q] [Quine's *104]
INC-Det. If INC[P] and INC[~P & Q)] then INC[Q].

These principles sanction the passage from two TAUT-theorems (M-theorems) to a third TAUT-theorem or from two INC-theorems to a third INC-theorem . In A-logic they apply to predicates as well as sentences, and commands as well as propositions.

There are also principles of logical inference for passing from assertions of the factual truth of two or more sentences to an assertion of factual truth of another sentence. Ax.7-5 is peculiar to truth-logic, and by prefixing T-operators it can justify the rules of detachment that M-logic associates with Modus Ponens. These include both "If (TP & T(P ⊃ Q)] then TQ" and "If (TP & (TP ⊃ TQ) then TQ" (which in A-logic are different). Either or both of these are usually called "modus ponens" in M-logic. But with TF-conditionals, they are synonymous with, and best understood as, variations of the Alternative syllogism: "If TP and T(~PvQ) then TQ", and "If TP and (~TPvTQ) then TQ".

We prove synonyms of |= [(TP & T(P ⊃ Q)] Cont TQ] and |= [(TP & (TP ⊃ TQ)) Cont TQ] in this chapter. We also prove, T7-45. [((~TP v TQ) & ~TQ) Cont ~TP], which is related to the Alternative Syllogism and is equivalent to |= [((TP ⊃ TQ) & ~TQ) Cont ~TP] which is related to M-logic's version of "Modus Tollens". These theorems, and others in this section, are all derived from Ax.7-5. In Chapter 8 they yield Validity-theorems with the main "If…then" a C-conditional.

Ax.7-5, [((TP & ~TP) v TQ) Cont TQ], expresses a kind of containment which is peculiar to truth-statements. In general (PvQ) does not contain Q or P, and without the T-operator prefixed to Q, ((TP & ~TP) v Q) does not contain Q. But with the 'T' prefixed to Q (or to the whole disjunction as in

---

30. |= [((TP & ~TP) v TQ) Syn T((TP & ~TP) v Q)]
Proof: 1) ((TP & ~TP) v TQ) Syn ((TP & ~TP) v TQ) [T1-11,U-SUB]
2) ((TP & ~TP) v TQ) Syn ((TP & **T~TP**) v TQ) [1),Ax.7-2,Df 'F',SynSUB]
3) ((TP & ~TP) v TQ) Syn ((**TTP** & T~TP) v TQ) [2),T7-14,SynSUB]
4) ((TP & ~TP) v TQ) Syn (**T(**TP & ~TP**)** v TQ) [3),T7-03,SynSUB]
5) ((TP & ~TP) v TQ) Syn **T(**(TP & ~TP) v Q**)** [4),T7-04,SynSUB]

'T((TP & ∼TP)vQ)' (which is Syn to ((TP&∼TP)vTQ)) we get a different result.[30] 'T((TP & ∼TP)vQ)' contains TQ, by the meaning of T. Because the concept of truth presupposes an objective domain of which no inconsistent statement can be true, and since '(TP&∼TP)' is inconsistent, a statement of the form ((TP&∼TP)vQ), can only be true if Q is true. Thus to say that ((TP&∼TP)vQ) is true, entails that Q is true. Note that in Axiom 7-5, TQ occurs POS in the antecedent and consequent, so it satisfies the requirement that if B is contained in A, then B has no components which occur POS (or NEG) in B unless they occur POS (or NEG) in A. All of the theorems derived from Ax.7-5 have this feature. All are based on the notion, contained in the concept of truth, of a field of reference in which at least one consistent statement is true and which has no inconsistent facts or events. The detachment principles are derived from Ax.7-5 as follows:

T7-44. [(TP & (∼TP v TQ)) Cont TQ] "Alternative Syllogism #1"

Proof: 1) [(P & (Q v R)) CONT ((P & Q) v R)] [T1-39]
2) [(TP&(∼TPvTQ)) Cont ((TP&∼TP)vTQ)] [1),U-SUB]
3) [((TP & ∼TP) v TQ) Cont TQ] [Ax.7-5]
4) [(TP & (∼TPvTQ)) Cont TQ] [2),3),CCC-Syll]

Using the definition of '⊃', this becomes, through T4-31,

T7-44⊃. [(TP & (TP ⊃ TQ)) Cont TQ] [T7-44,T4-31,SynSUB]

which will yield a Validity theorem which may be called "TF-Modus Ponens", since it is called "Modus Ponens" in M-logic.

Another version of the alternative syllogism is,

T7-45 [((∼TP v TQ) & ∼TQ) Cont ∼TP] "Alternative Syllogism #2"

Proof: 1) [(P & (Q v R)) Cont ((P & Q) v R)] [T1-39]
2) (∼TQ & (TQ v ∼TP)) Cont ((∼TQ & TQ) v ∼TP) [T1-39,U-SUB]
3) ((TQ & ∼TQ) v ∼TP) Cont ∼TP [Ax.7-5]
4) (∼TQ & (∼TP v TQ)) Cont ∼TP [2),3),CCC-SYLL]
5) ((∼TP v TQ) & ∼TQ) Cont ∼TP [4),&-COMM]

U-SUB on T1-39 in this case involves introducing contradictories where there were none. But this is OK since Unrestricted U-SUB in Cont-theorems, preserves CONT, and CONT preserves consistency—if the antecedent is consistent, then the consequent is, since the consequent has no more, and maybe less, conjuncts than the Antecedent.

Using the definition of '⊃' (through T4-31) on T7-45 it becomes,

T7-45⊃. [((TP ⊃ TQ) & ∼TQ) Cont ∼TP] [T7-45,T4-31,SynSUB]

which may be called "TF-Modus Tollens", since it would be called "Modus Tollens" in M-logic.

A weaker version of Alternative syllogism yields a weaker version of TF-Modus Ponens.

T7-46. [(TP & T(∼PvQ)) Cont TQ] "Alternative Syllogism #3"

Proof: 1) T(P ⊃ Q) Cont (TP ⊃ TQ) [T7-34]
2) T(∼PvQ) Cont (∼TPvTQ)) [1),T4-31,SynSUB(twice)]
3) (TP & T(∼PvQ)) Syn (TP & (T(∼PvQ)) [T1-11,U-SUB]

4) (TP & T(~PvQ)) Cont (TP & (~TPvTQ)) [3),2),SynSUB]
5) (TP & (~TPvTQ)) Cont TQ [T7-44]
6) (TP & T(~PvQ)) Cont TQ [4),5),CCC-Syll]

From this “TF-Modus Ponens #2” may be derived, using T4-31:

T7-46⊃. [(TP & T(P ⊃ Q)) Cont TQ] [T7-46,T4-31,SynSUB]

Correspondingly, there is a stronger version of the Alternative syllogism, which yields a stronger form of TF-Modus Tollens:

T7-47. [(T(~P v Q) & FQ) Cont FP] “Alternative Syllogism #3”
<u>Proof</u>: 1) [(P & (Q v R)) Cont ((P & Q) v R)] [T1-39]
2) (~Q & (Q v ~P)) Cont ((~Q & Q) v ~P) [T1-39,U-SUB]
3) T(~Q & (Q v ~P)) Cont T((~Q & Q) v ~P) [2),R7-1]
4) ((T~Q & T(Q v ~P)) Cont (T(~Q & Q) v T~P) [Ax.7-5]
5) (T(~Q & Q) v T~P) Cont T~P) [2),3),CCC-SYLL]
6) ((T(Q v ~P) & T~Q) Cont T~P [5),&-COMM]
7) (T(~P v Q) & T~Q) Cont T~P [6),v-COMM]
8) (T(~P v Q) & FQ) Cont FP [7),Df ‘F’(twice)]

From this “TF-Modus Tollens #2” may be derived, using T4-31:

T7-44⊃. [(T(P ⊃ Q) & FQ) Cont FQ] [T7-47,T4-31,SynSUB]

Many other interesting detachment-Theorems can be derived which are related to principles of propositional logic in Stoic Logic, Traditional Logic and M-logic. But we defer discussion of them until the next chapter when they will be presented as valid theorems of C-conditional inference in a truth-logic.

7.42125 Theorems about Expressions Which are Neither True nor False; from Df ‘0’

We use the symbol ‘0’ for the predicate, ‘<1> is not true and <1> is not false’, as in “P is neither true nor false”. In M-logic’s semantics this is an inconsistent predicate; it is never true and always false of any proposition (since a proposition by definition is either true or false exclusively[31]). In A-logic it may be predicated truly of many expressions, including predicates, questions, directives and certain categories of indicative statements and conditionals. A-logic is intended to cover these expressions as well as propositions. The task is to explain how, with the T-operator, we can talk consistently about expressions which are neither true nor false, as well as about others which are either true or false and can’t be both, and tell what follows logically in each case. The starting point is that ‘P is not true’ does not mean the same thing as ‘P is false’, and that ‘P is not false’ does not mean the same thing as ‘P is true’ although T7-13, TP Cont ~FP and T7-15, FP Cont ~TP hold.

---

31. “We call propositions those [sentences]only that have truth or falsity in them. A prayer is, for instance, a sentence but neither has truth nor falsity.” Aristotle, *On Interpretation*, 17a. (Loeb Lib)

The third value makes the TF-conditionals related to the converses false, hence the converse containments can not hold. If P in fact is 0, then to say it is not true does not entail or imply it is false, and to say it is not false does not entail or imply that it is true. The truth-tables show that the consequent can be true when the antecedent is not. Thus '∼TP' does not contain 'FP' and '∼FP does not Contain 'TP'.[32]

(∼TP ⊃ FP)	(∼FP ⊃ TP)
T F0 **F** F0	T F0 **F** F0

The next point is that the assertion that P is neither true nor false, symbolized by '0P' or '(∼TP&∼FP)', says that propositionality, or bivalence, (the traditional "Law of the Excluded Middle") does not always hold of P. It is synonymous with saying that P is not either true or false, ∼(TP v FP). This is true if P is an unsaturated predicate, an interrogatory sentence or a directive.

T7-48. [0P Syn ∼(TP v FP)]
<u>Proof</u>: 1) 0P Syn (∼TP & ∼FP) [Df '0']
2) 0P Syn ∼(TP v FP) [1), DeM]

Propositionality (or bivalence) can be attributed to an expression. Doing this is the same as denying that it is neither true nor false, i.e., with asserting that is either true or false.

T7-49. [∼0P Syn (TP v FP)]
<u>Proof</u>: 1) 0P Syn (∼TP & ∼FP) [Df.'0']
2) ∼0P Syn ∼(∼TP & ∼FP) [DR4-7]
3) ∼0P Syn (TP v FP) [2),Df 'v']

Two theorems which underly rules for the truth-table of 0;

T7-50 [0P Cont FTP] See 2nd row of truth-table for [0P]
<u>Proof</u>: 1) 0P Syn (∼TP & ∼FP) [Df.'0']
2) 0P Cont ∼TP [1),Df 'Cont']
3) 0P Cont FTP [2),Ax.7-1,SynSUB]

T7-51 [0P Cont FFP] See 3rd row of truth-table for [0P]
<u>Proof</u>: 1) 0P Syn (∼TP & ∼FP) [Df.'0']
2) 0P Cont ∼FP [1),Df 'Cont']
3) 0P Cont FFP [2),T7-17,SynSUB]

If P is neither true nor false, then its negation, ∼P, is neither true nor false.

T7-52 [0∼P Syn 0P]
<u>Proof</u>: 1) 0P Syn (∼TP & ∼FP) [Df.'0']
2) 0∼P Syn (∼T∼P & ∼F∼P) [1),U-SUB]
3) 0∼P Syn (∼FP & ∼TP) [2),Df 'F',DN]
4) 0∼P Syn (∼TP & ∼FP) [3),&-COMM]
5) 0∼P Syn 0P [1),4),SynSUB]

0∼P	0P
T00	**T**0
FFT	**F**T
FTF	**F**F

32. Rather, ∼TP Implies (FP v 0P) and ∼FP Implies (TP v OP).
Hint: (∼FP & ∼0P) Syn (∼FP & (TP v FP)), (∼FP & (TP v FP)) Cont TP,
.: ∼TP Impl (FP v 0P). See Section 7.423.

T7-53. [0P Syn T(0P)]

Proof: 1) 0P Syn (FTP & ~FP) [Df '0',Ax.7-02,SynSUB]
2) 0P Syn (FTP & FFP) [1),T7-17,SynSUB]
3) 0P Syn (T~TP & T~FP) [2),Df 'F']
4) 0P Syn T(~TP & ~FP) [3),Ax.7-3]
5) 0P Syn T(0P) [4),Df '0']

T7-54 . [~F(0P) Syn 0P]

Proof: 1) ~F(0P) Syn ~F(~TP & ~FP) [Df '0']
2) " Syn ~T~(~TP & ~FP) [1),Df 'F']
3) " Syn ~T(TP v FP) [2),Df 'v']
4) " Syn ~(TTP v TFP) [3,Ax.7-4]
5) " Syn ~(TP v TFP) [4),T7-20]
6) " Syn ~(TP v FP) [5),T7-21]
7) ~F(0P) Syn (~TP & ~FP) [6),DeM]
8) ~F(0P) Syn 0P [7),Df '0']

T7-55 [(0P v TP) Cont ~FP]

Proof: 1) (0P v TP) Syn (0P v TP) [T1-11,U-SUB]
2) (0P v TP) Syn ((~TP & ~FP) v TP) [1),Df '0']
3) (0P v TP) Syn (~TP & ~FP) v (TP & ~FP) [2),Ax.7-1,SynSUB]
4) (0P v TP) Syn ((~TP v TP) & ~FP) [3),v&-DIST]
5) (0P v TP) Cont ~F(P) [4),Df 'Cont']

T7-56 [(0P v FP) Cont ~TP]

Proof: 1) ((0~P v T~P) Cont ~F~P) [T7-55, U-SUB]
2) ((0~P v FP) Cont ~F~P) [1),Df 'F']
3) ((0P v FP) Cont ~F~P)) [2),T7-52]
4) ((0P v FP) Cont ~TP)) [3),T7-16]

There are several interesting theorems concerning the distribution of '0' over conjunction and disjunction. These provide grounds for certain rules of the truth-tables of '&', 'v' and '⊃'.

T7-57. (0P & 0Q) Syn (0(P&Q) & 0(PvQ))

Proof: 1) (0P & 0Q) Syn ((~TP & ~FP) & (~TQ & ~FQ)) [Df '0'(twice)]
2) " Syn ((~TP & ~TQ) & (~FP & ~FQ)) [1),&-ORD]
3) " Syn ~(TP v TQ) & ~(FP v FQ)) [2),DeM(twice)]
4) " Syn ~(TP v TQ) & ~(T~P v T~Q) [3),Df 'F]
5) " Syn ~T(P v Q) & ~T(~P v ~Q)) [4),Ax.7-4(twice)]
6) " Syn ~T(P v Q) & ~T~(P & Q)) [5),DeM]
7) (0P & 0Q) Syn ~T(P v Q) & ~F(P & Q)) [6),Df 'F']
8) (0~P & 0~Q) Syn ~T(~Pv~Q) & ~F(~P&~Q)) [7),U-SUB]
9) (0~P & 0~Q) Syn ~F(P&Q) & ~T(PvQ) [8),DeM]
10) (0P & 0Q) Syn ~F(P&Q) & ~T(PvQ) [9),T7-52,SynSUB(twice)]
11) (0P & 0Q) Syn ~T(PvQ) & ~F(P&Q) & ~F(P&Q) & ~T(PvQ) [7) and 10)]
12) (0P & 0Q) Syn ((~T(P&Q) & ~F(P&Q)) & (~T(PvQ) & ~F(PvQ))) [11),&-Ord]
13) (0P & 0Q) Syn (0(P&Q) & 0(PvQ)) [12),Df '0'(twice)]

The following Cont-theorems will be used in Section 7.4233 as the basis of some of the rules for trivalent truth-tables.

T7-58. (0P & 0Q) Cont 0(P & Q) [Cf. 1st Row, truth-table of '&']
Proof: T7-57, Df 'Cont'

T7-59. (0P & 0Q) Cont 0(P v Q) [Cf. 1st Row, truth-table of 'v']
Proof: T7-57, Df 'Cont'

T7-60. (0P & 0Q) Cont 0(P ⊃ Q) [Cf. 1st Row, truth-table of '⊃']
Proof: 1) (0∼P & 0Q) Cont (0(∼P v Q) [T7-59,U-SUB]
2) (0P & 0Q) Cont (0(∼P v Q) [1),T7-52,U-SUB]
3) (0P & 0Q) Cont (0(P ⊃ Q) [2),T4-31,SynSUB]

Prefixing 0 to 'T', 'F' or '0', always produces a contradiction. While sentences and other expressions can be neither true nor false, it can not be neither true nor false that a given expression is True, or neither true nor false that it is false, or neither true nor false that it is neither true nor false:

T7-61 [0TP Syn (∼TP & TP)]
Proof: 1) 0TP Syn (∼TTP & ∼FTP) [Df '0']
2) 0TP Syn (∼TP & ∼FTP) [1),7-20]
3) 0TP Syn (∼TP & TP) [3),Ax.7-2,DN]

T7-62 [0FP Syn (∼FP & FP)]
Proof: 1) 0FP Syn (∼TFP & ∼FFP) [Df '0']
2) 0FP Syn (∼FP & ∼FFP) [1),T7-21]
3) 0FP Syn (∼FP & ∼∼FP) [2),T7-17]
4) 0FP Syn (∼FP & FP) [3),DN]

T7-63 [0(0P) Syn (F(0P) & ∼F(0P)]
Proof: 1) 00P Syn (∼T(∼TP & ∼FP) & ∼F(∼TP & ∼FP)) [Df '0']
2) 00P Syn (FT(∼TP & ∼FP) & ∼F(∼TP & ∼FP)) [1),Ax.7-2]
3) 00P Syn (F(T∼TP & T∼FP) & ∼F(∼TP & ∼FP)) [2),Ax.7-3]
4) 00P Syn (F(FTP & FFP) & ∼F(∼TP & ∼FP)) [3),Df 'F'(twice)]
5) 00P Syn (F(∼TP & FFP) & ∼F(∼TP & ∼FP)) [4),Ax.7-2]
6) 00P Syn (F(∼TP & ∼FP) & ∼F(∼TP & ∼FP)) [5),T7-17]
7) 00P Syn (F(0P) & ∼F(0P)) [6),Df '0'(twice)]

Also ⊨ [0∼TP Syn 0TP], ⊨ [0∼FP Syn 0FP], and ⊨ [0∼0P Syn 00P].

Hence prefixing '0' to '∼T', '∼F' or '∼0' are also contradictions. Thus all truth-operators are covered; any prefixing of '0' to any truth-operator—i.e., '0' prefixed at 2nd-level to a 1st-level T-wff is a contradiction.[33] As a consequence, to prefix '∼0(...)' to any 1st-level **T-wff**, is to state a tautology, for it is the denial of an inconsistency. It follows from these remarks that '0' occurs significantly only when prefixed to wffs which are are not T-wffs.

The following three theorems provide a foundation for all of the analytic implication theorems in Section 7.423 including the A-implication theorems that provide rules of truth-tables. In each case a component with ‘0’ prefixed to it, is essential.

T7-64. [((T(P&Q) v F(P&Q) v 0(P&Q)) & FP) Cont F(P & Q)]
T7-65. [((T(P&Q) v F(P&Q) v 0(P&Q)) & ~TP) Cont ~T(P & Q)]
T7-66. [((T(P&Q) v F(P&Q) v 0(P&Q)) & (TP & 0Q)) Cont 0(P & Q)]

What is significant about all of three of these is the instantiation of the Law of Trivalence in the left-hand component.When it is removed what is left is an A-implication. All A-implications are elliptical. They implicitly presuppose the law of Trivalence, ⊨ [TP v FP v 0P]. Proofs of Implication-theorems must bring in T7-64, T7-65 or T7-66 to make the implicit presupposition on which any A-implication rests explicit. Proofs follow:

T7-64. [(T(P&Q)vF(P&Q)v0(P&Q)) & FP) Cont F(P&Q)]
Proof: 1) **(T(P&Q)vF(P&Q)v0(P&Q)) & FP)**
Syn (TP & TQ)vF(P&Q)v0(P&Q)) & FP) [T1-11, Ax.7-3,SynSUB]
2) “ Syn ((TP v F(PvQ)v0(P&Q)) & (TQ v F(PvQ)v0(P&Q)) & FP) [1),v&-DIST]
3) “ Cont ((TPvF(P&Q)v0(P&Q)) & FP) [2),Df‘Cont’]
4) “ Cont ((TP & ~FP)v(F(P&Q)v0(P&Q)) & FP) [3),Ax.7-01,SynSUB]
5) “ Cont (((TP v(F(P&Q)v0(P&Q)) & (~FP v (F(P&Q)v0(P&Q)) & FP) [4),v&-DIST]
6) “ Cont (~FP v (F(P&Q)v(0PvQ)) & FP) [5),Df‘Cont’]
7) (~T ~Pv(F(P&Q)v0(P&Q))& T ~P) Cont (F(P&Q)v0(P&Q)) [T7-44,U-SUB,&-COMM]
8) (~FPv(F(P&Q)v0(P&Q)) & FP) Cont ((F(P&Q)v0(P&Q)) & FP) [Df‘F’,&-IDEM,DR1-21]
9) (T(P&Q)vF(P&Q)v0(P&Q)) & FP) Cont ((F(P&Q)v0(P&Q)) & FP) [7),8),CCC-Syll]
10) (F(P&Q)v0(P&Q)) Syn (F(P&Q)v(~T(P&Q)& ~F(P & Q)) [Df‘0’,SynSUB]
11) (F(P&Q)v0(P&Q)) Syn (F(P&Q)v(~T(P&Q)& ~(FP v FQ)) [10),T7-22]
12) (F(P&Q)v0(P&Q)) Syn (F(P&Q)v(~T(P&Q)& ~FP & ~FQ) [11),DeM]
13) (F(P&Q)v0(P&Q)) Syn ((F(P&Q)v ~FP) & (F(P&Q)v ~FQ) & (F(P&Q)v ~T(P&Q)) [12),v&-DIST]
14) (F(P&Q)v0(P&Q)) Cont (F(P&Q)v ~FP) [13),Df‘Cont’]
15) ((F(P&Q)v0(P&Q) & FP) Cont ((F(P&Q)v ~FP) & FP) [14),DR1-21]
16) ((F(P&Q)v ~FP) & FP) Cont F(P&Q) [U-SUB,T7-44,v-ORD,Df‘F’]
17) ((F(P&Q)v0(P&Q) & FP) Cont F(P&Q)) [15),16),CCC-SYLL]
18) (T(P&Q)vF(P&Q)v0(P&Q)) & FP) Cont F(P&Q) [9),17),CCC-SYLL]

---

33. Another example:
1) 0T(P & ~TQ) Syn (~TT(P & ~TQ) & ~FT(P & ~TQ)) [Df ’0']
2) 0T(P & ~TQ) Syn (~T(TP & T ~TQ) & ~F(TP & ~TQ)) [1),Ax.7-3]
3) 0T(P & ~TQ) Syn (~(TTP & TT ~TQ) & ~F(TP & ~TQ)) [2)Ax.7-3]
4) 0T(P & ~TQ) Syn (~(TTP & TT ~TQ) & ~T ~(TP & ~TQ)) [3),Df ’F’]
5) 0T(P & ~TQ) Syn (~(TTP & TT ~TQ) & ~T(~TP v TQ)) [4),Demorgan]
6) 0T(P & ~TQ) Syn (~(TTP & TT ~TQ) & ~(T ~TP v TTQ)) [5),Ax.7-4]
7) 0T(P & ~TQ) Syn (~(TP & T ~TQ) & ~(T ~TP v TQ) [6),Eliminate TT for T]
8) 0T(P & ~TQ) Syn (~(TP & ~TQ) & ~(~TP v TQ)) [7),Eliminate 2nd-level T]
9) 0T(P & ~TQ) Syn ((~TP v TQ) & ~(~TP v TQ)) [DeMorgan]

T7-65. [((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) Cont ~T(P&Q)]

Proof: 1) (T(P&Q)vF(P&Q)v0(P&Q)) Syn (TP & TQ)vF(P&Q)v0 (P&Q)) [T1-11,T7-03,SynSUB]
2) (T(P&Q)vF(P&Q)v0(P&Q)) Syn ((TP v (F(P&Q)v0(P&Q))) & (TQ v (F(P&Q)v0(P&Q)))) [1),v&-DIST]
3) (T(P&Q)vF(P&Q)v0(P&Q)) Cont ((TP v (F(P&Q)v0(P&Q))) [2),Df'Cont']
4) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) Cont ((TP v (F(P&Q)v0(P&Q)) & ~TP) [3),DR1-21]
5) ((TP v (F(P&Q)v0(P&Q))) & ~TP) Cont (F(P&Q)v0(P&Q)) [U-SUB,T7-45,v-COMM]
6) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) Cont (F(P&Q)v0(P&Q)) [4),5),CCCSyll]
7) (F(P&Q)v0(P&Q)) Cont ~T(P&Q) [T7-56,U-SUB]
8) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) Cont ~T(P&Q) [6),7),CCC-Syll]

T7-66. [((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0Q)) Cont (0(P&Q)]

Proof: 1) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TQ) Cont ~T(P&Q) [T7-65]
2) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TQ & ~F(P&Q))) Cont (~T(P&Q) & ~F(P&Q)) [1),DR1-21,MP]
3) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TQ & ~F(P&Q)) Cont 0(P&Q) [2),Df '0',SynSUB]
4) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TQ & ~(FPvFQ)) Cont 0(P&Q) [3),T7-22,SynSUB]
5) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~TQ & ~FP& ~FQ) Cont 0(P&Q) [4),T4-17,SynSUB]
6) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~FP & (~TQ & ~FQ)) Cont 0(P&Q) [5),&-ORD]
7) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~FP & 0Q) Cont 0(P&Q) [6),Df '0',SynSUB]
8) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~FP & 0Q & TP) Cont (0(P&Q) & TP) [6),DR1-21,MP]
9) ((T(P&Q)vF(P&Q)v0(P&Q)) & ((TP& ~FP) & 0Q)) Cont (0(P&Q) & TP) [8),&-ORD,SynSUB]
10) ((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0Q)) Cont (0(P&Q) & TP) [9),Ax.7-1,SynSUB]
11) (0(P&Q) & TP) Cont 0(P & Q) [T1-36,U-SUB]
12) ((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0Q)) Cont 0(P&Q) [10),11),CCCSyll]

The last of the next three theorems are used in Section 7.4224, in which the Law of Trivalence, three Laws of Non-Contradition and three Laws of Excluded Middle are proved.

T7-67. [(0P v TP v FP) Syn ((~TP& ~FP) v (TP& ~FP) v (FP& ~TP)]

Proof: 1) (0P v TP v FP) Syn (0P v TP v FP) [T1-11,U-SUB]
2) (0P v TP v FP) Syn ((~TP & ~FP) v TP v FP) [1),Df '0',SynSUB]
3) (0P v TP v FP) Syn ((~TP & ~FP) v (TP & ~ FP) v FP) [2),Ax7-01]
4) (0P v TP v FP) Syn ((~TP & ~FP) v (TP & ~ FP) v (FP & ~TP)) [3), T7-14]

T7-68. [(0P v TP v FP) Syn ((~TP v TP) & (~TP v ~FP) & (~FP v FP))]

Proof: 1) [(0P v TP v FP) Syn ((~TP& ~FP) v (TP& ~FP) v (FP& ~TP) [T7-67]

2) [(0P v TP v FP) Syn ((~TP v TP v FP)	Syn	3) **((~TP v TP)**	[By T7-01]
& (~TP v TP v ~TP)		& (~TP v TP)	[By v-ORD]
& (~TP v ~FP v FP)		& **(~TP v ~FP)**	[By T7-01]
& (~TP v ~FP v ~TP)		& (~TP v ~ FP)	[By v-ORD]
& (~FP v TP v FP)		& **(~FP v FP)**	[By T7-14]
& (~FP v TP v ~TP)		& (~FP v ~TP)	[By T7-14]
& (~FP v ~FP v FP)		& (~FP v FP)	[By v-ORD]
& (~FP v ~FP v ~TP))		& (~FP v ~TP))	[By v-ORD]

4) ((0P v TP v FP) Syn ((~TP v TP) & (~TP v ~FP) & (~FP v FP)) [3),&-ORD (eliminating duplicates in 3)]

T7-69. [T(0P v TP v FP) Syn (T(~ TP v TP) & T(~ TP v ~ FP) & T(~ FP v FP))]
Proof: 1) [T(0P v TP v FP) Syn T((~ TP v TP) & (~ TP v ~ FP) & (~ FP v FP)) [T7-68,R7-1]
2) [T(0P v TP v FP) Syn (T(~ TP v TP) & T(~ TP v ~ FP) & T(~ FP v FP)) [1),Ax.7-04]

Other theorems of some interest include:

|= [~ 0P Syn T(P v ~ P)] |= [0P Syn ~ T(TP v FP)]
|= [~ 0P Syn T(TP v FP)] |= [0P Syn F(TP v FP)]

|= [0(P & Q) Cont ~ FP]
Proof: 1) 0(P&Q) Syn (~ T(P&Q) & ~ F(P&Q)) [Df '0']
2) 0(P&Q) Cont ~ F(P&Q) [1),Df.'Cont']
3) 0(P&Q) Cont ~ (FP v FQ) [2),T7-22,SynSUB]
4) 0(P&Q) Cont (~ FP & ~ FQ) [3),DeM]
5) 0(P&Q) Cont ~ FP [4),Df 'Cont']

|= [0(P v Q) Cont ~ TQ]
Proof: 1) 0(PvQ) Syn (~ T(PvQ) & ~ F(PvQ)) [Df '0']
2) 0(PvQ) Cont ~ T(P v Q) [1),Df''Cont']
3) 0(PvQ) Cont ~ (TP v TQ) [2),Ax.7-04]
4) 0(PvQ) Cont (~ TP & ~ TQ) [3),DeM]
5) 0(PvQ) Cont ~ TQ

7.422 Properties of T-wffs: Inconsistency, Unfalsifiability, Logical Truths and Presuppositions of A-logic

So far we have produced only Syn- and Cont-theorems between first or higher-level T-wffs of analytic truth logic. Next we consider other kinds of theorems with T-operators—Inc- and TAUT-theorems, and theorems assserting logical unfalsifiability, or unsatisfiability, logical truth or logical falsehood.

Putting R7-1 and its corrolaries together with the definitions of Inc and Taut, we can derive rules for inferring the inconsistency or tautology of T-wffs from the inconsistency or tautology of expressions without any T-operators, as well as from Syn- and Cont-theorems in this chapter. This is explored in Section 7.4221 starting with DR7-5 [If Inc(P) then Inc (TP)]. Since inconsistencies are derived from containments, the properties of Inconsistency and Tautology can also be ascribed to particular T-wffs on the basis of earlier Syn- and Cont-theorems.

The second Rule. R7-2. [If |= Inc(P) then |= ~ T(P)] and its equivalent for tautology, R7-2'. [If |= Taut(P) then |= ~ F(P)], permit inferences to the properties of logical non-Truth (logical Unsatisfiability) and logical non-falsehood (logical Unfalsifiability), from Inc- and TAUT-theorems, and therefore from Syn- and Cont-theorems of the earlier chapters. These are explored in Section 7.4222.

The properties of logical truth and logical falsehood figure prominently in the semantics of M-logic. Taking the meaning of "logical truth" as it is taken in M-logic,[34] many theorems of logical truth ("LT-theorems") and of logical falsehood ("LF-theorems") are expressible as 2nd-level T-wffs of A-logic. But, with a small number of exceptions, these are of little importance since they are not

34. See Quine's definition in Section 7.4223.

valid in A-logic's sense of 'valid' and are trivial and uniformative in actual truth-investigations. Different categories of logical truth and falsehood are examined in Section 7.4223.

Finally, in Section 7.4224 we examine and define a small number of exceptional logical truths, which are the important Presuppositions of analytic logic. The basic ones are the Law of Trivalence and its corrolaries, which include three distinct laws of Excluded Middle and three distinct Laws of Non-Contradiction, but do not include the Law of Bivalence called "Excluded Middle" in M-logic. The Law of Trivalence is involved essentially in the definition and differentiation of valid "analytic implications" as developed in Section 7.423.

7.4221 Inc- and TAUT-theorems in Truth-logic

Inc- and Taut- theorems in truth-logic are derivable in two ways. The first way prefixes 'T' or '∼F' to the wffs in Inc- or TAUT-theorems of previous chapters; for example, from T5-02. TAUT[P ⊃ (Q v P)] it follows that ⊨ TAUT[∼F(P ⊃ (Q v P))]. The second way is from Syn- and Cont-theorems for T-wffs unique to this chapter using U-SUB and the definitions of 'Inc' and 'Taut'; for example, from T7-07 [0(P) Syn (∼TP & ∼FP)] it follows that ⊨ Inc[0P & (TP v FP)]. We start with the first way.

It is not immediately apparent that if an expression is inconsistent, then the same expression with a 'T' prefixed to it is inconsistent, or that if any expression is tautologous then the same expression with '∼F' prefixed to it is tautologous. Inconsistency and tautology are not preserved if just any T-operator is prefixed, so we have here certain special principles, derivable from R7-1 with Df 'Inc', which are expressible in symbols as follows:

DR7-5.	[If Inc(P) then Inc(TP)]	"If P is inconsistent, then 'P is true' is inconsistent"
and DR7-5'.	[If Taut(P) then Taut(∼FP)]	"If P is tautologous, then 'P is not false' is tautologous"

DR7-5 holds because to say something is true, is to say it corresponds to a fact in an objective field of reference. If an expression is inconsistent, it is inconsistent with the concept of such a field to say that that expression corresponds to, or is true of, some fact in that field since, by definition, such a field has no inconsistencies. But **<u>none</u>** of the following are valid:

If Inc(P), then Inc (F(P)),	If Taut(P), then Taut(T(P)),
If Inc(P), then Inc (∼T(P)),	If Taut (P), then Taut(∼T(P)),
If Inc(P), then Inc(∼F(P)),	If Taut(P), then Taut(F(P)).

DR7-5 is derivable from R7-1 and the definitions of 'Inc' and 'Taut', which we repeat.

<u>Df 5-1</u>. 'Inc(P)' Syn_{df} (i) P Syn (Q & ∼R) & Q Cont R,
or (ii) P Syn (Q & R) & Inc Q,
or (iii) P Syn (Q v R) & Inc(Q) & Inc(R).

<u>Df 5 -2</u>. 'Taut(P)' Syn_{df} 'Inc(∼P)'.

DR7-5. If INC(P) then Inc(T(P))

<u>Proof</u>:	1) INC[P]	[Premiss]
	2) [(P Syn (Q & ∼R)) & (Q Cont R)]	[1),**Df 'Inc'Clause(i)**]
	3) [P Syn (Q & ∼R)]	[2),Simp]
	4) [Q Cont R]	[2),Simp]
	5) [Q Syn (Q & R)]	[4),Df 'Cont',]
	6) [P Syn ((Q & R) & ∼R)]	[3),5),SynSUB]

7) [TP Syn T((Q & R) & ~R)] [6),**R7-1**,MP]
8) [TP Syn (TQ & TR & T~R)] [7),Ax.7-3,SynSUB(twice)]
9) [T~R Syn (T~R & ~F~R)] [Ax.7-1,U-SUB]
10) [T~R Syn (T~R & ~T~~R)] [9),Df 'F']
11) [T~R Syn (T~R & ~TR)] [10),DN]
12) [TP Syn (TQ & TR & T~R & ~TR)] [8),11),SynSUB]
13) [TP Cont (TR & ~TR)] [12),Df.'Cont']
14) Inc[TR & ~TR] **[Df 'Inc',Clause(i)]**
15) Inc[TQ & TR & T~R & ~TR] **[Df 'Inc',Clause (ii)]**
16) Inc[TP] [15),12),SynSUB]
17) [If INC(P) then IncT(P)] [1) to 16),C.Pr.]

DR7-5 is used to establish the traditional version of the law of non-contradiction—that no expression is both true and false, i.e., |= F[TP & FP]. The Law of "Excluded Middle" taken as "Every sentence is either true or false", |= [TP v FP], is not derivable in A-logic; it does not follow from DR7-5 or any other rule or axiom. The companion of DR7-5 for tautology is:

DR7-5' [If Taut(P) then Taut(~F(P))]
<u>Proof</u>: 1) If Inc(~P) then Inc(T~P)) [DR7-5,U-SUB]
2) If Inc(~P) then Inc(~~T~P)) [1),DN,SynSUB]
3) If Taut(P) then Taut(~T~P)) [2),Df 'Taut',SynSUB]
4) If Taut(P) then Taut(~FP)) [3),Df 'F',SynSUB]

From DR7-5 and DR7-5', together with DR5-5a, we get derived rules, DR7-5a to DR7-5a', which take us from Cont-theorems to Inc- and TAUT-theorems about expressions with T-operators:

DR7-5a. [If P Cont Q then Inc(T(P & ~Q))]
<u>Proof</u>: 1) If [P Cont Q], then Inc[P & ~Q] [DR5-5a]
2) If Inc[P & ~Q], then Inc[T(P & ~Q)] [DR7-5]
3) If [P Cont Q], then Inc[T(P & ~Q)] [1),2),Hyp-Syll]

DR7-5a' . If P Cont Q then Taut(~F(~P v Q))
<u>Proof</u>: 1) [If (P Cont Q), then IncT(P & ~Q)] [DR7-5a]
2) [If (P Cont Q), then Inc[T~(~P v Q)] [1),DeM,SynSUB]
3) [If (P Cont Q), then Inc[~~T~(~P v Q)] [2),DN,SynSUB]
4) [If (P Cont Q), then Taut[~T~(~P v Q)] [3),Df 'Taut',SynSUB]
5) [If (P Cont Q), then Taut[~F(~P v Q)] [4),Df 'F',SynSUB]

and since Syn-theorem entail Cont-theorems, we can also infer Inc- and TAUT-theorems with T-operators from Syn-theorems, yielding DR7-5b to DR7-5d. For example,

DR7-5b. [If P Syn Q, then Inc(T(P & ~Q))]
DR7-5b'. [If P Syn Q, then Taut(~F(~P v Q))]

Finally, we have a principle concerning non-inconsistency:

DR7-5e. If not-Inc(P) then not-IncT(P)

Informally, the proof is as follows: Assume P is not inconsistent . If T is prefixed to P, this does not change the facts with respect to any component of P's occurring POS or NEG in P. If P is not inconsistent and T(P) does not change any component with respect to its occuring POS or NEG, then T(P) can not be inconsistent.

By DR7-5 if one prefixes a 'T' to an inconsistent expression as a whole, the result is inconsistent. By DR7-5' if one prefixes '~F' to a tautologous expression, the result is tautologous. But only 'T', of the four basic T-operators, can be prefixed if inconsistency is to be preserved, and only '~F' can be prefixed to a tautology if tautology is to be preserved. This is shown by truth-tables in the examples below.

In analytic truth-logic, expressions which are logically not-true are expressions which have no T's in their truth-tables. Thus <u>all</u> inconsistencies are logically not-true.

The result of prefixing T to an inconsistent wff is inconsistent (no T's in its truth-table):

$\models$ Inc(P & ~P)	$\models$ Inc[T(P & ~P)]	Syn	$\models$ Inc[TP & FP]
0	**F** 0		F0 **F** F0
F	**F** F		TT **F** FT
F	**F** F		FF **F** TF
^	^		^

Prefixing a F to an inconsistency does not preserve inconsistency.

$\models$ Inc(P & ~P), but	<u>Not-Inc</u>[F(P & ~P)].	[F(P & ~P) Syn	(TP v FP)] Syn	~0P
0	F 0	F 0	F F F	F T
F	**T** F	**T** F	T **T** F	**T** F
F	**T** F	**T** F	F **T** T	**T** F
^	^ ^	^ ^		

Prefixing a ~T to inconsistency does not preserves inconsistency:

$\models$ Inc(P & ~P), but	<u>not-Inc</u>[~T(P & ~P)].	[~T(P & ~P) Syn	(~TP v ~FP)]
0	**T**F 0	**T**F 0	TF **T** TF
F	**T**F F	**T**F F	FT **T** TF
F	**T**F F	**T**F F	TF **T** FT
^	^ ^	^	

Prefixing a ~F to an inconsistency does not preserve inconsistency.

$\models$ Inc(P & ~P), but	<u>not-Inc</u>[~F(P & ~P)].	[~F(P & ~P) Syn	(~FP & ~TP)	Syn 0P]
0	**T**F 0	**T**F 0	TF **T** TF	**T**
F	FT F	FT F	TF F FT	F
F	FT F	FT F	FT F TF	F
^	^ ^	^ ^		

Similarly, with respect to DR7-5', only '~F', if prefixed to a tautologous wff, will be tautologous. Given a wff P which is tautologous, prefixing 'T' or '~T' or 'F' to 'P' in Taut[P], does not preserve tautology, as the following examples will show. All tautologies are logically not- false. Expressions which are logically not-false have no F's in their truth-tables.

The result of prefixing an ~F to a tautologous wff is tautologous.

$\models$ Taut[~P v P]	$\models$ Taut[~F(~P v P)] Syn	$\models$ Taut[~TP v ~FP]	$\models$ Taut[~(TP & FP)]
0	**T**F 00 0 0	TF0 **T** TF0	**T** F0 F F0
T	**T**F FT T T	FTT **T** TFT	**T** TT F FT
T	**T**F TF T F	TFF **T** FTF	**T** FF F TF
^	^	^	^

But, prefixing a T to a tautologous expression does not preserve tautology; false cases can occur.

|= Taut(P v ∼P), but not-taut[T(P v ∼P)]; [T(P v ∼P) Syn (TP v FP) Syn ∼0P]
0 0 00 F 0 0 00

And prefixing a ∼T to tautology does not preserves tautology:

|= Taut (P v ∼P), but not-taut[∼T (P v ∼P)]; [∼T(P v ∼P) Syn (∼TP & ∼FP) Syn 0P]
T T FT FT T T FT

And prefixing a F to an tautology does not preserve tautology.

|= Taut(P v ∼P), but Not-Taut F(P v ∼P)]; [F(P v ∼P) Syn (TP & FP)]
F T TF F F T TF

Every Syn- and Cont-theorem in this book will immediately yield an Inc-theorem or Taut-theorem of the form '|= Inc[T(...)]' or '|= Taut [∼F(...)]' by means of one of the derived rules,

DR7-5a. [If (P Cont Q), then IncT(P&∼Q)]
DR7-5a'. [If (P Cont Q), then Taut∼F(∼P v Q)]
DR7-5b. [If (P Syn Q), then IncT(P&∼Q)]
DR7-5b'. [If (P Syn Q), then Taut∼F(∼P v Q)]
DR7-5c. [If (P Syn Q), then IncT(Q&∼P)]
DR7-5c'. [If (P Syn Q), then Taut∼F(∼Q v P)]
DR7-5d. [If (P Syn Q), then Inc(T(P&∼Q) & T(Q & ∼P))]
DR7-5d'. [If (P Syn Q), then Taut∼F(P ≡ Q)]

and each result would be amenable to numerous distinct but synonymous other theorems as a result of substituting synonyms for synonyms by SynSUB. It would be tedious and of no great importance to enumerate all of these theorems. But the following theorems will be referred to later.

The second way of deriving Inc- and TAUT-theorems in truth-logic is simply a matter of using Syn-and Cont-theorems of this chapter, and substituting T-wffs for predicate letters in the Derived Rules of Chapter 5. The tautologousness of the Law of Trivalence can be established in this manner.

T7-70 Taut[0P v TP v FP]

Proof:	1) [∼0P Syn (TP v FP)]	[T7-49]
	2) [If (P Syn Q) the Inc(P & ∼Q)]	[DR5-5b]
	3) [If (∼0P Syn (TP v FP)) then Inc(∼0P & ∼(TP v FP))]	[1), U-SUB]
	4) Inc[∼0P & ∼(TP v FP)]	[1),3), MP]
	5) Inc[∼∼(∼0P & ∼(TP v FP))]	[4),DN]
	6) Taut[∼(∼0P & ∼(TP v FP))]	[5),Df' 'Taut']
	7) Taut[(0P v TP v FP)]	[6),Df 'v']

From other theorems of this chapter with DR5-5a and DR5-5b we get the Inc-theorems,

|= Inc[TP & FP]	Three laws	[T7-13,DR5-5a]	
|= Inc[TP & ~TP]	of	[T7-11,DR5-5a]	
|= Inc[FP & ~FP]	non-contradiction	[T7-12,DR5-5a]	
|= Inc[(~TP & ~FP) & ~0P]		[T7-07,DR5-5a]	
|= Inc[~FTP & ~TTP]	(Syn '|= Inc[0(TP)]')	[T7-32,DR5-5a]	2nd-level
|= Inc[~FFP & ~TFP]	(Syn '|= Inc[0(FP)]')	[T7-33,DR5-5a]	2nd-level
|= Inc[~T0P & ~F0P]	(Syn '|= Inc[0(0P)]')	[T7-53,T7-54,DR5-5a]	2nd-level

From these with double negation, Df 'Taut', and Df 'v', we can derive,

|= Taut[(~TP v ~FP)]	Three laws	
|= Taut[(~TP v TP)]	of	
|= Taut[(~FP v FP)]	"excluded middle"	
|= Taut[(0P v TP v FP)]	Law of Trivalence	
|= Taut[FTP v TTP]	(Syn '|= Taut[~0(TP)]'	2nd-level
|= Taut[FFP v TFP]	(Syn '|= Taut[~0(FP)]')	2nd-level
|= Taut[F0P v T0P]	(Syn '|= Taut[~0 (0P)]')	2nd-level

The last three show that for 2nd-level T-wffs are always either True or False (bivalence).

There is little use for tautologies in A-logic. But proofs of inconsistency and of absence of inconsistency or satisfiability are important, both to avoid error and to establish validity as defined in A-logic, respectively. The primary objective of A-logic is to establish rules of valid inference.

7.4222 Unsatisfiability- and Unfalsifiability-Theorems

The second new principle of inference is R7-2 [If Inc(P) then |= ~T(P)]. This says "If any expression is Inconsistent then it is logically not true". To say that an expression is logically not true, is to say that by logic it is unsatisfiable—it can have no true substitution instances.

R7-2 is based on the idea that to say something is true, is to say it corresponds to a fact in an objective field of reference. The concept of an objective field of reference is of a field that has no inconsistencies. If an expression is inconsistent, it can not correspond to any fact in any objective field of reference which is presupposed by the word 'truth'. For similar reasons a Tautology, as a denial of an inconsistency, can not be false of any fact in an objective field of reference.

R7-2, with the definition of Tautology, quickly yields,
DR7-2'. [If Taut(P) then |= ~FP]

Proof:	1) [If Inc(~P) then ~T~P]	[R7-2,U-SUBa)]
	2) [If Inc(~P) then ~FP]	[1),Df 'F']
	3) [If Taut(P) then ~FP]	[2),Df 'Taut']

To derive the logical Unsatisfiability and Unfalsifiability of particular T-wffs, the first step is the step from Syn or Cont-theorems to Inc- or TAUT-theorems. The chief derived rules from Chapter 5, namely,

DR5-5a. [If P Cont Q, then Inc(P & ~Q)],
DR5-5a'. [If P Cont Q, then Taut(P ⊃ Q)],
DR5-5b. [If P Syn Q, then Inc(P & ~Q)],
and DR5-5b'. [If P Syn Q, then Taut(P ⊃ Q)],

say nothing about truth or falsehood. Applying DR5-5a and DR5-5b, with R7-2 and DR7-2' an Unsatisfiability theorem and an Unfalsifiability theorem follows from every Syn- or Cont-theorem.

From DR5-5a and R7-2 we get the following principles for going directly from a Syn- or Cont-theorem to a logical non-truth (an unsatisfiability-theorem) or of logical non-falsehood (an unfalsifiability-theorem).

DR7-2a. [If (P Cont Q) then ~T(P & ~Q)]

Proof: 1)	[If (P Cont Q) then Inc(P & ~Q)]	[DR5-5a]
2)	[If Inc(P & ~Q) then ~T(P & ~Q)]	[R7-2,U-SUB]
3)	[If (P Cont Q) then ~T(P & ~Q)]	[1),2) Hyp-SYLL]

DR7-2a'. [If (P Cont Q) then ~F(~P v Q)]

Proof: 1)	[If (P Cont Q) then ~T(P &~Q)]	[DR7-2a]
2)	[(P & ~Q) SYN ~(~Pv~Q)]	[T4–13(DeM)]
3)	[If (P Cont Q) then ~T~(~P v Q)]	[1),2),SynSUB]
4)	[If (P Cont Q) then ~F(~P v Q)]	[3)Df 'F',SynSUB]

Since [P Cont Q] if [P Syn Q], the following principles also hold:

DR7-2b. [If (P Syn Q), then ~T(P&~Q)]	[DR5-5b,R7-2,Hyp-SYLL]
DR7-2b'. [If (P Syn Q), then ~F(~P v Q)]	[DR7-2',DeM,Df 'F']

Thus every Syn- and Cont-theorem in this book will immediately yield an Unsatisfiability-theorem of the form '|= ~T[...]' and an Unfalsifiability-theorem of the form '|= ~F[...]' by means of one of the derived rules,

DR7-2a. [If (P Cont Q), then ~T(P&~Q)]
DR7-2a'. [If (P Cont Q), then ~F(~P v Q)]
DR7-2b. [If (P Syn Q), then ~T(P&~Q)]
DR7-2b'. [If (P Syn Q), then ~F(~P v Q)]

Applying these Rules to Syn and Cont-theorems of Chapters 1 through 5, in Section 7.421 we get, among many others,

1st-level T-wffs:	Unsatisfiability theorems:	|= ~T[P & ~P]	[T1-11,DR7-2b]
		|= ~T[P & Q & ~Q]	[T1-37,DR7-2a]
	Unfalsifiability theorems:	|= ~F[~P v P]	[T1-11,DR7-2b']
		|= ~F[~(P & Q) v Q)]	[T1-37,DR7-2a']

From every Syn and Cont-theorem in Chapters 1 through 5 we can get both a 1st-level unsatisfiability-theorem by DR7-2b or DR7-2a and a 1st-level unfalsifiability theorem by DR7-2b' or DR7-2a'. In addition, from Syn and Cont-theorems about T-wffs in the present chapter we can get 2nd-level Unsatisfiability and Unfalsifiability-theorems:

2nd-level T-wffs:

Unsatisfiability theorems:	$\models\ \sim T[TP\ \&\ \sim TP]$	[T7-11,DR7-2b]
	$\models\ \sim T[FP\ \&\ \sim FP]$	[T7-12,DR7-2b]
	$\models\ \sim T[TP\ \&\ FP]$	[T7-13,DR7-2a]
	$\models\ \sim T[\sim 0P\ \&\ \sim TP\ \&\ \sim FP]$	[T7-49,DR7-2b,Df 'v',DN]
Unfalsifiability theorems:	$\models\ \sim F[\sim TP \vee TP]$	[T7-11,DR7-2b']
	$\models\ \sim F[\sim FP \vee FP]$	[T7-12,DR7-2b']
	$\models\ \sim F[\sim TP \vee \sim FP]$	[T7-13,DR7-2a']
	$\models\ \sim F[0P \vee (TP \vee FP)]$	[T7-49,DR7-2b', DN]

In efforts to arrive at the lawlike statements of empirical science and common sense, the concept of, and ability to identify, the empirically unfalsifiability or unsatisfiability of a first level truth-claims is helpful. It would be tedious and of no great importance to enumerate lists of logically unfalsifiable or unsatisfiable truth-claims. Questionable cases can usually be quickly tested using the truth-table principles.

7.4223 Logical Truth and Logical Falsehood

"A logical truth" Quine says, "is definable as a sentence from which we get only truths if we substitute sentences for its simple sentences".[35] Elsewhere he explains,

> Logical truths are…not only true, but stay true even when we make substitutions upon their component words and phrases as we please, provided merely that the so-called "logical words" '= ', 'or', 'not', 'if-then', 'everything', 'something', etc., stay undisturbed. All that counts, when a statement is logically true, is its structure in terms of logical words. Thus it is that logical truths are commonly said to be true by virtue of the meanings of the logical words.[36]

In M-logic an unquantified sentence is Logically True if and only if it has only T's in the final column of its two-valued truth-table, but the definition above covers quantified statements as well. The concept of a Logical Truths as a *universal* truth is perhaps best exemplified in Theorems like $\models T[(\forall x)(TPx \vee \sim TPx)]$ which says in effect, "by logic it is true for all individuals, that either P is true of that individual or P is not true of that individual" where P can be replaced by any predicate.

We use a version of Quine's definition of 'Logical truth' and a corresponding definition for 'Logical Falsehood' throughout this section.[37] For unquantified wffs all substitution instances of the wff will be true, and the theorem will be a LT-theorem if and only if 1) it has the form '$\models T(\ldots)$' and 2) there is a T in the column under the left-most T at every row. The theorem will be a LF-theorem, if and only if it has

---

35. W.V.Quine, *Philosophy of Logic*, 1970, p. 50.

36. Quine, *Methods of Logic*, 4th Ed., Harvard University Press, 1982, p.4. (I do not include '= ' as a "logical word", since it has a substantive meaning)

37. In A-logic a wff or statement is unsatisfiable if there are no T's in its truth Table or if no substitution instances have the value True. This does not entail that it is logically false; i.e that all substitution instances are false. For if not-true, it may be neither true nor false. In M-logic, where every expression is either True or False, if an expression is unsatisfiable it must be logically false. In A-logic a wff or statement is unfalsifiable if there are no F's in its truthTable or if no substitution instance can be False. This does not entail that it is logically True, i.e that all substiution instances are True. For if not-false, it may be neither true nor false. In M-logic, where every expression is either True or False, if an expression is unfalsifiable it must be logically true.

the form '|= F(…)' and there is a T in the column under leftmost F at every row. For quantified wffs, every instantiation of its normal form first-level T-wff must satisfy the same requirements.

Quine also says 'Logical Truth' can be defined as sentences obtainable by substitution in theorems (valid schema) provable within an axiomatization of M-logic.[38] Schemata or wffs can not be logically true, because schemata are not sentences. Thus his theorems, called valid schemata, are not logical truths; rather, all sentences obtained by substitution in them are logical truths.

This kind of distinction is not needed in A-logic. The theorems of A-logic are all sentences, and all are intended to be taken as true sentences. The subject-terms are schemata in quasi-quotes, the predicate terms are "<1>is Syn to<2", '<1>is Taut", etc., and "<1>is True". Any theorem which is of the form '|= T[…]' may be read "By Logic, it is true that […]", i.e., an assertion that what replaces '…' is logically true in the sense of Quine. Such theorems maybe called Logical Truth-theorems. Similarly, '|= F[…]' signifies a Logical Falsehood-theorem. For simplicity, we sometimes call Logical Truth-theorems 'LT-theorems', and Logical falsehood- theorems 'LF-theorems'; so 'LT[…]' and 'LF[…]' can be taken as abbreviations for '|= T[…]' and '|= F[…]' respectively.

However, Quine's distinction between schemata, which are neither true nor false, and expressions which are instances of a schema which may be true, is correct. In A-logic 'T ~(P& ~P)' and 'F ~(P& ~P)' are both indicative sentences, but neither is true. For ' ~(P& ~P)' is just a schema so it is neither true nor false. So both 'T ~(P& ~P)' is false and 'F ~(P& ~P)' are false and '0 ~(P& ~P)' is a factual truth. By T7-52, it is synonymous with '0(P& ~P)' which is also true.

On the other hand, ' ~F[~(P& ~P)]' and its synonym ' ~T[P& ~P]' are true, but ' ~T[~(P& ~P)]' is false. Due to the brackets, the latter says "Any result of putting an expression for 'P' in ' ~(P& ~P)' is not true". But this is not correct. The result of putting an expression, 'Tito died', for 'P' in ' ~(P& ~P)' is "It is not the case that (both Tito died and it is not the case that Tito died)" which is the denial of a contradiction. By the law of non-contradiction this denial is true, and to say it is not true is false. Thus ~F[~(P& ~P)] is true and ~T[~(P& ~P)] is false, making 0[~(P& ~P)] false.

The rows in the truth-table of a quasi-quoted wff represent all of the possible truth-values, T, F, or 0, of expressions substituted for predicate letters in the expression. To assert that [P& ~P] or its synonym is not true, is to say there are no T's in the final column of its truth-table—i.e., that no matter what is substituted, the assertion that it is not-true, in that case, will be true.

Zero-level T-wffs		First-level T-wffs			
~[P ⊃ P]	[P ⊃ P]	F[P ⊃ P]	**~F[P ⊃ P]**	T[P ⊃ P]	~T[P ⊃ P]
~[~P v P]	[~P v P]	F[~P v P]	**~F[~P v P]**	T[~P v P]	~T[~P v P]
[P & ~P]	~[P & ~P]	T[P & ~P]	**~T[P & ~P]**	F[P & ~P]	~F[P & ~P]
0 **0** 00	**0** 0 0 00	**F** 0 0 00	**T**F 0 0 00	**F** 0 0 00	**T**F 0 0 00
T **F** FT	**T** T F FT	**F** T F FT	**T**F T F FT	**T** F F TT	**F**T T F FT
F **F** T F	**T** F F T F	**F** F F T F	**T**F F F T F	**T** T F T F	**F**T F F TF
^	^	^	^	^	^

No first-level T-wff will be a LT-theorem, i.e., none will both 1) have the form |= T(…) and 2) have all Ts in the final column of its truth-table. For 1st-level T-wffs always have an F in the truth-table in the row in which every elementary component of the zero-level T-wff takes 0. The closest 1st-level T-wffs come to logical truths are Unfalsifiability theorems, e.g., |= ~F[~Pv P] and synonymous Unsatisfiability-theorems like |= ~T[P & ~P]. From each Inc-theorem and Taut-theorem about a zero-

38. Ibid , pp 56-58

level T-wff, one can derive a first-level Unfalsifiability-theorem of the form ⊨ ~F(...) and a logically synonymous first-level T-wff which is an Unsatisfiability theorem of the form ⊨ ~T(...).

Only 2nd-level T-wffs can be LT-theorems or LF-theorems. Only 2nd- or higher- level T-wffs can have all T's in the final column of their truth-tables. But every 1st-level Unfalsifiability-theorem and Unsatisfiability-theorem yields a 2nd-level LT-theorem and an LF-theorem simply by prefixing a T.

T~T[P & ~P]	Syn	**T**~F[~P v P]
TT F 0 0 0 0		**T**T F 0 0 0 0
TT F T F F T		**T**T F F T T T
TT F F F T F		**T**T F T F T F

Logical-Truths and Logical Falsehoods are therefore readily derivable in analytic truth-logic, and they cover all of the "logical truths" that M-logic covers—and more.[39]

But, with a few exceptions, theorems of Logical Truth and Logicall Falsehood are of little use or importance in analytic truth-logic. The role of Logical Truth in M-logic through the association of M-valid TF-conditionals with 'implication' and valid inference, are supplanted in A-logic by valid C-conditionals which avoid the non-sequiturs, anomalies and paradoxes of M-logic. The concept of Logical Unfalsifiability comes in handy in dealing with the lawlike statements of the Empirical Sciences. But Logical Truths are of no use there. In Quine's words,

> When a schema is valid [*M-valid*], any statement whose form that schema depicts [*i.e., any "logical truth"*] is bound to be, in some sense trivial. It will be trivial in the sense that it conveys no real information regarding the subject matter whereof its component clauses speak.[40]

The primary exception to the triviality of Logical Truths is the Law of Trivalence, T7-70, ⊨ T[0P v TP v FP] and its entailments which are discussed in the next section.

In the meantime, to produce LT-theorems and LF-theorems from any Unfalsifiability-theorem or Unsatisfiability-theorem, simply prefix a T, and manipulate with DN, DeMorgan theorems, and Df 'F', until you get a Falsehood claim which is warranted by logic, and a synonymous Truth-claim which is warranted by logic. For example, from the Unfalsifiability theorem ⊨ ~F[P v ~P] and the Syn-Theorem, ⊨ [F(P v ~P) Syn (FP & F~P)] (By U-SUB on T7-23}, we get ⊨ ~(FP & F~P)] and prefix a T, to get the LT-theorem ⊨ T~(FP & F~P)] and which is synonymous (by Df 'F'), to the LF-theorem ⊨ F[FP & F~P] which is also synonymous with the LT-theorem ⊨ T[~FP v ~TP].

Since all Logically True schemata are synonymous with various Logically False schemata, there is no special importance to be attached to the set of Logically True schemata as compared with the set of logically false schemata.

In M-logic all logically true wffs are TF-equivalent (and called "logically equivalent"). But in A-logic they vary greatly with respect to synonymy; many distinctions can be made among Logically-True expressions for Logical-Truth-theorems are not in general synonymous with each other. Logical-Truth theorems can be divided into disjoint classes based on features of non-synonymy. Among such non-synonymous pairs of Logical Truths are pairs in which,

---

39. There are tautologies with C-conditionals in A-logic which yield 2nd-level Logical truths which have no analogues in M-logic. E.g., T[~T((P & ~P) => P)] vs.T[~T((P & ~P) ⊃ P)]

T TF 0 0 00 0 0	**T** TF 0 0 00 0 0
T TF T F FT 0 T	**F** FT T F FT T F
T TF F F TF 0 F	**F** FT F F TF T T

40. Quine, *Methods of Logic*, 4th Ed., Harvard University Press, 1982, p. 42. (Words in italics are added)

I. One has **n** atomic wffs **vs**. one has more than **n** atomic wffs
Examples: |= T[TP v ~TP] is not Syn to |= T[(~FP v TP v ~TP)]

II. All disjuncts are contingent **vs**. some disjuncts are not contingent
Examples: |= T[FP v ~FP] is not Syn to |= T[(TP& ~FP) v ~FP v FP]

III. Every disjunct is inconsistent with some other disjunct
vs. some disjuncts are not inconsistent with any other disjuncts
Examples: |= T[FP v ~FP v TQ v ~TQ] is not Syn to |= T[FP v ~FP v TQ]

IV. Every disjunct is inconsistent with all other disjuncts
vs. some disjuncts are not inconsistent with some other disjuncts
Examples: |= T[TQ v FQ v (~TQ &~FQ)] is not Syn to |= T[TQ v FQ v ~FQ]

Every LT-theorem in the examples just given has all T's in the final column of its trivalent truth table, but no two are logically synonymous. Many other distinctions due to non-synonymy can be drawn.

All 2nd-level Logical Truths have in common the property that all substitution instances are *de dicto* true. But having the same truth-tables does not make them synonymous. And among the various non-synonymous logical truths, some kinds are more important for logic than others.

7.4224 The Law of Trivalence and Presuppositions of Analytic Truth-logic

The Law of Trivalence—every expression is either true, or false, or neither-true-nor-false—is the basic presupposition guiding the search for truth. What we seek, of course, are Truths. Falsehoods and expressions that are neither true nor false are not the sort of results we seek, but we must recognize both what is false and what is irrelevant in order to avoid them in the truth search.

The Law of Trivalence is expressed in symbols as |= [0P v TP v FP]. It is a Tautology, it is Unfalsifiable, and it is also a Logical Truth.

Taut[0P v TP v FP] [T7-70]

|= [~F(0P v TP v FP)] [T7-70, DR7-2']

|= T[0P v TP v FP]

Proof: 1)	[T0P Syn T0P]	[T1-11,U-SUB]
2)	[Taut[T0P ⊃ T0P]	[1),DR5-5b',MP]
2)	[Taut[~T0P v T0P]	[2),T4-31,SynSUB]
3)	~F[~T0P v T0P]	[2),R7-2',MP]
4)	[T(~T0P v T0P) Syn ~F(~T0P v T0P)]	[T7-42]
5)	T[~T0P v T0P]	[3),4),SynSUB]
6)	T[~0P v 0P]	[5),T7-53,SynSUB(twice)]
7)	T[TP v FP v 0P]	[6),T7-49,SynSUB]

But among the T-wffs of Analytic Logic, and especially among the Logical Truths, the Law of Trivalence is unique in several ways. Let us see what marks it off.

It is, first of all, a Logical-Truth-theorem. This marks it off from all wffs which are contingent—i.e., have substitution instances that could be true or false. It is has no false instances.

T	[0P v	TP v	FP]
T	**T**0	T F0	T F0
T	FT	T **T**T	T FT
T	FF	F FF	T **T**F

<u>Secondly</u>, among LT-theorems, it has no inconsistent or tautologous proper disjuncts. Though as a whole it is logically true, each of its proper disjuncts is contingent—capable of having substitution instances which are either true or false. This is the case with '0P', 'TP', 'FP', '(0P v TP)', '(0P v FP)' and with '(TP v FP)'. Only the whole 1st-level T-wff, '(0P v TP v FP)' is tautologous and thus not-contingent. This separates Trivalence from many other sub-classes of Logical Truths, including those to which 'LT(∼TP v TP v ∼TP)', 'LT(∼TP v TP v 0P)' and 'LT(∼TP v TP v FQ)' belong.

<u>Third</u>, every disjunct is inconsistent with every other disjunct. That is, the conjunction of every pair of disjuncts is inconsistent (has no true substitution instances).

(TP & FP)	(TP & 0P)	(FP & 0P)	(TP & FP & 0P)
F0 **F** F0	F0 **F** T0	F0 **F** T0	F0 F F0 **F** T0
TT **F** FT	TT **F** FT	FT **F** FT	TT F FT **F** FT
FF **F** TF	FF **F** FF	TF **F** TF	FF F TF **F** FF

This distinguishes Trivalence from many other sub-classes of Logical Truths, including those to which 'T(∼TP v TP v ∼FP)' and 'T(∼TP v TP v ∼FP v FP)' and 'T(∼TP v TP v (FQ & ∼TQ))' belong.

<u>Fourth</u>, the negation of any one of its disjuncts is contained in the rest of the disjunction, as the following theorems show.

T7-55 [(0P v TP) Cont ∼FP] T7-56 [(0P v FP) Cont ∼TP] T7-49 [∼0P Syn (TP v FP)]

This is not true of LT-theorems like ⊨ T[(TP v ∼TP) v FP], ⊨ T[(TP v ∼TP) v 0P], etc.

<u>Fifth</u>, there are just eight truth-functions of a single variable in analytic truth logic. The six proper disjuncts of (0P v TP v FP) represent all six of the contingent truth-functions of a 1st-level single variable T-wff:

1	2	3	4	5	6			
TP	FP	0P	(TP v FP)	(TP v 0P)	(FP v 0P)	|	(TP v FP v 0P)	∼(TP v FP v 0P)
F0	F0	T0	F	T	T	|	T	F
TT	FT	FT	T	T	F	|	T	F
FF	TF	FF	T	F	T	|	T	F

The two other truth-functions of a T-wff with only a single atomic component P (shown on the right) are represented by (0P v TP v FP) which has all T's, and its negation which has all F's.

<u>Sixth</u>, of the four basic T-operators, TP, ∼TP, FP, and ∼FP, there are six non-synonymous disjunctive pairs. Three of them (1, 4, and 6) are the Logical-truth Theorems T7-72; T7-73 and T7-74. The others are not logically true.

1	2	3	4	5	6
<u>T(TP v ∼TP)</u>	<u>T(TP v FP)</u>	<u>T(TP v ∼FP)</u>	<u>T(∼TP v ∼FP)</u>	<u>T(∼TP v FP)</u>	<u>T(FP v ∼FP)</u>
T F0 T T F0	**F F0 F F0**	**T** F0 T T F0	**T** T F0 T T F0	**T** T F0 T F0	**T** F0 T T F0
T TT T F TT	**T** TT T FT	**T** TT T T FT	**T** F TT T T FT	**F F TT F FT**	**T** FT T T FT
T FF T T FF	**T** FF T TF	**F FF F F TF**	**T** T FT T F TF	**T** T FF T FT	**T** TF T F TF
^	^	^	^	^	^

T(OP v TP v FP) is synonymous with the conjunction of the three LT-wffs and it logically Contains all and only the three LT-theorems and their synonyms which are LF-theorems.

Although all of the three contained LT-theorems have the same truth-tables, they are not synonymous. In each pair of disjunctions there is at least one elementary first-level T-wff which occurs in one member of the pair but not in the other.

((~TP v TP) is not Syn to (~TP v ~FP)
and ((~FP v FP) is not Syn to (~TP v TP))
and ((~TP v ~FP) is not Syn to (~FP v FP)),

All three of them share the first four distinguishing characteristics 1) they are non-contingent T-wffs which are Logically true, 2) each of their proper disjuncts is contingent, 3) every disjunct is inconsistent with every other disjunct**,** 4) the negation of any one of its disjuncts is synonymous with the rest of the disjunction. Let us call any T-wff with these four characteristics a "presupposition".

Seventh, The Law of Trivalence logically contains all and only the three non-synonymous LT-theorems with just one predicate letter, which share the first four distinguishing characteristics listed above for the Law of Trivalence.

Given T7-71 T[0P v TP v FP]
and T7-69 [T(0P v TP v FP) Syn (T(~TP v TP) & T(~TP v ~FP) & T(~FP v FP))]

it follows that |= [T(~TP v TP) & T(~TP v ~FP) & T(~FP v FP))] and from this, by Simplification, we get Theorems T7-72 to T7-74, which constitute three **Laws of Excluded Middle** (but not "T(TP v FP)", the Bivalent Law)

T7-72. T[~TP v TP]
T7-73. T[~TP v ~FP]
T7-74. T[~FP v FP]

From these by DeMorgan rules, three **Laws of Non-Contradiction** (but not 'T[~(~TP& ~FP)]'):

T7-75. T[~(TP & ~TP)]	[T7-72, T4-16 (DeM6)]	Syn |= F[TP & ~TP]
T7-76. T[~(TP & FP)]	[T7-73, T4-18 (DeM8)]	Syn |= F[TP & FP]
T7-77. T[~(FP & ~FP)]	[T7-74, T4-16 (DeM6)]	Syn |= F[FP & ~FP]

These, then, are the basic presuppositions of Analytic Truth-logic and the Law of Trivalence is the most basic. It plays the role that the Law of Bivalence plays in M-logic. It is provable without paradox in A-logic from the definitions of 'T', 'F' and '0', while Bivalence is not provable in M-logic.

Finally, There is no conceivable alternative to the Law of Trivalence. Try to reject it, and one ends up affirming it. Its denial entails itself. To deny it one must deny each of the three disjuncts. This yields (~FP&~TP&~0P) which is Syn to (~FP&~TP& (TP v FP)) which is Syn to (0P & (TP v FP)) which entails (0P v (TP v FP)). This is not true of M-logic's law of Bivalence. If we deny (TP v FP) we get, (~TP & ~FP), which is OP (the contradictory of (TP v FP)) which entails (~TPv~FP) but not (TP v FP) itself. The denials of the three Laws of Excluded Middle also contain themselves. But while The Law Trivalence contains each of them, none of them contain the Law of Trivalence.

In M-logic, the models (i.e., truth-tables) which are used to justify the theorems exhaust the possible cases by presupposing that every wff is either true or false exclusively. The 2^n rows in the truth-table of any wff with **n** variables, then represent all possibilities. On finding that a wff comes out with T in the final column at every row, it is concluded that substitution instances for that wff will always be true.

In analytic truth-logic three "values", including "not-true and not-false", are needed if it is to cover logical properties and relations of all meaningful expressions, and solve the problem of conditionals. The Law of Trivalence provides the guiding principle for constructing trivalent truth-tables in A-logic, just as Bivalence provides the guiding principle for truth-table models in M-logic. Its trivalent truth-tables will have 3^n rows for any T-wff with **n** different atomic wffs. A necessary condition of synonymy in analytic truth-logic is that the two expressions be true, false or neither under the same conditions. Thus trivalent truth-tables provide a device for testing whether a necessary condition of logical synonymy is met: the final columns of the two truth- tables must be the same. For Containment the consequent must never be false if the antecedent is true. But no trivalent truth-table provides sufficient conditions to establish synonymity, containment, or validity, though they can provide decisive tests of inconsistency and tautology, unsatisfiability and unfalsifiability, and logical truth and logical falsehood as defined by Quine.

The Law of Trivalence is tautologous, unfalsifiable, and logically true, but none of these properties by themselves are of primary importance for logic. It is its role as a presupposition running through all of truth-logic which makes it valuable in defining the nature of conditionals, the distinction between logical validity and "empirical validity" of law-like statements, and the distinction between *de re* entailment and *de dicto* A-implication in the search for truth.

7.423 Implication Theorems of Chapter 7

The meaning of the term 'implication' in A-logic differs from its meaning in M-logic. All pairs of M-logic wffs such that A implies B in A-logic are also pairs such that A implies B in M-logic. But many pairs which are implicative in M-logic are not implications in A-logic. To avoid confusion we use 'A-implies' or 'A-implication' for the concept in A-logic and 'M-implies' or 'M-implication' for the concept of implication in M-logic, and we abbreviate 'P A-implies Q' as 'P Impl Q'. To distinguish an Impl-theorem from Cont- or Syn-theorems we will subscript the 'T' with 'i' in the theorem's name; e.g., Ti7-80 is the name of the theorem $\models$ [$\sim$TP Impl $\sim$T(P&Q)].

In M-logic, 'Implication' is a relation between two sentences such that a conjunction of the first with the negation of the second would be inconsistent. Every theorem of M-logic is a tautology, but theorems are not necessarily presented as TF-conditionals. However, every tautology is logically equivalent (and Syn) to some TF-conditional whose components are such that the antecedent M-implies the consequent.

If P M-implies Q, no relationship of meaning is required between the two sentences. If either the first sentence is inconsistent or the second is a denial of an inconsistency, the first sentence M-implies the second no matter what the other sentence may be. Thus every inconsistent sentence M-implies every sentence or expression, and every tautology (denial of an inconsistent expression) is M-implied by every sentence. These consequences have been called "Paradoxes of Strict Implication"; they are among the anomalies of M-logic. 'A-implication' requires a connection—a logical containment relation—between two expressions if one is to imply the other. This will elminate these "paradoxes of strict implication".

The problem for A-logic is that in ordinary language there appear to be many pairs of expressions which satisfy the condition that if the first is true the second must also be true, although the second is neither contained in the first nor synonymous with it. Two paradigmatic cases are:

1) If $\sim$TP then $\sim$T(P&Q) "If P is not true, then (P and Q) is not true"
2) If TP then T(PvQ) "If P is true, then (P or Q) is true."

The latter is usually called the Law of Addition and it appears (without the 'T's) as an axiom in most classical axiomatizations of M-logic.

We must account for these apparent laws, and also resolve the conflict between the fact that the consequent is *not* contained in the antecedent and A-logic's thesis that the consequent *must be* logically contained in the antecedent if it is to follow logically from the antecedent.

This problem is solved by first making a distinction between the relations of containment or entailment and the relation of "A-implication". A-implications are, on our definition, pairs of expressions such that the second follows logically from the first, though the first does not by itself logically contain the second. A-logic's requirement for containment is satisfied by interpreting A-implication as an *enthymematic* inference. Its unexpressed premiss is a substitution instance of the Law of Trivalence, the fundamental presupposition of Analytic truth-logic. To get to the underlying entailment from an A-implication one conjoins with the antecedent the substitution instance of the Law of Trivalence, ⊨ [0P v TP v FP] in which 'P' is replaced by the consequent, or conclusion, of the inference. All A-implications between wffs of M-logic are M-implications. But not all M-implications are A-implications, because those cases in which the antecedent logically Contains the consequent are not A-implications, by clause (ii) of the definition,

Df 'Impl'. [(TP Impl TQ) Syn_{df} (i) ((0QvTQvFQ) & TP) Cont TQ)
(ii) and Not: (TP Cont TQ)
(iii) and Not: (0QvTQvFQ) Cont TQ]

The last clause is necessary to avoid *non sequiturs*. If the implicandum (what is implied) is Contained in the presupposition, then we could put any expression for the antecedent, and the result of dropping the presupposition would be an implication which was a non- sequitur.

For example, suppose '(TR v ~TR)' is chosen as the consequent of an A-implication.
If '(TRv ~TR)' is put for 'P' in '(0P v TP v FP)' the result,
'0(TR&~TR) v T(TR&~TR) v F(TR&~TR)}', is Syn to '(TR v ~TR)'.
Therefore, Step 1) ⊨ [(T(TR&~TR) v F(TR&~TR) v 0(TR&~TR)] Cont T(TR v ~TR)]
Hence Step 2) ⊨ [((T(TR&~TR) v F(TR&~TR) v 0(TR&~TR)] **& TQ**) Cont T(TR v ~TR)]
Hence, if we drop the presupposition clause in accordance with Df 'Impl', we get
Step 3) ⊨ [TQ Impl T(TR v ~TR)] which is the kind of non-sequitur we must avoid.
(The proof of step1)in this proof is sketched out in the footnote).[41]

The principle forms of entailment from which A-implication can be drawn are the substitution instances of T7-64, T7-65 and T7-66. Thus since 'FP' does not contain 'F(P&Q)' (satisfying clause (ii)),

---

41. Sketch of Proof of ⊨ [((TR&~TR) v F(TR&~TR) v 0(TR&~TR)) Cont (TR v ~TR)]
1) ((TR&~TR)vF(TR&~TR)v 0(TR&~TR)) Syn (T(TR&~TR)vF(TR&~TR)v 0(TR&~TR))) [T1-11]
2) " Syn (T(TR&~TR) v (T~(TR&~TR) v (~T(TR&~TR) & ~F(TR&~TR))) [1),Df 'F',Df '0']
3) " Syn (((TR&~TR)v(~TRvTR)v((~TRvTR) & (TR&~TR})) [2) reduced , NF-Theorem]
4) " Syn (((TR&~TR)v(~TRvTR)v(~TRvTR)) & ((TR&~TR)v(~TRvTR)v(TR&~TR))) [3),v&-DIST]
5) " Syn (((TR&~TR)v(~TRvTR)) & ((TR&~TR)v(~TRvTR)) [4),v-ORD(twice)]
6) " Syn ((TR&~TR)v(~TRvTR)) [5),T119,T1-18]
7) ((TR&~TR)vF(TR&~TR)v0(TR&~TR)) Syn (TR v ~TR) [6),T1-19]
8) [((TR&~TR)vF(TR&~TR)v0(TR&~TR)) Cont (TR v ~TR)] [7),DR1-11]

and '((0(P&Q)vT(P&Q)vF(P&Q))' does not logically Contain 'F(P&Q)' (satisfying clause (iii), the implication " 'FP' implies 'F(P&Q)' " can be proved. For, when the Law of Trivalence (applied to '(P&Q)') is added to the antecedent, the resulting antecedent logically Contains the consequent.

1) |= [((T(P&Q)vF(P&Q)v0(P&Q)) & FP) Cont F(P&Q)] (Step 1 satisfies clause (i) Df 'Impl')
[T7-64,U-SUB{'(P&Q)'for 'Q'}]
2) Not: [(FP Cont F(P&Q)] [Inspection] (Step 2 satisfies clause (ii),Df 'Impl')
3) Not: [((0(P&Q)vT(P&Q)vF(P&Q)) Cont F(P&Q)] (Step 3 satisfies clause (iii),Df'Impl')
4) |= [FP Impl F(P&Q)] [1),2), 3), Df 'Impl']

Thus the conclusion, 4) '|= [FP Impl F(P&Q)]', may be viewed as an ellipsis, or abbreviation, of 1) with the substitution instance of the Law of Trivalence removed.

A-implications are never entailments. In an entailment the antecedent by itself must logically contain the consequent. In an A-implication the antecedent must *not* contain the consequent by clause (ii) in the definition of 'A-implication'. Entailments are never A-implications; for if PA-implies Q then P does not Contain Q, and if P entails Q then P must contain Q.

The deliberate separation of A-implication from entailment is to emphasize that A-implications are reliable only for *de dicto* inferences, whereas entailments can be relied on to convey *de re* relations even though they also depend on language. It separates out the process of adding irrelevant statements prototypically found in the Principle of Addition. The *truth* of [P v Q] follows from the *truth* of P because of the meaning of 'either or' and of 'True', even though Q may have no logical relation whatever to P. Q has nothing to do with it.

In A-implications truth-assertions *deduced* from the antecedent are not based on finding that the antecedent truly describes the facts in some objective (non-linguistic) field of reference based on observations in that field of reference. They are based on the meaning of "<1>is true" and the logical presuppositions of truth-logic. The meaning of "<1>is true" is implicitly involved in Axiom 7-5, which is always required in an A-implication so as to get the containment from the presupposition. In *de re* reasoning about non-linguistic factual truths, the containment of meanings of one expression in the meaning of another, corresponds to the containment of actual states of affairs in other actual states of affairs. It uses principles of logical inference in which one descriptive expression is logically Contained in another. How entailments differs from implications in *de re* reasoning, though both are *de dicto relations*, is discussed in more detail in Sect. 10.22.

7.4231 Basic Implication-theorems

Among the most basic Impl-theorems of analytic truth-logic are the following:

Ti7-80. [~TP Impl ~T(P & Q)]
Ti7-81. [FP Impl F(P & Q)]
Ti7-82. [~FP Impl ~F(P v Q)]
Ti7-83. [TQ Impl T(P v Q)]
Ti7-84. [TP Impl F(OP)] (i.e., [TP Impl (TPvFP)])
Ti7-85. [FP Impl F(OP)] (i.e., [FP Impl (TPvFP)])

In addition the following are provable:
|= [FP Impl F(P & ~Q)]
|= [~FQ Impl ~F(PvQ)]
|= [TQ Impl T(P ⊃ Q)]
|= [TP Impl T(PvQ)]
|= [FP Impl T(P ⊃ Q)]
|= [~FQ Impl ~F(P ⊃ Q)]
|= [(TP & 0P) Impl 0(P & Q)]

Proofs of Ti7-80 to Ti7-86 follow:

Ti7-80. [~ TP Impl ~ T(P&Q)]
Proof: 1) ((T(P&Q)vF(P&Q)v0(P&Q)) & ~ TP) Cont ~ T(P&Q) [T7-65]
2) Abbr. [~ TP Impl ~ T(P&Q)] [8),Df 'Impl']

Ti7-81. [FP Impl F(P&Q)]
Proof: 1) (T(P&Q)vF(P&Q)v0(P&Q)) & FP) Cont F(P&Q) [T7-64]
2) [FP Impl F(P&Q)] [1),Df 'Impl')]

Ti7-82. [~ FP Impl ~ F(PvQ)]
Proof: 1) ~ T ~ P Impl ~ T(~ P& ~ Q) [Ti7-80,U-SUBb]
2) ~ T ~ P Impl ~ T ~~ (~ P& ~ Q) [1),DN,SynSUB]
3) ~ FP Impl ~ F ~ (~ P& ~ Q) [2),Df 'F',SynSUB]
4) ~ FP Impl ~ F(PvQ) [3),Df 'v',SynSUB]

Ti7-83. [TP Impl T(P v Q)]
Proof: 1) F ~ P Impl F(~ P& ~ Q) [Ti7-81,U-SUBb]
2) T ~~ P Imp T ~ (~ P& ~ Q) [1),Df 'F',SynSUB(twice)]
3) TP Impl T ~ (~ P& ~ Q) [2),DN,SynSUB]
4) TP Impl T(PvQ) [3),Df 'v',SynSUB]

Ti7-84 TP Impl F(OP)
Proof: 1) ~ F(0P) Syn 0P [T7-54]
2) ~ F(~ TP& ~ FP) Syn (~ TP& ~ FP) [1),Df '0']
3) ~ F(~ TP& ~ FP) Cont ~ TP [2),Df 'CONT']
4) not-(TP CONT F(~ TP & ~ FP) [Inspection]
5) TP Impl F(~ TP& ~ FP) [3),4),Df 'Impl']
6) TP Impl F(0P) [5),Df.'0']

Ti7-85 FP Impl F(OP)
Proof: 1) ~ F(0P) Syn 0P [T7-54]
2) ~ F(~ TP& ~ FP) Syn (~ TP& ~ FP) [1),Df '0']
3) ~ F(~ TP& ~ FP) Cont ~ FP [2),Df 'CONT']
4) not -(FP CONT F(~ TP & ~ FP) [Inspection]
5) FP Impl F(~ TP& ~ FP) [3),4),Df 'Impl']
6) FP Impl F(0P) [5),Df.'0']

7.4232 Derived Inference Rules for A-implication

Several derived principles of inferences can be drawn from the definition of A-implication. Of course if A implies B, then A or B or any component in A or in B, can be replaced by a synonymous expression and the result is still an implication. This follows from the general principle of SynSUB. Replacements by SynSUB preserve all logical properties and relationships.

Proofs of Impl-theorems may be gotten, not only by satisfying the three clauses in

Df 'Impl'. [(TP Impl TQ) Syn_{df} (i) ((0QvTQvFQ) & TP) Cont TQ)
(ii) and Not: (TP Cont TQ)
(iii) and Not: (0QvTQvFQ) Cont TQ]

But also by the following rule: if P logically contains Q, but Q does not logically contain P and (0PvTPvFP) does not contain ~TP, then (~Q implies ~P). This is a qualified transposition principle. It proved as follows:

|= [If P Cont Q and not-(Q Cont P) and not-((0PvTPvFP) Cont, ~TP) then (~TQ Impl ~TP)

Proof: 1)	P Cont Q and not-(Q Cont P) and not-((0PvTPvFP) Cont, ~TP)	[Premiss]
2)	P Cont Q	[1),SIMP]
3)	(P Syn (Q&P))	[2),Df 'Cont']
4)	(P Cont (Q&P))	[3),DR1-11]
5)	not-(Q Cont P)	[1), SIMP]
6)	not-(Q Cont (Q&P))	[5),3),SynSUB]
7)	not-((0PvTPvFP) Cont, ~TP)	[1),SIMP]
8)	not-((0(Q&P)vT(Q & P) v F(Q&P)) Cont ~T(Q & P))	[7),3),U-SUB]
9)	|= [((0(Q&P)vT(Q&P) v F(Q&P)) & ~TQ) Cont ~T(Q&P)]	[T7-65,relettered]
10)	~TQ Impl ~T(Q&P)	[4),6),8),9),Df 'Impl']
11)	~TQ Impl ~TP	[10),3),SynSUB]
12)	[If P Cont Q and not-(Q Cont P) and not-((0PvTPvFP) Cont ~TP), then (~TQ Impl ~TP)]	[1) to 11),Cond. Pr.]

However, we shall not need to invoke this rule in what follows.

Additional rules can be derived which combine logical containment and A-implication; but these must be developed with caution. The first principle, which we call a CII-syllogism or "CIISyll", is: "If P CONT Q and TQ Impl TR and TP does not Contain TR, then TP Impl TR."

DRi7-4a. If P CONT Q and TQ Impl TR and Not-(TP Cont TR) then TP Impl TR "CIISyll"

Proof: 1)	a) (P CONT Q)	[Premiss]
	b) (TQ Impl TR)	[Premiss]
	c) not-(TP Cont TR)	[Premiss]
2)	TP Cont TQ	[1),a),R7-1d]
3)	(TP Syn (TP & TQ)	[2),Df 'Cont']
4)	((0RvTRvFR) & TQ) Cont TR)	[1),b),Df 'Impl']
5)	((0RvTRvFR) & (TP & TQ) Cont ((0RvTRvFR) & TQ)	[T1-11,Df 'Cont']
6)	(((0QvTQvFQ) & TP) Cont ((0RvTRvFR) & TQ)	[5),3),SynSUB]
7)	((0QvTQvFQ) & TP) Cont TR	[6),4),CCC-Syll]
8)	((0QvTQvFQ) & TP) Cont TR & not-(TP Cont TR)	[7),1)c),ADJ]
9)	[TP Impl TR]	[8),Df 'Impl']

The proviso Not-(TP Cont TR) is essential, for we have designed A-implication to account for just those kinds of logically valid conditional in which the consequent is <u>not</u> contained in the antecedent. Without this proviso it would follow from |= [(TP Impl (TP v TQ)] and |= [(P & Q) CONT P] that [(TP & TQ) <u>Impl</u> (TP v TQ)], although [(TP & TQ) Cont (TP v TQ)] by T1-38 and truth-logic. The separation of entailment and implication would be violated.

The second principle is called "the ICI-syllogism" or "ICISyll". The premisses of CII can appear in reverse order. Instead of the two premisses "P CONT Q and TQ Impl TR", ICISyll has the premisses "TP Impl TQ & Q CONT R". While CIISyll connects the consequent of the Cont-theorem with the antecedent of the Impl-theorem, ICISyll connects the consequent of the Impl-theorem with the antecedent of the Cont-theorem. The result is still valid. "If TP Implies TQ and Q CONT R and TP does not Contain

TR, then TP implies TR." (Obviously, if P is synonymous with Q or with R, this would violate the requirement that Implication and Containment be separated). ICISyll is proved as follows:

DRi7-4b. If TP Impl TQ and Q CONT R and Not-(TP Cont TR) then TP Impl TR "ICISyll"

Proof: 1)	a) (TP Impl TQ)	[Premiss]
	b) (Q CONT R)	[Premiss]
	c) not-(TP Cont TR)	[Premiss]
2)	TP Impl TQ	[1),a),SIMP]
3)	(Q SYN (R & Q)	[2),Df 'Cont']
4)	((0QvTQvFQ) & TP) Cont TQ)	[1),a),Df 'Impl']
5)	(((0QvTQvFQ) & TP) Cont T(R&Q)	[4),3),SynSUB]
6)	((0QvTQvFQ) & TP) Cont (TR&TQ)	[5),T7-03]
7)	((0QvTQvFQ) & TP) Cont TR	[6),T1-36,CCCSyll]
8)	((0QvTQvFQ) & TP) Cont TR & not-(TP Cont TR)	[7),1)c),ADJ]
9)	[TP Impl TR]	[8),Df 'Impl']

Again, the 'Not-(TP Cont TR)' clause is necessary to insure separation of implication and entailment.[42]

There are other valid principles of inference from combinations of Cont- and Impl-theorems but we will not pursue them here. However they will not include any statements which would make a Cont-theorem follow *simpliciter* from an Implication. '⊨ [P Impl Q]' can not entail or imply '⊨ [P Cont Q]', and '⊨ [P Cont Q]' can not entail or imply '⊨ [P Impl Q]'. And because P does not contain Q, if P Impl Q,

it is not the case that either a) if P Impl Q, then P Implies or contains Q
I.e., not: 'P Impl Q' **Impl** 'P Cont Q'
not: 'P Impl Q' **Cont** 'P Cont Q'
or b) if 'P cont Q' then 'P implies Q'
I.e., not: 'P Cont Q' **Impl** 'P Impl Q'
not: 'P Cont Q' **Cont** 'P Impl Q'

Hence not: 1) If P Impl Q & Q cont R, then P Cont R = No ICC-SYLL
2) If P Cont Q & Q Impl R, then P Cont R = No CIC-SYLL
3) If P Impl Q and Q Impl R, then P Cont R = No IIC- SYLL
4) If P Cont Q and Q Cont R, then P Impl R. = No CCI-SYLL
(For by definition, P only implies R if P does not Cont R.)

With 'S' for 'Syn', transitivity also fails for ISS-syll, ICS-syll, CIS-syll, CCS-syll and CSS-syll.

Finally, A-implications are **valid** if and only if they are satisfiable. Consequently, 'Valid[P Impl Q]' does not contain or imply 'Valid[∼Q Cont ∼P]'. For example Valid_I (TP => (TP v ∼TP)) is established in in Chapter 8, but [∼(TP v ∼TP) => ∼TP] is not-Valid because it is INC.

42. Without this clause, we would have only, "If ((TP Impl TQ) & Q CONT R) then (TP Impl TR)". But if we put '(P&R)' for 'P', and '((P&R)v(R&S))' for 'Q', in this conditional, then
1) 'TP Impl TQ' becomes '⊨ [T(P&R) Impl T((P&R)v(R&S))]'
which is Syn to '⊨ [T(P&R) Impl T(R&(PvS))]'
2) and 'Q CONT R' becomes '⊨ [((P&R)v(R&S)) CONT R]'
3) so that 'TP Impl TR' becomes '⊨ [T(P&R) Impl TR]', although ⊨ [T(P&R) Cont TR],
So 3) violates clause (ii) of Df 'Impl'.

7.4233 Principles Underlying Rules of the Truth-tables

In A-logic, implication-theorems are important in the explication of truth-tables. Truth- tables are devices which represent *de dicto* relations between compound expressions and their components. Each row in a truth-table represents a *de dicto* valid conditional; one which is valid because based on logical synonymy or containment or implication.

Nine Syn- and Cont-theorems provide principles of the truth-table rules. But the remaining 18 rules for '&', 'v' and '⊃' are based on implications which are purely *de dicto* principles. The nine containment or synonymy theorems provide grounds for the rules governing three of the nine rows in each of the trivalent truth-tables for '&', 'v' and '⊃':

In (P & Q) table:	In (P v Q) table:
1st row: T7-58. (0P & 0Q) Cont 0(P & Q)	T7-59. (0P & 0Q) Cont 0(P v Q)
5th row: Ax.7-3. (TP & TQ) Syn T(P & Q)	T7-38. (TP & TQ) Cont T(P v Q)
9th row: T7-37. (FP & FQ) Cont F(P & Q)	T7-23. (FP & FQ) Syn F(P v Q)

In (P ⊃Q) Table:
1st row: T7-60. (0P & 0Q) Cont 0(P ⊃ Q)
6th row: T7-39. (FP & TQ) Cont T(P ⊃ Q)
8th row: T7-40. (TP & FQ) Syn F(P ⊃ Q)

The other 18 rules in the truth-tables for '&', 'v' and '⊃' are based on implication-theorems established in this section. Valid inferential *C-conditionals* for all rows will be developed in full in the next chapter, based on the Cont-theorems or Impl-theorems in this chapter.

All rules for the singulary operators ' ~ ', 'T', and 'F' are based on containments and synonymies. Two rules for the truth-table for '0' are based on implications which are proved below. Truth-tables for the singulary operators:

	~P		T P
T7-52: 0P Syn 0(~ P)	0 0	T7-50: 0P Cont F(TP)	F 0
T7-16: TP Syn F(~ P)	F T	T7-20: TP Cont T(TP)	T T
Df 'F': FP Syn T(~ P)	T F	T7-18: FP Syn F(TP)	F F

	F P		0 P
T7-51: 0P Cont F(FP)	F 0	T7-53: 0P Syn T(0P)	T 0
T7-19: TP Cont F(FP)	F T	Ti7-84: TP Impl F(0P)	F T
T7-21: FP Syn T(FP)	T F	Ti7-85: FT Impl F(0P)	F F

It is readily seen, given the definition of valid conditionals in the preceding chapter, how these theorems may prepare the way (in the next chapter) for deriving valid C-conditionals such as "If P is 0, then ~ P is 0", "If P is true then ~ P is false", etc.

A complete set of Syn-, Cont-, and Impl-theorems sufficient to establish the rules of the truth-tables for '&', 'v' and '⊃' in Chapter 8, follows. These theorems are named as Chapter 7 theorems, ("T7-" or "Ti7-") with the connective of the table, and the number of the row (in our way of arranging the rows) to which the theorem applies. Rules for the **&-table** are based on the theorems:

T7-&R1. [(0P & 0Q) Cont 0(P & Q)] [T7-58]

Ti7-&R2. [(TP & 0Q) Impl 0(P & Q)]
<u>Proof</u>: 1) ((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0Q)) Cont 0(P&Q) [T7-66]
2) ((TP & 0Q) Impl 0(P&Q)) [1),Df 'Impl']

Ti7-&R3. [(FP & 0Q) Impl F(P & Q)]
<u>Proof</u>: 1) (FP & 0Q) Cont FP [T1-36,U-SUB]
2) FP Impl F(P & Q) [T7-81]
3) (FP & 0Q) Impl F(P & Q) [1),2),CII-SYLL]

Ti7-&R4. [(0P & TQ) Impl 0(P & Q)]
<u>Proof</u>: 1) [(TQ & 0P) Impl 0(Q & P)] [Ti7-&R2,U-SUB]
2) [(0P & TQ) Impl 0(Q & P)] [1),&-COMM,SynSUB]
3) [(0P & TQ) Impl 0(P & Q)] [2),&-COMM,SynSUB]

T7-&R5. [(TP & TQ) Syn T(P & Q)] [Ax.7-3]

Ti7-&R6. [(FP & TQ) Impl F(P & Q)]
<u>Proof</u>: 1) (FP & TQ) Cont FP [T1-36,U-SUB]
2) FP Impl F(P & Q) [T7-81]
3) (FP & 0Q) Impl F(P & Q) [1),2),CII-SYLL]

Ti7-&R7. [(0P & FQ) Impl F(P & Q)]
<u>Proof</u>: 1) [(0P & FQ) Cont FQ] [T1-37,U-SUB]
2) [FQ Impl F(Q & P)] [Ti7-81,U-SUB]
3) [(FQ Impl F(P & Q)] [2),&-COMM,SynSUB]
4) [(0P & FQ) Impl F(P & Q)] [1),3),CII-SYLL]

T$_I$7-&R8. [(TP & FQ) Syn F(P & Q)]
<u>Proof</u>: 1) [(TP & FQ) Cont FQ] [T1-37,U-SUB]
2) [(FQ Impl F(Q & P)] [Ti7-81,U-SUB]
3) [(FQ Impl F(P & Q)] [2),&-COMM,SynSUB]
4) [(TP & FQ) Impl F(P & Q)] [1),3),CII-SYLL]

T7-&R9. [(FP & FQ) Cont F(P & Q)] [T1-37]

Rules for the **v-table** are based on the next nine theorems:

T7-vR1. [(0P & 0Q) Cont 0(P v Q)] [T7-59]

T7-vR2. [(TP & 0Q) Impl T(P v Q)]
<u>Proof</u>: 1) (F~P & 0~Q) Impl F(~P & ~Q) [T7-&R3,U-SUB]
2) (F~P & 0~Q) Impl T~(~P & ~Q) [1),Df 'F',SynSUB]
3) (TP & 0~Q) Impl T~(~P & ~Q) [2),T7-16,SynSUB]
4) (TP & 0~Q) Impl T(P v Q) [3),Df 'v',SynSUB]

5) (0~Q Syn 0Q) [T7-52,U-SUB]
6) (TP & 0Q) Impl T(P v Q) [4),5),SynSUB]

Ti7-vR3. [(FP & 0Q) Impl 0(P v Q)]
Proof: 1) [(T~P & 0~Q) Impl 0(~P & ~Q)] [T7-&R2,U-SUB]
2) (FP & 0~Q) Impl 0(~P & ~Q) [1),T7-16,SynSUB]
3) (FP & 0~Q) Impl 0~(P v Q) [2),T4-17,SynSUB]
4) (FP & 0Q) Impl 0(P v Q) [3),T7-52,SynSUB(twice)]

Ti7-vR4. [(0P & TQ) Impl T(P & Q)]
Proof: 1) [(TQ & 0P) Impl T(Q v P)] [T7-vR2,U-SUB]
2) [(0P & TQ) Impl T(Q v P)] [1),&-COMM,SynSUB]
3) [(0P & TQ) Impl T(P v Q)] [2),v-COMM,SynSUB]

T7-vR5. [(TP & TQ) Cont T(P v Q)] [T7-38]

Ti7-vR6. [(FP & TQ) Impl T(P v Q)]
Proof: 1) (FP & TQ) Cont TQ [T1-36,U-SUB]
2) [(TQ Impl T(Q v P)] [Ti7-83,U-SUB]
3) [(TQ Impl T(P v Q)] [2),v-COMM,SynSUB]
4) (FP & TQ) Impl T(P v Q) [1),3),CII-SYLL]

Ti7-vR7. [(0P & FQ) Impl 0(P v Q)]
Proof: 1) [(0~P & T~Q) Impl 0(~P & ~Q)] [Ti7-&R2,U-SUB]
2) [(0~P & T~Q) Impl 0~(P v Q)] [1),T4-17,SynSUB]
3) [(0~P & FQ) Impl 0~(P v Q)] [1),Df 'F',SynSUB]
4) [(0P & TQ) Impl 0(P v Q)] [3),T7-52(twice),SynSUB]

Ti7-vR8. [(TP & FQ) Impl T(P v Q)] [T1-37]
Proof: 1) [(FQ & TP) Impl T(Q v P)] [T7-&R6,U-SUB]
2) [(TP & FQ) Impl T(Q v P)] [1),&-COMM,SynSUB]
3) [(TP & FQ) Impl T(P v Q)] [2),v-COMM,SynSUB]

T7-vR9. [(FP & FQ) Syn F(P v Q)] [T7-23]

Rules for the ⊃ **table** are all derivable from principles of rows in the v-table.

T7-⊃R1 [(0P & 0Q) Cont 0(P ⊃ Q)] [T7-60]
Proof: 1) (0~P & 0Q) Cont 0(~P v Q) [T7-vR1,U-SUB]
2) (0P & 0Q) Cont 0(~P v Q) [1),T7-52,SynSUB]
3) (0P & 0Q) Cont 0(P ⊃ Q) [1),T4-31,SynSUB]

Ti7-⊃R2 [(TP & 0Q) Impl 0(P ⊃ Q)]
Proof: 1) (F~P & 0Q) Impl 0(~P v Q) [Ti7-vR3,U-SUB]
2) (TP & 0Q) Impl 0(~P v Q) [1),T7-16,SynSUB]
3) (TP & 0Q) Impl 0(P ⊃ Q) [2),T4-31,SynSUB]

Ti7-⊃R3 [(FP & 0Q) Impl T(P ⊃ Q)]
Proof: 1) (T ~ P & 0Q) Impl T(~ P v Q) [Ti7-vR2,U-SUB]
2) (FP & 0Q) Impl T(~ P v Q) [1),T7-16,SynSUB]
3) (FP & 0Q) Impl T(P ⊃ Q) [2),T4-31,SynSUB]

Ti7-⊃R4 [(0P & TQ) Impl T(P ⊃ Q)]
Proof: 1) (0 ~ P & TQ) Impl T(~ P v Q) [Ti7-vR4,U-SUB]
2) (0P & TQ) Impl T(~ P v Q) [1),T7-52,SynSUB]
3) (0P & TQ) Impl T(P ⊃ Q) [2),T4-31,SynSUB]

Ti7-⊃R5 [(TP & TQ) Impl T(P ⊃ Q)]
Proof: 1) (F ~ P & TQ) Impl T(~ P v Q) [Ti7-vR6,U-SUB]
2) (TP & TQ) Impl T(~ P v Q) [1),T7-16,SynSUB]
3) (TP & TQ) Impl T(P ⊃ Q) [2),T4-31,SynSUB]

T7-⊃R6. (FP & TQ) Cont T(P ⊃ Q) [T7-39]
Proof: 1) (T ~ P & TQ) Cont T(~ PvQ) [T7-vR5,U-SUB]
2) (FP & TQ) Cont T(~ P v Q) [1),T7-16,SynSUB]
3) (FP & TQ) Cont T(P ⊃ Q) [2),T4-31,SynSUB]

Ti7⊃R7 [(0P & FQ) Impl 0(P ⊃ Q)]
Proof: 1) (0 ~ P & TQ) Impl 0(~ P v Q) [Ti7-vR7,U-SUB]
2) (0P & TQ) Impl 0(~ P v Q) [1),T7-52,SynSUB]
3) (0P & TQ) Impl 0(P ⊃ Q) [2),T4-31,SynSUB]

T7⊃R8 [(TP & FQ) Syn F(P ⊃ Q)] [T7-40]
Proof: 1) (F ~ P & FQ) Syn F(~ PvQ) [T7-vR9,U-SUB]
2) (TP & FQ) Syn F(~ P v Q) [1),T7-16.SynSUB]
3) (TP & FQ) Syn F(P ⊃ Q) [2),T4-31,SynSUB]

Ti7⊃R9 [(FP & FQ) Impl T(P ⊃ Q)]
Proof: 1) (T ~ P & FQ) Impl T(~ PvQ) [Ti7-vR8,U-SUB]
2) (FP & FQ) Impl T(~ P v Q) [1),Df 'F']
3) (FP & FQ) Impl T(P ⊃ Q) [2),T4-31,SynSUB]

Summarizing: For each row in the trivalent truth-tables of conjunction and disjunction, and theTF-conditional, the truth value (including '0') assigned to the compound is either synonymous with, or contained in, or implied by, an assignment of values to its components:

	Table for Conjunction	Table for Disjunction
Row 1:	(0P & 0Q) Cont 0(P&Q) [T7-58]	(0P & 0Q) Cont 0(PvQ) [T7-59]
Row 2:	(TP & 0Q) Impl 0(P&Q) [Ti7-&R2]	(TP & 0Q) Impl T(PvQ) [Ti7-vR2]
Row 3:	(FP & 0Q) Impl F(P&Q) [Ti7-&R3]	(FP & 0Q) Impl 0(PvQ) [Ti7-vR3]
Row 4:	(0P & TQ) Impl 0(P&Q) [Ti7-&R4]	(0P & TQ) Impl T(PvQ) [Ti7-vR4]

Row 5: (TP & TQ) Syn T(P&Q) [Ax.7-3]	(TP & TQ) Cont T(PvQ) [T7-38]
Row 6: (FP & TQ) Impl F(P&Q) [Ti7-&R6]	(FP & TQ) Impl T(PvQ) [Ti7-vR6]
Row 7: (0P & FQ) Impl F(P&Q) [Ti7-&R7]	(0P & FQ) Impl 0(PvQ) [Ti7-vR7]
Row 8: (TP & FQ) Impl F(P&Q) [Ti7-&R8]	(TP & FQ) Impl T(PvQ) [Ti7-vR8]
Row 9: (FP & FQ) Cont F(P&Q) [T7-37]	(FP & FQ) Syn F(PvQ) [T7-23]

Table for the TF-conditional

Row 1: (0P & 0Q) Cont 0(P ⊃ Q) [T7-60]
Row 2: (TP & 0Q) Impl T(P ⊃ Q) [Ti7⊃R2]
Row 3: (FP & 0Q) Impl 0(P ⊃ Q) [Ti7⊃R3]
Row 4: (0P & TQ) Impl T(P ⊃ Q) [Ti7⊃R4]
Row 5: (TP & TQ) Impl T(P ⊃ Q) [Ti7⊃R5]
Row 6: (FP & TQ) Cont T(P ⊃ Q) [T7-39]
Row 7: (0P & FQ) Impl 0(P ⊃ Q) [Ti7⊃R7]
Row 8: (TP & FQ) Syn T(P ⊃ Q) [T7-40]
Row 9: (FP & FQ) Impl F(P ⊃ Q) [Ti7⊃R9]

7.4234 A-implication in Quantification Theory

In Section 7.423 we defined 'TP Impl TQ' as follows,

[(TP implies TQ) Syn_{df} (i) ((0QvTQvFQ) & TP Cont TQ
(ii) and not: (TP Cont TQ)
(iii) and not: (0QvTQvFQ) Cont TQ]

From any implication rule represented by a row in the trivalent truth-tables for logical constants, we can derive a quantified implication. For example,

From Ti7-&R2. [(TP & 0Q) Impl 0(P & Q), |= [(∀x)(TPx & 0Qx) Impl (∀x)0 (Px & Qx)
From Ti7-vR3. [(FP & 0Q) Impl 0(P v Q)], |= (∀x)(FPx & 0Qx) Impl (∀x)0 (Px v Qx), etc.,

Also the basic implications discussed in Section 7.423 can be generalized by prefixing quantifiers to the antecedent and the consequent.

Ti7-80. [~TP Impl ~T(P & Q)],	hence, |= [(∀x)~TPx Impl (∀x)~T(Px & Qx)]
Ti7-81. [FP Impl F(P & Q)],	hence, |= [(∀x) FPx Impl (∀x) F(Px & Qx)]
Ti7-82. [~FP Impl ~F(P v Q)],	hence, |= [(∀x)~FPx Impl (∀x)~F(Px v Qx)]
Ti7-83. [TQ Impl T(P v Q)],	hence, |= [(∀x) TPx Impl (∀x) T(Px v Qx)]
Ti7-84. [TP Impl F(0P)],	hence, |= [(∀x) TPx Impl (∀x) (TPx v FQx)]
Ti7-85. [FP Impl F(0P)],	hence, |= [(∀x) FPx Impl (∀x) (TPx v FQx)]

For, by the definition of 'Impl' any Impl-theorem, |= [P Impl Q], entails a constructible Cont-theorem of the form |= [(0Q v TQ v FQ) & TP Cont TQ].

If ⊨ [(P<1> Impl Q<1>) then ⊨ ((0Q<1> v TQ<1> v FQ<1>) & TP<1>) Cont TQ<1>]

And the quantifications follow by the derived rule,

DR3-3e. If [P<1> Cont Q<1>], then [(∀x)Px Cont (∀x)Qx], (see Section 3.323)

yielding, ⊨ [(∀x)(**(0Qx v TQx v FQx)** & TPx) Cont (∀x)TQx]

from which the clause in **bold** expressing the presupposition of Trivalence can be dropped, leaving,

⊨ [(∀x)TPx Impl (∀x)TQx]

If the indicated Containment relation holds between the predicates of P and Q, it will hold, within any given domain, between the conjunctive generalization of P<1∀> and a conjunctive generalization of Q<1>. The Boolean expansions of whatever antecedent is put for 'P<1>' in '(∀x)Px', will contain for each individual in the domain, a conjunct of the over-all form '(TQa_i v FQa_i v $0Qa_i$)'. These conjuncts are instantiations of the Law of Trivalence which may be removed as presuppositions, allowing the elliptical expressions, [TPa_i Impl TQa_i]. Thus we can get the Derived Rule for generalized A-implication:

[If P and Q are normal form T-wffs and ⊨ (P Impl Q) then ⊨ (∀x)Px Impl (∀x)Qx)]

Assuming these implications have met all of the conditions in Df 'Impl', they remain merely *de dicto*. They have a role to play and the rules hold, but they must not be confused with *de re* entailments.

Of particular interest are *de dicto* principles which cause problems when confusedly taken as *de re* scientific laws and generalizations, including:

Ti7-86. [(∀x)TQx Impl (∀x)(0Px ⊃ TQx)
Ti7-87. [(∀x)TQx Impl (∀x)(TPx ⊃ TQx) Ti7-83a,U-SUB,T4-31,UG]
Ti7-88. [(∀x)TQx Impl (∀x)(FPx ⊃ TQx)
Ti7-89. [(∀x)∼TPx Impl (∀x)(TPx ⊃ FQx)
Ti7-90. [(∀x)FPx Impl (∀x)(TPx ⊃ FQx)
Ti7-91. [(∀x)TPx Impl (∀x)(∼TPx ⊃ TQx)

As we said, A-implications are only valid *de dicto*; they are true by reference to the meaning we given to negation with other syntactical and meta-lingistic elements in our language. They are not valid *de re*. They can not be relied upon to make deductions about facts in fields of reference. The examples above introduce confusion in the logic of inductive inference, confirmation and contrary-to-fact conditionals if they are proposed for use in empirical truth-searchs. These confusions can be eliminated by recognizing the distinction between implication and containment and refraining from using the latter in inferences to matter of fact.

In A-logic the Rule called "Existential Generalization" in M-logic is, as we have said, misnamed, being essentially a *de dicto* rule, rather than *de re*. Essentially, it proceeds from a particular case to a disjunction of all cases in a field of reference. This adds no information; the conclusion is less informative than its premiss. It says that if a particular statement is true, then the disjunction of that statement with many other statements is true. In this, like the rule of addition, the meaning of 'T' and 'either or' and the Presupposition of Trivalence make the indefinite disjunction true.[43]

Ti7-92. T(Pa$_i$) Impl T(∃x)Px, (Hence, T(Pa$_i$) Impl (∃x)TPx))

Proof: In any domain of n individuals, if 1 ≤ i ≤ n,

1) T(∃x)Px Syn T(Pa$_1$ v … v Pa$_n$) [Df '∃']
2) T ~ (∃x)Px Syn T(∀x) ~ Px [T4-25(Q-Exch2),R7-1]
3) 0(∃x)Px Syn (~ T(∃x)Px & ~ F(∃x)Px) [Df '0']
4) 0(∃x)Px Syn (~ T(∃x)Px & T(∀x) ~ Px) [3),2),SynSUB]
5) (T(∃x)Px v F(∃x)Px v 0(∃x)Px) Syn (T(∃x)Px v T(∀x) ~ Px v (~ T(∃x)Px & T(∀x) ~ Px)) [1),2),4) SynSUB]
6) (T(∃x)Px v F(∃x)Px v 0(∃x)Px) Syn (T(∃x)Px v T(∀x) ~ Px v ~ T(∃x)Px) & (T(∃x)Px v T(∀x) ~ Px v T(∀x) ~ Px)) [5),v&-DIST]
7) (T(∃x)Px v F(∃x)Px v 0(∃x)Px) Cont (T(∃x)Px v T(∀x) ~ Px) [6),Df 'Cont']
8) (T(∃x)Px v T(∀x) ~ Px) Syn (T(∃x)Px v (FPa$_1$ & **FPa$_i$** & … & FPa$_n$)) [7), Df '∀',Df 'F']
9) (T(∃x)Px v T(∀x) ~ Px) Syn ((T(∃x)Px v FPa$_1$) & (**T(∃x)Px v FPa$_i$**) & … & (T(∃x)Px v FPa$_1$)) [8),Gen &vDIST]
10) (T(∃x)Px v T(∀x) ~ Px) Cont (**T(∃x)Px v FPa$_i$**) [9),Df 'Cont']
11) ((T(∃x)Px v T(∀x) ~ Px) & **TPa$_i$** Cont (T(∃x)Px v FPa$_i$) & **TPa$_i$**) [10),DR1-21]
12) (T(∃x)Px v F(∃x)Px v 0(∃x)Px) & **TPa$_i$** Cont (T(∃x)Px v T(∀x) ~ Px) & **TPa$_i$**) [7),DR1-21]
13) (T(∃x)Px v F(∃x)Px v 0(∃x)Px) & TPa$_i$ Cont (T(∃x)Px v FPa$_i$) & TPa$_i$) [12),11),CCC-Syll]
14) [(TP & T(~ P v Q)) Cont TQ] [T7-46]
15) [(TP & T(Q v ~ P)) Cont TQ] [14), v-COMM]
16) [(TP & (TQ v FP)) Cont TQ] [15),Ax.7-4, Df 'F']
17) [(TQ v FP) & TP) Cont TQ] [16) &-COMM]
18) [(**T(∃x)Px v FPa$_i$**) & TPa$_i$) Cont T(∃x)Px] [17),U-SUB]
19) [(T(∃x)Px v F(∃x)Px v 0(∃x)Px) & TPa$_i$) Cont T(∃x)Px] [13),18),CCCSyll]
20) Not: ((T(∃x)Px v F(∃x)Px v 0(∃x)Px) Cont T(∃x)Px) [Inspection]
21) Not: (TPa$_i$ Cont T(∃x)Px) [Inspection]
22) [TPa$_i$ Impl T(∃x)Px] [19),20),21),Df'Im[pl']

Ti7-93. [F(Pa$_i$) Impl F(∀x)Px] (Hence, F(Pa$_i$) Impl (∃x)FPx)

Proof: In any domain of n individuals, if 1 ≤ i ≤ n,

1) T ~ Pa$_i$ Impl (∃x)T ~ Px [Ti7-92,U-SUB]
2) FPa$_i$ Impl (∃x)FPx [1),Df 'F']
3) FPa$_i$ Impl F(∀x)Px [2),T7-29, SynSUB]

Ti7-94. ~ T(Pa$_i$) Impl ~ T(∀x)Px,

Proof: In any domain of n individuals, if 1 ≤ i ≤ n,

1) T ~ T(∀x)Px Syn (∃x) ~ TPx [T4-25(Q-Exch2),R7-1]
2) F ~ T(∀x)Px Syn T(∀x)TPx [Df 'F',DN,SynSUB]
3) 0 ~ T(∀x)Px Syn (~ T ~ T(∀x)Px & ~ F ~ T(∀x)Px) [Df '0']

43. The philosophical point is related to Russell's view that there are no disjunctive facts: "I do not suppose there is in the world a single disjunctive fact correspnding to 'P or Q' ". From "The Philosophy of Logical Atomism", Section III, in Bertrand Russell, *Logic and Knowledge*, 1956, p. 209]

4) 0 ~ T(∀x)Px Syn (T(∀x)Px & ~ T(∀x)Px) [Df 'F',DN,T7-20,Ax.7-2,SynSUB]
5) (T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) Syn (~ T(∀x)Px v T(∀x)Px v (T(∀x)Px & ~ T(∀x)Px) [1),2),4) SynSUB]
6) (T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) Syn ((~ T(∀x)Px v T(∀x)Px v T(∀x)Px) & (~ T(∀x)Px v T(∀x)Px v ~ T(∀x)Px)) [5),v&-DIST]
7) (T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) Cont ((~ T(∀x)Px v T(∀x)Px) [6),Df 'Cont']
8) ((T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) & **~ TPa$_i$**) Cont ((~ T(∀x)Px v T(∀x)Px) & **~ TPa$_i$**] [7),DR1-21]
9) ((~ T(∀x)Px v T(∀x)Px) Syn (~ T(∀x)Px v (TPa$_1$ & **TPa**$_i$ & … & Tpa$_n$)) [Df '∀']
10) ((~ T(∀x)Px v T(∀x)Px) Syn ((~ T(∀x)Px v TPa$_1$) & (~ T(∀x)Px v **TPa**i) &…& (~ T(∀x)Px v TPa$_n$)) [9),Gen &vDIST]
11) ((~ T(∀x)Px v T(∀x)Px) Cont **(~ T(∀x)Px v TPa$_i$)** [10),Df 'Cont']
12) ((~ T(∀x)Px v T(∀x)Px) & **~ TPa$_i$** Cont ((~ T(∀x)Px v TPa$_i$) & **~ TPa$_i$**) [11),DR1-21]
13) ((T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) & ~ TPa$_i$) Cont ((~ T(∀x)Px v TPa$_i$) & ~ TPa$_i$) [8),12),CCCSyll]
14) [((~ TP v TQ) & ~ TQ) Cont ~ TP] [T7-45]
15) (~ T(∀x)Px v T(∀x)Px) & ~ TPa$_i$) Cont ~ T(∀x)Px [14),U-SUB]
16) ((T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) & ~ TPa$_i$) Cont ~ T(∀x)Px] [13),15),CCCSyll]
17) Not: ((T ~ T(∀x)Px v F ~ T(∀x)Px v 0 ~ T(∀x)Px) Cont ~ T(∀x)Px [Inspection]
18) Not: (~ TPa$_i$ Cont ~ T(∀x)Px) [Inspection]
19) [~ TPa$_i$ Impl ~ T(∀x)Px] [16),17),18),Df 'Impl']

From Ti7-94 the following theorems also follow:

|= [~ TPa$_i$ Impl ~ (∀x)TPx] [Ti7-94,T7-24,SynSUB]
|= [~ TPa$_i$ Impl (∃x) ~ TPx] [Q-Exch]

Ti7-95. [~ FPa$_i$ Impl ~ F(∃x)Px]

<u>Proof</u>: In any domain of n individuals, if $1 \leq i \leq n$,

1) [~ T ~ Pa$_i$ Impl ~ T(∀x) ~ Px] [Ti7-94,U-SUB]
2) [~ T ~ Pa$_i$ Impl ~ T ~ (∃x)Px] [1),Q-Exch]
3) [~ FPa$_i$ Impl ~ F(∃x)Px] [2),Df 'F']

From Ti7-95 the following theorems also follow:

|= [~ FPa$_i$ Impl (∃x) ~ FPx] [Ti7-95,T7-27,SynSUB]
|= [~ FPa$_i$ Impl ~ (∀x)FPx] [Q-Exch]

7.424 Valid Inference Schemata

With the introduction of T-operators, the set of ***valid inferences*** is increased beyond that of Chapter 5. It is increased in many ways. We first present Validity theorems based on entailment, and then present Validity-theorems based on A-implications.

7.4241 Valid Inference Schemata Based on Entailments

One way to derive a Validity-theorem of truth-logic, is to prefix T-operators to all atomic wffs in the valid inference Schemata of Section 5.52, using U-SUBab. Each of the 110 valid inferences schemata listed in Section 5.52, were based on SYN- and CONT-theorems in Chapters 1 to 4. Each of them becomes a theorem of truth-logic by using U-SUBab to prefix a T-operator to each predicate letter. For example,

from T5-431b. VALID [(~PvQ), .: (P ⊃ Q)] [T4-31,DR5-6b]
we derive ⊨ Valid [(~TP v TQ), .: (TP ⊃ TQ)] [U-SUB with 'TP' for 'P' and 'TQ' for 'Q']

and from T5-437a. VALID [(∀x)(Px ⊃ Qx), .: ((∀x)Px ⊃ (∀x)Qx)] [T4-37,DR5-6a]
we derive ⊨ Valid [(∀x)(TPx ⊃ TQx), .: ((∀x)TPx ⊃ (∀x)TQx)]
[U-SUBa with 'TP' for 'P' and 'TQ' for 'Q']

In converting purely formal wffs into T-wffs for truth-logic, U-SUBab can start by substituting 'TP' for 'P' uniformly throughout purely formal wffs by U-SUBa, and then replace 'P' by '~P' under U-SUBb to get 'T~P; or it can replace 'P' by 'TP' at all occurrences of '~P' to get '~TP'. Since U-SUB requires that such substitutions be made at all occurrences of P this rule can not result in any atomic wff's occurring both POS and NEG if it did not so occur before the substitution, so it can not introduce any new inconsistencies.

All 110 validity theorems of Chapter 5 can be converted into Validity theorems of analytic truth-logic in this way. Because non-syncategorematic T-operators are introduced, "VALID" changes to "Valid". Thus the first five of the 110 theorems listed yield,

⊨ Valid [TP, .: (TP & TP)]	[&-IDEM]	[T5-101b,U-SUBa,(P/TP)]
⊨ Valid [TP, .: (TP v TP)]	[v-IDEM]	[T5-102b,U-SUBa,(P/T~P)]
⊨ Valid [(TP & TQ), .: (TQ & TP)]	[&-COMM]	[T5-103b,U-SUBa,(P/T~P)]
⊨ Valid [(TP v TQ), .: (TQ v TP)]	[v-COMM]	[T5-104b,U-SUBa,(P/T~P)]
⊨ Valid [(TP & (TQ & TR)), .: ((TP & TQ) & TR)]	[&-ASSOC]	[T5-105b,U-SUBa,(P/T~P)]

Replacing 'P' by '~P' under U-SUBb in these theorems (or 'P' by 'T~P' in theorems from Chapter 5),

⊨ Valid [T~P, .: (T~P & T~P)]	[&-IDEM]	[T5-101b,U-SUBa,(P/T~P)]
⊨ Valid [T~P, .: (T~P v T~P)]	[v-IDEM]	[T5-102b,U-SUBa,(P/T~P)]
⊨ Valid [(T~P & TQ), .: (TQ & T~P)]	[&-COMM]	[T5-103b,U-SUBa,(P/T~P)]
⊨ Valid [(T~P v TQ), .: (TQ v T~P)]	[v-COMM]	[T5-104b,U-SUBa,(P/T~P)]
⊨ Valid [(T~P & (TQ & TR)), .: ((T~P & TQ) & TR)]	[&-ASSOC]	[T5-105b,U-SUBa,(P/T~P)]

Replacing P by ~P under U-SUBb in the initial Ch 1-5 theorems yields

⊨ Valid [~P, .: (~P & ~P)]	[&-IDEM]	[T5-101b,U-SUBb (P/~P)]]
⊨ Valid [~P, .: (~P v ~P)]	[v-IDEM]	[T5-102b,U-SUBb (P/~P)]
⊨ Valid [(~P & Q), .: (Q & ~P)]	[&-COMM]	[T5-103b,U-SUBb (P/~P)]
⊨ Valid [(~P v Q), .: (Q v ~P)]	[v-COMM]	[T5-104b,U-SUBb (P/~P)]
⊨ Valid [(~P & (Q & R)), .: ((~P & Q) & R)]	[&-ASSOC]	[T5-105b,U-SUBb (P/~P)]

then Replacing 'P' by 'TP' in the theorems just given, (or 'P' by '∼TP' in the initial theorems),

|= Valid [∼TP, **.:** (∼TP & ∼TP)] [&-IDEM] [T5-101b,U-SUBa,(P/∼TP)]
|= Valid [∼TP, **.:** (∼TP v ∼TP)] [v-IDEM] [T5-102b,U-SUBa,(P/∼TP)]
|= Valid [(∼TP & TQ), **.:** (TQ & ∼TP)] [&-COMM] [T5-103b,U-SUBa,(P/∼TP)]
|= Valid [(∼TP v TQ), **.:** (TQ v ∼TP)] [v-COMM] [T5-104b,U-SUBa,(P/∼TP)]
|= Valid [(∼TP & (TQ & TR)), **.:** ((∼TP & TQ) & TR)] [&-ASSOC] [T5-105b,U-SUBa,(P/∼TP)]

Replacing P by ∼P by U-SUBb in the theorems just above (or 'P' by '∼T∼P' in the initial Chapter 5 theorems),

|= Valid [∼T∼P, **.:** (∼T∼P & ∼T∼P)] [&-IDEM] [T5-101b,U-SUBa,(P/∼T∼P)]
|= Valid [∼T∼P, **.:** (∼T∼P v ∼T∼P)] [v-IDEM] [T5-102b,U-SUBa,(P/∼T∼P)]
|= Valid [(∼T∼P & TQ), **.:** (TQ & ∼T∼P)] [&-COMM] [T5-103b,U-SUBa,(P/∼T∼P)]
|= Valid [(∼T∼P v TQ), **.:** (TQ v ∼T∼P)] [v-COMM] [T5-104b,U-SUBa,(P/∼T∼P)]
|= Valid [(∼T∼P & (TQ & TR)), **.:** ((∼T∼P & TQ) & TR)]
[&-ASSOC] [T5-105b,U-SUBa,(P/∼T∼P)]

Of course any or all occurrences of 'T∼P' can be replaced by 'FP', by Df 'F' and SynSUB.

By using U-SUBab, theorems derived in this way continue to satisfy the non-Inc requirement for validity which was satisfied by the 110 valid inferences schemata listed in Section 5.52.

A second way to derive Valid inference schemata of truth-logic, is to begin with the Syn- and Cont-theorems peculiar to this chapter. These theorems already have only T-wffs as components. So derivations can use DR5-6a and DR5-6b or 6c or 6d with U-SUBab to get the desred results. The justification on the right will read, '[T7-01,DR5-6b,U-SUB]', for example, to signify that the theorem numbered T7-701 can be proved by U-SUB and DR5-6a provided the 'not-inc' clause in DR5-6a and DR5-6b is satisfied. Thus

T7-701 Valid[TP, **.:** (TP & ∼FP)] [T7-01,DR5-6b,U-SUB]

stands for the proof,

T7-701 Valid [TP, **.:** (TP & ∼FP)]
Proof: 1) (TP Syn (TP & ∼FP)) [**T7-01**]
2) If (P Syn Q) & (ii) not-Inc(P&Q) then Valid (P, **.:**Q)] **[DR5-6b]**
3) If (TP Syn (TP & ∼FP) & not-Inc(TP & TP & ∼FP) then Valid [TP, **.:**(TP & ∼FP)]
[2),**U-SUB**,'TP'for 'P','(TP & ∼FP)'for'Q']
4) Not-Inc(TP & TP & ∼FP) [Inspection]
tt t tt **t** t t t
5) ((TP Syn (TP & ∼FP)) & not-Inc(TP & TP & ∼FP) [1),4) ADJ]
6) Valid [TP, **.:**(TP & ∼FP)] [5),3),MP]

Step 4) can be done in each of the Validity-theorems.below, but will be left for the reader to check. The Syn- and Cont-theorems of this chapter yield the following validity theorems:

From CHAPTER 7, SYN- and CONT-theorems (with C-conditionals)

Df 'F'. Valid [F(P), .:T ~ P] [Df 'F',DR5-6b,U-SUB]
Df '0'. Valid [0(P), .: (~ TP & ~ FP)] [Df '0',DR5-6b,U-SUB]

Syn and Cont-theorems of Chapter 7.

T7-701 Valid [TP, .: (TP & ~ FP)] [T7-01,DR5-6b,U-SUB]
T7-702. Valid [FTP, .: ~ TP] [T7-02,DR5-6b,U-SUB]
T7-703. Valid [T(P & Q), .: (TP & TQ)] [T7-03,DR5-6b,U-SUB]
T7-704. Valid [T(P v Q), .: (TP v TQ)] [T7-04,DR5-6b,U-SUB]
T7-705. Valid [T((TP & ~ TP) v Q)), .: TQ] [T7-05,DR5-6a,U-SUB]
T7-706. Valid [F(P), .: T ~ P] [T7-06,DR5-6b,U-SUB]
T7-707. Valid [0(P), .: (~ TP & ~ FP)] [T7-07,DR5-6b,U-SUB]

From 7.42121. Valid Inference-Theorems from Axioms 7-1 and Df 'F'

T7-711. Valid [TP, .: TP] [T7-11,DR5-6b,U-SUB]
T7-712. Valid [FP, .: FP] [T7-12,DR5-6b,U-SUB]
T7-713. Valid [TP, .: ~ FP] [T7-13,DR5-6a,U-SUB]
T7-714. Valid [FP, .: (FP & ~ TP)] [T7-14,DR5-6b,U-SUB]
T7-715. Valid [FP, .: ~ TP] [T7-15,DR5-6a,U-SUB]
T7-716. Valid [F ~ P, .: TP] [T7-16,DR5-6b,U-SUB]

From 7.42122 Validity-Theorems for Reduction to Normal Form T-wffs from Ax.7-1 to 7-4

T7-717. Valid [FFP, .: ~ FP] [T7-17,DR5-6b,U-SUB]
T7-718. Valid [FP, .: FTP] [T7-18,DR5-6a,U-SUB]
T7-719. Valid [TP, .: FFP] [T7-19,DR5-6a,U-SUB]
T7-720. Valid [TTP, .: TP] [T7-20,DR5-6b,U-SUB]
T7-721. Valid [TFP, .: FP] [T7-21,DR5-6b,U-SUB]
T7-722. Valid [F(P & Q), .: (FP v FQ)] [T7-22,DR5-6b,U-SUB]
T7-723. Valid [F(P v Q), .: (FP & FQ)] [T7-23,DR5-6b,U-SUB]
T7-724. Valid [T(∀x)Px, .: (∀x)TPx] [T7-24,DR5-6b,U-SUB]
T7-725. Valid [T(∃x)Px, .: (∃x)TPx] [T7-25,DR5-6b,U-SUB]
T7-726. Valid [~ F(∀x)Px, .: (∀x) ~ FPx] [T7-26,DR5-6b,U-SUB]
T7-727. Valid [~ F(∃x)Px, .: (∃x) ~ FPx] [T7-27,DR5-6b,U-SUB]
T7-728. Valid [F(∃x)Px, .: (∀x)FPx] [T7-28,DR5-6b,U-SUB]
T7-729. Valid [F(∀x)Px, .: (∃x)FPx] [T7-29,DR5-6b,U-SUB]
T7-730. Valid [~ T(∃x)Px, .: (∀x) ~ TPx] [T7-30,DR5-6b,U-SUB]
T7-731. Valid [~ T(∀x)Px, .: (∃x) ~ TPx] [T7-31,DR5-6b,U-SUB]

From 7.42123—Other Validity-theorems from Axioms 7-1 to 7-4

T7-732. Valid [~ FTP, .: TTP] [T7-32,DR5-6b,U-SUB]
T7-733 .Valid [~ FFP, .: TFP] [T7-33,DR5-6b,U-SUB]
T7-734. Valid [T(P ⊃ Q), .: (TP ⊃ TQ)] [T7-34,DR5-6a,U-SUB]
T7-735. Valid [T(P ⊃ Q), .: T(~ Q ⊃ ~ P)] [T7-35,DR5-6b,U-SUB]
T7-736. Valid [(TP ⊃ TQ), .: (~ TQ ⊃ ~ TP)] [T7-36,DR5-6b,U-SUB]
T7-737. Valid [(FP & FQ), .: F(P & Q)] (For &-table, Row 9) [T7-37,DR5-6a,U-SUB]
T7-738. Valid [(TP & TQ), .: T(P v Q)] (For v-table, Row 5) [T7-38,DR5-6a,U-SUB]
T7-739. Valid [(FP & TQ), .: T(P ⊃ Q)] (For ⊃-table, Row 6) [T7-39,DR5-6a,U-SUB]
T7-740. Valid [(TP & FQ), .: F(P ⊃ Q)] (For ⊃-table, Row 8) [T7-40,DR5-6b,U-SUB]

T7-741. Valid [F(TP & ∼TQ), .: ∼T(TP & ∼TQ)] [T7-41,DR5-6b,U-SUB]
T7-742. Valid [T(∼TP v TQ), .: ∼F(∼TP v TQ)] [T7-42,DR5-6b,U-SUB]
T7-743. Valid [T(∀x)(Px ⊃ Qx), .: (T(∀x)Px ⊃ T(∀x)Qx] [T7-43,DR5-6a],U-SUB]

From 7.42124 Validity-Theorems of Detachment, from Ax.7-5)

T7-744. Valid [(TP & (∼TPvTQ)), .: TQ] "Alternative Syllogism #1" [T7-44,DR5-6a,U-SUB]
T7-745. Valid [((∼TPvTQ) & ∼TQ), .: ∼TP] "Alternative Syllogism #2" [T7-45,DR5-6a,U-SUB]
T7-746. Valid [(TP & T(∼PvQ)), .: TQ] "Alternative Syllogism #3" [T7-46,DR5-6a,U-SUB]
T7-747. Valid [(T(∼PvQ) & FQ), .: FP] "Alternative Syllogism #4" [T7-47,DR5-6a,U-SUB]

From 7.42125. Validity-Theorems about not-true-and-not-false expressions, from Df '0'

T7-748. Valid [0P, .: ∼(TP v FP)] [T7-48,DR5-6b,U-SUB]
T7-749. Valid [∼0P, .: (TP v FP)] [T7-49,DR5-6b,U-SUB]
T7-750. Valid [0P, .: ∼TP] [T7-50,DR5-6a,U-SUB]
T7-751. Valid [0P, .: ∼FP] [T7-51,DR5-6a,U-SUB]
T7-752. Valid [0∼P, .: 0P] [T7-52,DR5-6b,U-SUB]
T7-753. Valid [0P, .: T(0P)] [T7-53,DR5-6b,U-SUB]
T7-754. Valid [∼F0P, .: 0P] [T7-54,DR5-6b,U-SUB]
T7-755. Valid [(TP v 0P), .: ∼FP] [T7-55,DR5-6a,U-SUB]
T7-756. Valid [(FP v 0P), .: ∼TP] [T7-56,DR5-6a,U-SUB]
T7-757. Valid [(0P & 0Q), .: (0(P&Q) & 0(PvQ))] [T7-57,DR5-6b,U-SUB]
T7-758. Valid [(0P & 0Q), .: 0(P&Q)] [T7-58,DR5-6a,U-SUB]
T7-759. Valid [(0P & 0Q), .: 0(PvQ)] [T7-59,DR5-6a,U-SUB]
T7-760. Valid [(0P & 0Q), .: 0(P ⊃ Q)] [T7-60,DR5-6a,U-SUB]
T7-761. Valid [0TP, .: (∼TP & TP)] [T7-61,DR5-6b,U-SUB]
T7-762. Valid [0FP, .: (∼FP & FP)] [T7-62,DR5-6b,U-SUB]
T7-763. Valid [00P, .: (F0P & ∼F0P)] [T7-63,DR5-6b,U-SUB]
T7-764. Valid [((T(P&Q) v F(P&Q) v 0(P&Q)) & FP), .: F(P&Q)] [T7-64,DR5-6a,U-SUB]
T7-765. Valid [(T(P&Q) v F(P&Q) v 0(P&Q)) & ∼TP), .: ∼T(P&Q)] [T7-65,DR5-6a,U-SUB]
T7-766. Valid [((T(P&Q) v F(P&Q) v 0(P&Q)) & (TP & 0P)), .: 0(P&Q)] [T7-66,DR5-6a,U-SUB]
T7-767 Valid [(0PvTPvFP), .: ((∼TP&∼FP) v (TP&∼FP) v (FP&∼TP)] [T7-67,DR5-6a,U-SUB]
T7-768 Valid [(0PvTPvFP), .: ((∼TP vTP) & (∼TPv∼FP) & (∼FpvFP))] [T7-68,DR5-6a,U-SUB]
T7-769 Valid [T(0PvTPvFP), .: (T(∼TPvTP) & T(∼TPv∼FP) & T(∼FPvFP))] [T7-69,DR5-6a,U-SUB]

A third way to get Valid inference schemata for truth-logic prefixes T-operators to antecedent and consequent in the Cont- and Syn theorems, and yeilds validity theorems of Truth-logic whenever the consistency requirement is met. It is based on R7-1 and its derived rules (which preserve theoremhood while prefixing T-operators to antecedent and consequent in SYN- and CONT-theorems of Chapter 1 to 4) together with the derived rules based on Df 'Valid' which we have just been using:

DR5-6a. [If (P CONT Q) and not-Inc (P&Q), then Valid (P, .:Q)]
DR5-6b. [If (P SYN Q) and not-Inc (P&Q), then Valid (P, .:Q)]

Each of the 110 valid inferences schemata listed in Section 5.52, were based on SYN- and CONT-theorems in Chapters 1 to 4. Each of them yields a Valid inference schema of truth-logic by prefixing a 'T' to the antecedent and consequent to these SYN- and CONT-theorems (rather than prefixing T-operators to elementary components). Thus,

from T4-31[(~ P v Q) SYN (P ⊃ Q)]
we derive |= Valid [T(~ P v Q), .: T(P ⊃ Q)]
rather than |= Valid [(~ TP v TQ), .: (TP ⊃ TQ)] (T7-431)

The following rules facilitate derivations of Valid inference schemata from Syn- and Cont- theorems by this third method.

DR7-6a [If (P Cont Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]
DR7-6b. [If (P Syn Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]
DR7-6c. [If (P Syn Q) and not-Inc (P&Q) then Valid (TQ, .:TP)]
DR7-6d. [If (P Syn Q) and not-Inc (P&Q) then Valid ((TP, .:TQ) & (TQ, .:TP)]

Many, but not all, of the Validity theorems of truth-logic which are derived from Validity- theorems of Chapter 5 by U-SUBab with T-operators can also be derived from the SYN and CONT theorems of Chapters 1 through 4, using DR7-6a to DR7-6d. The theorems that can not be gotten in this way are those in which '$\sim P_i$' occurs as an elementary wff in the initial normal form. This is because Rule 7-1 and DR7-6a to DR7-6d (which are based on it) introduce T- operaters on the outside (to the left of) any complex wffs. Distribution inward by SynSUB through Axioms 7-03 or 7-04 moves the 'T' inward to the components of a conjunction or disjunction. But there is no Syn-theorem.that allows 'T ~ P' to be interchangeable with ' ~ TP'. (If there were, the distinction between "false" and "not-true" would evaporate and analytic A-logic would collapse into M-logic.) To be sure, we have |= [T ~ P Cont ~ TP] from T7-15 and [FP Cont ~ TP], hence T8-715.Valid[T ~ P => ~ TP], but Containment is not always grounds for replacement.

The derived rules DR7-6a to DR7-6d are proved as follows:

DR7-6a [If (P CONT Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]

<u>Proof</u>:	1) (P CONT Q) and not-Inc (P&Q)	[Premiss]
	2) P CONT Q	[1),SIMP]
	3) If (P CONT Q) then (TP Cont TQ)	[R7-1d]
	4) (TP Cont TQ)	[2),3),MP]
	5) Not-Inc(P & Q)	[1),SIMP]
	6) Not-IncT(P & Q)	[5),DR7-5e]
	7) Not-Inc(TP & TQ)	[6),Ax.7-03,SynSUB]
	8) (TP Cont TQ) & not-Inc(TP & TQ)	[4),7),ADJ]
	9) |= [If (TP Cont TQ) and not-Inc (TP&TQ) then Valid (TP, .:TQ)]	[DR5-6a,U-SUB]
	10) |= Valid(TP, .:TQ)]	[8),10),MP]
	11) |= [If (P CONT Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]	[1) to 11) Cond.Pr.]

DR7-6b [If (P Syn Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]

<u>Proof</u>:	1) (P Syn Q) and not-Inc (P&Q)	[Premiss]
	2) (P Syn Q)	[1),SIMP]
	3) (P Cont Q)	[2),DR1-11]
	4) not-Inc (P&Q)	[1),SIMP]
	5) (P Cont Q) & not-Inc(P&Q)	[3),4),ADJ]
	6) |= Valid (TP, .:TQ)]	[2),DR7-6a,MP]
	7) If (P Syn Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]	[1),6), Cond.Pr.]

DR7-6c [If (P Syn Q) and not-Inc (P&Q) then Valid (TQ, .:TP)]

Proof:	1) (P Syn Q) and not-Inc (P&Q)	[Premiss]
	2) (P Syn Q)	[1),SIMP]
	3) (Q Cont P)	[2),DR1-12]
	4) not-Inc (P & Q)	[1),SIMP]
	5) not-Inc (Q & P)	[4),&COMM,SynSUB]
	6) (Q Cont P) & not-Inc(Q & P)	[3),4),ADJ]
	7) Valid (TQ, .:TP)]	[2),DR7-6a,MP]
	8) If (P Syn Q) and not-Inc (P&Q) then Valid (TQ, .:TP)]	[1) to 6), Cond.Pr.]

DR7-6d [If (P Syn Q) and not-Inc (P&Q) then Valid ((TP, .:TQ) &(TQ, .:TP)]

Proof:	1) (P Syn Q) and not-Inc (P&Q)	[Premiss]
	2) Valid (TP, .:TQ)]	[1),DR7-6b, MP]
	3) Valid (TQ, .:TP)]	[1),DR7-6c, MP]
	4) Valid ((TP, .:TQ) & Valid (TQ, .:TP))	[2),3),ADJ]
	5) Valid ((TP, .:TQ) & (TQ, .:TP))	[4), Df 'Valid&']
	6) If (P Syn Q) and not-Inc(P&Q) then Valid ((TP, .:TQ)&(TQ, .:TP))	[1) to 6),Cond.Pr.]

Another derived rule, to establish the validity of inferences based on A-implication, is

DR7-6g. [If (TP Impl TQ) and not-Inc (P&Q) then Valid_I (TP, .:TQ)]

The proof of this rule will be given at the beginning of Section 7.4242, which follows.

Every Cont-theorem in Chapters 1 to 4 yields a Validity-theorem of analytic truth-logic, by DR7-6a. Every Syn-theorems in Chapters 1 to 4 yields a Validity-theorem of analytic truth- logic, by DR7-6b, or DR7-6c, or DR7-6d. The first three and the last three of the 110 Validity- theorems gotten by this method are:

⊨ Valid [(TP, .:T(P & P)) & T(P &P), .:TP)]	[&-IDEM]	[Ax.1-01,DR7-6d]
⊨ Valid [(TP, .:T(P v P)) & (T(P v P), .:TP]	[v-IDEM]	[Ax.1-02,DR7-6d]
= Valid [(T(P&Q), .:T(Q & P)) & (T(Q & P), .:T(P & Q))]		[Ax.1-03,DR7-6d]
. . . . .		
⊨ Valid [T(∀x)(Px ⊃ Qx), .:T((∃x)Px ⊃ (∃x)Qx)]	ML*149	[T4-39,DR7-6a]
⊨ Valid [T((∃x)Px ⊃ (∃x)Qx), .:T(∃x)(Px ⊃ Qx)]	ML*150	[T4-40,DR7-6a]
⊨ Valid [T((∀x)Px ⊃ (∀x)Qx), .: (∃x)(Px ⊃ Qx)]	ML*151	[T4-41,DR7-6a]

From results of these derivations many other theorems are derivable by moving the T-operators inward towards the Normal Form for T-wffs. However, we do not enumerate all these results here, since they wil be interchangeable with results in Chapter 8 where '.:' is replaced by '=>' in validity theorems, by the VC\VI rule, R6-6.

We started in Chapter 1 through 3 with negation-free wffs, and introduced negation into theorems of Chapter 4 only through Axiom 5, definitions, and the restricted rule U-SUBab, all of which preserve the feature that no atomic wffs occur both POS and NEG. Consequently, the theorems of Chapter 1 through 4 have no inconsistent components at all in their wffs. In Chapter 5, we had to use unrestricted U-SUB in order to produce the inconsistent and tautologous components which make up the "theorems" of M-logic. The Syn-and Cont- theorems of Chapter 6 are not considered in the present chapter because they all contain occurrences of ' => ' or ' <=> ' which are deliberately excluded from Chapter 7.

7.4242 Valid Inference Schemata Based on A-implication

A-implication-theorems, which are elliptical for containment theorems with the Trivalence Presupposition conjoined, can also yield *de dicto* **valid** inference schemata. Since the Containment requirement is implicitly met the only additional requirement is that the not-inc clause be satisfied. But A-implications only occur in truth-logic, since by definition they are about T-wffs and require truth-operators.Their proofs always require theorems which are not derivable in earlier chapters.

The derived rule DR7-6g says that if P Implies Q and (P&Q) is not inconsistent, then the inference from P to Q is Valid. This does not follow from the definition of either 'Impl' or 'Valid', for not all A-implications meet the not-inc requirement for validity. Thus the following Lemma, establishing a conventional use of 'Valid$_I$**'**, is required. The word 'Valid' is subscripted with an 'I' to remind us that these theorems rely upon an unexpressed linguistic presupposition and are *de dicto* only.

Lemma I: **Valid$_I$**(P, **.:** Q) Syn (i) **Valid**((0Q v TQ v FQ) & P), **.:** Q) &
(ii) & not: (P Cont Q)
(iii) & not: (0QvTQvFQ) Cont Q) &
(iv) & not-Inc(P & Q)

The proof of DR7-6g follows:

DR7-6g [If (P Impl Q) and not-Inc (P&Q) then Valid$_I$ (P, **.:**Q)]

Proof: 1) P Impl Q) and not-Inc (P&Q) [Premiss]
2) P Impl Q[1),SIMP]
3) ((0Q v TQ v FQ) & P) Cont Q)
& not-(P Cont Q)
& not-((0QvTQvFQ) Cont Q) [2),Df 'Impl']
4) ((0Q v TQ v FQ) & P) Cont Q) [3),SIMP]
5) not-(P Cont Q) [3),SIMP]
6) not-((0QvTQvFQ) Cont Q) [3),SIMP]
7) not-inc(P & Q) [1),SIMP]
8) $\models$ T(0Q v TQ v FQ) [T7-71]
9) T(0Q v TQ v FQ) & not-Inc(P&Q)) [6),5),ADJ]
10) not-inc(((0Q v TQ v FQ) & P) & Q) [5),8)][44]
11) Valid(((0Q v TQ v FQ) & P), **.:** Q) [4),8),DR5-6a]
12) Valid$_I$ (P, **.:**Q) [11),5),6),7), Lemma I]
13) If (P Impl Q) and not-Inc (P&Q) then Valid$_I$ (P, **.:**Q)] [1) to 12),Cond.Pr.]

Thus by DR7-6g we can derive Validity-theorems for each Impl-theorem which meets the consistency requirement for validity. The Impl-theorems referred to in the antecedent of DR7-6g are only found in Chapter 7 so far; so their names all begin with 'Ti7- … '. As in cases above, the satisfaction of the

44. To explain Step 10: since, by 9) (0Q v TQ v FQ) is logically true (true in every possible case) and by 7) P and Q are jointly not-Inc, hence satisfiable (possibly true together in at least one case), the conjunction in 10), '(((0Q v TQ v FQ) & P) & Q)' must be satisfiable in at least one case, thus is not-Inc.

"not-Inc"clause can be checked by assignments of 'T', 'F' or '0' to each atomic wff so as to make the conjunction of premiss and conclusion take the value T. This is possible in each case below, but is left to the reader. In the proofs which follow, a line like

Ti7-780.Valid$_I$[∼TP,.:∼T(P&Q)] [Ti7-80,DR7-6g]

stands for a proof like,

Ti7-780. Valid$_I$ [∼TP, .:∼T(P & Q)]

Proof: 1) [∼TP Impl ∼T(P & Q)] **[Ti7-80]**

2) not-Inc(∼TP & ∼T(P & Q)
T FF T T F F F T [Inspection]

3) [If (P Impl Q) and not-Inc (P&Q) then Valid$_I$ (P, .:Q)] **[DR7-6g]**

4) [If (∼TP Impl ∼T(P&Q)) and not-Inc(∼TP& ∼T(P&Q) then Valid$_I$ (∼TP, .: ∼T(P&Q)]
[3).U-SUB('∼TP'for 'P', '∼T(P&Q)' for 'Q']

5) ((∼TP Impl ∼T(P & Q)) & not-Inc (∼TP& ∼T(P&Q))) [1),2),ADJ]

6) Valid$_I$ [∼TP, .:∼T(P & Q) [5),4),MP]

From 7.4231 Basic Implication Theorems

Ti7-780. Valid$_I$ [∼TP, .:∼T(P & Q)]	[Ti7-80,DR7-6g]
Ti7-781. Valid$_I$ [FP, .: F(P & Q)]	[Ti7-81,DR7-6g]
Ti7-782. Valid$_I$ [∼FP, .: ∼F(P v Q)]	[Ti7-82,DR7-6g]
Ti7-783. Valid$_I$ [TQ, .: T(P v Q)]	[Ti7-83,DR7-6g]
Ti7-784. Valid$_I$ [TP, .: (TP v FP)]	[Ti7-84,DR7-6g]
Ti7-785. Valid$_I$ [FP, .: (TP v FP)]	[Ti7-85,DR7-6g]

From 7.4233∼ Principles of the truth-tables (some are not A-implications).

For the truth∼table of (P& Q)

T7-758a.	Valid [(0P & 0Q), .: 0(P&Q)	(For &∼Row 1)	[T7-58,DR5-6a,U-SUB]
Ti7-7&R2.	Valid$_I$ [(TP & 0Q), .: 0(P & Q)]	(For &∼Row 2)	[Ti7-&R2,DR7-6g]
Ti7-7&R3.	Valid$_I$ [(FP & 0Q), .: F(P & Q)]	(For &∼Row 3)	[Ti7-&R3,DR7-6g]
Ti7-7&R4.	Valid$_I$ [(0P & TQ), .: 0(P & Q)]	(For &∼Row 4)	[Ti7-&R4,DR7-6g]
T7-703c.	Valid [TP & TQ), .: (T(P &Q)]	(For &∼Row 5)	[Ax. 7-3,DR5-6c,U-SUB]
Ti7-7&R6.	Valid$_I$ [(FP & TQ), .: F(P & Q)]	(For &∼Row 6)	[Ti7-&R6,DR7-6g]
Ti7-7&R7.	Valid$_I$ [(0P & FQ), .: F(P & Q)]	(For &∼Row 7)	[Ti7-&R7,DR7-6g]
Ti7-7&R8.	Valid$_I$ [(TP & FQ), .: F(P & Q)]	(For &∼Row 8)	[Ti7-&R8,DR7-6g]
T7-737a.	Valid [(FP & FQ), .: F(P & Q)]	(For &∼Row 9)	[T7-37,DR5-6a,U-SUB]

For the truth-table of (P v Q)

T7-759a.	Valid [(0P & 0Q), .: 0(P v Q)]	(For v∼Row 1)	[T7-59,DR5-6a,U-SUB]
Ti7-7vR2.	Valid$_I$ [(TP & 0Q), .: T(P v Q)]	(For v∼Row 2)	[Ti7-vR2,DR7-6g]
Ti7-7vR3.	Valid$_I$ [(FP & 0Q), .: 0(P v Q)]	(For v∼Row 3)	[Ti7-vR3,DR7-6g]
Ti7-7vR4.	Valid$_I$ [(0P & TQ), .: T(P v Q)]	(For v∼Row 4)	[Ti7-vR4,DR7-6g]
T7-738a.	Valid [(TP & TQ), .: T(P v Q)]	(For v∼Row 5)	[T7-38,DR5-6a,U-SUB]
Ti7-7vR6.	Valid$_I$ [(FP & TQ), .: T(P v Q)]	(For v∼Row 6)	[Ti7-vR6,DR7-6g]

Ti7-7vR7.	Valid$_I$ [(0P & FQ), **.:** 0(P v Q)]	(For v ~ Row 7)	[Ti7-vR7,DR7-6g]
Ti7-7vR8.	Valid$_I$ [(TP & FQ), **.:** T(P v Q)]	(For v ~ Row 8)	[Ti7-vR8,DR7-6g]
T7-723c.	Valid [(FP & FQ), **.:** F(P v Q)]	(For v ~ Row 9)	[T7-23,DR5-6c,U-SUB]

For the truth-table of (P ⊃ Q)

T7-760a.	Valid [(0P & 0Q), **.:** 0(P ⊃ Q)]	(For ⊃-Row 1)	[T7-60,DR5-6a,U-SUB]
Ti7-7⊃-R2.	Valid$_I$ [(TP & 0Q), **.:** 0(P ⊃ Q)]	(For ⊃-Row 2)	[Ti7⊃-R2,DR7-6g]
Ti7-7⊃-R3.	Valid$_I$ [(FP & 0Q), **.:** T(P ⊃ Q)]	(For ⊃-Row 3)	[Ti7⊃-R3,DR7-6g]
Ti7-7⊃-R4.	Valid$_I$ [(0P & TQ), **.:** T(P ⊃ Q)]	(For ⊃-Row 4)	[Ti7⊃-R4,DR7-6g]
Ti7-7⊃-R5.	Valid$_I$ [(TP & TQ), **.:** T(P ⊃ Q)]	(For ⊃-Row 5)	[Ti7⊃-R5,DR7-6g]
T7-739a.	Valid [(FP & TQ), **.:** T(P ⊃ Q)]	(For ⊃-Row 6)	[T7-39,DR5-6a,U-SUB]
Ti7-7⊃-R7.	Valid$_I$ [(0P & FQ), **.:** 0(P ⊃ Q)]	(For ⊃-Row 7)	[Ti7⊃-R7,DR7-6g]
T7-740a.	Valid [(TP & FQ), **.:** F(P ⊃ Q)]	(For ⊃-Row 8)	[T7-40,DR5-6b,U-SUB]
Ti7-7⊃-R9.	Valid$_I$ [(FP & FQ), **.:** T(P ⊃ Q)]	(For ⊃-Row 9)	[Ti7⊃-R8,DR7-6g]

From 7.4234—A-implication in Quantification Theory

Ti7-786.	Valid$_I$ [(∀x)TQx, **.:** (∀x)(0Px ⊃ TQx)]	[Ti7-86,DR7-6g]
Ti7-787.	Valid$_I$ [(∀x)TQx, **.:** (∀x)(TPx ⊃ TQx)]	[Ti7-87,DR7-6g]
Ti7-788.	Valid$_I$ [(∀x)TQx, **.:** (∀x)(FPx ⊃ TQx)]	[Ti7-88,DR7-6g]
Ti7-789.	Valid$_I$ [(∀x) ~ TPx, **.:** (∀x)(TPx ⊃ FPx)]	[Ti7-89,DR7-6g]
Ti7-790.	Valid$_I$ [(∀x)FPx, **.:** (∀x)(TPx ⊃ FQx)]	[Ti7-90,DR7-6g]
Ti7-791.	Valid$_I$ [(∀x)TPx, **.:** (∀x)(~ TPx ⊃ TQx)]	[Ti7-91,DR7-6g]
Ti7-792.	Valid$_I$ [T(Pai), **.:**T(∃x)Px]	[Ti7-92,DR7-6g]
Ti7-793.	Valid$_I$ [F(Pai), **.:** F(∀x)Px]	[Ti7-93,DR7-6g]
Ti7-794.	Valid$_I$ [~ T(Pai), **.:** ~ T(∀x)Px]	[Ti7-94,DR7-6g]
Ti7-795.	Valid$_I$ [~ F(Pai), **.:** ~ F(∃x)Px]	[Ti7-95,DR7-6g]

With the legitimation of Detachment Theorems, and *de dicto* valid inference schemata associated with Addition and other Impl-theorems, validity-theorems of analytic truth-logic are very close to the set of plausible valid inference schemata which gave M-logic its plausibility.

7.5 Consistency and Completeness of Analytic Truth-logic re M-logic

Is Analytic Logic complete with respect to the theorems of M-logic? The answer is yes if M-logic is considered to be the set of wffs which it calls "theorems" of propositional logic and first-order predicate logic expressed in zero-level T-wffs—i.e., without T-operators. It s correlate in A-logic is a set of tautologous zero-level T-wffs. For, as we showed in Chapter 5, all and only those wffs which are "M-valid" and "theorems" of M-logic, are wffs which occur as the subject terms in TAUT-theorems of A-logic.

Is Analytic *Truth*-logic complete with respect to M-logic? Here the answer is more slippery, since M-logic does not have a Truth-operator, and has an odd, ambivalent concept of truth. In its semantic theory it claims that each zero-level wff is equivalent to an expression which predicates truth of it, but does not allow a truth-operator in its object-languge, although it uses a truth-predicate to predicate truth of its zero-level sentences in its semantics. Analytic Truth-logic consists of theorems about first-level T-wffs.

Suppose we interpret all of the wffs of M-logic as having a T prefixed to every atomic wff, in accordance with Tarski's convention T. Among the resulting T-wffs of M-logic, *is it true* that all and only those which are theorems of M-logic are the subjects of TAUT-theorems of A-logic? The answer is yes. Taking Tarski at his word, let us prefix a T to every Predicate Letter and Sentence letter. Thus Thomason axioms, AS1 to AS6, become

AS1. TAUT[TP ⊃ (TQ ⊃ TP)]
AS2. TAUT[(TP ⊃ (TQ ⊃ TR)) ⊃ ((TP ⊃ TQ) ⊃ (TP ⊃ TR)]
AS3. TAUT[(~ TP ⊃ ~ TQ) ⊃ (TQ ⊃ TP)]
AS4. TAUT[(∀x)(TP ⊃ TQx) ⊃ (TP ⊃ (∀x)TQx)]
AS5. TAUT[(∀x)(TPx ⊃ TQx) ⊃ ((∀x)TPx ⊃ (∀x)TQx))
AS6. TAUT[(∀x)TPx ⊃ TPa]

The Rule of Inference, "From ⊢ TP and ⊢ (TP ⊃ TQ), Infer ⊢ TQ" becomes "From Taut[TP] and Taut[TP ⊃ TQ], infer Taut[TQ]", and the rule of U-SUB requires that each sentence letter or predicate letter be prefixed with a T in the expression to be substituted.

The use of SynSUB preserves Tautology and sameness of truth-tables, since the truth-tables of synonyms are the same. The proofs in Chapter 5 of consistency and completeness of TAUT-theorems *re* theorems of M-logic go through precisely as before, except for using 'TP_i' for 'P_i' throughout. In the Axiomatic System of this chapter everything else remains the same.

Let us call the wffs which are subject to the restriction above (every atomic wff has a T prefixed to it) Tarski-Type wffs. We have just isolated the fragment which is restricted to the Tarski-Type wffs. The result is a sub-system of analytic truth-logic which is consistent and complete with respect to M-logic. Soundness and completeness of this sub-system system are established in the usual ways. The set of wffs in TAUT-theorems is the set of theorems of M-logic.

But in this sub-system one will not be able to derive TAUT[TP v FP] as it is understood in A-logic. For there will be no elementary wffs of the form T ~ P, or FP. Since 'F(TP)'Syn_{df}' ~ (TP)' in A-logic, ' ~ TP' and 'FTP' are interchangeable in this sub-system as they are supposed to be in M-logic. F(TP) has the same truth-table as ~ T(TP), and T(TP) has the same truth-table as ~ F(TP), so that (~ T(TP) & ~ F(TP)), i.e., 0(TP), has all 'F's in its trivalent truth-table, while its contradictory, (T(TP)vF(TP)), has all T's. Further if P SYN TP, then the Law of Bivalence holds.

TP Syn ~FTP		FTP Syn ~TP		**Excluded**: FP Syn ~TP		~0(TP)	0(TP)
(TP)	~F(TP)	F(TP)	~(TP)	**T~P**	~TP	[T(TP) v F(TP)]	[~T(TP) & ~F(TP)]
F0	FT F0	T F0	T F0	**F** 0 0	**T** F0	F F0 T T F0	T F F0 F F T F0
TT	TF TT	F TT	F TT	F F T	F TT	T TT T F TT	F T TT F T F TT
FF	FT FF	T FF	T FF	T T F	T FF	F FF T T FF	T F FF F F T FF

The novelties of A-logic with T-wffs, e.g., that '(TP v FP)' is not Taut, are absent from this fragment since 'T ~ P' with an atomic P is excluded. There are only four distinct truth-functions for one variable, as in M-logic. The third value can not be expressed, makes no difference, and may be dropped.

The answer is also Yes if every wff of M-logic is interpreted as being false (i.e, T is prefixed to the negation of each atomic wff). In each of these cases, taken separately, there is a Bivalent Law |= [~ TP v TP] or |= [~ FP v FP] which behaves like M-logic's two-valued "Law of Excluded Middle". But neither of these make allowance for a difference between ' ~ TP' and 'FP' i.e., 'T ~ P'.

When the difference between '~TP' and 'FP' is allowed, and '(~TP & ~FP)' is capable of being true, then the semantic presupposition of M-logic, '(TP v FP)', fails to be logically true, and A-logic is not complete with respect to the semantic theory of M-logic because it lacks this basic principle of M-logic. This lack of completeness is, of course, premeditated. It is necessary to exclude the narrower principle of bivalence in order to allow the third "value", 0. The third value '0' is what is needed to account for the contrary-to-fact conditionals that science uses, and a host of other things Logic should cover .

Thus 1) the unquantified portion of A-logic is complete in the sense that every zero-level wff of M-logic that is M-valid occur in TAUT-theorems of A-logic, and no wff of M-logic which is not M-valid occurs in a Taut-theorem of A-logic.

But 2) A-logic is not confined to, and thus is not complete with respect to, the narrower presuppositions that lie behind M-logic. Analytic Truth-logic omits many metatheorems from M-logic's Semantics; most notably [TP v FP] which is neither logically true nor tautologous in A-logic.

Further, the overall system of Analytic Logic does not grant A-validity to any "theorems" of M-logic, since it defines validity in an utterly different way which it holds is closer to traditional and ordinary usage.

If proponents of M-logic choose to treat '~TP' as equivalent to 'FP' i.e., to 'T~P', then they will hold that Analytic truth-logic is not consistent, for they will hold that '(~TP v TP)' and '(FP v TP)' say the same thing, and thus that A-logic holds that the same thing is both Tautologous and not Tautologous. But such a finding of inconsistency would be due to a failure to recognize the difference in meaning that A-logic gives to the terms '~T' and 'F'.

Completeness, in its most significant sense, is always relative to some model or field of reference which lies outside of the formal axiomatic system. The most important field of reference is the class of case histories in ordinary and scientific language wherein 'and', 'not', 'if...then', 'all' and 'some' have been used in many brilliant processes of deduction over at least two millenia. Logic approaches completeness—or falls short of it—to the extent it captures or fails to capture and formalize the brilliant logical deductions of the past. Determining the degree to which a logic is complete in this sense is a matter of judgement that is not, of course, reducible to any simple formula. But it is not difficult to see that M-logic is incomplete in several areas—particularly in the empirical sciences. Its proponents would like to have established a completeness, relative to the use of logic in the natural scieinces, but they admittedly failed.

In order to be complete relative to an outside sytem some other kinds of completeness are required that **do** depend on logic alone. One of these, when we are deal with truth-tables, is functional completeness. Each truth-table for a particular wff represents in its final column one truth-function. If truth-logic is to deal with every kind of truth-claim, it must be able to present a distinct wff for every possible truth-table—i.e, to express every possible truth-function. With trivalent truth tables, there will be 3 to the 3^n power truth-functions for any n variables. This means that for 1 variable, with 3 truth-values, there are $3^1 = 3$ rows in the truth-table, and 3 to the 3^1 power, or $3^3 = 27$ truth-functions. With two variables there are $3^2 = 9$ rows in the truth-table and 3 to the 3^2, i.e., $3^9 = 19{,}683$ truth-functions. In this chapter, the truth-tables constructible from truth tables of '&' and '~' and 'T' only cover 12 truth-functions.[45] With the truth table for '=>' added in the next chapter, each of the 27 truth-functions

45. Namely, FP ~FP 0P ~0P

P	~P	(P&~P)	(~PvP)	T(P)	~T(P)	T~P	~T~P	(~TP&~FP)	(TPvFP)	(TP&~TP)	(~TPvTP)
0	0	0	0	F	T	F	T	T	F	F	T
T	F	F	T	T	F	F	T	F	T	F	T
F	T	F	T	F	T	T	F	F	T	F	T

for one variable and each of the 19,683 truth-functions for two variables can be expressed by a wff using only the primitive operators, '~', '=>' and 'T'.[46] From these results it can be deduced that truth-functions for any number of variables are all expressible. But it is clear that Analytic Truth-logic, which presupposes three truth-values, can not be functionally complete with only truth-tables for '&', 'v', and '~'. The more general question of the completeness of Analytic truth-logic will not be pursued further in this book, but may be deferred to other investigations.

Finally, the logic of this chapter is also deliberately incomplete with respect to analytic truth-logic because the C-conditional, '=>', has been excluded in order to first clarify the relation of the truth-operator to M-logic. Whether the addition of the C-conditional with appropriate axioms, renders the axiomatic system of analytic truth-logic complete with respect to its own semantic theory, is another area of investigation.

Truth-tables have been used to establish the consistency of the propositional calculus in M-logic. But there is no need to designate these tables as "truth-tables" or their values as "truth-values" to prove consistency. If 'T' is replaced by '1' and 'F' by '2', so that the values are '1' and '2', and '1' is labeled the "designated value", both consistency tests can be carried through as before. The POS-NEG tables introduced in Chapter 4, provided two-valued tables which can be used in A-logic or M-logic to test for Tautology (M-logic's "validity"), Inconsistency, and *prima facie* Contingency of wffs.

An axiomatic *system* is consistent if there are no two theorems A and theorem B, such that [A&B] is inconsistent. Since we deal in this chapter only with T-wffs of M-logic supplemented by T-operators, the consistency of A-logic can be tested here by the usual methods, using any binary system of values. Each fragment of A-logic is **inconsistent** if there is any wff such that both it and its negation are subject terms of an Inc-theorem, or of a Taut-theorem, or of an Unfalsifiability-theorem, or of an Unsatisfiabilitity theorem, or of a Logical truth-theorem, or of a Logical-Falsehood theorem, or of a Validity-theorem. Further, the set of Syn- and Cont-theorems will be inconsistent if any pair of wffs or T-wffs are proven to be both Syn and not-Syn. Since all of the earlier properties, including validity, depend on Syn-theorems, the consistency of Syn-theorems is basic to all of them. To prove that there are no inconsistent cases of any of these sorts, is to prove A-logic consistent. However, the question of the consistency of Analytic truth-logic generally is best considered in the light of the next chapter which covers more fully the range of possible T-wffs. Though we will deal with methods of preserving consistency throughout, we will not present a formal proof.

---

46. Given the truth-function of 'P' as 0TF, of '~' as 0FT, of 'T' as FTF and of '8 ' as 0000T00F0, the 27 truth-functions of one variable are expressed by the following wffs ('F' abbreviating 'T~'):

000 = ((P => ~P) => P)	T00 = ((~TP => ~FP) => ~TP)	F00 = ((~TP => ~FP) => TP)
00T = (~P => ~P)	T0T = (~TP => ~TP)	F0T = (~TP => FP)
00F = (~P => P)	T0F = (~TP => ~FP)	F0F = (~TP => TP)
0T0 = (P => P)	TT0 = (~FP => ~FP)	FT0 = (~FP => TP)
0TT = (~F(~TP => FP) => ~F(P => P))	TTT = ~F(P => P)	FTT = ~F(~TP => FP)
0TF = P	TTF = ~FP	FTF = TP
0F0 = (P => ~P)	TF0 = (~FP => TP)	FF0 = (~FP => FP)
0FT = ~P	TFT = ~TP	FFT = FP
0FF = (~F(~TP => FP) => F(P => P))	TFF = F(~TP => FP)	FFF = F(P => P)

The truth-function of '(P&Q)', i.e., 00FOTFFFF, is expressed by:

= ((~T(F(~TP => FP) => TQ) => ~((~T(F (~TQ => FQ) => ~FP) => ~T(P => Q))

the other truth-functions, for 'v', '⊃' etc., can be gotten from those of conjunction and denial.

Chapter 8

Analytic Truth-logic

8.1 Introduction

In this chapter the logic of C-conditionals with truth-operators is added so as to obtain a full account of analytic truth-logic. The logistic base for this consists of the base for A-logic in Chapter 6, plus the additions for truth-logic in Chapter 7, plus the following two Axioms:

Ax.8-01. T(P ⇒ Q) Syn T(P&Q)
Ax.8-02. F(P ⇒ Q) Syn T(P& ~ Q)

The resulting system provides solutions to a wide variety of problems which have confronted M-logic because of its reliance on the truth-functional interpretation of conditionals. In particular, it offers a solution to the problems of subjunctive and contrary-to-fact conditionals.

8.11 Inference-vehicles vs. the Truth of Conditionals

The "truth-logic of C-conditionals" covers two very different kinds of expressions: a) inferential conditionals with truth-operators prefixed to its component parts only, and b) statements in which a truth-operator is prefixed to a conditional as a whole.

a) The truth-logic of inferential conditionals has components which assert truth or non-truth of sentences in the antecedent and the consequent. In symbols these are conditionals of the forms:

(TP ⇒ TQ),	(TP ⇒ ~ TQ),	(TP ⇒ FQ),	(TP ⇒ ~ FQ),
(FP ⇒ TP),	(FP ⇒ ~ TQ),	(FP ⇒ FQ),	(FP ⇒ ~ FQ),
(~ TP ⇒ TQ),	(~ TP ⇒ ~ TQ),	(~ TP ⇒ FQ),	(~ TP ⇒ ~ FQ),
(~ FP ⇒ TP),	(~ FP ⇒ ~ TQ),	(~ FP ⇒ FQ),	(~ FP ⇒ ~ FQ),

These conditionals are "inference vehicles";[1] their utility lies in providing schemata for passing from the truth-status of statements of one logical form to the truth-status of statements of another logical form. The fundamental laws of logic—all of its principles of valid inference—as well as the general laws recognized by common sense and natural science concerning truth about the actual world, are best expressed with this type of conditional. Validity or credibility are the primary properties wanted in these laws.

b) The logic of truth assertions about C-conditionals is different. In addition to expressions like the T(P&Q), T(PvQ), ~T(P), T(~P), etc., of Chapter 7 we now have expressions of the forms,

$$T(P \Rightarrow Q),\quad F(P \Rightarrow Q),\quad \sim T(P \Rightarrow Q),\quad \sim F(P \Rightarrow Q),\ T(\forall x)(Px \Rightarrow Qx),\ \ldots$$

which assert that conditionals are true or not true, etc. An assertion that a conditional statement is true, false, not true or not-false, is very different from an assertion that it is valid or credible or unfalsifable. These are not conditional statements. They are truth-claims or denials of truth-claims about conditionals. Just as '~(P&Q)' is not a conjunction but a denial of a conjunction, when a truth-operator is prefixed to an inferential conditional the result is a truth-claim and the expression as a whole is no longer a conditional.

The meaning of T[P ⇒ Q] is both something less and something more than the meaning of [P ⇒ Q]. Prefixing 'It is true that…' to a conditional robs the conditional of some of its meaning. The order relationship which makes 'If P then Q' have a different meaning than 'if Q then P' is ignored. The aspects of a conditional's meaning that suggest some kind of connection between the referents of its components is by-passed. All that counts is correspondence of its two components with facts. But prefixing a truth-operater also adds something to the meaning; the idea of an objective field of reference with facts that correspond, or not, to the component parts of the conditional.

The distinctions between 1) describing facts in an objective field of reference, 2) accepting and using inferential conditionals as valid or credible, and 3) determining the truth or falsehood of a conditional statement in a particular application, are important for the organon of logical reasoning, This is especially so where *de re* reasoning involves subjunctive or "contrary-to-fact" conditionals.

In describing the facts of any objective field of reference, categorical descriptions are conjoined to ascribe properties or relations to entities in the field.

In the conditional statements we carry around with us in our heads, neither the antecedent nor the consequent are true most of the time. But these conditionals are highly useful for predicting, retrodicting and controlling events. They are also used simply to convey certain relations of ideas.[2]

Establishing the truth or falsity of particularized conditionals is important in deciding on the reliability of the inferential conditional for predicting and/or controlling events. Ax.8-01 and Ax.8-02 are statements about what it means to say a conditional is true or false. Conditional statements are not mere descriptions of facts. To say what it means to say they are true or false is not to explain fully what a conditional means. An explanation of their meanng was given in Chapter 6 independently of truth.

---

1. Or, "inference tickets". See the reference to Ryle's introduction of that term in the discussion of Lawlike Statements, Section 6.211.

2. Without truth-operators (implicit or explicit) the conditional does not purport to describe anything except the thoughts we are trying to convey. E.g., Consider Elizabeth Barrett Browning's conditional statement, "If thou must love me, let it be for naught except for love's sake only." This is not the same as "If it is true that you love me, then it is true that you love me for nothing but love's sake alone"(unfortunately).

These distinctions are tied to the sharp difference drawn in A-logic between a conditional's being valid and it's being true or not. In M-logic validity has been identified with universal truth; M-validity is ascribed to disjunctions and conjunctions. In A-logic logical validity is a property only of certain inferential C-conditionals and is independent of whether these conditionals are true in fact.

Sections 8.2 through 8.23 develop theorems which involve conditionals and T-operators. Section 8.21 provides the logistic base for Syn-, Cont- and Impl-theorems based on Ax.8-01 and Ax.8-02. Section 8.22 derives Syn- and Cont-theorems (Section 8.221) and Impl-Theorems(Section 8.222). Section 8.23 establishes Validity-theorems based on **entailments** which can reflect *de re* relations (Section 8.231), and C-conditionals which are only valid *de dicto* including valid **implications** (Section 8.232). Section 8.24 establishes some new Inc- and Taut- theorems which have no analogues in M-logic, Section 8.25 derives with related theorems of logical unsatisfiability and unfalsifiability. Section 8.26 deals with Logical Truth- and Logical Falsehood- Theorems. "Logical truths" (statements always true by logic alone) are distinquished from "truths of logic".

In Section 8.3 we deal with two interesting problems for Analytic Truth-logic: Transpositiion and various Squares of Opposition.

In Chapter 9 we will consider how analytic truth-logic applies to reasoning with contingent conditionals—substantive C-conditionals which attempt to describe relationships among factual truths. Here we are concerned with the role of logic in establishing "empirically valid" generalizations, lawlike scientific statements, causal principles, and probabilistic laws, based on ascribing of truth or falsity to particular conditionals for the confirmation or disconfirmation of general statements.

In Chapter 10 we review in detail how analytic truth-logic solves the major problems which have beset efforts to apply mathematical logic to empirical science and common sense.

8.12 Philosophical Comments on Innovations

Why should the statements in Ax.8-01 and Ax.8-02 about the meaning of asserting a conditional true or false, be accepted?

Philosophical justifications for logical axioms are based partly on appeals to ordinary language and partly on arguments for the utility of adopting certain non-ordinary conventions.

Because ordinary language is amorphous and often ambiguous it is not always the deciding factor. But past usage contains many examples of the jobs conditionals can do. It is an important source of information on the useful employment of conditionals. Notable among these is the development of conditional scientific laws on the basis of observations of particular facts.

The primary question is whether A-logic's axioms will work better than M-logic's alternatives in performing the important jobs we believe logic and conditionals can do. The fundamental argument in favor of Axiom 8-01 is that it, together with the the distinction between inferential conditionals and truth claims about conditionals, accounts in a thoroughly rigorous way for dispositional (conditional) predicates, the confirmation and testing of law-like conditional statements, the calculus of probabilities, and counterfactual conditionals. M-logic fails in those areas.

Nevertheless, there are good reasons for examining Axioms 8-01 and 8-02 in terms of their relationship to ordinary usage, as well as to the usage prescribed by M-logic.

There is little or no disagreement about Ax.8-02; a conditional is false if and only if the antecedent is true and the consequent false. This seems to accord with common usage everywhere and with the criterion adopted by M-logic for the TF-conditional. By Ax.8-02, "[P $\Rightarrow$ Q] is false" is synonymous with "P is true and Q is false", and this is what "[P $\supset$ Q] is false" has been taken to mean in M-logic also. These points are expressed in theorem T7-40 [(TP & FQ) Syn F(P $\supset$ Q)] and in T8-25, [F(P $\Rightarrow$ Q) Syn F(P $\supset$ Q)] and hence $\models$[F(P $\Rightarrow$ Q) Syn (TP & FQ)] .

The synonymy of Ax.8-02 is due to the truth-operator, and not to the meaning of "if then" alone. This is clear from counter-examples in which we replace the operator "it is true that…" with the operators 'O…' for "it ought to be that…" or 'L…' for "it is necessary that" in a schema:

Ax.8-02. T~(Pa ⇒ Qa) <u>is Syn to</u> T(Pa & ~Qa)
E.g., "*It is true that* it is not the case that
if a steals, then a's hand is cut off"
<u>is Syn to</u>
"*It is true that* a steals and a's hand is not cut off."

But, O~(Pa ⇒ Qa) <u>is not Syn to</u> O(Pa & ~Qa);
E.g., "*It ought to be that* it is not the case that
if a steals, then a's hand is cut off"
<u>is not Syn to</u>
"*It ought to be that* a steals and a's hand is not cut off."

L~(Pa ⇒ Qa) <u>is not Syn to</u> L(Pa& ~Qa)
E.g., "*Necessarily,* it is not the case that
if a = 5 then a = 6"
<u>is not Syn to</u>
"*Necessarily*, (a=5 and not (a=6))"

Thus due to the substantive meanings of "true" and "false" and to the meaning of the C-conditional "if…then", F(P ⇒ Q) is Synonymous with T(P& ~Q) and its Synonym (TP & FQ). With these distinctions in A-logic, Ax.8-02 does not conflict with ordinary language or M-logic.

It is otherwise with Ax.8-01. Chief among differences between the truth-conditions of C-conditionals in A-logic and those of the TF-conditional in M-logic are the following:

In M-logic: If the antecedent of a TF-conditional is **false**
or the consequent is **true,**
then the TF-conditional as a whole is **true**;

In A-logic: If the antecedent of the C-conditional is **not true**
or the consequent is **neither true nor false**,
then the C-conditional as a whole is **neither true nor false**.

The characterization of the C-conditional in Axiom 8-01, can only be formalized if we distinguish 'not-true' from 'false' and 'not-false' from 'true', and admit 'neither true nor false' as a third "truth-value". If we have only two truth-values, T and F, every sentence is either one or the other (none being neither) and there is nothing left to do but assign either T's or F's to the cases in which the antecedent of a conditional is F.

Judgements of the truth and falsity of simple statements and of conjunctions of such statements are essential in recording facts about the world. But M-logic's requirement that either T or F be assigned to the whole conditional for each assignment of T or F to its components conflicts with the ordinary concept that when the antecedent of a conditional is false we should not say either that the conditional is true, or that it is false. Since M-logic excludes the possibility of a statement's being neither true nor false, this concept can not even be expressed in M-logic's language.

Proponents of M-logic recognize that TF-conditionals deviate from ordinary usage. Addressing the question, "Under what circumstances, then, should a conditional as a whole be regarded as true and under what circumstances false?" Quine writes that common attitudes suggest "a conditional with true antecedent and true consequent will count as true, and conditional with a true antecedent and a false consequent will count as false."[3] This is precisely what the Axioms 8-01 and 8-02 for C-conditionals say:

$$\models[T(P \Rightarrow Q) \text{ Syn } (TP \,\&\, TQ)] \qquad [Ax.8\text{-}01, Ax.7\text{-}03, SynSUB]$$
$$\text{and } \models[F(P \Rightarrow Q) \text{ Syn } (TP \,\&\, FQ)] \qquad [Ax.8\text{-}02, Ax.7\text{-}04, Df'F', SynSUB]$$

Quine goes on to say that according to everyday attitudes, "If on the other hand the antecedent turns out to be false, our conditional affirmation [of the form 'if p then q'] is as if it had never been made."[4] This seems to go a bit too far—if we made the affirmation, do we deny we made it? Rather we say that if the antecedent is not true, then the conditional is not shown to be true (which is not the same as saying it is false) and not shown to be false (which is not the same as saying it is true). In other words, it is not counted as either true or false; i.e., it is counted as neither true nor false in such cases.

Despite Quine's first comments, like most proponents of M-logic he defends the conditional of M-logic, though he allows that it is "arbitrary" and a departure from "everyday attitudes":

> Where the antecedent is false, on the one hand, the adoption of a truth value for the conditional becomes rather more arbitrary; but the decision which proves most convenient is to regard all conditionals with false antecedents as true.[5]

Quine's comments are in accord with our claim that C-conditionals are closer to ordinary usage than TF-conditionals in the following respects: in contrast to the TF-conditional, the truth of a C-conditional is never derivable from the falsity or inconsistency of its antecedent, or from the mere truth or tautologousness of its consequent.

However, the relationship of Ax.8-01 [T(P⇒Q) Syn T(P&Q)] to ordinary uses of conditionals is not simple. It conflicts with some others of our ordinary language intuitions.

From Ax. 8-01 it follows that the *truth* of [P & Q] entails the *truth* of [if P then Q] and of [if Q then P]. as well as the *truth* of [P if and only if Q]. Since we want to say that each of [if P then Q], [if Q then P], and [P if and only if Q] have distinct meanings, it may be asked: how can they all be established as true from the same single premiss, T[P&Q]? If this suggests a conflict between A-logic and ordinary intuitions, it is one which is shared with M-logic; A-logic and M-logic have the following analogous theorems:

$$\models \text{Valid}[T(P \,\&\, Q) \Rightarrow T(P \Rightarrow Q)]$$
$$\vdash [\,(P \,\&\, Q) \supset (P \supset Q)\,]$$

$$\models \text{Valid}[T(P \,\&\, Q) \Rightarrow T(Q \Rightarrow P)]$$
$$\vdash [\,(P \,\&\, Q) \supset (Q \supset P)\,]$$

$$\models \text{Valid}[T(P \,\&\, Q) \Rightarrow T(P \Leftrightarrow Q)]$$
$$\vdash [\,(P \,\&\, Q) \supset (P \equiv Q)\,]$$

(Where each of the conditionals in the M-logic theorems are said, by Tarski, to mean the same as as assertions of their truth—assertions which are made explicit in the A-logic theorems.)

---

3. W.V.Quine, *Methods of Logic,* 4th Ed, Harvard, 1982, Ch.2, p. 21.
4. Ibid.
5. Ibid.

In A-logic the question above is answered by explaining that though Ax.8-01 makes '**T**[If P then Q]' synonymous with '**T**[P and Q]', '[If P then Q]' without the 'T' prefixed, does not mean the same as '[P and Q]'. Despite Tarski, 'T(P & Q)' and '(P & Q)' are *not* synonymous, nor are 'T(P $\Rightarrow$ Q)' and '(P $\Rightarrow$ Q)'. Thus while T(P $\Rightarrow$ Q) and T(P & Q) are synonymous; (P $\Rightarrow$ Q) and (P & Q) differ radically in meaning. In A-logic 'If…then' by itself is not synonymous with any expression using only conjunction, disjunction and negation. Differences in the trivalent truth-tables of (P $\Rightarrow$ Q) and (P & Q) show this. There are no axioms or theorems which say that (P & Q) Contains or is Synonymous with (P $\Rightarrow$ Q), or that [(P & Q) $\Rightarrow$ (P $\Rightarrow$ Q)] (as distinct from [T(P & Q) $\Rightarrow$ T(P $\Rightarrow$ Q)]) is logically valid. Thus A-logic can explain the purported divergence from common usage within its logical scheme.

M-logic does not have these distinctions. The truth-predicate evaporates and the TF- conditional is equated with a form of disjunction. Consequently M-logic has no way to say that [if P then Q], [if Q than P] and [P if and only if Q] by themselves do not follow from [P & Q] or from the truth of [P & Q]. It is stuck with the three theorems above. They all conform to the rule that when the consequent is true then the TF-conditional as a whole is true. But the *explanation* that a conditional is true *because* the consequent is true is counter-intuitive and does not reflect any ordinary usage.

Perhaps it is felt that something more than the mere truth of [P and Q] should be required to establish the truth of [P $\Rightarrow$ Q],—perhaps something that shows a "connection" between P and Q or a "dependency" of Q on P. From this point of view, the truth-conditions in Ax.8-01 might be viewed as too loose—not strong enough. We argue in the next chapter that this is not necessary; being true together (co-existing in an objective field of reference) establishes a sufficient kind of "connection" to satisfy these objections.[6] But at present we point out only that Ax. 8-01 is much better off in this respect than the TF- conditional. The truth-conditions in M-logic for T(P $\supset$ Q) warrant the inference to T(P $\supset$ Q) not only from T(P & Q), but also from T($\sim$P&Q) and T($\sim$P&$\sim$Q). As Quine said, the latter two are not in accord with common attitudes. The truth-conditions for the C-conditional are stronger and more limited; the truth of both antecedent and consequent is the *one and only* sufficient condition for the truth of the C-conditional as a whole. In contrast, the truth of the TF-conditional follows from three of the four cases in the bivalent truth table (either (P&Q) or ($\sim$P&Q) or ($\sim$P&$\sim$Q)), and from five of the nine cases in its trivalent truth-table.

The requirement that the antecedent be true for a C-conditional to be true greatly reduces the number of C-conditional statements which can be true. Most instantiations of '$\Rightarrow$'-for-'$\supset$' analogues are neither true nor false, rather than being either true or else false as in M-logic. Thus the objection that the truth-conditions for the C-conditional are too loose can not be raised by defenders of the TF-conditional; for it applies even more to truth-conditions for the TF-conditional.[7] There is a further disadvantage in having three sufficient truth-conditions for the truth of a TF-conditional; it creates obstacles to anyanalysis

---

6. In fact it works well in both A-logic and M-logic to hold that both T(P$\Rightarrow$Q) and T(Q $\Rightarrow$P) follow from T(P&Q), particularly in the logic of confirmation. Otherwise there is no way to get from observations of true particulars, to the generalized conditionals which are treated as laws of nature (See Section 9. 3).

7. Given ten distinct propositions, 5 true and 5 false, there are 1,048,576 C-conditionals of the form '(P_1 & P_2) $\Rightarrow$ (P_3 & P_4)', (where each P_i is one of the ten). Of these 1,015,812 take the value 0, 1,024 take T, and 31744 take F. Putting '$\supset$'-for-'$\Rightarrow$', we get: none have the value 0, 1,016,832 take T, and 31744 take F. Thus there are 993 times (99,300%) more true TF-conditionals than true C-conditionals of this form; only 0.001% of true TF-conditionals would be true C-conditionals. 97% are neither true nor false.

(Suppose: P1, P2, P3, P4, P5, P6, P7, P8, P9, P10
T T T T T F F F F F

There are 2^{10} = 1,024 pairs which are antecedents of which 32 are true; 2^{10} = 1,024 pairs are consequents, of which 32 are true and 992 are false. Thus 32X32 = 1024 are true, and 32X992 = 31,744 are false.)

analysis of causal statements using those conditionals. (See "Paradoxes of Truth-functional Causation", Section. 9.225.)

Does the falsity of the antecedent, or truth of the consequent entail the truth of a conditional? A-logic says No. M-logic says Yes. Although A-logic rejects M-logic's semantic theorems regarding truth-conditions of conditionals because it does not accept the truth-functional interpretation of conditionality, M-logic's principles can be re-interpreted and expressed in A-logic as Cont-theorems, Impl-theorems, or Validity-theorems in the following ways:

Though these do not hold:	|<——— These do hold	———>|
Not Valid [T(~P v Q) ⇒ T(P ⇒ Q)]	**but:** [T(~P v Q) Syn T(P ⊃ Q)],	**and** Valid[T(~P v Q) ⇒ T(P ⊃ Q)]
Not Valid [T(P v ~Q) ⇒ T(Q ⇒ P)]	**but:** [T(P v ~Q) Syn T(Q ⊃ P)],	**and** Valid[T(P v ~Q) ⇒ T(Q ⊃ P)]
Not Valid[T(~P & Q) ⇒ T(P ⇒ Q)]	**but:** [T(~P& Q) Cont T(P ⊃ Q)],	**and** Valid[~T(P & Q) ⇒ T(P ⊃ Q)]
Not Valid[T(P & ~Q) ⇒ T(Q ⇒ P)]	**but:** [T(P & ~Q) Cont T(Q ⊃ P)],	**and** Valid[T(P & ~Q) ⇒ T(Q ⊃ P)]
Not Valid[T(~P&~Q) ⇒ T(P ⇔ Q)]	**but:** [T(~P&~Q) Impl T(P ≡ Q)],	**and** Valid$_I$ [T(~P&~Q) ⇒ T(P ≡ Q)]
Not [T(P & Q) Cont T(P ⊃ Q)]	**but:** [T(P & Q) Impl T(P ⊃ Q)],	**and** Valid$_I$ [T(P & Q) ⇒ T(P ⊃ Q)]
Not [T(P & Q) Cont T(Q ⊃ P)]	**but:** [T(P & Q) Impl T(Q ⊃ P)],	**and** Valid$_I$ [T(P & Q) ⇒ T(Q ⊃ P)]
Not [T(P & Q) Cont T(P ≡ Q)]	**but:** [T(P & Q) Impl T(P ≡ Q)],	**and** Valid$_I$ [T(P & Q) ⇒ T(P ≡ Q)]

A second principle derived from Ax.8-01, "If T[if P then Q] then TP and TQ", does not hold in M-logic. Does T(If P then Q) entail TP and TQ? With the TF-conditional M-logic says no and A-logic agrees that the truth of a *TF-conditional* T(P ⊃ Q) does not either Contain or Imply the conjunction T(P & Q). But with the C-conditional A-logic says, yes; to say that a conditional is true, is to say that both its antecedent and its consequent are true.

Two other objections might be raised, one against Ax.8-02 and one against Ax.8-01.

Ax.8-02 may appear to violate the A-logic's concept of logical containment since Q occurs unnegated in 'F(P ⇒ Q)' but negated in 'T(P&~Q)'. But the concept of "falsehood" over-rides this. Upon replacing 'F' in Ax.8-02 with its definiens 'T~' in 2), the consequent Q occurs in the scope of a negation sign, on both sides of 'Syn' thus it does not violate the rule.

1) [F(Pa ⇒ Qb) Syn T(Pa & ~Qb)]	[Ax.8-02]
2) [T~(Pa ⇒ Qb) Syn T(Pa & ~Qb)]	[1),Df 'F']
3) [T~(Pa ⇒ Qb) Syn T(Pa ⇒ ~Qb)]	[2),Ax.8-01]

On the other hand the antecedent Pa lies in the scope of a negation sign on the left but not on the right side of 'Syn'. But this does not matter in C-conditionals, for the conditional is neither true nor false if the antecedent is not fulfilled. To negate the conditional is to negate the consequent if and when the antecedent is fulfilled.

But perhaps the strongest objection that has been raised to Ax.8-01, is that it is incompatible with Transposition in the forms,

[T(P ⇒ Q) if and only if T(~Q ⇒~P)]
[(TP ⇒ TQ) if and only if (FQ ⇒ FP)]

This charge is true. If ⊨[T(P⇒Q) Syn T(P & Q)] and ⊨[T(~Q⇒~P) Syn T(~Q&~P)] are both accepted, then transposition can not hold. For if it did hold, then from Ax.8-01 and Ax.8-02 by substitution of synonyms, it woud follow that [T(P & Q) Syn T(~Q & ~P)], and therefore, [(TP & TQ) Syn (FP & FQ)'. This would make contrary expressions logically equivalent to each other! But the fault is not with Axiom 8-01 but with an over-simplified conception of Transposition.

The failure of this version of transposition has been offered as a conclusive objection to the C-conditional.[8] However, there are good arguments against accepting Transposition in the forms just given and for accepting instead certain weaker forms of transposition which are provably valid and useful. Transposition is not an absolute principle of A-logic. With TF-conditionals it holds, even in A-logic: $\models$[(TP ⊃ TQ) Syn (~TQ ⊃ ~TP)] holds, and [(TP ⊃ TQ) <=> (~TQ ⊃ ~TP)] is Valid. But [TP => TQ] is not Syn to [~TQ => ~TP], and [(TP => TQ) <=> (~TQ => ~TP)]' is not valid. For the weaker forms of transposition with C-conditionals that are both valid and useful see Section 8.32.

Finally, there is one bit of ordinary usage which actually runs counter to the sharp separation we advocate between a) predicating Truth of (P => Q) and b) asserting the validity of (TP => TQ). This is the general tendency to conflate the validity of a conditional with the truth of that conditional. This tendency is formalized and exacerbated in M-logic's semantics where *validity* is identified with *logically true* statements. It must be over ridden if A-logic is to be accepted. The fundamental flaw in this usage of 'true' with respect to conditionals lies in perpetuating an ambiguity in the word 'true'. The validity of an inferential conditional is confused with assertions that that conditional as a whole is true. These are very different situations. Consider the statements,

(1) "If Joe is in Paris and whoever is in Paris is in France, then Joe is in France".
(2) "If Joe is in Paris and whoever is in Paris is in France, then Joe is at the Louvre."

All logicians recognize that (1) is a valid conditonal and that (2) is not.valid. The tendency is to think that a conditional's being valid, is just to think of it as being true in a special way—i.e., as being "necessarily **true**" or "universally **true**", or being **true regardless** of whether its antecedent and consequent are true. This stance is embedded in M-logic which associates validity with universal truth and does not distinguish falsehood from non-truth in its semantics or its truth-tables.

According to Axiom 8-01 to find that (2) is true at some point in time, we must establish that 'Joe is in Paris' is true and 'whoever is in Paris is in France' is true and 'Joe is at the Louvre' is true at that moment. But (2) can also be false. According to Axioms 8-02 (and the truth-tables for the TF-condtional), if we find Joe is in Paris and that whoever is in Paris is in France, but Joe is not at the Louvre, then (2) is false. According to A-logic's three-valued truth-tables, if Joe is not in Paris at some point in time, neither (1) nor (2) is true or false as of that moment (hence not-**true**). If Joe has died and never has been or will be in Paris, (2) is forever not-true and not- false.

To say (1) is true, at any point in time, is to say that both 'Joe is in Paris' and 'whoever is in Paris is in France' and 'Joe is in France' are true at that time. If Joe is not in Paris, or Paris is not in France, (1) is neither true nor false. But (1) has a property that (2) does not have: it is ***valid***. This means that unlike (2) it can not be false even if it is not true. Its validity (vs. its truth) does not depend on whether its components are both true. We can not *consistently* imagine that 'Joe is in Paris' and 'whoever is in Paris is in France' and 'Joe not in France' are all true together; but we can consistently imagine that Joe is in Paris and whoever is in Paris is in France, but that Joe is not at the Louvre. Validity entails unfalsifiability, not truth. In A-logic, unfalsifiability is not the same as universal truth; in M-logic it is. Further, (1) remains ***valid*** (though not-**true**) even if both antecedent and consequent taken by themselves are false; and it remains ***valid*** (though not-**true**) even when it is neither true nor false. A-validity requires containment relations which are relations of ideas; truth does not.

---

8. Hempel argued that this aspect of the C-conditional (which he called "Nicod's criterion of confirmation" of a conditional), is untenable because it conflicts with laws of transposition ("Studies in the Logic of Confirmation", Sect. 3, MIND, 1945). His argument is presented and dealt with in Section 10.331.

Ordinary usage allows people to assert "It is true that If Joe **were** in Paris Joe **would be** in France" even if Joe is not in Paris, is dead, or we don't know where Joe is. Ordinary usage also awkwardly allows people to claim that even if Joe is neither in Paris nor in France, that "it is necessarily true that **If** Joe is in Paris, **then** Joe is in France"—and that the conditional "**If** Joe is in Paris **then** Joe goes to the Louvre" may express a "true" statement about Joe's actual modes of behavior. But to say a conditional is true "regardless of whether the antecedent is true or false" contradicts Axiom 8-01 and the common attitude Quine pointed to—that a conditional is counted as true only if both its antecedent is true and its consequent is true. But all valid conditionals are not true by the criterion of common attitides. By definition, a valid *counterfactual* conditional does not have both true antecedents and true consequents. Traditionally, a valid argument need not have true premisses or a true conclusion. To say that a conditional is both (i) valid and (ii) true *when antecedent and consequent are not both true*, is to make two conflicting assertions. If Axiom 8-01 is accepted, it is an ozymoron. This facet of ordinary usage, which confuses truth and validity, involves an inconsistency in ordinary usage probably due to the fact that the traditional logical concept of "valid" argument has never been fully incorporated into ordinary usage.

It is either true or false that a given conditional, or conditional schema, is logically valid. But, that '[(P & Q) $\Rightarrow$ Q] is valid' *is true* does not entail that '[(P&Q) $\Rightarrow$ Q]' *is true*'; and the assertion that the statement "If any x is human, x has a human father" *is always true* in fact, does not entail that "If any x is human, x has a human father" is *logically valid*. Quite possibly human cloning will some day show that generalization to be false.

Thus according to A-logic, ordinary usage and M-logic ascribe "truth" to conditionals in two different, incompatible ways. A-logic's solution lies in the distinctions and relations between truth, validity and conditionals expressed in A-logic's definitions of validity and truth (which make the truth and the validity of conditionals quite independent of each other) and Axiom 8-01.

This confusion of validity with truth in ordinary usage may be the source of some objections, since Ax.8-01 stands against that use. A-logic admittedly deviates from some ordinary discourse as well as from M-logic in this respect. But the advantages of maintaining the separation between (i) predicating truth of (P $\Rightarrow$ Q), and (ii) asserting validity of (TP $\Rightarrow$ TQ) overrides the principle of respect for common usage in this case and is a necessary step in constructing a sound logic of truth-assertions.

8.2 Theorems

All theorems are derived from the logistic base (Section 8.21) which adds the new items in the base of Chapters 7 and 8 to that of Chapter 6. The basic theorems of analytic truth-logic are presented in Section 8.22 which covers Syn-, Cont- and Impl-theorems added by virtue of Axioms 8-01and 8-02. Section 8.23 is the most important section, covering Validity-theorems. Section 8.24 covers Inc- and TAUT-theorems; Section 8.25, theorems of Logical Unsatisfiability and Unfalsifiability; and Section 8.25, theorems of Logical Truth.and Logical Falsehood.

8.21 The Logistic Base: Definitions, Axioms, Rules of Inference

I. Primitives:	Grouping devices: (,), [,], < , > .		
	Predicate letters: $P_1, P_2, \ldots, P_n$.	[Abbr. 'P', 'Q', 'R']	PL
	Argument Place holders: 1, 2, 3,…		APH
	Individual Constants: $a_1, a_2, \ldots a_n$,	[Abbr. 'a', 'b', 'c']	IC
	Individual variables: $x_1, x_2, \ldots x_n$,	[Abbr. 'x', 'y', 'z']	IV
	Predicate operators: & ~ $\Rightarrow$ T		
	Conjunctive Quantifier: $(\forall v_i)$		
	Primitive (2nd Order) Predicate of Logic: Syn		

<u>II. Formation Rules</u>:
FR1. $[P_i]$ is a wff
FR2. If P and Q are wffs, [(P&Q)] is a wff.
FR3. If P_i is a wff and $t_j (1 \leq j \leq k)$ is an APH or a IC, then $P_i < t_1, \ldots, t_k >$ is a wff.
FR4. If $P_i < 1 >$ is a wff, then $[(\forall v_j) P_i v_j]$ is a wff.
FR5. If P is a wff, $[\sim P]$ is a wff.
FR6. If P and Q are wffs, $[(P \Rightarrow Q)]$ is a wff.
FR7. If A is wff, T(A) is wff

<u>III. Abbreviations, Definitions</u>

<u>Predicate Operators</u>:

Df 1-1.	[(P & Q & R)	SYN_{df} (P & (Q & R))]	
Df 3-1.	$[(\forall_k x)Px$	SYN_{df} $(Pa_1 \,\&\, Pa_2 \,\&\, \ldots \,\&\, Pa_k)]$	[Df '$(\forall x)$']
Df 4-1.	[(P v Q)	SYN_{df} $\sim(\sim P \,\&\, \sim Q)]$	[Df 'v',DeM1]
Df 4-2.	$[(P \supset Q)$	SYN_{df} $\sim(P \,\&\, \sim Q)]$	[Df '$\supset$']
Df 4-3.	$[(P \equiv Q)$	SYN_{df} $((P \supset Q) \,\&\, (Q \supset P))]$	[Df '$\equiv$']
Df 4-4.	$[(\exists x)Px$	SYN_{df} $\sim(\forall x) \sim Px]$	[Df '$(\exists x)$']
Df 6-1.	$[(P \Leftrightarrow Q)$	SYN_{df} $((P \Rightarrow Q) \,\&\, (Q \Rightarrow P))]$	[Df '$\Leftrightarrow$']
Df 7-1.	[F(P)	Syn_{df} $T(\sim P)]$	[Df 'F']
Df 7-2.	[0(P)	Syn_{df} $(\sim TP \,\&\, \sim FP)]$	[Df '0']

<u>Logical Predicates</u>

Df 'Cont'. [(P Cont Q) Syn_{df} '(P Syn (P&Q))]
Df 'Inc'. [Inc(P) Syn_{df} ((P Syn (Q&-R)) & (Q cont R))
v ((P Syn (Q&R)) & Inc(R))
v ((P Syn (QvR)) & Inc(Q) & Inc(R))
v ((P Syn $(Q \Rightarrow R)$ &Inc(Q&R)]
Df 'Taut' [Taut(P) Syn_{df} $Inc(\sim P)]$
Df 'P Ent Q' [(P Ent Q) Syn_{df} $((P \text{ Cont } Q) \,\&\, \sim Inc(P\&Q))]$
Df 'Impl' [(TP Impl TQ) Syn_{df} (i) ((0Q v TQ v FQ) & TP) Cont TQ)
(ii) & Not: (TP Cont TQ)
(iii) & Not: (0Q v TQ v FQ) Cont TQ]
Df 'Valid'[Valid$(P \Rightarrow Q)$ Syn_{df} ((P Ent Q) v (P Impl Q)) & not-Inc(P&Q))]

<u>IV. Axioms and Transformation Rules</u>

1. <u>Axioms</u>

Ax.6-01.	[P SYN (P&P)]	"&-IDEM1"
Ax.6-02.	[(P&Q) SYN(Q&P)]	"&-COMM"
Ax.6-03.	[(P&(Q&R)) SYN ((P&Q)&R)]	"&-ASSOC1"
Ax.6-04.	[(P&(QvR)) SYN ((P&Q)v(P&R))]	"&v-DIST1"
Ax.6-05.	$[P \text{ SYN } \sim\sim P]$	"DN"
Ax 6-06.	$\models [((P \Rightarrow Q) \,\&\, P) \text{ SYN } (((P \Rightarrow Q) \,\&\, P) \,\&\, Q)]$	
Ax.7-01.	$[TP \text{ Syn } (TP \,\&\, \sim FP)$	
Ax.7-02.	$[FTP \text{ Syn } \sim TP$	
Ax.7-03.	[T(P & Q) Syn (TP & TQ)	
Ax.7-04.	[T(P v Q) Syn (TP v TQ)	
Ax.7-05.	$[(TP \,\&\, \sim TP) \text{ v } TQ))$ Cont TQ]	
Ax.8-01.	$T(P \Rightarrow Q)$ Syn T(P&Q)	
Ax.8-02.	$F(P \Rightarrow Q)$ Syn $T(P\& \sim Q)$	

2. Principles of Inference

R1. If $\models$P and $\models$[Q Syn R] then $\models$P(Q//R) [SynSUB]

R3-2. If = R and $P_i<t_1,\ldots,t_n>$ occurs in R, [U-SUB]
and Q is an h-adic wff, where h $\geq$ n,
and Q has occurrences of all numerals 1 to n,
and no variable in Q occurs in R or S
then $\models$[R($P_i<t_1,\ldots,t_n>$/Q)

R2-3. If $\models$[P<1>] then $\models$[Pa] "INST"

R6-6. Valid[P $\Rightarrow$Q] if and only if Valid[P, .: Q] "VC\VI"

R7-1. If (P SYN Q) then (TP Syn TQ)

R7-2. If Inc(P) then $\models\sim$T(P).

We also have the derived rule, DR6-10, "Conditional Proof". A Conditional Proof is a proof that a conditional is valid. It is used to establish the Validity of Derived Rules.

DR6-10. If A is a consistent ordered set of wffs, $<A_1, A_2,\ldots,A_n>$ ($1\leq i \leq n$)
and for each A_i, (i >1) either $\models A_i$
or (g, h < i) and A_h is [$A_g \Rightarrow A_i$] and $\models$Valid[$A_g \Rightarrow A_i$]
then Valid($A_1 \Rightarrow A_i$) "Conditional Proof"

The seven extra axioms with T-operators and the extra rules of inference R7-1 and R7-2, cover material comparable to that in M-logic's Semantic Theory.

Analytic truth-logic, like M-logic, treats all expressions as truth-claims. A well-formed formula of truth-logic—a "T-wff"—is one in which every atomic component lies in the scope of a T-operator. All of the wffs we ascribe logical properties or relations to in truth logic are T-wffs.

T-operators are introduced into the wffs of theorems in earlier chapters in three ways.

First, by U-SUB. Since the result of prefixing a 'T' to any wff is a wff, (Formation Rule 7), any wff in any theorem of the earlier chapters may be converted into a theorems about T-wffs.by replacing each predicate letter P_i by 'TP_i.' and leaving everything else the same. This is brought about by U-SUB, but it does not exhaust all the possibilities, since it makes no provision for '$T\sim P_i$' or 'FP_i'.

Second, by Rule R7-1 and its derived rules. Each Cont- and Syn-theorem from previous chapterrs can be converted into a theorem about T-wffs by prefixing T-operators to their antecedent and consequent in accordance with R7-1 and DR7-1a to DR7-1e. And,

Third, by SynSUB. T-operators can be moved inward or outward by the Axioms 7-03 and 7-04 using SynSUB; by these and other axioms of Chapter 7 and 8 the positions of T-operators can be shifted from the atomic wffs outward, or from the over-all wffs inward, so that they can be placed or changed into all sorts of different positions while remaining a T-wff.

The move from T(P $\Rightarrow$Q) to (TP $\Rightarrow$TQ) is not logically valid. The converse move from (TP $\Rightarrow$TQ) to T(P $\Rightarrow$ Q) is also not valid. In this respect, the truth logic of C-conditionals differs from that of conjunctions or disjunctions in which a prefixed T can be distributed over their components, as in Ax.7-03. [T(P & Q) Syn (TP & TQ)] and Ax.7-04. [T(P v Q) Syn (TP v TQ)].

Nevertheless, DR7-NF, the Normal Form Theorem for T-wffs without C-conditionals can be extended to DR8-NF, which covers wffs with C-conditionals as well as '$\sim$', '&', '($\forall$x)', etc**.:**

DR8-NF. Every T-wff in Analytic Truth-logic is reducible to a Synonymous expression in which 'T' is prefixed only to negated or unegated atomic wffs.

By definition, every atomic component in a T-wff lies in the scope of a T-operator. If a C-conditional has a T-operator prefixed to it, it is synonymous to a T-wff which is a conjunction. Both T(P ⇒ Q) and F(P ⇒ Q) are, by Axioms 8-01 and 8-02, synonymous via Ax.7-03 with (TP & TQ) and (TP & FQ) respectively, making ∼T(P ⇒ Q) and ∼F(P ⇒ Q) synonymous (via DeMorgan and DN) to ∼(TP & TQ) and ∼(TP & FQ) respectively. Thus every wff which has a T-operator prefixed to it, will be reducible to a normal form T-wff without any C-conditionals.

If a T-wff is a conditional without a T-operator prefixed to it, then both its antecedent and its consequent must be T-wffs and will be reducible to normal form T-wffs, whether they are conjunctions, disjunctions or conditionals. If a C-conditional has no T-operator prefixed and has an antecedent or consequent which is not a T-wff, it is not a T-wff by definition.[9]

Thus every T-wff in A-logic is reducible to a synonymous T-wff in which T is prefixed to all and only elementary wffs.

This analysis preserves the difference between inferential conditionals and truth-assertions. Inferential conditionals are not reducible to categorical assertions if they do not lie in the scope of a T-operator. For though the T-operators distribute inwardly, reducing conditionals prefixed by T-operators to conjunctions as they go, they can not be distributed outwardly from conditionals which do not have a T-operator prefixed to them as a whole. If a conjunction or disjunction has only purely inferential conditionals, then reductions to normal form T-wffs all take place within the scope of conditional signs. But if a T-operator is prefixed to such a conjunction or disjunction, then the whole expression is reducible to the a truth-assertion in M-logic. These points are illustrated in the steps of the derivation which follows.

1) ((T((Pa & Qb) ⇒ TRab) & (TPa ⇒ T(∼Qb & ∼TRab)))	[Premiss]
2) Syn ((T((Pa & Qb) & TRab) & (TPa ⇒ T(∼Qb v ∼TRab)))	[1),Ax.8-01,SynSUB]
3) Syn ((T((Pa & Qb) & TRab) & (TPa ⇒ (**T**∼Qb **v** **T**∼TRab)))	[2),Ax.7-4,SynSUB]
4) Syn ((**T**(Pa & Qb) & **T**TRab & (TPa ⇒ (T∼Qb v T∼TRab)))	[3),Ax.7-3,SynSUB]
5) Syn ((T(Pa & Qb) & TTRab & (TPa ⇒ (**F**Qb v **F**TRab)))	[4),Df ‘F’,SynSUB]
6) Syn ((**T**Pa & **T**Qb & TTRab & (TPa ⇒ (FQb v FTRab)))	[5),Ax.7-3,SynSUB]
7) Syn ((TPa & TQb & **T**Rab & (TPa ⇒ (FQb v FTRab)))	[6),T7-20,SynSUB]
8) Syn (TPa & TQb & TRab & (TPa ⇒ (FQb v ∼TRab)))	[7),Ax.7-02,SynSUB]

Step 8) is a Normal Form T-wff. The left-most ‘⇒’ in the initial wff lay in the scope of ‘T’ and was eliminated; the right-most ‘⇒’ did not lie in the scope of T-operator and is preserved.

The same kind of process reduces quantified T-wffs to normal form. We will sometimes presuppose DR8-NF in making derivationa, so the results can come out with T’s and F’s prefixed only to atomic wffs.

8.22 Syn-, Cont-, and Impl-theorems From Ax.8-01 and Ax.8-02

As was suggested in the preceding section, every Syn- and Cont- theorem of Chapters 1, through 6, can be converted into several theorems of analytic truth-logic, simply by introducing T-operators by U-SUB or Rule 7-01 and its derivatives, followed by SynSUB with Axioms 7-03 and 7-04.

By changing the earlier purely logical structures into T-wffs we change their meanings. We add the concept that they are being related to some objective field or fields of reference. The difference between

9. E.g., ‘(P ⇒ (TQ ⇒ TR))’, ‘((P ⇒ Q) ⇒ (TP ⇒ TR)’, ‘((TP ⇒ TQ) ⇒ (P & Q))’ are not T-wffs.

so viewing them was exhibited in part by the special additional theorems derivable from the five axioms added for truth-logic in Chapter 7. In this chapter we add additional Syn, Cont and Impl theorems which are also not reducible to any theorem preceding them. These are based on the relationship between C-conditionals and the concepts of truth, non-truth, falsehood and non-falsehood.

Ax.8-01 and Ax.8-02 tell us how to confirm or refute truth-claims about conditionals. They reduce both the statement that a conditional is true, and the statement that a conditional is false, to descriptive assertions about what is true in an objective field of reference. They signify a conceptual relationship between *the idea that a conditional is true*, and the co-existence of facts in an objective field of reference which are independent of our imaginations. They show how determinations of actual facts would be relevant in considering the acceptability of conditional statements about the world.

This section contains no purely inferential conditionals, thus no validity-theorems. But in Section 8.23 the Syn-, Cont- and Impl-theorems of this section are used to prove the validity of inferential conditionals.

8.221 Syn- and Cont-theorems

There are a variety of Syn- and Cont-theorems which spell out the logical consequences of Ax.8-01 and Ax.8-02. We begin with commutations of both:

T8-01. [T(P&Q) Syn T(P⇒Q)]	[Ax.8-01,DR1-01]
T8-02. [T(P&~Q) Syn F(P⇒Q)]	[Ax.8-02,DR1-01]

The claim that a conditional is true contains the claim that its antecedent is true:

T8-11. [T(P ⇒Q) Cont TP]	
Proof: 1) [T(P ⇒Q) Syn T(P & Q)]	[Ax.8-01]
2) [T(P ⇒Q) Syn (TP & TQ)]	[1),Ax.7.03,SynSUB]
3) [T(P ⇒Q) Cont TP]	[2),Df 'Cont']

To claim a conditional is false also entails the claim that its antecedent is true:

T8-12. [F(P ⇒Q) Cont TP]	
Proof: 1) [F(P ⇒Q) Syn T(P & ~Q)]	[Ax.8-01]
2) [F(P ⇒Q) Syn (TP & T~Q)]	[1),Ax.7-03, SynSUB]
3) [F(P ⇒Q) Syn (TP & FQ)]	[2),Df'F',SynSUB]
4) [F(P ⇒Q) Cont TP]	[3),Df 'Cont']

The claim that a conditional is true entails that its consequent is true:

T8-13. [T(P ⇒Q) Cont TQ]	
Proof: 1) [T(P ⇒Q) Syn T(P & Q)]	[Ax.8-01]
2) [T(P & Q) Syn (TP & TQ)]	[Ax.7-03]
3) [T(P ⇒Q) Syn (TP & TQ)]	[1),2),SynSUB]
4) [T(P ⇒Q) Cont TQ]	[3),Df 'Cont']

The claim that a conditional is false entails that its consequent is false:

T8-14. [F(P ⇒ Q) Cont FQ]
Proof: 1) [F(P ⇒ Q) Syn T(P & ~Q)] [Ax.8-02]
2) [T(P & ~Q) Syn (TP & T~Q)] [Ax.7-03]
3) [F(P ⇒ Q) Syn (TP & T~Q)] [1),2),SynSUB]
4) [T(P ⇒ Q) Cont T~Q] [3),Df 'Cont']
5) [T(P ⇒ Q) Cont FQ] [4),Df 'F']

Something which might be mistaken for Hypothetical Syllogism is expressed by

T8-15. [(T(P ⇒ Q) & T(Q ⇒ R)) Cont T(P ⇒ R)]
Proof: 1) (T((P & Q & Q & R) Cont T(P & R))
2) (T((P & Q) & (Q & R) Cont T(P & R)) [1),&-ORD]
3) ((T(P & Q) & T(Q & R) Cont T(P & R)) [2),Ax.7-03,SynSUB]
4) ((T(P ⇒ Q) & T(Q ⇒ R) Cont T(P ⇒ R)) [3),Ax.8-01,SynSUB(thrice)]

But T8-15 is synonymous with [(TP & TQ & TR) Cont (TP & TR)] as can be seen by steps 1), 2) and 3). This would trivialize Hypothetical Syllogism, reducing it to Simplication. The genuine principle of Hypothetical Syllogism, must have inferential conditionals, not truth claims—as in
DR6-119. [VALID (P ⇒ Q) & VALID(Q ⇒ R)) ⇒ VALID(P ⇒ R)] and by U-SUBa,
DR8-119. [Valid(TP ⇒ TQ) & Valid(TQ ⇒ TR)) ⇒ Valid(TP ⇒ TR)], and
DR6-120.[VALID(P_1 ⇒ P_2) & VALID(P_2 ⇒ P_3}&…& VALID(P_{n-1} ⇒ P_n)) ⇒ VALID(P_1 ⇒ P_n]

If, under Ax.8-01, T(P ⇒ Q) means (TP & TQ) — i.e., "'If P then Q' is true" means 'P is true and Q is true' — then clearly, "'If P then Q' is true" means the same as "'If Q then P' is true" and both mean the same as "'P if and only if Q'is true".

T8-16. [T(P ⇒ Q) Syn T(Q ⇒ P)]
Proof: 1) T(P ⇒ Q) Syn T(P&Q)] [Ax.8-01]
2) T(P ⇒ Q) Syn T(Q&P)] [1),&-COMM, SynSUB]
3) T(P ⇒ Q) Syn T(Q ⇒ P)] [2),Ax.8-01, SynSUB]

By U-SUBab on T8-16 it follows that |= [T(P ⇒ ~Q) Syn T(~Q ⇒ P)], and |= [T(~P ⇒ Q) Syn T(Q ⇒ ~P)] and |= [T(~P ⇒ ~Q) Syn T(~Q ⇒ ~P)], although T(P ⊃ ~Q) is <u>not</u> Syn with T(~Q ⊃ P), i.e., T(~P v ~Q) is <u>not</u> Syn with T(Q v P). Also the truth-of a conditional is synonymous with the truth of its biconditional,

|= [T(P ⇒ Q) Syn T(Q <=> P)]
Proof: 1) [T(P ⇒ Q) Syn (T(P ⇒ Q) & T(P ⇒ Q))] [Ax.1-01]
2) [T(P ⇒ Q) Syn (T(P ⇒ Q) & T(Q ⇒ P))] [1),T8-16,SynSUB]
3) [T(P ⇒ Q) Syn T((P ⇒ Q) & (Q ⇒ P))] [2),Ax.7-03,SynSUB]
4) T (P ⇒ Q) Syn T(Q <=> P)] [2),Df '<=>']

But despite Tarski's Convention T, 'P is true' does not mean the same as 'P' and 'If P then Q' does not mean "'if P then Q' is true". The wffs (P ⇒ Q), (TP ⇒ TQ) and T(P ⇒ Q) all say different things. The first two are inferential conditionals, the last is not. Their trivalent truth-tables differ hence no two are synonmyous. T[P => Q] can be false when [P => Q] or [TP => TQ] are valid and cannot be false.

Because validity of conditionals is different than the truth of conditionals, Valid[P => Q] is not synonymous with Valid[Q => P] or Valid[P <=> Q]. While T8-15 is trivial, the transitivity of valid C-conditionals in DR6-119 and DR8-119 is not.

However, '[TP => TQ] <u>is true</u>' says the same thing as '[P => Q] <u>is true</u>' :

T8-17. [T(P => Q) Syn T(TP => TQ)]
<u>Proof</u>: 1) T(P => Q) Syn TT(P & Q) [Ax.8-01,T7-20,SynSUB]
2) T(P => Q) Syn T(TP & TQ) [1),Ax.7-3,SynSUB]
3) T(P => Q) Syn T(TP => TQ) [2),T8-01,SynSUB]

T(P => Q)			T(TP => TQ)		
F	F	F	F	F	F
F	**T**	F	F	**T**	F
F	F	F	F	F	F

But T[P => Q] does not imply Valid[TP => TQ]; and Valid[TP => TQ] does not imply T[P => Q]. However, Valid[P => Q] does entail ~F[TP => TQ].

From Ax.8-01 and Ax.8-02 together it follows that '[P => Q] is true' means '[P => ~Q] is false' and '[P => Q] is false' means '[P => ~Q] is true'. This resolves at one level Goodman's first problem of counterfactual conditionals.[10] If a C-conditional is true, it follows that a conditional with the same antecedent and the denial of its consequent is false.This follows from analytic truth-logic of C-conditionals. To assert a C-conditional true is synonymous with asserting that the conditional with the same antecedent and its negated consequent is false:

T8-18. [T(P => Q) Syn F(P => ~Q)]
<u>Proof</u>: 1) [T(P => Q) Syn T(P & Q)] [Ax.8-01]
2) [T(P => Q) Syn T(P & ~~Q)] [1),DN]
3) [T(P => Q) Syn F(P => ~Q)] [2),Ax.8-02]

T8-19. [F(P => Q) Syn T(P => ~Q)]
<u>Proof</u>: 1) [F(P => Q) Syn T(P & ~Q)] [Ax.8-02]
2) [F(P => Q) Syn T(P => ~Q)] [1),Ax.8-01,SynSUB]

It does not follow from the synonymies of T8-18 and T8-19, that [P => Q] and [~(P => ~Q] mean the same thing. '(P => Q)' is <u>not</u> Syn with '~(P => ~Q)' though they have identical truth-tables.

It is worthwhile examining why T8-18 and T8-19 are theorems but '[(P => Q) Syn ~(P => ~Q)]' is not. Why are some pairs of wffs which have identical truth-tables provably logically Synonymous, as in T8-18 and T8-19 while others pairs, with identical truth-tables are not logically synonymous?

F(P=>~Q)			T(P=>Q)			~(P=>~Q)			(P=>Q)		
F	F	F	F	F	F	0	0	0	0	0	0
F	**T**	F	F	**T**	F	0	T	F	0	T	F
F	F	F	F	F	F	0	0	0	0	0	0

1) First, two expressions may have different truth-tables taken by themselves but have identical truth-tables when 'T' is prefixed to them: Truth assertions may be referentially synonymous—may refer to the same states of affairs—though the meanings of the expressions by themselves are not synonymous.

---

10. "…the problem is to define the circumstances under which a given counterfactual [conditional] holds while the opposing conditional with the contradictory consequent fails to hold." Nelson Goodman, "The Problem of Counterfactual Conditionals," *Journal of Philosophy*, February 1947, p 114.

This is particularly clear in the case of the meanings of conditional statements vs the meaning of asserting them true. The meaning of asserting a conditional true, is the same as the meaning of asserting the conjunction of antecedent and consequent true, but the meaning of the conditional and the meaning of the conjunction are very different.

Example: ((P ⇒ Q)	(P & Q))	Ax.8-01 T(P ⇒ Q)	Syn T(P & Q)
0 0 0	0 0 **F**	F F F	F F F
0 **T** **F**	0 **T** **F**	F **T** F	F **T** F
0 0 0	**F** **F** **F**	F F F	F F F
(Different truth-tables)		(Identical truth-tables)	

That (P ⇒ Q) and (P & Q) do not have the same truth-tables, though T(P ⇒ Q) and T(P & Q) do have the same truth-tables, shows that the synonymy of the latter is due in part to the meaning of the word “True” as it applies to ‘ ⇒’, and not to the meaning of ‘ ⇒’ and ‘&’ alone. Expressions with different meanings may be true under the same conditions; but being true under the same conditions does not entail that the expressions by themselves have the same meaning.

2) The meanings of truth-claims, which assert correspondances with states of affairs in objective fields of reference, are fundamentally different than the meanings of inferentiall conditionals *simpliciter*. None of the meanings of the inferential conditionals (P ⇒Q), (TP ⇒TQ), or their denials ~(P ⇒ ~Q) and ~(TP ⇒ ~TQ) are synonymous, even though all truth assertions about them, i.e., T(P ⇒Q), T(TP ⇒TQ), and T~(P⇒~Q) and T(TP ⇒~TQ), are synonymous with each other.

Inferential conditionals and their denials				Truth-claims about inferenctial conditionals,etc.			
~(P ⇒~Q)	(P⇒Q)	(TP⇒TQ)	~(TP⇒ ~TQ)	T~(P⇒ ~Q)	T(P⇒Q)	T(TP⇒TQ)	T~(TP⇒ ~TQ)
0 0 0	0 0 0	0 0 0	0 0 0	F F F	F F F	F F F	F F F
0 T F	0 T F	0 T F	0 T F	F T F	F T F	F T F	F T F
0 0 0	0 0 0	0 0 0	0 0 0	F F F	F F F	F F F	F F F
None of these are synonymous with any other				All of these are synonymous with each other.			

Since (P ⇒Q) is POS, it is not Syn to ~(P ⇒ ~Q), which is NEG, altho their trivalent truth-tables are the same. With T-operators T(P ⇒Q) Syn F(P ⇒ ~Q); and both are POS due to ‘T’. In the four wffs on the right the truth-tables are necessary for synonymy; but for the four wffs on the left sameness of truth-tables does not produce synonymy.

3) If two expressions are synonymous they must have the same truth-tables, but having the same truth-tables does not necessarily make them synonymous. All of the valid theorems listed on the right have the same truth-table, but no two of them are synonymous. If two inferential conditionals are synonymous, then their antecedents should be synonymous and their consequents should be synonymous. No two consequents in these wffs are synonymous and one antecedent is not synonymous with any of the other five. Determination of two expressions' containment or synonymy is logically prior to determination of their truth-functions and truth-tables. This is clearly the case if the antecedent of a conditional is

[T(P⇒Q)	⇒	~F(TP⇒TQ)]
[T(P & Q)	⇒	TP]
[(TP & (FP v TQ))	⇒	TQ]
[T(P⇒Q)	⇒	~F(~Q⇒~P)]
[(TP⇒TQ)	⇒	~F(~Q ⊃ ~P)]
[(TP⇒TQ)	⇒	T(P⇒Q)]
0	0	0
0	T	0
0	0	0

inconsistent, for then regardless of the meaning of any terms in the antecedent or consequent the truth-table will be the same, having only '0's in all cases.

4) There are other systematic reasons why '(P ⇒ Q) Syn ~(P ⇒~ Q)' is not a theorem of A-logic. If (P ⇒ Q) Syn ~(P ⇒ ~ Q) were a theorem then (i) C-conditionals could be Taut, and (ii) contradictories could both be Inc, or both Taut, and (iii) the same conditional could be both inconsistent and tautologous, as the following derivation shows. If '((P ⇒ Q) Syn ~(P ⇒ ~ Q))' were a theorem, 'Inc ~((P& ~ P) ⇒ ~ P)' would follow from 'Inc((P& ~ P) ⇒ P)' and by Df 'Taut',

Inc[(P& ~ P) ⇒ P], Inc[~((P& ~ P) ⇒ ~ P], .: Taut[((P& ~ P) ⇒ ~ P]
Inc[(P& ~ P) ⇒ ~ P], Inc[~((P& ~ P) ⇒ P], .: **Taut[((P& ~ P) ⇒ P]**

These results must be avoided. They can not be derived from T8-18, T(P ⇒ Q) Syn F(P ⇒ ~ Q)] but they could have been derived if '((P ⇒ Q) Syn ~(P ⇒ ~ Q))' were a theorem.

Since denials of synonyms are synonymous, the denials of synonymous truth-assertaions are synonymous:

|= [~T(P ⇒ Q) Syn ~F(P ⇒ ~ Q)] [Proof: T8-18,DR4-1]
|= [~F(P ⇒ Q) Syn ~T(P ⇒ ~ Q)] [Proof: T8-19,DR4-1]

And since a conditional is not false if true, and not true if false,

|= [T(P ⇒ Q) Cont ~F(P ⇒ Q)] [Proof: T7-36,U-SUB]
|= [F(P ⇒ Q) Cont ~T(P ⇒ Q)] [Proof: T7-37,U-SUB]

Hence if a C-conditional is true, the opposing conditional is not-true:

T8-20. [T(P ⇒ Q) Cont ~T(P ⇒ ~ Q)]
Proof: 1) [T(P ⇒ Q) Cont ~F(P ⇒ Q)] [T7-36,U-SUB]
2) [~F(P ⇒ Q) Syn ~T(P ⇒ ~ Q)] [T8-19,DR4-1]
3) [T(P ⇒ Q) Cont ~T(P ⇒ ~ Q)] [1),2),SynSUB]

and if a C-conditional is false, the opposing conditional is not false (though it may not be true):

T8-21. [F(P ⇒ Q) Cont ~F(P ⇒ ~ Q)]
Proof: 1) [F(P ⇒ Q) Cont ~T(P ⇒ Q)] [T7-37,U-SUB]
2) [~T(P ⇒ Q) Syn ~F(P ⇒ ~ Q)] [T8-18,DR4-1]
3) [F(P ⇒ Q) Cont ~F(P ⇒ ~ Q)] [1),2),SynSUB]

Ax.8-01 amd Ax.8-02 preclude that version of the transposition principle which reads, "[T(P ⇒ Q) if and only if T(~Q ⇒ ~P)]". For T(P ⇒ Q) and T(~Q ⇒ ~P) are contraries. It is inconsistent to hold that they are both true at the same time, for by Ax.8-01 and Ax.8-02, they are synonymous with (TP & TQ) and (FP & FQ) respectively. To argue that 'if P then Q' is true if and only if 'if not ~Q then not ~P' is true, is an oxymoron if Axioms 8-01 is accepted.

However, milder versions of Transposition with C-conditionals are derivable: [P ⇒ Q] and [~Q ⇒~ P] are both false or not-false under the same conditions. For what makes [~Q ⇒~ P] false, according to Ax.8-02, is FQ and TP; and that is what makes [P ⇒ Q] false and not true.

T8-22. [F(P⇒Q) Syn F(~Q⇒~P),

Proof: 1) [F(P⇒Q) Syn T(P&~Q)] [Ax.8-02]
2) [F(P⇒Q) Syn T(~Q&P)] [1),&-COMM]
3) [F(P⇒Q) Syn T(~Q&~~P)] [2),DN]
4) [F(P⇒Q) Syn F(~Q⇒~P)] [Ax.8-02,SynSUB]

Since negations of synonyms are synoymous,

T8-23. [~F(P⇒Q) Syn ~F(~Q⇒~P)] [T8-22,DR4-1]

M-logic equates truth with non-falsity, so that 'T' is equivalent to '~F'. Perhaps this conflation explains why its proponents think T(if P then Q) entails T(if ~Q then ~P) based on T8-23.

The closest we get in A-logic to the first version of a transposition principle at this level is that If [P⇒Q] is true, then [~Q ⇒ ~P] can not be false. This is based on T8-24.

T8-24. [T(P⇒Q) Cont ~F(~Q⇒~P)]

Proof: 1) [T(P ⇒Q) Cont ~F(P ⇒Q)] [T7-36,U-SUB]
2) [~F(P⇒Q) Syn ~F(~Q⇒~P) [T8-23]
3) [T(P⇒Q) Cont ~F(~Q⇒~P)] [1),2),SynSUB]

A C-conditional and a TF-conditional with the same components are false under the same conditions. To say one is false is synonymous with saying the other is false. In both cases, a true antecedent and false consequent is synonymous with the falsehood of the conditional.

T8-25 [F(P⇒Q) Syn F(P ⊃ Q)]

Proof: 1) [F(P⇒Q) Syn T(P &~Q)] [Ax.8-02]
2) [F(P⇒Q) Syn T~~(P &~Q)] [1),DN]
3) [F(P⇒Q) Syn T~(P ⊃ Q)] [2),Df '⊃']
4) [F(P⇒Q) Syn F(P ⊃ Q)] [3),Df 'F']

However, the C-conditional and the TF-conditional are not true under the same conditions and thus can't be synonymous (as their truth-tables show).

(P⇒Q)	(P⊃Q)
0 0 0	0 T 0
0 T F	0 T F
0 0 0	T T T

If a C-conditional is either true or false, then its antecedent is true and its consequent either true or false. This is stated in T8-26.

T8-26. [(T(P⇒Q) v F(P⇒Q)) Syn (TP & (TQ v FQ))]

Proof: 1) (T(P⇒Q) v F(P⇒Q)) Syn (T(P ⇒Q) v F(P ⇒Q)) [T1-11]
2) (T(P⇒Q) v F(P⇒Q)) Syn (T(P & Q) v F(P ⇒Q)) [1),T8-01,SynSUB]
3) (T(P⇒Q) v F(P⇒Q)) Syn (T(P & Q) v T(P & ~Q)) [2),T8-02,SynSUB]
4) (T(P⇒Q) v F(P⇒Q)) Syn ((TP & TQ) v (TP & T~Q)) [3),Ax.7-3,SynSUB(twice)]
5) (T(P⇒Q) v F(P⇒Q)) Syn (TP & (TQ v T~Q)) [4),&v-DIST,SynSUB]
6) (T(P⇒Q) v F(P⇒Q)) Syn (TP & (TQ v FQ)) [5),Df 'F']

Since '(T(P⇒Q) v F(P⇒Q)' is synonymous with '~0(P⇒Q)', T8-26, yields two containment theorems. First, if (P ⇒ Q) is either true or false (.i.e, not neither true nor false), then the antecedent P must be true.

T8-27. [~0(P⇒Q) Cont TP]

Proof: 1) ~0(P⇒Q) Syn (T(P⇒Q) v F(P⇒Q)) [T7-49,U-SUB]
2) ~0(P⇒Q) Syn (TP & (TQ v FQ)) [1),T8-26,SynSUB]
3) ~0(P⇒Q) Cont TP [2),Df 'Cont']

Secondly if (P⇒Q is either true or false then the consequent, Q is either true or false; or synonymously, if it is not the case that (P⇒Q) is neither true nor false, then it is not the case that consequent, Q, is neither true nor false. This is expressed and proven in T8-28:

T8-28. [~0(P⇒Q) Cont ~0Q]

Proof: 1) ~0(P⇒Q) Syn ~(~T(P⇒Q) & ~F(P⇒Q)) [Df '0',U-SUB]
2) ~0(P⇒Q) Syn (T(P⇒Q) v F(P⇒Q)) [1),T4-12,SynSUB]
3) ~0(P⇒Q) Syn (TP & (TQ v FQ)) [2),T8-26,SynSUB]
4) ~0(P⇒Q) Cont (TQ v FQ) [3),Df 'Cont']
5) ~0(P⇒Q) Cont ~(~TQ & ~FQ) [4),Df 'v']
6) ~0(P⇒Q) Cont ~0Q [5),Df '0']

It also follows from T8-26 that if (P⇒Q) is neither true nor false, then either its antecedent is not true, or its consequent is neither true nor false. This is expressed and proven in T8-29.

T8-29. [0(P ⇒Q) Syn (~TP v 0Q)]

Proof: 1) (T(P⇒Q) v F(P⇒Q)) Syn (TP & (TQ v FQ)) [T8-26]
2) ~(T(P⇒Q) v F(P⇒Q)) Syn ~(TP & (TQ v FQ)) [1),DR4-1]
3) ~(T(P⇒Q) v F(P⇒Q)) Syn (~TP v ~(TQ v FQ)) [2),T4-18]
4) (~T(P⇒Q) & ~F(P⇒Q)) Syn (~TP v (~TQ & ~FQ)) [3),T4-17,SynSUB(twice)]
5) 0(P⇒Q) Syn (~TP v 0Q) [4),Df '0'(twice)]

If a C-conditional is neither true nor false, the the opposite conditional with the same antecedent but a contradictory conclusion, is also neither true nor false. This is expressed in T8-30:

T8-30. [0(P⇒Q) Syn 0(P⇒~Q)]

Proof: 1) 0(P ⇒Q) Syn (~T(P ⇒Q) & ~F(P ⇒Q)) [Df '0']
2) 0(P ⇒Q) Syn (~T(P ⇒Q) & ~T(P & ~Q)) [1),Ax.8-02,SynSub]
3) 0(P ⇒Q) Syn (~T(P & Q) & ~T(P & ~Q)) [2),Ax.8-01,SynSub]
4) 0(P ⇒Q) Syn (~T(P & ~~Q) & ~T(P & ~Q)) [3),DN,SynSUB]
5) 0(P ⇒Q) Syn (~T(P & ~~Q) & ~T(P ⇒ ~Q)) [4),Ax.8-01,SynSub]
6) 0(P ⇒Q) Syn (~F(P ⇒ ~Q) & ~T(P ⇒ ~Q)) [5),Ax.8-02,SynSub]
7) 0(P ⇒Q) Syn 0(P ⇒ ~Q) [6),Df '0',SynSUB]

The next entailment theorems are selected because of their role in establishing the basis of principles for the trivalent truth-table for (P⇒Q).The following Cont- and Syn-theorems provide grounds for four of the nine rows in the truth-table of the C-conditional. Proofs follow the truth-table. The basis of the other truth-table principles for (P⇒Q) are established in Section 8.222.

		P	Q	(P ⇒ Q)		
T8-31. [(0P & 0Q) Cont 0(P ⇒Q)]	➔	0	0	**0**	**0**	**0**
(For Row 1, of (P⇒Q) Table)		T	0	T	0	0
T8-32. [(FP & 0Q) Cont 0(P ⇒Q)]	➔	F	**0**	**F**	**0**	**0**
(For Row 3, of (P⇒Q) Table)		0	T	0	0	T
T8-33. [(TP & TQ) Syn T(P ⇒Q)]	➔	**T**	**T**	**T**	**T**	**T**
(For Row 5, of (P⇒Q) Table)		F	T	F	0	T
T8-34. [(TP & FQ) Syn F(P ⇒Q)]		0	F	0	0	F
(For Row 8, of (P⇒Q) Table)	➔	**T**	**F**	**T**	**F**	**F**
		F	F	F	0	F

T8-31. [(0P & 0Q) Cont 0(P ⇒Q)] [For Row 1, truth-table of '⇒']
Proof: 1) (0P & 0Q) Syn ((~TP & ~FP) & 0Q) [T1-1,Df '0']
2) (0P & 0Q) Syn (~FP & (~TP & 0Q)) [1),&-0RG]
3) (0P & 0Q) Syn (~FP & ((~TP & 0Q) & (~TP v 0Q)) [2),T1-23,SynSub]
4) (0P & 0Q) Cont (~TP v 0Q) [3),Df 'CONT']
5) (0P & 0Q) Cont 0(P ⇒Q) [4),T8-29,SynSub]

T8-32. [(FP & 0Q) Cont 0(P ⇒Q)] [For Row 3, truth-table of '⇒']
Proof: 1) (FP & 0Q) Syn (T~P & 0Q) [T1-11,U-SUB,Df 'F']
2) (FP & 0Q) Syn (T~P & ~T~~P & 0Q)) [1),T7-01,&-ORG]
3) (FP & 0Q) Syn (T~P & (~TP & 0Q)) [2),DN]
4) (FP & 0Q) Syn (FP & ((~TP&0Q) & (~TPv0Q))) [3),T1-23,SynSub]
5) (FP & 0Q) Cont (~TP v 0Q) [4),Df 'Cont']
6) (FP & 0Q) Cont 0(P ⇒Q) [5),T8-29,SynSub]

T8-33. [(TP & TQ) Syn T(P ⇒Q)] [For Row 5, truth-table of '⇒']
Proof: 1) (TP & TQ) Syn T(P & Q) [Ax.7-3]
2) (TP & TQ) Syn T(P ⇒Q) [1),Ax.8-01]

T8-34. [(TP & FQ) Syn F(P ⇒Q)] [For Row 8, truth-table of '⇒']
Proof: 1) (TP & FQ) Syn (TP & FQ) [T1-11]
2) (TP & FQ) Syn (TP & T~Q) [1),Df'F']
3) (TP & FQ) Syn T(P & ~Q) [2),Ax.7-3]
4) (TP & FQ) Syn F(P ⇒Q) [3),Ax.8-02]

Principles T8-31 to T8-34 are the basis of the valid conditionals which give the rules for Rows 1, 3, 5 and 8 of the truth table for '⇒'.

The next three Syn- and Cont-theorems provide the basis for A-implication-theorems with negated C-conditionals in their consequents in next section. Those A-implication-theorems, in turn, are the basis for the other five conditional rules for the Truth-table of the C-conditional in analytic truth-logic.

T8-35. [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn ((TPv0(P⇒Q)) & (~0Qv0(P⇒Q)))]
T8-36. [((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & ~TP) Cont ~T(P⇒Q)]
T8-37. [((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & 0Q) Cont (0(P⇒Q)]

T8-35. [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn ((TPv0(P⇒Q)) & (~0Qv0(P⇒Q)))]

<u>Proof</u>: 1) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn (T(P⇒Q) v F(P⇒Q) v 0(P⇒Q)] [T1-11]
2) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn (T(P⇒Q) v F(P⇒Q) v (**~TPv0Q**))] [1),T8-29]
3) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn (T(P⇒Q) v **T(P&~Q)** v (~TPv0Q))] [2),T8-02]
4) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn (**T(P&Q)** v T(P&~Q) v (~TPv0Q))] [3),T8-01]
5) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn ((**TP&TQ**) v (TP&FQ) v (~TPv0Q)] [4)Ax7-03]
6) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn (**(TP&(TQvFQ))** v (~TPv0Q))] [5),&v-DIST]
7) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn ((TP&~0Q) v (~TPv0Q))] [6),T7-49]
8) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn ((TPv(~TPv0Q)) & (~0Qv(~TPv0Q)))] [7),v&-DIST]
9) [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Syn ((TPv0(P⇒Q)) & (~0Qv0(P⇒Q)))] [8),T-28]

T8-36. [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & ~TP) Cont 0(P⇒Q)]

<u>Proof</u>: 1) (~TP & (TPvT0(P⇒Q))) Cont (~TP & TP) v T0(P⇒Q)) [T1-39, U-SUB]
2) ((~TP & TP) v T0(P⇒Q)) Cont T0(P⇒Q)) [Ax.7-05,U-SUB]
3) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Cont (TPv 0(P⇒Q)) [T8-35,Df'Cont']
4) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & ~TP) Cont ((TPv0(P⇒Q)) & ~TP) [3),DR1-21]
5) (") Cont (~TP & (TPv0(P⇒Q))) [4),&-COMM]
6) (") Cont ((~TP & TP) v T0(P⇒Q)) [4),1),CCCSyll]
7) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & ~TP) Cont T0(P⇒Q) [6),2),CCCSyll]
8) T0(P⇒Q) Syn 0(P⇒Q) [T7-53. U-SUB]
9) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & ~TP) Cont 0(P⇒Q)) [7),8),SynSUB]

T8-37. [(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & 0Q) Cont 0(P⇒Q)]

<u>Proof</u>: 1) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Cont (~0Q v 0(P⇒Q)) [T8-35,Df 'Cont']
2) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & 0Q) Cont((~0Q v 0(P⇒Q)) & 0Q) [1),DR1-21]
3) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & 0Q) Cont (0Q & (~0Q v 0(P⇒Q)) [2),&-COMM]
4) (T0Q & (~T0Q v T0(P⇒Q))) Cont T0(P⇒Q)) [T7-44,U-SUB]
5) T0(P⇒Q) Syn 0(P⇒Q) [T7-53,U-SUB]
6) (T0Q & (~T0Q v 0(P⇒Q))) Cont 0(P⇒Q)) [4),5), SynSUB(twice)]
7) T0Q Syn 0Q
8) (0Q & (~0Q v 0(P⇒Q))) Cont 0(P⇒Q)) [6),7), SynSUB(twice)]
9) ((T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & 0Q) Cont 0(P⇒Q) [3),8),CCCSyll]

The next theorem, T8-38 is preparatory to an implication-theorem which is A-logic's version of Modus Tollens, namely Ti8-44. Valid_I [((TP ⇒TQ) & ~TQ) ⇒~TP]. If we have an assertion of an inferential conditional of truth-logic with an assertion that the consequent is not true, the non-truth of the antecedent is <u>not</u> contained in these two assertions. However, it **is** contained in these two statements together with the presupposition that Law of Trivalence applied to that inferential conditional. Ti8-44 is an ellipsis based on dropping out the trivalence presupposition in T8-38,

T8-38. [(**T(TP⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ)** & (TP⇒TQ) & ~TQ) Cont ~TP].

The proof of T8-38 is a bit complicated so we do it in two steps, beginning with Lemma 1, which shows that the Law of Trivalence for (TP⇒TQ), contains the formula (TP v ~TP vTQ):

<u>Lemma 1</u> (T(TP⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ) Cont (TP v ∼TP v TQ)

<u>Proof</u>:1) (T(TP⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ)
Syn (TTP&TTQ)v(TTP&FTQ)v(∼T(TP⇒TQ)&∼F(TP⇒TQ)))
2) (“) Syn ((TP&TQ)v(TP&∼TQ)v(∼(TTP&TTQ)&∼T(TP&∼TQ)))
3) (“) Syn ((TP&TQ)v(TP&∼TQ)v(∼(TP&TQ)&∼(TP&∼TQ)))
4) (“) Syn ((TP&TQ)v(TP&∼TQ)v((∼TPv∼TQ)&(∼TPvTQ)))
5) (“) Cont (((**TP**&TQ)v(**TP**&∼TQ)) v **(∼TPvTQ)**))
6) (“) Cont ((**TP** & (TQv∼TQ)) v **(∼TPvTQ)**)) [5),&v-Dist,SynSUB]
7) (“) Cont ((**TP** v **(∼TPvTQ)**)) & ((TQv∼TQ)) v (∼TPvTQ))) [6),v&-Dist]
8) (T(TP ⇒ TQ)v(F(TP⇒TQ)v(0(TP⇒TQ) Cont (TPv∼TPvTQ) [7),Df'Cont']

We next conjoin the antecedent of the intended implication, ((TP⇒TQ) & ∼TQ) to the antecedent and consequent in Lemma 1, and then show that ∼TP is contained in ((TPv∼TPvTQ) & (TP⇒TQ) & ∼TQ). In the move from Step 5) to Step 6), we will use Ax.6-06.

T8-38. [(T(TP⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ) & (TP⇒TQ) & ∼TQ) Cont ∼TP]

<u>Proof</u>: 1) (T(TP⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ) Cont ((TP v ∼TP v TQ) [Lemma 1]
2) [(TP & (TP⇒TQ)) Syn (TP & (TP⇒TQ) & TQ)] [Ax.6-06,U-SUBa]
3) ((T(TP⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ) & (TP⇒TQ) & ∼TQ))
Cont (((TPv∼TPv**TQ**) & (TP⇒TQ) & **∼TQ**) [1),DR1-21]
4) (“) Cont (((**TPv∼TP)** & (TP⇒TQ)) & ∼TQ) [3),T7-45,CCC-Syll]
5) (“) Cont (((TP&**(TP⇒TQ)**)v(∼TP&**(TP⇒TQ)**)) & ∼TQ) [4),&v-DIST]
6) (“) Cont (((TP&(TP⇒TQ)**&TQ**)v(∼TP&(TP⇒TQ)) & ∼TQ) [5,2),SynSUB]
7) (“) Cont ((((TP&(TP⇒TQ))v(∼TP&(TP⇒TQ)))
&(TQv(∼TP&(TP⇒TQ))) & ∼TQ) [6),v&DIST]
8) (“) Cont (**TQv**(∼TP & (TP⇒TQ))) & **∼TQ**) [7),Df 'Cont']
9) (“) Cont (∼TP & (TP⇒TQ)) [8),DisjSyll T7-45]
10) (T (TP ⇒TQ)v(F(TP⇒TQ)v(0(TP⇒TQ) & (TP⇒TQ) &∼TQ) Cont∼TP [9),Df 'Cont']

We will not develop many Quantification Theorems. However there are two Cont-theorems in quantification theory with T-operators and C-conditionals which help to locate where A-logic will go with respect to '⇒'-for-'⊃' analogues of M-logic Axioms.

T5-437. TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx), is A-logic's version of Quine's *101, Rosser's Axiom 4, and Thomason's Axiom AS5. And T5-09. TAUT[(∀y)((∀x)Pxy ⊃ Pyy)] was A-logic's version of Rosser's Ax.6, used in lieu of Quine *103. Related to these theorems we have the following theorems in Analytic Truth Logic:

T8-39. [((∀x)(TPx ⇒TQx) & (∀x)TPx) Cont (∀x)TQx]

<u>Proof</u>: 1) [(TP ⇒TQ) & TP) Cont TQ] [T8-613]
2) [(TPa ⇒TQa) & TPa) Cont TQa] [1),U-SUB]
3) [(∀x)((TPx ⇒TQx) & TPx) Cont (∀x)TQx] [2),DR3-3e]
4) [(∀x)((TPx ⇒TQx) & TPx) Syn ((∀x)(TPx ⇒TQx) & (∀x)TPx)] [T3-13,U-SUB]
5) [(∀x)((TPx ⇒TQx) & (∀x)TPx) Cont (∀x)TQx)] [4),3),SynSUB]

From T8-39 we will getT8-839. Valid [((∀x)(TPx ⇒TQx) & (∀x)TPx) ⇒(∀x)TQx]. But the '⇒'for'⊃' analogue of Quine's *101, (∀x)(TPx ⇒TQx) ⇒((∀x)TPx) ⇒(∀x)TQx)' is not valid, because the antecedent does not Contain the consequent.[11]

Rosser's Axiom 6, which may be used in lieu of Quine's *103, has a '⇒'-for-'⊃' analogue with a C-conditional, '⊨Valid [(∀y) ((∀x)TPxy ⇒TPyy)]'. This theorem is derivable by substituting 'TP' for 'P' in T3-48: [(∀y) ((∀x)Pxy CONT Pyy)], using U-SUBa.

8.222 Impl-Theorems

Some theorems which are based on the definition of A-implication in Chapter 7 have occurrences of both '⇒' and a T-operator and are the basis of rules in the trivalent truth-table for '⇒'. Two important ones are Ti8-40, [∼TP Impl 0(P⇒Q)] and Ti8-43, [0Q Impl 0(P ⇒Q)]. They are based on Cont-theorems T8-36 and T8-37 respectively. Together these cover all grounds on which one can hold that a C-conditional is neither true nor false: namely, either the antecedent is not true, or the consequent is neither true nor false.

It is easy to establish "if the antecedent is **not true** then the conditional is **not true**", either from T8-37 with Df 'Impl', or from Ti7-80 with Ax. 8-01 as follows:

⊨[∼TP Impl ∼T(P⇒Q)] "If it is not true that P, then [P⇒Q] is not true."
Proof: 1) ∼TP Impl ∼T(P & Q) [Ti7-80]
2) ∼TP Impl ∼T(P ⇒Q) [1),Ax.8-01,SynSUB]

From this many other implications will follow by SynSUB. Note that the consequent must be the denial of a truth-assertion, not merely the denial of a conditional.

	and byT7-15 and CII-Syll:	
⊨[∼TP Impl ∼T(P⇒Q)]	⊨[FP Impl ∼T(P⇒Q)]	but not: [∼TP Impl ∼(P⇒Q)]
⊨[∼TP Impl ∼T(TP⇒TQ)]	⊨[FP Impl ∼T(TP⇒TQ)]	but not: [∼TP Impl ∼(TP⇒TQ)]

But the stronger principle we want is Ti8-40, which is based on T8-36 and says, "If P is **not true** then (P⇒Q) is **neither true nor false**".

Ti8-40. [∼TP Impl 0(P⇒Q)]
Proof: 1) (T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) & ∼TP) Cont 0(P⇒Q) [T8-36]
2) Not-(∼TP Cont 0(P⇒Q))
3) Not-(T(P⇒Q)vF(P⇒Q)v0(P⇒Q)) Cont 0(P⇒Q
4) ∼TP Impl 0(P⇒Q) [1),2),3),Df 'Impl']

---

11. **Consider:** "If every x is such that if it is true that x is a Pacifist, then it is true that x is a Quaker and for every x it is true that x is a Pacifist" entails "for every x, it is true that x is a Quaker". [by T8-39].

But, "For every x if it is true that x is a pacifist, then it is true that x is a Quaker" does not entail "If, for every x, it is true that x is a pacifist, then, for every x, it is true that x is a Quaker". I.e. (∀x)(TPx ⇒TQx) does not entail ((∀x)TPx ⇒(∀x)TQx). Why not? Because there can be cases in which the first wff is empirically valid—has true cases and no false ones, though the second is false or null. For example, in a given domain, it may be that 1 percent of the domain is pacifist and all members of that 1% are Quakers (verifying the first wff) but the second is null in that domain, because the antecedent fails (not 100% are pacifists). Also even if 100% were to become pacifists, it doesn't follow that they would all become Quakers. The empirical validity of the first, does not entail the validity of the second.

This theorem, like other Impl-theorems that follow, satisfies the requirement of Clauses (ii) (Not: (TP Cont TQ)) and (iii) (Not: (0PvTPvFP) Cont TQ)) in the definition of 'Impl'; but to save space we do not include demonstrations of this in their proofs.[12] From Ti8-40 it follows that if P is false or P is neither true nor false, the conditional as a whole is neither true nor false.

Ti8-41. [FP Impl 0(P⇒Q)]

Proof: 1)	FP Cont ∼TP	[T7-15]
2)	∼TP Impl 0(P⇒Q)	[Ti8-40]
3)	FP Impl 0(P⇒Q)	[1),2),CII-Syll]

Ti8-42. [0P Impl 0(P⇒Q)]

Proof: 1)	0P Cont ∼TP	[T7-50]
2)	∼TP Impl 0(P⇒Q)	[Ti8-40]
3)	0P Impl 0(P⇒Q)	[1),2),CII-Syll]

It is easy to prove that if the consequent is not false, then the conditional is not false:

⊨[∼FQ Impl ∼F(P⇒Q)]

Proof: 1)	∼T∼Q Impl ∼T(P & ∼Q)	[Ti7-80,U-SUB(b)]
2)	∼T∼Q Impl ∼F(P ⇒Q)	[1),Ax.8-2,SynSUB]
3)	∼FQ Impl ∼F(P ⇒Q)	[2),Df 'F']

But the stronger principle we need is: "If the consequent is neither true nor false, then the conditional is neither true nor false". This is Ti8-43 which comes directly from T8-37**:.**

Ti8-43. [0Q Impl 0(P⇒Q)] [T8-37, Df 'Impl']

The "paradoxes of material implication" have no '⇒'-for-'⊃' analogues among A-implications:

Although '⊨[∼TP Impl (TP ⊃ TQ)]'	is a theorem,	'[∼TP Impl (TP ⇒TQ)]'	is not a theorem.
Although '⊨[TQ Impl (TP ⊃ TQ)]'	is a theorem,	'[TQ Impl (TP⇒TQ)]'	is not a theorem.
Although '⊨[FP Impl (TP ⊃ TQ)]'	is a theorem,	'[FP Impl (TP ⇒TQ)]'	is not a theorem.
Although '⊨[FQ Impl (TP ⊃ FQ)]'	is a theorem,	'[FQ Impl (TP ⇒FQ)]'	is not a theorem.

---

12. The following qualify as A-implications (satisfying clause (ii), Not: (TP Cont TQ)), because in each the consequent has a component that does not occur in the antecedent:

Ti8-40. [∼TP Impl 0(P⇒Q)]	No Q in antecedent
Ti8-41 [FP Impl 0(P⇒Q)]	No Q in Antecedent
Ti8-42. [0P Impl 0(P⇒Q)]	No Q in antecedent
Ti8-43. [0Q Impl 0(P⇒Q)]	No P in antecedent

Note: 0(P ⇒ Q) Syn (∼TP v (∼TQ & ∼FQ)) [T8-29,Df '0']

Ti8⇒R2. [(TP & 0Q) Impl 0(P ⇒ Q)	No ∼TP in antecedent
Ti8⇒R4. [(0P & TQ) Impl 0(P ⇒ Q)]	No ∼TQ in antecedent
Ti8⇒R6. [(FP & TQ) Impl 0(P ⇒ Q)]	No ∼TQ in antecedent
Ti8⇒R7. [(0P & FQ) Impl 0(P ⇒ Q)]	No ∼FQ in antecedent
Ti8⇒R9. [(FP & FQ) Impl 0(P ⇒ Q)]	No ∼FQ in antecedent

Finally, the following Impl-theorem is the basis of A-logic's version of the rule of Modus Tollens. Its consequent does not occur in the antecedent of the implication, but it does occur in the MOCNF of the antecedent of T8-38, from which it is derived.

Ti8-44. [(TP ⇒ TQ) & ~TQ) Impl ~TP] [T8-38,Df 'Impl']

Ti8-44 may be read as: "(If P is true, then Q is true) and Q is not true' implies that P is not true. It follows from this with T7-15, that [((TP ⇒ TQ) & FQ) Impl ~TP] is a theorem. But it does not follow that [((TP ⇒ TQ) & FQ) ⇒ FP] is a valid implication, for FP is not logically contained in the result of replacing 'TP' by 'FP' in the antecedent of T8-38. If (TP ⇒ TQ) is Valid and Q is false, this proves P is not true, but not that P is false. Truth-tables support this:

Valid$_I$[(~F(TP ⇒ TQ) & FQ) ⇒ ~TP]
(TF(F0 0 FF) T TF) **T** T F0

<u>but Not</u>: Valid$_I$[~F(TP ⇒ TQ) & FQ) ⇒ FP]
TF F0 0 FF T TF **F** F0

Theorems which express the principles of the truth-table of '⇒' depend on Axioms 8-01 and 8-02 and the Principle of Trivalence. The following Syn-, Cont- and Impl-theorems are the grounds of the valid C-conditionals in Section 8.232331 which provide a rule for each row in the trivalent truth-table of '⇒'. Each of these implication-theorems is named for the row of the truth-table of '⇒' which it represents. E.g., 'Ti8⇒R 2' is the name of the the theorem expressing the rule for the second row of the truth-table for '⇒', etc.

T8-31.	[(0P & 0Q) Cont 0(P ⇒ Q)]
Ti8 ⇒r2.	[(TP & 0Q) Impl 0(P ⇒ Q)]
T8-32.	[(FP & 0Q) Cont 0(P ⇒ Q)]
Ti8 ⇒r4.	[(0P & TQ) Impl 0(P ⇒ Q)]
T8-33.	[(TP & TQ) Syn T(P ⇒ Q)]
Ti8 ⇒r6.	[(FP & TQ) Impl 0(P ⇒ Q)]
Ti8 ⇒r7.	[(0P & FQ) Impl 0(P ⇒ Q)]
T8-34.	[(TP & FQ) Syn F(P ⇒ Q)]
Ti8 ⇒r9.	[(FP & FQ) Impl 0(P ⇒ Q)]

The principles for rows 1, 3, 5, and 8 are based on the four Syn- or Cont-theorems T8-31, T8-32, T8-33 and T8-34. The principles of the remaining rows, 2, 4, 6, 7, and 9, are Impl-theorems. In each of these the consequent contains an occurrence of a wff which is not a conjunct of the antecedent — specifically '~TP' or '~TQ' or '~FQ'.

Ti8⇒r2. [(TP & 0Q) Impl 0(P ⇒ Q) [For Row 2, truth-table of '⇒']

<u>Proof</u>: 1) (TP & 0Q) Cont 0Q [T1-37,U-SUB]
2) 0Q Impl 0(P⇒Q) [Ti8-43]
3) ((TP & 0Q) Impl 0(P ⇒ Q)) [1),2),CII-Syll]

Ti8⇒r4. [(0P & TQ) Impl 0(P ⇒ Q)] [For Row 4, truth-table of '⇒']

<u>Proof</u>: 1) ((~TP & ~FP) & TQ) Cont ~TP) [Df 'Cont']
2) (0P & TQ) Cont ~TP) [1),Df '0P',SynSUB]
3) (~TP Impl 0(P⇒Q)) [Ti8-40]
4) (0P & TQ) Impl (0(P ⇒ Q) [2),3),CII-Syll]

Ti8⇒r6. [(FP & TQ) Impl 0(P ⇒ Q)] [For Row 6, truth-table of '⇒']
Proof: 1) (FP & TQ) Cont FP [T1-36,U-SUB]
2) FP Impl 0(P⇒Q) [Ti8-41]
3) (FP & TP) Impl 0(P ⇒ Q) [2),3),CII-SYLL]

Ti8⇒r7. [(0P & FQ) Impl 0(P ⇒ Q)] [For Row 7, truth-table of '⇒']
Proof: 1) (0P & TQ) Cont 0P [T1-36,U-SUB]
2) (0P Impl 0(P ⇒ Q) [Ti8-42]
3) (0P & FQ) Impl 0(P ⇒ Q) [1),2),CII-Syll]

Ti8⇒r9. [(FP & FQ) Impl 0(P ⇒ Q)] [For Row 9, truth-table of '⇒']
Proof: 1) (FP & T ~ Q) Cont FP [T1-36,U-SUB]
2) (FP Impl 0(P ⇒ Q)) [Ti8-41]
3) (FP & FQ) Impl 0(P ⇒ Q)) [1),2),CII-Syll]

Thus we have complete set of Syn-, Cont- and Impl- theorems, sufficient to establish a conditional rule for each row of the trivalent truth-tables for '⇒'. The Validity-theorems, which express the "Truth-conditions" of '⇒' in Valid Conditional theorems of analytic logic with '⇒' as the conditional are derived from them. (See Section 8.232331.)

There is a trivial sense in which various forms which may appear to be Hypothetical Syllogisms and Sorites are presented as theorems involving truth-assertions about C-conditionals. For example, T8-15. [(T(P⇒Q) & T(Q ⇒R)) Cont T(P⇒R)]. These are trivial because the proofs reduce hypothetical syllogisms and sorities to Simplification. We want hypothetical syllogisms and sorites with inferential conditionals, e.g., a proof that [$((TP_1 \Rightarrow TP_2)$ & $(TP_2 \Rightarrow TP_3)$ & $(TP_3 \Rightarrow TP_4)) \Rightarrow (TP_1 \Rightarrow TP_4)$] is valid. Validity-theorems of this sort will be established in Sections 8.231 and 8.232, but with conditions concerning satisfiability.

A wff which may look like Modus Tollens with the <u>truth</u> of the conditional posited in the antecedent can never be valid in A-logic because its antecedent is always inconsistent. Thus by Axiom 8-01 with SynSUB '((T(P ⇒Q) & FQ) ⇒FP)' and '((T(P ⇒Q) & ~TQ) ⇒ ~TP)' become '((TP & **TQ & FQ**) ⇒ FP)' and '((TP & **TQ & ~TQ**) ⇒ ~TP)' which are inconsistent by Df 'Inc⇒', and not valid. To have "If P then Q" *true* and the consequent Q *not true* is an oxymoron according to Axiom 8-01, but to have (TP ⇒ TQ) *valid* and Q not true is possible.

Inc((T(P ⇒Q) & ~TQ)		Valid(TP ⇒TQ)	&	~TQ
Proof: 1) Inc[TP & TQ & ~TQ]	[Df 'Inc']	(I.e.,the truth-table	&	(In fact
2) Inc[T(P & Q) & ~TQ]	[1), Ax.7-03,SynSUB]	of 'TP⇒TQ' has		Q is not
3) Inc[T(P ⇒Q) & ~TQ]	[2), Ax.8-01,SynSUB]	a T and no F's)		True)

Therefore, if Modus Tollens is interpreted as assuming (P⇒Q) is true, it can not be A-valid due to violation of the satisfiability clause in Df 'Valid'. However restricted versions of the principle of Modus Tollens with inferential conditionals can be established without inconsistency, since inferential conditionals are not committed to either the truth or falsity of their components. A-logic's version of Modus Tollens is proven in Section 8.2321 as

T_I8-844: $Valid_I$ [(TP ⇒TQ) &~TQ) ⇒ ~TP], from Ti8-44. [(TP⇒TQ) & ~TQ) Impl ~TP].

Though Inc-theorems and TAUT-theorems (which include all theorems of M-logic) are true statements of logic, no wffs or statements which are Taut are **valid** in the sense of A-logic, since only conditional (C-conditional) statements can be A-valid statements.

8.23 Validity Theorems

VALIDITY theorems of various *inference schemata* were established in Chapter 5. VALIDITY-theorems with *C-conditionals* were established in Chapter 6 based on Syn- and Cont- theorems in Chapters 1 to 6 which met the consistency requirement for validity. Chapter 7 had truth operators but no C-conditionals, thus there were no valid C-conditionals in Chapter 7. However, Chapter 7 introduced T-operators and had Syn- and Cont- theorems with T-wffs from which the Valid *inference schemata in truth-logic* were derived. The present chapter has both T-operators and C-conditionals and this yields new validity-theorems consisting of Conditional statements about truth-claims. These are the special business of a full Analytic Truth-logic.

For theorems which assert the Validity of C-conditionals in truth-logic the problem is to find wffs to replace 'A' and 'C' in '(TA ⇒ TC)' such that the result would be logically valid according to Df 'Valid' and the meaning of 'T'. All of the sixteen forms,

(TA ⇒ TC),	(TA ⇒ ~TC),	(TA ⇒ FC),	(TA ⇒ ~FC),
(FA ⇒ TA),	(FA ⇒ ~TC),	(FA ⇒ FC),	(FA ⇒ ~FC),
(~TA ⇒ TC),	(~TA ⇒ ~TC),	(~TA ⇒ FC),	(~TA ⇒ ~FC),
(~FA ⇒ TA),	(~FA ⇒ ~TC),	(~FA ⇒ FC),	(~FA ⇒ ~FC),

can be viewed as instances of the form '(TA ⇒ TC)', since ⊨[FP Syn **T**FP], ⊨[~TP Syn **T**~TP], and ⊨[~FP Syn **T**~FP].

Validity in truth-logic is not a question of whether the conditional as a whole is ***true***, but of a relationship between the idea of the antecedent's being true or not, and the idea of the consequent's being true or not. The results depend on the meanings of logical forms, and in some case on the particular meaning of the word 'true'.

We first derive valid entailment theorems which are central to *de re* reasoning, then derive valid implication theorems which are essentially only *de dicto* vehicles. In both cases containment of some sort is required, but *de dicto* validity is based on containments of substantive linguistic concepts, while entailments use only the syncategorematic words, 'and', 'not', 'if…then', 'all' etc. The logical validity of a C-conditional in Truth-logic is an *a priori* guarantee, based on meanings, that if it should happen to be the case that the antecedent is, was, will be, or or would be true, then it will be the case that the consequent is, was, will be, or or would be true. Whether the antecedent, the consequent, or the whole conditional is true in fact is never entailed or implied by its validity, but to say a conditional is valid implies that it has no false instances.

8.231 De Re *Valid Conditionals—Based on Entailment-theorems*

Valid entailment theorems are C-conditionals in which the containment relation between antecedent and consequent is based on the purely syncategorematic words 'and', 'or', 'not', 'all' and 'if…then'.

This kind of valid conditional is of great use in *de re* reasoning, because the relation of logical containment is able to mirror the way we conceive that smaller facts can be contained (*de re*) as parts of a larger fact in objective reality. The antecedent of a logical containment is basically a conjunction, and the consequent is a conjunct of that conjunction. The concept of an objective field of reference is the concept of a conjunction of facts, with larger facts containing smaller facts as parts or components. The most familiar concept of an objective field of reference is the common sense concept of an external physical world. It, with its history, its people, is what common sense calls "reality". In that field of reference, we think of facts as having locations in time and in space. Some facts are facts of change taking place in specific time periods; some are thought of as fixed or static facts. Events are facts of change, or factual changes. Once they are fixed in history, they do not change. The event of Lincoln's assasination, or the event of the World War II, was composed of may smaller events, each fixed in time and place. The conjunction constituting the ordered sequence of the smaller facts is the stuff of which the larger facts are constituted. Laws of nature are about general relationships between these facts.

Not all entailments can mirror *de re* containments. For many entailments contain inconsistent components and the concept of objective reality entails that it contains no inconsistencies. And of course many logically valid containments, which have no inconsistencies, are conceivable ones that do not apply, in fact, to anything we believe could be the case in objective reality. For example, "If some body has played chess with every person who ever lived, then every person who ever lived has played chess with some one," is a VALID statement of the form '((∃x)(∀y)Rxy ⇒ (∀y)(∃x)Rxy)' although its antecedent and consequent correspond to nothing that anyone would expect to ever be a fact.

We will examine the Syn- and Cont-theorems that have been presented in all chapters up to this point, seeking to derive Valid entailment-conditionals of truth-logic, which could be useful in *de re* reasoning.

8.2311 Validity-Theorems from Syn- and Cont-theorems in Chapter 7 and 8

Both Chapter 7 and the present Chapter are about truth-logic, so all of their Syn- and Cont-theorems are about logical relations of T-wffs and truth-claims. Thus the problem in producing Valid conditionals is how to introduce conditionals in place of 'Cont' or 'Syn'. Conditional Validity-theorems of truth-logic can be derived from Syn- and Cont- theorems in Chapters 7 and 8 in which T-wffs occur, using U-SUB with 'T' and the principles,

DR6-6a. If (P Cont Q) & not-Inc(P&Q) then Valid [P ⇒ Q]
DR6-6b. If (P Syn Q) & not-Inc(P&Q) then Valid [P ⇒ Q]
DR6-6c. If (P Syn Q) & not-Inc(P&Q) then Valid [Q ⇒ P]
DR6-6d. If (P Syn Q) & not-Inc(P&Q) then Valid [P ⇔ Q]

Thus for example, we can prove T8-801 as follow:

T8-801. Valid [T(P&Q) ⇔ T(P ⇒ Q)]

<u>Proof</u>: 1) (T(P&Q) Syn T(P ⇒ Q)) **[Ax.8-01]**
2) Not-Inc(T(P&Q) & T(P ⇒ Q))
t t t t **t** t t t t [Inspection]
3) ((T(P&Q) Syn T(P ⇒ Q)) & not-Inc(T(P&Q) & T(P ⇒ Q))) [1),2) ADJ]
4) If (P Syn Q) & not-Inc(P&Q) then Valid (P ⇔ Q)] **[DR6-6d]**
5) If (T(P&Q) Syn T(P ⇒ Q)) & Not-Inc[T(P&Q) & T(P ⇒ Q)
then Valid [T(P&Q) ⇔ T(P ⇒ Q)] [1),**U-SUB**,'T(P&Q)'for 'P', 'T(P ⇒ Q)'for'Q']
6) Valid [T(P&Q) ⇔ T(P ⇒ Q)] [3),5),MP]

But instead of giving a full proof for T8-801we will write, T8-801. Valid [T(P&Q) ⇔ T(P ⇒ Q)] [Ax.8-01,DR6-6b,U-SUB]

The cited Syn or Cont-theorem satisfies the Containment requirement for validity. The joint-satisfiability requirement is satisfied in all of those listed below, since we showed that the Axioms from which the theorems were derived satisfied this requirement, and all of the theorems were derived from these axioms by using U-SUB, used only the restricted form (U-SUBab) which insures the preservation of satisfiability. Therefore it is sufficient to cite the Syn-theorem or Cont-theorem, though we could also include a check for consistency in each proof, like the truth-table check in Step 2 of the proof for T8-801.

All of the following can be proven by similar proofs and we use '[T..., DR6-6a,U-SUB]' or '[T..., DR6-6d,U-SUB]' to indicate this.

T8-801d. Valid [T (P & Q) ⇔ T(P ⇒ Q)] [T8-01,DR6-6d,U-SUB]
T8-802d. Valid [T(P&~Q) ⇔ F(P ⇒ Q)] [T8-02, DR6-6d,U-SUB]
T8-811a. Valid [T(P ⇒ Q) ⇒ TP] [T8-11, DR6-6a,U-SUB]
T8-812a. Valid [F(P ⇒ Q) ⇒ TP] [T8-12,DR6-6a,U-SUB]
T8-813a. Valid [T(P ⇒ Q) ⇒ TQ] [T8-13,DR6-6a,U-SUB]
T8-814a. Valid [F(P ⇒ Q) ⇒ FQ] [T8-13,DR6-6a,U-SUB]
T8-815a. Valid [(T(P ⇒ Q) & T(Q ⇒ R)) ⇒ T(Q ⇒ P)] [T8-15,DR6-6a,U-SUB]
T8-816d. Valid [T(P ⇒ Q) ⇔ T(Q ⇒ P)] [T8-16,DR6-6d,U-SUB]
T8-817d. Valid [T(P ⇒ Q) ⇔ T(TP ⇒ TQ)] [T8-17,DR6-6d,U-SUB]
T8-818d. Valid [T(P ⇒ Q) ⇔ F(P ⇒ ~Q)] [T8-18,DR6-6d,U-SUB]
T8-819d. Valid [F(P ⇒ Q) ⇔ T(P ⇒ ~Q)] [T8-19,DR6-6d,U-SUB]
T8-820d. Valid [T(P ⇒ Q) ⇒ ~T(P ⇒ ~Q)] [T8-20,DR6-6a,U-SUB]
T8-821d. Valid [F(P ⇒ Q) ⇒ ~F(P ⇒ ~Q)] [T8-21,DR6-6a,U-SUB]
T8-822d. Valid [F(P ⇒ Q) ⇔ F(~Q ⇒ ~P)] [T8-22,DR6-6d,U-SUB]
T8-823d. Valid [~F(P ⇒ Q) ⇔ ~F(~Q ⇒ ~P)] [T8-23,DR6-6d,U-SUB]
T8-824a. Valid [T(P ⇒ Q) ⇒ ~F(~Q ⇒ ~P)] [T8-24,DR6-6a,U-SUB]
T8-825d. Valid [F(P ⇒ Q) ⇔ F(P ⊃ Q)] [T8-25,DR6-6d,U-SUB]
T8-826d. Valid [(T(P ⇒ Q) v F(P ⇒ Q)) ⇔ (TP & (TQ v FQ))] [T8-26,DR6-6d,U-SUB]
T8-827a. Valid [~0(P ⇒ Q) ⇒ TP] [T8-27,DR6-6a,U-SUB]
T8-828a. Valid [~0(P ⇒ Q) ⇒ ~0Q] [T8-28,DR6-6a,U-SUB]
T8-829d. Valid [0 (P ⇒ Q) ⇔ (~TP v 0Q)] [T8-29,DR6-6d,U-SUB]
T8-830d. Valid [0 (P ⇒ Q) ⇔ 0(P ⇒ ~Q)] [T8-30,DR6-6d,U-SUB]

Using the same rules, we derive the following Validity-theorems from the Syn- and Cont-theorems in the various sections of Chapter 7. The only difference is that in Chapter 7 '⇒' does not occur in any wff on either side of 'Cont' or 'Syn', but in Chapter 8 it, or '⇔', does.

<u>From Axioms and Definitions in the Base of Truth-logic</u>.
T8-701d. Valid [TP ⇔ (TP & ~FP)] [T7-01,DR6-6d,U-SUB]
T8-702d. Valid [FTP ⇔ ~TP] [T7-02,DR6-6d,U-SUB]
T8-703d. Valid [T(P & Q) ⇔ (TP & TQ)] [T7-03,DR6-6d,U-SUB]
T8-704d. Valid [T(P v Q) ⇔ (TP v TQ)] [T7-04,DR6-6d,U-SUB]
T8-705a. Valid [T((TP & ~TP)vQ)) ⇒ TQ] [T7-05,DR6-6a,U-SUB]
T8-706d. Valid [F[P] ⇔ T~P] [T7-06,DR6-6d,U-SUB]
T8-707d. Valid [0[P] ⇔ (~TP & ~FP)] [T7-07,DR6-6d,U-SUB]

From Section 7.42121. Theorems derived from Axioms 7-1 and Df'F'

T8-711d. Valid [TP ⇔ TP]		[T7-11,DR6-6d,U-SUB]
T8-712d. Valid [FP ⇔ FP]		[T7-12,DR6-6d,U-SUB]
T8-713a. Valid [TP ⇒ ~FP]		[T7-13,DR6-6a,U-SUB]
T8-714d. Valid [FP ⇔ (FP & ~TP)]		[T7-14,DR6-6d,U-SUB]
T8-715a. Valid [FP ⇒ ~TP]		[T7-15,DR6-6a,U-SUB]
T8-716d. Valid [F~P ⇔ TP]		[T7-16,DR6-6d,U-SUB]

From Sect. 7.42122. Theorems for Reduction to Normal Form T-wffs (from Ax, 7-1 to Ax. 7-4)

T8-717d. Valid [FFP ⇔ ~FP]		[T7-17,DR6-6d,U-SUB]
T8-718a. Valid [FP ⇒ FTP]		[T7-18,DR6-6a,U-SUB]
T8-719a. Valid [TP ⇒ FFP]		[T7-19,DR6-6a,U-SUB]
T8-720d. Valid [TTP ⇔ TP]		[T7-20,DR6-6d,U-SUB]
T8-721d. Valid [TFP ⇔ FP]		[T7-21,DR6-6d,U-SUB]
T8-722d. Valid [F(P & Q) ⇔ (FP v FQ)]		[T7-22,DR6-6d,U-SUB]
T8-723d. Valid [F(P v Q) ⇔ (FP & FQ)]	(For v-table, Row 9)	[T7-23,DR6-6d,U-SUB]
T8-724d. Valid [T(∀x)Px ⇔ (∀x)TPx]		[T7-24,DR6-6d,U-SUB]
T8-725d. Valid [T(∃x)Px ⇔ (∃x)TPx]		[T7-25,DR6-6d,U-SUB]
T8-726d. Valid [~F(∀x)Px ⇔ (∀x)~FPx]		[T7-26,DR6-6d,U-SUB]
T8-727d. Valid [~F(∃x)Px ⇔ (∃x)~FPx]		[T7-27,DR6-6d,U-SUB]
T8-728d. Valid [F(∃x)Px ⇔ (∀x)FPx]		[T7-28,DR6-6d,U-SUB]
T8-729d. Valid [F(∀x)Px ⇔ (∃x)FPx]		[T7-29,DR6-6d,U-SUB]
T8-730d. Valid [~T(∃x)Px ⇔ (∀x)~TPx]		[T7-30,DR6-6d,U-SUB]
T8-731d. Valid [~T(∀x)Px ⇔ (∃x)~TPx]		[T7-31,DR6-6d,U-SUB]

From Section 7.42123. Other Syn- and Cont-theorems from Axioms 7-1 to 7-4

T8-732d. Valid [~FTP ⇔ TTP]		[T7-32,DR6-6d,U-SUB]
T8-733d. Valid [~FFP ⇔ TFP]		[T7-33,DR6-6d,U-SUB]
T8-734a. Valid [T(P ⊃ Q) ⇒ (TP ⊃ TQ)]		[T7-34,DR6-6a,U-SUB]
T8-735d. Valid [T(P ⊃ Q) ⇔ T(~Q ⊃ ~P)]		[T7-35,DR6-6d,U-SUB]
T8-736d. Valid [(TP ⊃ TQ) ⇔ (~TQ ⊃ ~TP)]		[T7-36,DR6-6d,U-SUB]
T8-737a. Valid [(FP & FQ) ⇒ F(P & Q)]	(For &-table, Row 9)	[T7-37,DR6-6a,U-SUB]
T8-738a. Valid [(TP & TQ) ⇒ T(P v Q)]	(For v-table, Row 5)	[T7-38,DR6-6a,U-SUB]
T8-739a. Valid [(FP & TQ) ⇒ T(P ⊃ Q)]	(For ⊃ table, Row 6)	[T7-39,DR6-6a,U-SUB]
T8-740d. Valid [(TP & FQ) ⇔ F(P ⊃ Q)]	(For ⊃ table, Row 8)	[T7-40,DR6-6d,U-SUB]
T8-741d. Valid [F(TP & ~TQ) ⇔ ~T(TP & ~TQ)]		[T7-41,DR6-6d,U-SUB]
T8-742d. Valid [T(~TP v TQ) ⇔ ~F(~TP v TQ)]		[T7-42,DR6-6d,U-SUB]
T8-743a. Valid [T(∀x)(Px ⊃ Qx) ⇒ (T(∀x)Px ⊃ T(∀x)Qx]		[T7-43,DR6-6a,U-SUB]

From Section 7.42124. Detachment theorems from Ax.7-5)
(Note that TF-Modus Ponens is derivable with a C-conditional as the main connective.)

T8-744a. Valid [(TP & (~TPvTQ)) ⇒ TQ]	"Alternative Syllogism #1"	[T7-44,DR6-6a,U-SUB]
⊨ [(TP & (TP ⊃ TQ)) ⇒ TQ]	"TF-Modus Ponens #1"	[T8-744a,T4-31,SynSUB]
T8-745a. Valid [((~TPvTQ) & ~TQ) ⇒ ~TP]	"Alternative Syllogism #2"	[T7-45,DR6-6a,U-SUB]
⊨ [((TP ⊃ TQ) & ~TQ) ⇒ ~TP]	"TF-Modus Tollens #1"	[T8-745a,T4-31,SynSUB]
T8-746a. Valid [(TP & T(~P v Q)) ⇒ TQ]	"Alternative Syllogism #3"	[T7-46,DR6-6a,U-SUB]
⊨ [(TP & T(P ⊃ Q)) ⇒ TQ]	"TF-Modus Ponens #2"	[T8-746a,T4-31,SynSUB]

T8-747a. Valid [(T(~P v Q) & FQ) ⇒ FP]	"Alternative Syllogism #4"	[T7-47,DR6-6a,U-SUB]
⊨ [(T(P ⊃ Q) & FP) ⇒ TQ]	"TF-Modus Ponens #2"	[T8-747a,T4-31,SynSUB]

<u>From Section 7.42125. Theorems about non-true and non-false expressions, from Df '0'</u>

T8-748d. Valid [0P ⇔ ~(TP v FP)]		[T7-48,DR6-6d,U-SUB]
T8-749d. Valid [~0P ⇔ (TP v FP)]		[T7-49,DR6-6d,U-SUB]
T8-750a. Valid [0P ⇒ ~TP]	not-Valid: [~TP ⇒ OP]	[T7-50,DR6-6a,U-SUB]
T8-751a. Valid [0P ⇒ ~FP]	not-Valid: [~FP ⇒ OP]	[T7-51,DR6-6a,U-SUB]
T8-752d. Valid [0~P ⇔ 0P]	0P ⇔ 0~P ⇔ T0P ⇔ F0P	[T7-52,DR6-6d,U-SUB]
T8-753d. Valid [0P ⇔ T(0P)]	~0P ⇔ ~T0P ⇔ F0P	[T7-53,DR6-6d,U-SUB]
T8-754d. Valid [~F(0P) ⇔ 0P]		[T7-54,DR6-6d,U-SUB]
T8-755a. Valid [(TP v 0P) ⇒ ~FP]		[T7-55,DR6-6a,U-SUB]
T8-756a. Valid [(FP v 0P) ⇒ ~TP]		[T7-56,DR6-6a,U-SUB]
T8-757d. Valid [(0P & 0Q) ⇔ (0(P&Q) & 0(PvQ))]		[T7-57,DR6-6d,U-SUB]
T8-758a. Valid [(0P & 0Q) ⇒ 0(P&Q)]	(For &-table, Row 1)	[T7-58,DR6-6a,U-SUB]
T8-759a. Valid [(0P & 0Q) ⇒ 0(PvQ)]	(For v-table, Row 1)	[T7-59,DR6-6a,U-SUB]
T8-760a. Valid [(0P & 0Q) ⇒ 0(P ⊃ Q)]	(For ⊃-table, Row 1)	[T7-60,DR6-6a,U-SUB]
T8-761d. Valid [0TP ⇔ (~TP & TP)]		[T7-61,DR6-6d,U-SUB]
T8-762d. Valid [0FP ⇔ (~FP & FP)]		[T7-62,DR6-6d,U-SUB]
T8-763d. Valid [00P ⇔ (F0P & ~F0P)]		[T7-63,DR6-6d,U-SUB]
T8-764a. Valid [((T(P&Q)vF(P&Q)v0(P&Q)) & FP) ⇒ F(P&Q)]		[T7-64,DR6-6a,U-SUB]
T8-765a. Valid [((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) ⇒ ~T(P&Q)]		[T7-65,DR6-6a,U-SUB]
T8-766a. Valid [((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0P)) ⇒ (0(P&Q)]		[T7-66,DR6-6a,U-SUB]

In Chapter 7 the introduction of T-operators provided a ground for allowing atomic wffs to occur both POS and NEG in a way that is useful. The definition of '0P' recognizes the possibility that an expression may be neither true nor false; shielded by the T-operator, P occurs both POS and NEG in '(~TP & ~T~P)'.[13] In Axiom 5, based on the principle that an inconsistent expression can not be true, the assertion, 'T((~P & P) v Q)', is said to entail that Q is true. Throughout analytic truth-logic, it is logically possible for disjunctive statements which have inconsistent disjuncts to be true, provided they have other disjuncts which are not inconsistent.

8.2312 Valid Conditionals from Syn- and Cont-theorems in Chapter 6

In Chapter 6 over one hundred VALIDITY-theorems with C-conditionals were derived from the SYN- and CONT-theorems of Chapters 1 through 6. What these theorems had, but Chapter 7 lacked, was C-conditionals. What Chapter 6 theorems lacks that Chapters 7 and 8 have, is T-operators. To bring the theorems of Chapter 6 into truth-logic, we must convert all wffs to T-wffs. This can be done in two ways, with slightly different results.

T-operators can be introduced into these theorems by U-SUBab. If 'P_i<1,2,...,n>'occurs in any Validity-theorem then '~(P_i<1,2,...,n>)', 'T(P_i<1,2,...,n>)', '~T(P_i<1,2,...,n>)', or 'T(~P_i<1,2,...,m>' can be substituted at all such occurrences under U-SUBab. Since the conjunction of antecedent and consequent of the initial theorem is not inconsistent, the result of any such substitution on the atomic wffs will also be valid and not inconsistent.

---

13. However, the presupposition of 'T' is that there is some entity of which P is true or ~P is true, making 'TP and 'T~P' both Pos, so that '(~TP & ~T~P) is Neg.

The act of prefixing a 'T' to all occurrences of each elementary wff in the normal form of these theorems can not introduce a new negation of any elementary wffs that exists in the wff, and therefore can not introduce an inconsistency where there was none before.

Thus a Valid C-conditional of truth-logic is derivable from each Syn- or Cont-theorems in Chapter 6. Each ascribes validity to an inferential conditional or biconditional.

⊨Valid [(TP & (TP ⇒TQ)) ⇔ ((TP & TQ)&(TP ⇒TQ))] [T6-11,U-SUBa]
⊨Valid [(TP & (TP⇒TQ)) ⇒(TP&TQ)] [T6-12,U-SUBa]
⊨Valid [((TP⇒TQ) & TP) ⇒TQ] **'MP', "Modus Ponens"** [T6-13,U-SUBa]
⊨Valid [(TP⇔TQ) ⇒(TQ⇒TP)] [T6-14,U-SUBa]
⊨Valid [(TP⇔TQ) ⇒(TP⇒TQ)] [T6-15,U-SUBa]
⊨Valid [((∀x)((TPx ⇒TQx) & TPx) ⇒(∀x)TQx)] [T6-20,U-SUBa]
⊨Valid [((∀x)((TPx ⇒TQx) & (∀x)TPx) ⇒(∀x)TQx] [T6-21,U-SUBa]

These may be introduced as steps in proofs of theorems, or may be adopted amd named as theorems in their own right. They can be supplemented, using U-SUBab, by theorems which prefix some other T-operator than 'T' to this or that predicates letter in any formula. For example, given T6-13.VALID [((P⇒Q) & P) ⇒Q], the following derivations are possible:

1) ⊨Valid [((∼P ⇒Q) & ∼P) ⇒Q] [T6-13, U-SUBb ('∼P' for 'P')]
2) ⊨Valid [((TP ⇒TQ) & TP) ⇒TQ] [T6-13,U-SUBa ('TP' for 'P', 'TQ' for 'Q')]
3) ⊨Valid [((∼TP ⇒TQ) & ∼TP) ⇒TQ] [1),U-SUBa ('TP' for 'P', 'TQ' for 'Q')]
4) ⊨Valid [((T∼P ⇒TQ) & T∼P) ⇒TQ] [T8-613,U-SUBb ('∼P' for 'P', 'TQ' for 'Q')]
5) ⊨Valid [((FP ⇒TQ) & FP) ⇒TQ] [4), Df'F'(twice)]

Every derivation of this kind must be based on a valid theorem of Chapter 6 and/or must satisfy both requirements to preserve validity.

A second way to produce Validity theorems with T-operators is to take a Syn- or Cont-theorem of Chapter 1 through 6 and apply one of the following derived rules from Chapter 7.

DR7-6a [If (P Cont Q) and not-Inc (P & Q) then Valid (TP, **.:**TQ)]
DR7-6b. [If (P Syn Q) and not-Inc (P & Q) then Valid (TP, **.:**TQ)]
DR7-6c. [If (P Syn Q) and not-Inc (P & Q) then Valid (TQ, **.:**TP)]
DR7-6d. [If (P Syn Q) and not-Inc (P&Q) then Valid ((TP, **.:**TQ) & (TQ, **.:**TP)]

This method prefixes a T-operator to the antecedent and consequent of the Syn- or Cont-theorem and results in a validity theorem about an inference schema if the consistency requirement is met, or, if we then use R6-6, the VC\VI principle, a valid conditional. For example,

T8-613d. Valid [T((P⇒Q) & P) ⇒TQ] **'MP', "Modus Ponens"** [T6-13,U-SUBa]
<u>Proof</u>: 1) [((P⇒Q) & P) CONT Q] [T6-13]
2) Not-Inc(((P⇒Q) & P) & Q] [Inspection]
T T T T T **T** T
3) Valid [T((P⇒Q) & P), **.:** TQ] [1),2), **DR7-6a]**
4) Valid [T((P⇒Q) & P) ⇒TQ] [3), R6-6, MP]

The following theorems about inference schemata are derivable by DR7-6a and DR7-6b from SYN- and CONT-theorems in Chapter 6:

|= Valid [T(P & (P ⇒ Q), .: T((P & Q)&(P ⇒ Q))] [T6-11,DR7-6b]
|= Valid [T(P & (P ⇒ Q), .: T(P&Q)] [T6-12,DR7-6a]
|= Valid [T((P ⇒ Q) & P, .: TQ] **'MP', "Modus Ponens"** [T6-13,DR7-6a]
|= Valid [T(P <=> Q, .: T(Q ⇒ P)] [T6-14,DR7-6a]
|= Valid [T(P <=> Q, .: T(P ⇒ Q)] [T6-15,DR7-6a]
|= Valid [T((∀x)((Px ⇒ Qx) & Px, .: T(∀x)Qx)] [T6-20,DR7-6a]
|= Valid [T((∀x)((Px ⇒ Qx) & (∀x)Px, .: T(∀x)Qx] [T6-21,DR7-6a]

From DR7-6a to DR7-6d, together with DR6-6f, we get the following derived rules of Chapter 8 which yield Valid conditionals from the same premisses:

DR8-6a [If (P Cont Q) and not-Inc (P&Q) then Valid (TP ⇒ TQ)]
DR8-6b. [If (P Syn Q) and not-Inc (P&Q) then Valid (TP ⇒ TQ)]
DR8-6c. [If (P Syn Q) and not-Inc (P&Q) then Valid (TQ ⇒ TP)]
DR8-6d. [If (P Syn Q) and not-Inc (P&Q) then Valid (TP **<=>** TQ)]

Their proofs are:

DR8-6a [If (P Cont Q) and not-Inc (P&Q) then Valid (TP ⇒ TQ)]
Proof: 1) |= [If (P Cont Q) and not-Inc (P & Q) then Valid (TP, .:TQ)] [DR7-6a]
2) |= [If Valid(P, .: Q) then Valid(P ⇒ Q)] [DR6-6f]
3 |= [If (P Cont Q) and not-Inc (P & Q) then Valid (TP ⇒ TQ)] [2),3),HypSYLL]

DR8-6b [If (P Syn Q) and not-Inc (P&Q) then Valid (TP ⇒ TQ)]
Proof: 1) |= [If (P Syn Q) and not-Inc (P & Q) then Valid (TP, .:TQ)] [DR7-6b]
2) |= [If Valid(P, .: Q) then Valid(P ⇒ Q)] [DR6-6f]
3 |= [If (P Syn Q) and not-Inc (P & Q) then Valid (TP ⇒ TQ)] [2),3),HypSYLL]

DR8-6c [If (P Syn Q) and not-Inc (P&Q) then Valid (TQ ⇒ TP)]
Proof: 1) |= [If (P Syn Q) and not-Inc (P & Q) then Valid (TQ, .:TP)] [DR7-6c]
2) |= [If Valid(Q, .: P) then Valid(Q ⇒ P)] [DR6-6f (re-lettered)]
3 |= [If (P Syn Q) and not-Inc (P & Q) then Valid (TP ⇒ TQ)] [2),3),HypSYLL]

DR8-6d [If (P Syn Q) and not-Inc (P&Q) then Valid (TP <=> TQ)]
Proof: 1) (P Syn Q) and not-Inc (P&Q) [Premiss]
2) Valid (TP ⇒ TQ)] [1),DR8-6b, MP]
3) Valid (TQ ⇒ TP)] [1),DR8-6c, MP]
4) Valid ((TP ⇒ TQ) & Valid (TQ ⇒ TP)) [2),3),ADJ]
5) Valid ((TP ⇒ TQ) & (TQ ⇒ TP)) [4), Df 'Valid&',SynSUB]
6) If (P Syn Q) and not-Inc (P&Q) then Valid ((TP ⇒ TQ) & (TQ ⇒ TP) [1) to 5), Cond.Pr.]
7) If (P Syn Q) and not-Inc (P&Q) then Valid (TP <=> TQ) [6),Df.' <=> ',SynSUB]

Another derived rule, DR8-6g, to get valid conditionals from Impl-theorems, will be introduced in Section 8-232. The following numbered conditional theorems are derivable by R8-6a and DR8-6d from SYN- and CONT-theorems in Chapter 6:

T8-611d. Valid [T(P & (P ⇒ Q)) ⇔ T((P & Q &(P ⇒ Q))]		[T6-11,DR8-6d]
T8-612a. Valid [TP & (P ⇒ Q)) ⇒ T(P&Q)]		[T6-12,DR8-6a]
T8-613a. Valid [T(P ⇒ Q) & P) ⇒ TQ]	'MP', "Modus Ponens"	[T6-13,DR8-6a]
T8-614a. Valid [T(P ⇔ Q) ⇒ T(Q ⇒ P)]		[T6-14,DR8-6a]
T8-615a. Valid [T(P ⇔ Q) ⇒ T(P ⇒ Q)]		[T6-15,DR8-6a]
T8-620a. Valid [T((∀x)((Px ⇒ Qx) & Px) ⇒ T(∀x)Qx)]		[T6-20,DR8-6a]
T8-621a. Valid [T((∀x)((Px ⇒ Qx) & (∀x)Px) ⇒ T (∀x)Qx]		[T6-21,DR8-6a]

Many additional Validity theorems can be derived from these theorems using U-SUBab, and/or SynSUB.

8.2313 Valid Conditionals from SYN- and CONT-theorems in Chapters 1 thru 4

The 110 SYN- and CONT-theorems of Chapters 1 to 4 have neither T-operators nor C-conditionals.

In Chapter 5 theorems T5-101 to T5-441a established the VALIDITY of 110 *inference schemata* based on those theorems using derived rules DR5-6a and DR5-6b.

In Chapter 6 theorems T6-101 to T6-441 derived the VALIDITY of 110 *C-conditionals* based on those theorems using rules DR6-6a and DR6-6d.

In Chapter 7 theorems T7-101 to T7-441a established the Validity of 110 *inference schemata* between wffs prefixed by T based on those theorems using rules DR7-6a and DR7-6b.

We now complete the job for analytic truth logic.

First, Validity-theorems of Analytic Truth-logic with C-conditionals, can be gotten using U-SUBab to prefix some T-operator to every occurrence of each predicate letter any of the VALIDITY-theorems of Chapter 6. For example, by U-SUBa., putting 'TP' for 'P' throughout
T6-101d. VALID [P ⇔ (P & P)], we derive ⊨Valid [TP ⇔ (TP & TP)]. Thus the list begins,

⊨Valid[TP ⇔ (TP & TP)]	[&-Idem]	[T6-101,U-SUBa (P/TP)]
⊨Valid[TP ⇔ (TP v TP)]	[v-Idem]	[T6-102,U-SUBa (P/TP)]
⊨Valid[(TP&TQ) ⇔ (TQ & TP)]	[&-Comm]	[T6-103,USUBa (P/TP, Q/TQ)]
⊨Valid[(TPvTQ) ⇔ (TQ v TP)]	[v-Comm]	[T6-104,U-SUBa (P/TP,Q/TQ)]
⊨Valid[(TP&(TQ&TR)) ⇔ ((TP&TQ)&TR)	[&-Assoc]	[T6-105,-SUBa (P/TP,Q/TQ,R/TR)]

and ends with

⊨Valid[((∀x)TPx ⊃ (∀x)TQx) ⇒ (∃x)(TPx ⊃ TQx)] [T6-441,U-SUBa (P/TP,Q/TQ)]

which comes from T6-441. VALID[((∀x)Px ⊃ (∀x)Qx) ⇒ (∃x)(Px ⊃ Qx)]

As was mentioned in Chapter 7, any T-operator, 'T', 'F', '∼T' or '∼F' can be substituted via U-SUBab.

Secondly, T8-101 to T8-441 can be derived from the SYN- and CONT-theorems of Chapter 1 to 4, by using DR8-6a and DR8-6d to derive the Validity of 110 C-conditionals in which antecedent and consequent are 1st-level T-wffs with a T prefixed to each. The full proofs from axioms, rules and definitions of Chapter 8 can easily be reconstructed by going back through the proof of theorems referred to in the name of the theorem. (E.g., to get a complete proof of T8-415, incorporate the proof of T4-15 instead of taking T4-15 as premiss.)

<u>From SYN- and CONT-theorems in Chapter 1</u>:

T8-101d. Valid[TP <=> T(P&P)] [&-IDEM] [Ax.1-01, DR8-6d]
T8-102d. Valid[TP <=> T(PvP)] [v-IDEM] [Ax.1-02, DR8-6d]
T8-103d. Valid[T(P&Q) <=> T(Q&P)] [&-COMM] [Ax.1-03, DR8-6d]
T8-104d. Valid[T(PvQ) <=> T(QvP)] [v-COMM] [Ax.1-04, DR8-6d]
T8-105d. Valid[T(P&(Q&R)) <=> T((P&Q)&R)] [&-ASSOC] [Ax.1-05, DR8-6d]
T8-106d. Valid[T(Pv(QvR)) <=> T((PvQ)vR)] [v-ASSOC] [Ax.1-06, DR8-6d]
T8-107d. Valid[T(Pv(Q&R)) <=> T((PvQ)&(PvR))] [v&-DIST-1] [Ax.1-07, DR8-6d]
T8-108d. Valid[T(P&(QvR)) <=> T((P&Q)v(P&R))] [&v-DIST-1] [Ax.1-08, DR8-6d]
T8-111d. Valid[TP <=> TP] [T1-11,DR8-6d]
T8-112d. Valid[T((P&Q) & (R&S)) <=> T((P&R) & (Q&S))] [T1-12,DR8-6d]
T8-113d. Valid[T((PvQ) v (RvS)) <=> T((PvR) v (QvS))] [T1-13,DR8-6d]
T8-114d. Valid[T(P & (Q&R)) <=> T((P&Q) & (P&R))] [T1-14,DR8-6d]
T8-115d. Valid[T(P v (QvR)) <=> T((PvQ) v (PvR))] [T1-15,DR8-6d]
T8-116d. Valid[T(Pv(P&Q)) <=> T(P&(PvQ))] [T1-16,DR8-6d]
T8-117d. Valid[T(P&(PvQ)) <=> T(Pv(P&Q))] [T1-17,DR8-6d]
T8-118d. Valid[T(P&(Q&(PvQ))) <=> T(P&Q)] [T1-18,DR8-6d]
T8-119d. Valid[T(Pv(Qv(P&Q))) <=> T(PvQ)] [T1-19,DR8-6d]
T8-120d. Valid[T(P&(Q&R)) <=> T(P&(Q&(R&(Pv(QvR)))))] [T1-20,DR8-6d]
T8-121d. Valid[T(Pv(QvR)) <=> T(Pv(Qv(Rv(P&(Q&R)))))] [T1-21,DR8-6d]
T8-122d. Valid[T(Pv(P&(Q&R))) <=> T(P&((PvQ)&((PvR)&(Pv(QvR)))))] [T1-22,DR8-6d]
T8-123d. Valid[T(P&(Pv(QvR))) <=> T(Pv((P&Q)v((P&R)v(P&(Q&R)))))] [T1-23,DR8-6d]
T8-124d. Valid[T(Pv(P&(Q&R))) <=> T(P&(Pv(QvR)))] [T1-24,DR8-6d]
T8-125d. Valid[T(P&(Pv(QvR))) <=> T(Pv(P&(Q&R)))] [T1-25,DR8-6d]
T8-126d. Valid[T(P&(PvQ)&(PvR)&(Pv(QvR))) <=> T(P&(Pv(QvR)))] [T1-26,DR8-6d]
T8-127d. Valid[T(Pv(P&Q)v(P&R)v(P&(Q&R))) <=> T(Pv(P&(Q&R)))] [T1-27,DR8-6d]
T8-128d. Valid[T((P&Q)v(R&S)) <=> T(((P&Q)v(R&S)) & (PvR))] [T1-28,DR8-6d]
T8-129d. Valid[T((PvQ)&(RvS)) <=> T(((PvQ)&(RvS)) v (P&R))] [T1-29,DR8-6d]
T8-130d. Valid[T((P&Q)&(RvS)) <=> T((P&Q) & ((P&R)v(Q&S)))] [T1-30,DR8-6d]
T8-131d. Valid[T((PvQ)v(R&S)) <=> T((PvQ) v ((PvR)&(QvS)))] [T1-31,DR8-6d]
T8-132d. Valid[T((PvQ)&(RvS)) <=> T(((PvQ)&(RvS)) & (PvRv(Q&S))] "Praeclarum" [T1-32,DR8-6d]
T8-133d. Valid[T((P&Q)v(R&S)) <=> T(((P&Q)v(R&S)) v (P&R&(QvS))] [T1-33,DR8-6d]
T8-134d. Valid[T((P&Q)v(R&S)) <=> T(((P&Q)v(R&S)) & (PvR) & (QvS))] [T1-34,DR8-6d]
T8-135d. Valid[T((PvQ)&(RvS)) <=> T(((PvQ)&(RvS)) v (P&R) v (Q&S))] [T1-35,DR8-6d]

<u>From CONT-theorems:</u>

T8-136a. Valid [T(P&Q) => TP] [T1-36,DR8-6a]
T8-137a. Valid [T(P&Q) => TQ] [T1-37,DR8-6a]
T8-138a. Valid [T(P&Q) => T(PvQ)] [T1-38),DR8-6a]
T8-122c(1) Valid[T(Pv(P&(Q&R))) => TP] [T1-22c(1),DR8-6a]
T8-122c(1) Valid[T(Pv(P&(Q&R))) => T(PvQ)] [T1-22c(2),DR8-6a]
T8-122c(2) Valid[T(Pv(P&(Q&R))) => T(PvR)] [T1-22c(3),DR8-6a]
T8-122c(4) Valid[T(Pv(P&(Q&R))) => T(Pv(QvR))] [T1-22c(4),DR8-6a]
T8-122c(1,2) Valid[T(Pv(P&(Q&R))) => T(P&(PvQ))] [T1-22c(1,2),DR8-6a]
T8-122c(1,3) Valid[T(Pv(P&(Q&R))) => T(P&(PvR))] [T1-22c(1,3),DR8-6a]
T8-122c(2,3) Valid[T(Pv(P&(Q&R))) => T((PvQ)&(PvR))] [T1-22c(2,3),DR8-6a]

T8-122c(2,4) Valid[T(Pv(P&(Q&R))) ⇒T((PvQ)&(Pv(QvR)))]	[T1-22c(2,4),DR8-6a]
T8-122c(3,4) Valid[T(Pv(P&(Q&R))) ⇒T((PvR)&(Pv(QvR)))]	[T1-22c(3,4),DR8-6a]

From CHAPTER 3 SYN and CONT theorems with Quantifiers

From SYN-theorems in Chapter 3:

		Analogues of Quine's Metatheorems:	
T8-311d. Valid[T((∀x) Px) ⇔T(Pa_1 & P_2 &…& P_n)]			[T3-11,DR8-6d]
T8-312d. Valid[T((∃x) Px) ⇔T(Pa_1 v P_2 v…v P_n)]		|	[T3-12,DR8-6d]
T8-313d. Valid[T(∀x)(Px & Qx) ⇔T((∀x)Px & (∀x)Qx)]		ML*140	[T3-13,DR8-6d]
T8-314d. Valid[T(∃x)(Px v Qx) ⇔T((∃x)Px v (∃x)Qx)]		ML*141	[T3-14,DR8-6d]
T8-315d. Valid[T(∀x)(∀y)Rxy ⇔T(∀y)(∀x)Rxy]		ML*119	[T3-15,DR8-6d]
T8-316d. Valid[T(∃x)(∃y)Rxy ⇔T(∃y)(∃x)Rxy]		ML*138	[T3-16,DR8-6d]
T8-317d. Valid[T(∀x)(P & Qx) ⇔T(P & (∀x)Qx)	}	ML*157	[T3-17,DR8-6d]
T8-318d. Valid[T(∃x)(P v Qx) ⇔T(P v (∃x)Qx)]	"Rules of	ML*160	[T3-18,DR8-6d]
T8-319d. Valid[T(∃x)(P & Qx) ⇔T(P & (∃x)Qx)]	Passage"	ML*158	[T3-19,DR8-6d]
T8-320d. Valid[T(∀x)(P v Qx) ⇔T(P v (∀x)Qx)]		ML*159	[T3-20,DR8-6d]
T8-321d. Valid[T(∀x)Px ⇔T((∀x)Px & (∃x)Px)]			[T3-21,DR8-6d]
T8-322d. Valid[T(∃x)Px ⇔T((∃x)Px v (∀x)Px)]			[T3-22,DR8-6d]
T8-323d. Valid[T(∃x)(Px & Qx) ⇔T((∃x)(Px & Qx) & (∃x)Px)]			[T3-23,DR8-6d]
T8-324d. Valid[T(∀x)(Px v Qx) ⇔T((∀x)(Px v Qx) v (∀x)Px)]			[T3-24,DR8-6d]
T8-325d. Valid[T((∀x)Px & (∃x)Qx) ⇔T((∀x)Px & (∃x)(Px & Qx))]			[T3-25,DR8-6d]
T8-326d. Valid[T((∃x)Px v (∀x)Qx) ⇔T((∃x)Px v (∀x)(Px v Qx))]			[T3-26,DR8-6d]
T8-327d. Valid[T(∃y)(∀x)Rxy ⇔T((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]			[T3-27,DR8-6d]
T8-328d. Valid[T(∀y)(∃x)Rxy ⇔T((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]			[T3-28,DR8-6d]
T8-329d. Valid[T(∀x)(Px v Qx) ⇔T((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx)]			[T3-29,DR8-6d]
T8-330d. Valid[T(∃x)(Px & Qx) ⇔T((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx)]			[T3-30,DR8-6d]
T8-331d. Valid[T(∀x)(∀y)Rxy ⇔T((∀x)(∀y)Rxy & (∀x)Rxx)]			[T3-31,DR8-6d]
T8-332d. Valid[T(∃x)(∀y)Rxy ⇔T((∃x)(∀y)Rxy & (∃x)Rxx)]			[T3-32,DR8-6d]

From CONT-theorems in Chapter 3:

T8-333a. Valid [T(∀x)Px ⇒TPa		[T3-33,DR8-6a]
T8-334a. Valid [T(∀x)(∀y)Rxy ⇒T(∀x)Rxx		[T3-34,DR8-6a]
T8-335a. Valid [T(∃x)(∀y)Rxy ⇒T(∃x)Rxx		[T3-35,DR8-6a]
T8-336a. Valid [T(∀x)Px ⇒T(∃x)Px]	ML*136	[T3-36,DR8-6a]
T8-337a. Valid [T(∃y)(∀x)Rxy ⇒T(∀x)(∃y)Rxy]	ML*139	[T3-37,DR8-6a]
T8-338a. Valid [T((∀x)Px v (∀x)Qx) ⇒T(∀x)(Px v Qx)]	ML*143	[T3-38,DR8-6a]
T8-339a. Valid [T(∀x)(Px v Qx) ⇒T((∃x)Px v (∀x)Qx))]	ML*144	[T3-39,DR8-6a]
T8-340a. Valid [T(∀x)(Px v Qx) ⇒T((∀x)Px v (∃x)Qx))]	ML*145	[T3-40,DR8-6a]
T8-341a. Valid [T((∀x)Px v (∃x)Qx) ⇒T(∃x)(Px v Qx)]	ML*146	[T3-41,DR8-6a]
T8-342a. Valid [T((∃x)Px v (∀x)Qx) ⇒T(∃x)(Px v Qx)]	ML*147	[T3-42,DR8-6a]
T8-343a. Valid [T(∀x)(Px & Qx) ⇒T((∃x)Px & (∀x)Qx)]	ML*152	[T3-43,DR8-6a]
T8-344a. Valid [T(∀x)(Px & Qx) ⇒T((∀x)Px & (∃x)Qx)]	ML*153	[T3-44,DR8-6a]
T8-345a. Valid [T((∀x)Px & (∃x)Qx) ⇒T(∃x)(Px & Qx)]	ML*154	[T3-45,DR8-6a]
T8-346a. Valid [T((∃x)Px & (∀x)Qx) ⇒T(∃x)(Px & Qx)]	ML*155	[T3-46,DR8-6a]
T8-347a. Valid [T(∃x)(Px & Qx) ⇒T((∃x)Px & (∃x)Qx)]	ML*156	[T3-47,DR8-6a]

From SYN-theorems in Chapter 4:

T8-411d. Valid [T(P&Q) <=> T~(~Pv~Q)]		[DeM2]	[T4-11,DR8-6d]
T8-412d. Valid [T(PvQ) <=> T~(~P&~Q)]	[Df ‘v’]	[DeM1]	[T4-12,DR8-6d]
T8-413d. Valid [T(P&~Q) <=> T~(~PvQ)]		[DeM3]	[T4-13,DR8-6d]
T8-414d. Valid [T(Pv~Q) <=> T~(~P&Q)]		[DeM4]	[T4-14,DR8-6d]
T8-415d. Valid [T(~P&Q) <=> T~(Pv~Q)]		[DeM5]	[T4-15,DR8-6d]
T8-416d. Valid [T(~PvQ) <=> T~(P&~Q)]		[DeM6]	[T4-16,DR8-6d]
T8-417d. Valid [T(~P&~Q) <=> T~(PvQ)]		[DeM7]	[T4-17,DR8-6d]
T8-418d. Valid [T(~Pv~Q) <=> T~(P&Q)]		[DeM8]	[T4-18,DR8-6d]
T8-419d. Valid [TP <=> T(PvP)]		[v-IDEM]	[T4-19,DR8-6d]
T8-420d. Valid [T(PvQ) <=> T(QvP)]		[v-COMM]	[T4-20,DR8-6d]
T8-421d. Valid [T(Pv(QvR)) <=> T((PvQ)vR)		[v-ASSOC]	[T4-21,DR8-6d]
T8-422d. Valid [T(P&(QvR)) <=> T((P&Q)v(P&R))]		[v&-DIST]	[T4-22,DR8-6d]
T8-424d. Valid [T(∃x)~Px <=> T~(∀x)Px]	[Q-Exch2]	ML*130	[T4-24,DR8-6d]
T8-425d. Valid [T(∀x)~Px <=> T~(∃x)Px]	[Q-Exch3]	ML*131	[T4-25,DR8-6d]
T8-426d. Valid $[T(Ex_1)\ldots(Ea_n)\sim P<x_1,\ldots,x_n> \;<\!=\!>\; T\sim(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n>]$	[Q-Exch4]	ML*132	[T4-26,DR8-6d]
T8-427d. Valid $[T(\forall x_1)\ldots(\forall x_n)\sim P<x_1,\ldots,x_n> \;<\!=\!>\; T\sim(Ex_1)\ldots(Ex_n)P<x_1,\ldots,x_n>]$	[Q-Exch5]	ML*133	[T4-27,DR8-6d]

From CHAPTER 4, SYN- and CONT-theorems (with TF-conditionals).

T8-430d. Valid [T(P ⊃ Q) <=> T~(P&~Q)]	[Df ‘⊃’]		[T4-30,DR8-6d]
T8-431d. Valid [T(~PvQ) <=> T(P ⊃ Q)]			[T4-31,DR8-6d]
T8-432d. Valid [T(P ⊃ Q) <=> T(~Q ⊃ ~ P)]			[T4-32,DR8-6d]
T8-433d. Valid [T(∃x)(Px ⊃ Qx) <=> T((∀x)Px ⊃ (∃x)Qx)]		ML*142	[T4-33,DR8-6d]
T8-434d. Valid [T(∃x)(Px ⊃ Q) <=> T((∀x)Px ⊃ Q)]	“Rules of	ML*162	[T4-34,DR8-6d]
T8-435d. Valid [T(∀x)(Px ⊃ Q) <=> T((∃x)Px ⊃ Q)]	Passage”	ML*161	[T4-35,DR8-6d]
T8-436a. Valid [T((∃x)Px ⊃ (∀x)Qx) => T(∀x)(Px ⊃ Qx)]		ML*148	[T4-36,DR8-6a]
T8-437a. Valid [T(∀x)(Px ⊃ Qx) => T((∀x)Px ⊃ (∀x)Qx)]		ML*101	[T4-37,DR8-6a]
T8-438a. Valid [T(∀x)(Px ≡ Qx) => T((∀x)Px ≡ (∀x)Qx)))]		ML*116	[T4-38,DR8-6d]
T8-439a. Valid [T(∀x)(Px ⊃ Qx) => T((∃x)Px ⊃ (∃x)Qx)]		ML*149	[T4-39,DR8-6a]
T8-440a. Valid [T((∃x)Px ⊃ (∃x)Qx) => T(∃x)(Px ⊃ Qx)]		ML*150	[T4-40,DR8-6a]
T8-441a. Valid [T((∀x)Px ⊃ (∀x)Qx) => T(∃x)(Px ⊃ Qx)]		ML*151	[T4-41,DR8-6a]

These results can be extended many times by one or more of the follow means: (i), using other T-operators than ‘T’ with U-SUBab, (ii) using different T-operators on different atomic wffs, (iii) using U-SUBa, which allows any negation-free wff to be uniformly substituted for P_i, provided it has no Predicate letters which occur elsewhere in the theorem, (iv) redistributing T-operators when possible, to synonymous T-wffs using SynSUB.

8.232 De Dicto *Valid Conditionals*

All Principles of Inference which warrant the Rules and Derived Rules of Inference set forth in this book are conditionals which are implicitly asserted to be Valid. Since they all talk about properties and relationships which belong to the meanings of linguistic terms, they may be described as valid *de dicto* theorems of truth-logic. A conditional or an inference is “valid *de dicto*” if it is valid and is about such purely linguistic properties and relationships. Section 8.2321 discusses the possibility of an extension of A-logic in which these principles are derived by principles of A-logic from definitions of the concepts involved.

In Analytic Truth-logic there are also A-implication theorems which are Valid *de dicto* but not *de re*, because they are based on Impl-theorems in which the logical Containment depends on the Law of Trivalence,—i.e., the linguistic presupposition that every expression has either the property of being true, or the propert of being false, or has neither property. A-implication theorems will be established in Section 8.2322.

Section 8.2323 distinguishes some appropriate and inappropriate uses of A-implications in connection with *de re* reasoning about what is true or not true.

8.2321 Principles of Inference as Valid Conditionals *De Dicto*

Principles of inference are Validity-theorems of A-logic in the broad sense. They are not theorems of pure formal logic, because what they talk about—the content of the clauses in antecedent and consequent—are not pure logical structures containing only syncategorematic words. They have predicate terms with substantive meanings—'Syn', 'Cont', 'Inc', 'Taut', 'Valid', etc. They belong to the special logic of logic ("metalogic"). This logic is comparable to the logic of mathematics (based on the meanings of 'is equal to', 'is the successor of',etc.), the logic of physics (based on the meaning of 'time' 'distance', 'volume', 'mass', 'velocity' etc.) and Truth-logic, which is built on the concept of 'truth' with its presupposition of objective fields of reference.

This special logic of logic, or metalogic, derives theorems which are valid *de dicto*. It is about properties and relationships of the meanings of linguistic entities. And as long as there are individuals who assign clear meanings to the words and expressions that they use, there will be cases in which it is true, or false, for all who accept those meaning assignments, that certain expressions are referentially synonymous with, or logically contain, certain other expressions, and are inconsistent or not with still other expressions. It will also be true or false that the meanings of certain expressions are both consistent-with and logically-contained-in the meanings of other expressions—i.e., it will be *true* that some inferences are logically *valid* along with the related C-conditionals and that other inferences and conditionals are not logically valid.

As an example, consider what DR1-12, "[If P Syn Q, then Q Cont P]" is intended to convey. The implicit meaning of DR1-12 is expressed more explicitly in the symbolism of A-logic as "$\models$Valid[T(P Syn Q) $\Rightarrow$ T(Q Cont P)]". This has not been our notation up to this point because "[If P Syn Q, then Q Cont P]" is simpler and the intelligent reader understands that it is offered as a valid conditional, such that the consequent would be true, if and when the antecedent is true. But making its meaning fully explicit, "$\models$Valid[T(P Syn Q) $\Rightarrow$ T(Q Cont P)]" means

1) according to the system of A-logic the following statements are valid:
(signified by '$\models$Valid')
2) when any expressions are put for 'P' and 'Q' in '(If P Syn Q, then Q Cont P)'
(signified by brackets '[' and ']' around the latter)
3) if the result of the substitution in 'P Syn Q' is a **true** statement, <u>then</u>
4) the result of like substitution in 'P Cont Q' is a **true** statement.

And when specific wffs or expressions are substituted for 'P' and 'Q' we can prove, by definitions of 'Syn' and 'Cont' and the axioms and rules of inference which have been given, whether the result of the substitution is true or false. Thus if we put '[(P&(Q&(PvQ)))' for 'P' and '(P&Q)]' for 'Q' we get, as substitution instance of DR1-12, [If T((P&(Q&(PvQ))) Syn (P&Q)) then T((P&(Q&(PvQ))) Cont (P&Q))]

and since we have proven

T1-18 T((P&(Q&(PvQ))) Syn (P&Q))

it follows by Modus Ponens, that

$\models$T((P&(Q&(PvQ))) Cont (P&Q))

But if we put '((P&(Q&(PvQ))) for 'P' and '(R&Q)' for 'Q' the antededent is not true and substiution instance of DR1-12 is neither true nor false. No substitution instance of DR1-12 can be false, though many substitution instances are not-true because the antecedent is not true.

The statement "$\models$Valid[T(P Syn Q) $\Rightarrow$ T(Q Cont P)]", and all similar formulations of principles of Inference, is not the same as asserting the principle to be a logical truth. I.e., it does not mean the same as "T[T(P Syn Q) $\Rightarrow$ T(Q Cont P)]" or similar expressions for other principles of inference. According to Ax.8-01 for the latter to be logically *true*, every substitution instance of [T(P Syn Q) & T(Q Cont P)] would have to be true, and of course this is not the case at all. To be *valid* in A-logic, all that is required of Ax.8-01 is that there be ***some*** instances such that it is true that what is substituted for P is Syn to what is substituted for Q, and that the result of making the same substitutions in 'P Cont Q' is also true. The meaning we have given to 'Syn' and 'Cont' insures that if the substitutions are right, this metatheorme can not be false.

Proofs that it is true that a certain conditional expressing a principle of inference is valid (or not), are no different in kind than proofs of validity of in *de re* reasoning. Unlike M-logic, which can not apply directly to its own semantics, Analytic Truth-logic applies to its own principles without getting into inconsistency.[14] The analytic truth-logic applies to the logic of mathematics, the logic of physics, the logic of common sense and to A-logic itself and its semantics.

The many informal proofs and derivations of derived rules which have been presented along the way, can be put into the formal language of analytic truth-logic with C-conditionals and T-operators. The principles that we have used, like SIMP, MP, Hyp-SYll, and Conditional Proof, have only been fully established in this latest chapter. In earlier chapters they were assumed. Now they are able to be proven. We have come full circle.

There is a mistaken idea among some philosophers that one can not use a principle until it is proven, just as one must build the foundations before one can erect a physical building. But the physical metaphor is a bad one. We know the principle of mathematical induction is sound, and we use it in proofs for purely formal quantification theory even though the full analytic explication the principle of mathematical induction must come much later after we have developed a logic of mathematics. The fact that its full explanation is not derivable from purely formal logic (which does not contain essential mathematical concepts), does not entail that it can not be used to prove theorems of purely formal logic.

Thus it would be possible to list all the principles we have called Rules and Derived Rules of Inference, together with the definitions for each of the predicates we have used to represent logical properties and relations; and the principles could be derived from the Syn-theorems which come out of the definitions we have given. Proofs would proceed in the same way we have derived principles of basic

14. The difficulties of self-referentiality in M-logic are analyzed and the reason these difficulties do not occur in A-logic is explained in Section 9.11.

A-logic and of analytic truth-logic. The rough outlines have been suggested.[15] But we leave this task to another time.

Conditionals like Principles of Inference which are only valid *de dicto* can not be used directly in *de re* reasoning. But some of their substitution instances can. The instances in which the wffs substituted are purely formal—containing only syncategorematic locutions—can be used for *de re* deductions. Instances which presuppose, or are based on presuppositions about purely linguistic properties and relations can not.

8.2322 Valid Conditionals from A-implications in Chapters 7 and 8

The property of being an A-implication depends on presuppositions about extra-logical operators; in the case of truth-logic, about T-operators. All A-implications below are part of truth-logic. The concept of being an A-implication applies in truth-logic; but it does not apply in pure formal logic and it does not apply in all other branches of logic. Since no C-conditionals were introduced in Chapter 7, implication was presented there only as a logical relation between expressions with T-operators. The purely formal conditionals of Chapter 6, by themselves, can not express an implication.

With both T-wffs and C-conditionals in this chapter the relation between A-implication and valid conditionals can be laid out precisely. The general Rule of Inference is: If P A-implies Q and (P&Q) are capable of being true together, then [P $\Rightarrow$ Q] is a valid implicative conditional. In short, if [P Impl Q] and (P&Q) is satisfiable then Valid_I [P $\Rightarrow$ Q]. This is the principle, DR8-6g:

DR8-6g [If (P Impl Q) and not-Inc(P&Q) then Valid_I (P $\Rightarrow$ Q)]

<u>Proof</u>:	1) [If (P Impl Q) and not-Inc (P&Q) then Valid_I (P, **.:**Q)]	[DR7-6g]
	2) If Valid (P, **.:**Q) then Valid (P $\Rightarrow$ Q)]	[DR6-6f, U-SUB]
	3) [If (P Impl Q) and not-Inc (P&Q) then Valid_I (P $\Rightarrow$ Q)]	[1),2),HypSYLL]

Thus putting '$\sim$TP' for 'P' and '$\sim$T(P&Q)' for 'Q' in DR8-6g we derive ***Valid****$_I$*[$\sim$TP $\Rightarrow$ $\sim$T(P&Q)] from Ti7-80. [$\sim$TP Impl $\sim$T(P&Q)] and not-Inc ($\sim$TP **&** $\sim$T(P&Q)).

The validity of implicational conditionals is a function of the *language* of negation and conjunction *and of 'truth'* which is a property of some linguistic entities. In order for the antecedent to be fulfilled, the substitutions for 'P' and 'Q' must be T-wffs.The principle DR8-6g can not be applied to theorems of Chapter 1 to 6 since A-implication occurs only among T-wffs, and truth-operators were not introduced until Chapter 7.

The rule DR8-6g allows the derivation of a valid conditional [P $\Rightarrow$Q] directly from any Impl-theorem of this chapter or Chapter 7 which satisfies the consistency requirement for validity. The P and Q must be T-wffs, for Implication in truth-logic depends on the presupposition that every wff is a T-wff. It is possible of course for P to imply Q when (P & Q) is inconsistent. For example, (T(P&$\sim$P) A-implies T((P&$\sim$P)vQ). But (T(P&$\sim$P) $\Rightarrow$T((P&$\sim$P)vQ) is not valid because (T(P&$\sim$P) & T((P&$\sim$P)vQ)) is inconsistent.

The following Validity Theorems are established immediately by DR8-6g from the Impl-theorems presented in the truth-logic of Chapter 7 and this chapter. Most of them are discussed later as the basis of rules for the trivalent truth-tables of A-logic.

15. The Appendices include for each chapter the definitions we have given for the logical predicates, and lists of principles of inference that have been presented.

Ti8-780. Valid$_I$ [~TP ⇒ ~T(P & Q)]		[Ti7-80,DR8-6g]
Ti8-780a. Valid$_I$ [~TP ⇒ ~T(P⇒Q)]		[Ti8-780,Ax.8-01,SynSUB]
Ti8-781. Valid$_I$ [FP ⇒ F(P & Q)]		[Ti7-81,DR8-6g]
Ti8-782. Valid$_I$ [~FP ⇒ ~F(P v Q)]		[Ti7-82,DR8-6g]
Ti8-783. Valid$_I$ [TQ ⇒ T(P v Q)]		[Ti7-83,DR8-6g]
Ti8-784. Valid$_I$ [TP ⇒ (TP v FP)]	Syn Valid$_I$ [TP ⇒F0P]	[Ti7-84,DR8-6g]
Ti8-785. Valid$_I$ [FP ⇒ (TP v FP)]	Syn Valid$_I$ [TP ⇒F0P]	[Ti7-85,DR8-6g]

<u>From 7.4233 - Principles of the truth-tables</u>

Ti8-7&R3. Valid$_I$ [(FP & 0Q) ⇒ F(P & Q)]	(For &-Row 3)	[Ti7-&R3,DR8-6g]
Ti8-7&R6. Valid$_I$ [(FP & TQ) ⇒ F(P & Q)]	(For &-Row 6)	[Ti7-&R6,DR8-6g]
Ti8-7&R7. Valid$_I$ [(0P & FQ) ⇒ F(P & Q)]	(For ⊃-Row 7)	[Ti7-&R7,DR8-6g]
Ti8-7&R8. Valid$_I$ [(TP & FQ) ⇒ F(P & Q)]	(For &-Row 8)	[Ti7-&R8,DR8-6g]
Ti8-7vR2. Valid$_I$ [(TP & 0Q) ⇒ T(P v Q)]	(For v-Row 2)	[Ti7-vR2,DR8-6g]
Ti8-7⊃ R3. Valid$_I$ [(FP & 0Q) ⇒ T(P ⊃ Q)]	(For ⊃-Row 3)	[Ti7⊃ R3,DR8-6g]
Ti8-7vR4. Valid$_I$ [(0P & TQ) ⇒ T(P v Q)]	(For v-Row 4)	[Ti7-vR4,DR8-6g]
Ti8-7⊃ R4. Valid$_I$ [(0P & TQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 4)	[Ti7⊃ R4,DR8-6g]
Ti8-7vR6. Valid$_I$ [(FP & TQ) ⇒ T(P v Q)]	(For v-Row 6)	[Ti7-vR6,DR8-6g]
Ti8-7⊃ R5. Valid$_I$ [(TP & TQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 5)	[Ti7⊃ R5,DR8-6g]
Ti8-7vR8. Valid$_I$ [(TP & FQ) ⇒ T(P v Q)]	(For v-Row 8)	[Ti7-vR8,DR8-6g]
Ti8-7⊃ R9. Valid$_I$ [(FP & FQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 9)	[Ti7⊃ R8,DR8-6g]
Ti8-7&R4. Valid$_I$ [(0P & TQ) ⇒ 0(P & Q)]	(For &-Row 4)	[Ti7-&R4,DR8-6g]
Ti8-7&R2. Valid$_I$ [(TP & 0Q) ⇒ 0(P & Q)]	(For &-Row 2)	[Ti7&R2,DR8-6g]
Ti8-7vR7. Valid$_I$ [(0P & FQ) ⇒ 0(P v Q)]	(For v-Row 7)	[Ti7-vR7,DR8-6g]
Ti8-7⊃ R7. Valid$_I$ [(0P & FQ) ⇒ 0(P ⊃ Q)]	(For ⊃-Row 7)	[Ti7⊃ R7,DR8-6g]
Ti8-7vR3. Valid$_I$ [(FP & 0Q) ⇒ 0(P v Q)]	(For v-Row 3)	[Ti7-vR3,DR8-6g]
Ti8-7⊃ R2. Valid$_I$ [(TP & 0Q) ⇒ 0(P ⊃ Q)]	(For ⊃-Row 2)	[Ti7⊃ R2,DR8-6g]

<u>From 7.4234 - A-implication in Q-Theory</u>

Ti8-786. Valid$_I$ [(∀x)TQx ⇒ (∀x)(0Px ⊃ TQx)]	[Ti7-86,DR8-6g]
Ti8-787. Valid$_I$ [(∀x)TQx ⇒(∀x)(TPx ⊃ TQx)]	[Ti7-87,DR8-6g]
Ti8-788. Valid$_I$ [(∀x)TQx ⇒(∀x)(FPx ⊃ TQx)]	[Ti7-88,DR8-6g]
Ti8-789. Valid$_I$ [(∀x)~TPx ⇒(∀x)(TPx ⊃ FQx)]	[Ti7-89,DR8-6g]
Ti8-790. Valid$_I$ [(∀x)FPx ⇒(∀x)(TPx ⊃ FQx)]	[Ti7-90,DR8-6g]
Ti8-791. Valid$_I$ [(∀x)TPx ⇒(∀x)(~TPx ⊃ TQx)]	[Ti7-91,R8-6g]
Ti8-792. Valid$_I$ [T(Pai) ⇒T(∃x)Px] … (∃x)TPx]	[Ti7-92,DR8-6g]
Ti8-793. Valid$_I$ [F(Pai) ⇒F(∀x)Px] … (∃x)FPx]	[Ti7-93,DR8-6g]
Ti8-794. Valid$_I$ [~T(Pai) ⇒~T(∀x)Px]	[Ti7-94,DR8-6g]
Ti8-795. Valid$_I$ [~F(Pai) ⇒~F(∃x)Px]	[Ti7-95,DR8-6g]

<u>From Section 8.2112 - A-implications</u>

Ti8-840. Valid$_I$ [~TP ⇒0(P⇒ Q)]	[Ti8-40,DR8-6g]
Ti8-841. Valid$_I$ [FP ⇒0(P⇒ Q)]	[Ti8-41,DR8-6g]
Ti8-842. Valid$_I$ [0P ⇒0(P⇒ Q)]	[Ti8-42,DR8-6g]
Ti8-843. Valid$_I$ [0Q ⇒0(P⇒ Q)]	[Ti8-43,DR8-6g]
Ti8-844. Valid$_I$ [(TP ⇒TQ) & ~TQ) ⇒~TP]	[Ti8-44,DR8-6g]

A-Implications expressing principle of the truth-table for " ⇒ ."

Ti8-8 ⇒ r2. Valid$_I$ [(TP & 0Q) ⇒ 0(P ⇒ Q)] [Ti8 ⇒ r2,DR8-6g]
Ti8-8 ⇒ r4. Valid$_I$ [(0P & TQ) ⇒ 0(P ⇒ Q)] [Ti8 ⇒ r4,DR8-6g]
Ti8-8 ⇒ r5. Valid$_I$ [(FP & TQ) ⇒ 0(P ⇒ Q)] [Ti8 ⇒ r6,DR8-6g]
Ti8-8 ⇒ r7. Valid$_I$ [(0P & FQ) ⇒ 0(P ⇒ Q)] [Ti8 ⇒ r7,DR8-6g]
Ti8-8 ⇒ r9. Valid$_I$ [(FP & FQ) ⇒ 0(P ⇒ Q)] [Ti8 ⇒ r9,DR8-6g]

8.2323 Remarks About A-implications and De Re Reasoning

The Valid conditionals in the Section 8.231 were all derived from Entailments, in which the consequent is contained in the antecedent. The A-implications of Section 8.2321, if consistent, can also yield valid conditionals. But for some implications the conjunction of antecedent and consequent is unsatisfiable. If P implies Q is satifiable *and* (P and Q) is satisfiable, then [P ⇒ Q] is valid.

The paradigmatic examples of implication, from Section 7.4231,

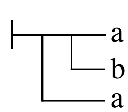
Ti7-80. [∼TP Impl ∼T(P & Q)] "(P is not true) implies ((P&Q) is not true)"
Ti7-83. [TQ Impl T(P v Q)] "(P is true) implies ((PvQ) is true)"

yield, by DR8-6g, the Valid C-conditionals,

Ti8-780. Valid$_I$ [∼TP ⇒ ∼T(P & Q)] [Ti7-80,DR8-6g]
"[If it is not true that P then it is not true that P and Q] is a valid implication."
Ti8-783. Valid$_I$ [TQ ⇒ T(P v Q)] [Ti7-83,DR8-6g]
"[If P is true, then (P or Q) is true] is a valid implication."

The latter is called "Addition" and appears as an axiom with a TF-conditional in most axiomatizations of M-logic; it has many guises, but usually it is expressed as ⊢ [Q ⊃ (P ⊃ Q)], or ⊢ [Q ⊃ (P v Q)].[16] Many of the problems which have confronted proponents when they wanted to apply M-logic to *de re* reasoning in ordinary discourse or in the empirical sciences, stem from failing to distinguish between A-implication which does not fit *de re* requirements, and entailments which can mirror what we conceive to be the properties of objective realities.

In all implications it is difficult to deny that the consequent follows from the antecedent. We would say: If P were true in fact, then that fact would make [P v Q] true. The essential ingredients are: (i) the occurrence of P as both the antecedent and as a disjunct in the consequent and (ii) the prefix 'It is true that…' before both 'P' and '(Pv…)'. We would not say the conditional was valid if 'v' is replaced by '&', or if 'T' is replaced by 'F', or by 'O' (for "It ought to be that…"). But what we put in the blank space of [P v __] does not matter when the prefix is "It is true that…". The blank can be filled by any wff

16. E.g. The axioms of Frege 

Begriffsschrift,1879

Russell & Whitehead	*1.2. q .⊃ . p v q	Principia Mathematica, 1910
Tarski	CpCqp	1921
Hilbert	b) X ⊃ X v Y	1922
Church	✝202. p ⊃ .q ⊃ p	Intro. to Math Logic,1956
Thomason	AS1. A ⊃ .B ⊃ A	Symbolic Logic, 1970

Rosser's axioms do not include Addition, but of course Addition is derivable as a theorem.

or its denial. The validity of the whole is not affected in any way by the meaning, or the truth, falsity, inconsistency, or tautologousness of what is put in the blank. Put differently, it is not required that either the antecedent, or the concept that the antecedent is true, provide a ground for what is put for 'Q'.

The class of conditional wffs I have isolated for treatment are those satisfiable wffs which are reducible to the over-all form 'TP ⇒T(PvQ)' in which the consequent is not contained in the antecedent. Some substitution instances of 'TP ⇒T(P v Q)' do not fall in this category; e.g., 'TP ⇒T(P v P)' and 'T(P v Q) ⇒T((P v Q) v Q)'. In such cases what is put for Q is something logically contained in the antecedent. All of these substitution instances are cases of entailment as covered in the Section 8.231, and are not implications. Thus the class of valid implicational conditionals I wish to define are those in which the consequent follows from the antecedent, but the consequent is not contained in the antecedent.

8.23231 A-implications are Only Valid *De Dicto*

What do we mean when we say A-entailments can be valid *de re* valid, but A-implication can only be valid *de dicto*?

In truth-logic A-entailment is valid *de re* because, the f*acts* which make a conjunctive statement true, must contain the *facts* which make any of the conjuncts in that statement true. Thus whatever facts (if any) makes the antecedent true must include facts that make the consequent true. In '(T(P&Q) ⇒ TP)' the affixing of T to P in the consequent is warranted because the facts needed to make (P&Q) true, must include facts which would make P true.

But in A-implication, the facts in the field of reference which warrant affixing 'It is true that…' to P in the antecedent, TP, do not include any facts which warrant affixing 'It is true that…' to the consequent Q in '(TPvTQ)', or to ~Q in '(TPvT~Q)', or to ~TQ in '(TPvT~TQ)' or ~FQ in '(TPvT~FQ)'. All of these disjunctions follow from TP; but only because of the meaning given to 'v' and its connection with the meaning of 'T'. It is a linguistic convention: a disjunction is true if one of its disjuncts is true.

Thus we say that A-implication is only valid *de dicto*, while A-entailment is both valid *de re* and valid *de dicto*. The latter, but not the former, keeps us on track when we are reasoning *de re*. Used in reasoning about matters of fact, it confines the moves from statements about facts to other statements about facts never going beyond what truth-assertions based on facts would allow. A-implication allows the introduction of irrelevant alternatives asserting truths or falsehood of utterly unrelated sentences because of conventions of meaning.

Let us call any component in the consequent which is not contained in the antecedent of an implication, an '**irrelevant component**'; it doesn't matter what they are or what is substituted for them. Unrestricted U-SUB is permitted on irrelevant components without affecting the relationship of A-implication provided what is substituted is not contained in the antecedent.

It remains true that A-implications can be valid, e.g., 'TP ⇒ (TP v TQ)'. They can never be false and they are capable of having instances which are true, and the consequent does follow from the antecedent with the presupposed Law of Trivalence. But A-implication is valid *de dicto* only. It is true only because of linguistic meanings of 'v' and 'T'. While A-implications are useful in reasoning about relations of meanings, it can create confusion or defeat our purposes, when used in reasoning about matters of fact. Among others, two such confusions occur in formulations of Gettier's problem and in Goodman's "new riddle of induction"; the latter is discussed in Section 10.2.

8.23232 Inappropriate uses of A-implication

If P A-implies Q, then one can confidently pass from an assertion that P is true to an assertion that Q is true; and it will never be the case that P is in fact true, and Q is false.

But since P does not contain Q if P A-implies Q, components of the implicans (conclusion, consequent) may be utterly irrelevant; i.e., utterly unconnected to the implicandum (premisses, antecedent). There are endless possibilities for meaningless chains of irrelevant valid implications:

$$
\begin{aligned}
& \text{Valid}_I\ [TP \Rightarrow T(P \vee Q)] \\
.:\ & \text{Valid}_I\ [T(P \vee Q) \Rightarrow T(P \vee Q \vee R)] \\
.:\ & \text{Valid}_I\ [T(P \vee Q \vee R) \Rightarrow T(P \vee Q \vee R \vee S)] \\
& \quad \ldots. \\
.:\ & \text{Valid}_I\ [TP1 \Rightarrow T(P_1 \vee P_2 \vee P_3 \vee \ldots \vee P_n)]
\end{aligned}
$$

The validity of these A-implications is preserved even if every P_i $(i > 1)$ in the last formula is replaced by an irrelevant wff or statement, provided only that (P1 & (P1vP2vP3v…vPn)) is free of inconsistency.[17] But in all cases, the consequent (conclusion) does not follow *de re* from the antecedent (premiss); it only follows *de dicto*. An inference by A-implication conveys new information about language, but except in special cases to be discussed, the consequent conveys no new factual information. Indeed, in these examples information about facts is diminished, for "TP or TQ" provides less information than TP or TQ by themselves. From (TP vTQ) one can not know whether TP is true or false, or whether TQ is true or false.

Rules of inference in logic should facilitate sound logical reasoning. In cases like this the Rule of Addition can introduces irrelevancies, tangents and obstacles to efficient deductive reasoning. If Addition is used as a logical rule for *de re* inferences, every law-like statement that is sound yields a proliferation of other lawlike statements with unlimited alternatives in the consequent. Systems of classification in science and common sense (designed to mirror in language similarities and differences found in nature) are reduced to chaos if Addition is applied as a *de re* rule of inference. Persuasive arguments and reasoning, even in philosophy, have been based on a fallacious use of Addition as a *de re* principle of inference.[18]

8.23233 Appropriate uses of A-implications in Reasoning

A-implication, being a *de dicto* relation, is appropriately used when determining the applicability of predicates which are defined as having alternative conditions of application, i.e., are said to apply if any one of several alternative conditions obtains.

In nature, often objects which bear certain abstract properties can be classified in two or more subclasses. Such is the tree of Porphyry and the general method of classification in botany, zoology, chemistry, as well as in man-made accounting systems, systems of naming geographical regions and sub-regions, etc. Here it is often the case that if P applies, either Q or R will apply. "If A is a human, then either A is male or A is female", has a form similar to an A-implication, $(TP \Rightarrow (TQ \vee TR))$, as does "If A lives in Philadelphia, then A either lives East of Broad Street or West of Broad Street". Assuming that the humans could be differentiated from other beings and adequately defined without reference to sexual features, this is not an analytic truth but one based on observations of fact. Thus the

---

17. e.g., "If it is true that a_1 is a living organism, then either it is true that a_1 is an animal or it is true that a_1 is vegetable" A-implies "If a_1 is a living organism, then either it is true that a_1 is an stone,or it is true that a_1 is a book, or it is true that a_1 is an animal, or it is true that a_1 is a number, or it is true that a_1 is vegetable, or it is true that a_1 is an electric light bulb, or … etc"

18. See Chapter 10, Sections 10.22 and 10.23.

inference from A is human, to A is male or female, is based on a classificatory system which reflects distinctions found in nature among entities of the same kind.The disjunctionn describe *de re* alternatives. We shall have more to say on this in the next chapter, which deals with conditionals and findings of fact.

The proofs for truth-table conditionals in the following section proceed on the assumption of trivalence; that every sentence in truth-logic will be either true or false or neither. Components in the consequent of the implicative principles which do not occur in the antecedent are chosen because of the task that has been undertaken. We want to determine what the truth value of given expression with a binary connective would be, given each of the nine possible assignments of truth-values to its components. The component in the consequent which is not contained in the antecedent is not just any thing we happen to place there. It it an essential part of the task we are trying to accomplish. Thus appropriate uses of A-implication take place in a context of inquiry in which any components of consequents which are not contained in the antecedent are put there by clearly specifiable demands in the context of inquiry. They are not components that are replaceable by *any* expression without affecting desired properties of the expression in which they occur.

8.232331 Implication, and Proofs of Truth-Table Rules for Conditionals

We have said each row of a truth-table represents a short-hand way of asserting an analytically valid conditional statement. We now provide the appropriate C-conditionals for each row in the various truth-tables and prove them valid.Five or six of the nine rows in each truth-table are based on A-implications.

The basis for proofs of the truth-table conditionals has already be laid in the Syn-, Cont- and Impl-theorems of this chapter and Chapter 7. It remains only to prove that the antecedent and consequents of those theorems are jointly satisfiable. An unquantified wff can be proven satisfiable (capable of having true instances) by reduction to its MODNF. For example, "not-Inc((TP & 0Q) & 0(P & Q))", can be shown to be satisfiable as follows:

1) ((TP & 0Q) & 0(P & Q)) Syn (TP & (∼TQ&∼FQ) & (∼T(P&Q) & ∼F(P&Q))) [Df '0']
2) ((") Syn (TP & ∼TQ & ∼FQ & ∼(TP&TQ) & ∼T(∼P v∼Q)) [Ax.7-3]
3) ((") Syn (TP & ∼TQ & ∼FQ & ∼(TP&TQ) & ∼(T∼P vT∼Q)) [Ax.7-4]
4) ((") Syn (TP & ∼TQ & ∼FQ & ∼(TP&TQ) & ∼(FP v FQ)) [Df 'F']
5) ((") Syn (TP & ∼TQ & ∼FQ & ∼(TP&TQ) & ∼FP & ∼FQ)) [DeM]
6) ((") Syn (TP & ∼TQ & ∼FQ & (∼TPv∼TQ) & ∼FP & ∼FQ)) [DeM]
7) ((") Syn ((TP & ∼FP & ∼TQ & ∼FQ) & (∼TP v∼TQ)) [&-ORG]
8) ((") Syn
MOCNF((TP & ∼FP & ∼TQ & ∼FQ & **∼TP**) v **(**TP & ∼FP & ∼TQ & ∼FQ **& ∼TQ**))
|——Not-Inc——|
Hence, |——Satisfiable——|

However, satisfiability can be established in an easier way, simply by presenting an assignment of values which would make the conjunction of antecedent and consequent come out true. For example, by

((TP & 0Q) & 0(P & Q))
tt t t0 **T** t t 0 0

or, synonymously by ((TP & (∼TQ & ∼T∼Q)) & (∼T(P & Q) & ∼T∼(P & Q))).
((t t t (t f 0 t t f 0 0)) **T**((tf t 0 0) t (t f 0 t 0 0))

Thus by adding proofs of satisfiability to proven Syn-, Cont- or Impl-theorems we can derive the validity of each truth-table conditional in the truth-tables of '∼', 'T', '&', 'v', '⊃' and '⇒': For example, the validity-proof for the first conditional in the truth table of '(∼P)' is as follows:

T8-752. Valid[0P ⇒ 0(~ P)] “If P is 0, then [~ P] is 0”

Proof: 1) 0 ~ P Syn 0P [T7-52]
2) Not-Inc[OP & 0 ~P] Syn Not-Inc[(~TP & ~T~P) & (~T ~P & ~T~~P))
t 0 T t 00 t f0 t t f 00 T t f 00 t t f 0 00 [inspection]
3) Valid[0P, .: 0 ~ P] [1),2),Df ‘Valid’]
4) Valid[0P ⇒ 0 ~ P] [3),VC\VI]

At step 2) truth-tables show one row with a ‘T’; so the conjunction of antecedent and consequent is satisfiable. The conditional for this rule of the 2nd row in the truth-table for &, is proved valid as follows:

Ti8-7&R2. Valid((TP & 0Q) ⇒ 0(P & Q)) “If P is T and Q is 0, then (P&Q)is 0”

Proof: 1) (TP & 0Q) Impl 0(P & Q) [Ti7-&R2]
2) Sat((TP & 0Q) & 0(P & Q))
t t t t 0 t t t 0 0 [Inspection]
3) Valid[(TP & 0Q) ⇒ 0(P & Q)] [1),2),DR8-6g]

Proofs of validity of the *inference schemata* for truth-tables for ‘ ~ ’, ‘T’, ‘&’, ‘v’ and ‘⊃’ were given in Chapter 7. These may be used below to prove that the satisfiability condition is met, and thus that the conditional for that row of a truth-table is valid. It remains to prove the validity of conditionals governing the truth-table for ‘ ⇒’. The reader may supply the truth-table tests indicated by “Inspection”.

T8 ⇒ r1. Valid[(0P & 0Q) ⇒ 0(P ⇒ Q)]

Proof: 1) (0P & 0Q) Cont 0(P ⇒ Q) [T8-31]
2) Sat(0P & 0Q) & 0(P ⇒ Q) [Inspection]
3) Valid[(0P & 0Q) ⇒ 0(P ⇒ Q)] [1),2),Df.’Valid’,VC\VI]

Ti8-8 ⇒ r2. Valid$_I$[(TP & 0Q) ⇒ 0(P ⇒Q)]

Proof: 1) (TP & 0Q) Cont 0Q [T1-37,U-SUB]
2) 0Q Impl 0(P ⇒ Q) [Ti8-42]
3) ((TP & 0Q) Impl 0(P ⇒ Q)) [1),2),CIISyll]
4) Sat((TP & 0Q) & 0(P ⇒ Q)) [Inspection]
5) Valid$_I$[(0P & TQ) ⇒ 0(P ⇒ Q)] [3),4),DR8-6g]

T8 ⇒ r3. Valid[(FP & 0Q) ⇒ 0(P ⇒ Q)]

Proof: 1) (FP & 0Q) Cont 0(P ⇒ Q) [T8-32]
2) Sat((FP & 0Q) & 0(P ⇒ Q)) [Inspection]
3) Valid[(FP & 0Q) ⇒ 0P ⇒ Q)] [1),2),Df.’Valid’,VC\VI]

Ti8 ⇒ r4. Valid$_I$[(0P & TQ) ⇒ 0(P ⇒ Q)]

Proof: 1) (0P & TQ) Cont 0P [T1-36,U-SUB]
2) 0P Impl 0(P ⇒ Q) [Ti8-42]
3) ((0P & TQ) Impl 0(P ⇒ Q)) [1),2),CIISyll]
4) Sat((0P & TQ) & 0(P ⇒ Q)) [Inspection]
5) Valid$_I$[(0P & TQ) ⇒ 0(P ⇒ Q)] [3),4),DR8-6g]

T8 ⇒ r5. Valid[(TP & TQ) ⇒ T(P ⇒ Q)]

Proof: 1) (TP & TQ) Cont T(P ⇒ Q) [T8-33]
2) Sat((TP & TQ) & T(P ⇒ Q)) [Inspection]
3) Valid[(TP & TQ) ⇒ T(P ⇒ Q)] [1),2),Df.’Valid’,VI/VC]

Ti8⇒r6. Valid$_I$[(FP & TQ) ⇒ 0(P ⇒ Q)]
Proof: 1) (FP & TQ) Cont FP [T1-36,U-SUB]
2) FP Impl 0(P ⇒ Q) [Ti8-41]
3) ((FP & TQ) Impl 0(P ⇒ Q)) [1),2),CIISyll]
4) Sat((FP & TQ) & 0(P ⇒ Q))
tf t tt t tf 0 t [Inspection]
5) Valid$_I$[(FP & TQ) ⇒ 0(P ⇒ Q)] [3),4),DR8-6g]

Ti8⇒r7. Valid$_I$[(0P & FQ) ⇒ 0(P ⇒ Q)]
Proof: 1) (0P & FQ) Cont 0P [T1-36,U-SUBb]
2) 0P Impl 0(P ⇒ Q) [Ti8-42]
3) (0P & FQ) Impl 0(P ⇒ Q)) [1),2),CIISyll]
4) Sat((0P & FQ) & 0(P ⇒ Q))
t0 t tf t t0 0 f [Inspection]
5) Valid$_I$[(0P & FQ) ⇒ 0(P ⇒ Q)] [3),4),DR8-6g]

T8⇒r8. Valid[(TP & FQ) ⇒ F(P ⇒ Q)]
Proof: 1) (TP & FQ) Syn F(P ⇒ Q) [T8-34]
2) Sat((TP & FQ) & F(P ⇒ Q))
tt t tf t t t f f [Inspection]
3) Valid[(TP & FQ) ⇒ F(P ⇒ Q)] [1),2),Df 'Valid⇒']

Ti8⇒r9. Valid[(FP & FQ) ⇒ 0(P ⇒ Q)]
Proof: 1) (FP & FQ) Cont FP [T1-36,U-SUB]
2) FP Impl 0(P ⇒ Q) [Ti8-41]
3) ((FP & FQ) Impl 0(P ⇒ Q)) [1),2),CIISyll]
4) Sat((FP & FQ) & 0(P ⇒ Q))
tf t tf t tf 0 t [Inspection]
5) Valid[(FP & FQ) ⇒ 0(P ⇒ Q)] [3),4),DR8-6g]

Given the joint satisfiability of antecedent and consequent (which the reader may check), the valid conditionals for the truth-table of '⇒' are based on the Syn-, Cont- or Impl-theorems,

T8-31. [(0P & 0Q) Cont 0(P ⇒Q)]
Ti8⇒r2. [(TP & 0Q) Impl 0(P ⇒Q)]
T8-32. [(FP & 0Q) Cont 0(P ⇒Q)]
Ti8⇒r4. [(0P & TQ) Impl 0(P ⇒Q)]
T8-33. [(TP & TQ) Syn T(P ⇒Q)]
Ti8⇒r6. [(FP & TQ) Impl 0(P ⇒Q)]
Ti8⇒r7. [(0P & FQ) Impl 0(P ⇒Q)]
T8-34. [(TP & FQ) Syn F(P ⇒Q)]
Ti8⇒r9. [(FP & FQ) Impl 0(P ⇒Q)]

Our task was to prove that each truth-table conditional is valid, either by virtue of containment or by implication. TABLES 8-1 and 8-2 show respectively 1) how the conditionals are on the relevant Syn-Cont- or Impl-theorems and the principles of DR6-6a, DR6-6d with U-SUBab, or DR8-6g, and 2) how the results correspond with the natural language interpretation of the theorems and of the meaning of the rows in the truth-table.

TABLE 8-1. DERIVATIONS by DR8-6a, DR8-6d and DR8-6g

T7-52. (0P Syn 0~P)	.:	r1 T8-752. Valid[0P ⇔ 0~P]	[DR8-6d]
T7-16. (TP Syn F~P)	.:	r2 T8-716. Valid[TP ⇔ F(~P)]	[DR8-6d]
T7-06. (FP Syn T~P)	.:	r3 T8-706. Valid[FP ⇔ T(~P)]	[DR8-6d]
T7-50. (0P Cont FTP)	.:	r1 T8-750. Valid[0P ⇒ FTP]	[DR8-6a]
T7-20. (TP Cont TTP)	.:	r2 T8-720. Valid[TP ⇒ TTP]	[DR8-6a]
T7-18. (FP Cont FTP)	.:	r3 T8-718. Valid[FT ⇒ FTP]	[DR8-6a]
T7-58. (0P & 0Q) Cont 0(P & Q)	.:	r1 T8-758. Valid[(0P & 0Q) ⇒ 0(P & Q)]	[DR8-6a]
Ti7-&R2. (TP & 0Q) Impl 0(P & Q)	.:	r2 Ti8-7&R2. Valid[(TP & 0Q) ⇒ 0(P & Q)]	[DR8-6g]
Ti7-&R3. (FP & 0Q) Impl F(P & Q)	.:	r3 Ti8-7&R3. Valid[(FP & 0Q) ⇒ F(P & Q)]	[DR8-6g]
Ti7-&R4. (0P & TQ) Impl 0(P & Q)	.:	r4 Ti8-7&R4. Valid[(0P & TQ) ⇒ 0(P & Q)]	[DR8-6g]
T7-03. (TP & TQ) Syn T(P & Q)	.:	r5 T8-703. Valid[(TP & TQ) ⇔ T(P & Q)]	[DR8-6d]
Ti7-&R6. (FP & TQ) Impl F(P & Q)	.:	r6 Ti8-7&R6. Valid[(FP & TQ) ⇒ F(P & Q)]	[DR8-6g]
Ti7-&R7. (0P & FQ) Impl F(P & Q)	.:	r7 Ti8-7&R7. Valid[(0P & FQ) ⇒ F(P & Q)]	[DR8-6g]
Ti7-&R8. (TP & FQ) Impl F(P & Q)	.:	r8 Ti8-7&R8. Valid[(TP & FQ) ⇒ F(P & Q)]	[DR8-6g]
T7-37. (FP & FQ) Cont F(P & Q)	.:	r9 T8-737. Valid[(FP & FQ) ⇒ F(P & Q)]	[DR8-6a]
T7-59. (0P & 0Q) Cont 0(P v Q)	.:	r1 T8-759. Valid[(0P & 0Q) ⇒ 0(P v Q)]	[DR8-6a]
Ti7-vR2. (TP & 0Q) Impl T(P v Q)	.:	r2 Ti8-7vR2. Valid[(TP & 0Q) ⇒ T(P v Q)]	[DR8-6g]
Ti7-vR3. (FP & 0Q) Impl 0(P v Q)	.:	r3 Ti8-7vR3. Valid[(FP & 0Q) ⇒ 0(P v Q)]	[DR8-6g]
Ti7-vR4. (0P & TQ) Impl T(P v Q)	.:	r4 Ti8-7vR4. Valid[(0P & TQ) ⇒ T(P v Q)]	[DR8-6g]
T7-38. (TP & TQ) Cont T(P v Q)	.:	r5 T8-738. Valid[(TP & TQ) ⇒ T(P v Q)]	[DR8-6a]
Ti7-vR6. (FP & TQ) Impl T(P v Q)	.:	r6 Ti8-7vR6. Valid[(FP & TQ) ⇒ T(P v Q)]	[DR8-6g]
Ti7-vR7. (0P & FQ) Impl 0(P v Q)	.:	r7 Ti8-7vR7. Valid[(0P & FQ) ⇒ 0(P v Q)]	[DR8-6g]
Ti7-vR8. (TP & FQ) Impl T(P v Q)	.:	r8 Ti8-7vR8. Valid[(TP & FQ) ⇒ T(P v Q)]	[DR8-6g]
T7-23. (FP & FQ) Syn F(P v Q)	.:	r9 T8-723. Valid[(FP & FQ) ⇔ F(P v Q)]	[DR8-6d]
T7-60. (0P & 0Q) Cont 0(P ⊃ Q)	.:	r1 T8-760. Valid[(0P & 0Q) ⇒ 0(P ⊃ Q)]	[DR8-6a]
Ti7⊃ R2. (TP & 0Q) Impl 0(P ⊃ Q)	.:	r2 Ti8-7⊃ R2. Valid[(TP & 0Q) ⇒ 0(P ⊃ Q)]	[DR8-6g]
Ti7⊃ R3. (FP & 0Q) Impl T(P ⊃ Q)	.:	r3 Ti8-7⊃ R3. Valid[(FP & 0Q) ⇒ T(P ⊃ Q)]	[DR8-6g]
Ti7⊃ R4. (0P & TQ) Impl T(P ⊃ Q)	.:	r4 Ti8-7⊃ R4. Valid[(0P & TQ) ⇒ T(P ⊃ Q)]	[DR8-6g]
Ti7⊃ R5. (TP & TQ) Impl T(P ⊃ Q)	.:	r5 Ti8-7⊃ R5. Valid[(TP & TQ) ⇒ T(P ⊃ Q)]	[DR8-6g]
T7-39. (FP & TQ) Cont T(P ⊃ Q)	.:	r6 T8-739. Valid[(FP & TQ) ⇒ T(P ⊃ Q)]	[DR8-6a]
Ti7⊃ R7. (0P & FQ) Impl 0(P ⊃ Q)	.:	r7 Ti8-7⊃ R7. Valid[(0P & FQ) ⇒ 0(P ⊃ Q)]	[DR8-6g]
T7-40. (TP & FQ) Syn F(P ⊃ Q)	.:	r8 T8-740. Valid[(TP & FQ) ⇔ F(P ⊃ Q)]	[DR8-6d]
Ti7⊃ R9. (FP & FQ) Impl T(P ⊃Q)	.:	r9 Ti8-7⊃ R9. Valid[(FP & FQ) ⇒ T(P ⊃ Q)]	[DR8-6g]
T8-31. (0P & 0Q) Cont 0(P ⇒Q)	.:	r1 T8-831. Valid[(0P & 0Q) ⇒ 0(P ⇒ Q)]	[DR8-6a]
Ti8 ⇒ r2. (TP & 0Q) Impl 0(P ⇒Q)	.:	r2 Ti8-8 ⇒ r2. Valid[(TP & 0Q) ⇒ 0(P ⇒Q)]	[DR8-6g]
T8-32. (FP & 0Q) Cont 0(P ⇒Q)	.:	r3 T8-832. Valid[(FP & 0Q) ⇒ 0(P ⇒Q)]	[DR8-6a]
Ti8 ⇒ r4. (0P & TQ) Impl 0(P ⇒Q)	.:	r4 Ti8-8 ⇒ r4. Valid[(0P & TQ) ⇒ 0(P ⇒Q)]	[DR8-6g]
T8-33. (TP & TQ) Syn T(P ⇒Q)	.:	r5 T8-833. Valid[(TP & TQ) ⇔ T(P ⇒Q)]	[DR8-6d]
Ti8 ⇒ r6. (FP & TQ) Impl 0(P ⇒Q)	.:	r6 Ti8-8 ⇒ r6. Valid[(FP & TQ) ⇒ 0(P ⇒Q)]	[DR8-6g]
Ti8 ⇒ r7. (0P & FQ) Impl 0(P ⇒Q)	.:	r7 Ti8-8 ⇒ r7. Valid[(0P & FQ) ⇒ 0(P ⇒Q)]	[DR8-6g]
T8-34. (TP & FQ) Syn F(P ⇒Q)	.:	r8 T8-834. Valid[(TP & FQ) ⇔ F(P ⇒Q)]	[DR8-6d]
Ti8 ⇒ r9. (FP & FQ) Impl 0(P ⇒Q)	.:	r9 Ti8-8 ⇒ r9. Valid[(FP & FQ) ⇒ 0(P ⇒Q)]	[DR8-6g]

TABLE 8-2 - TRUTH-TABLES

Row Truth-table

Row	P	~P	(The rows may be read as saying:)	I.e., the theorem:
Row 1	0	0	"If P is 0 then [~ P] is 0"	[T8-752] Valid[0P ⇒ 0 ~ P]
Row 2	T	F	"If P is T then [~ P] is F"	[T8-716] Valid[TP ⇒ F ~ P]
Row 3	F	T	"If P is F then [~ P] is T"	[T8-706] Valid[FP ⇒ T ~ P]

Row	P	TP	(The rows may be read as saying:)	I.e., the theorem:
Row 1	0	F	"If P is 0 then [TP] is F"	[T8-750] Valid[0P ⇒ FTP]
Row 2	T	T	"If P is T then [TP] is T"	[T8-720] Valid[TP ⇒ TTP]
Row 3	F	F	"If P is F then [TP] is F"	[T8-718] Valid[FT ⇒ FTP]

Row	P	Q	(P&Q)	(The rows may be read as saying:)	I.e., the theorem:
Row 1)	0	0	0	"If P is 0 and Q is 0, then [P&Q] is 0"	[T8-758]
Row 2)	T	0	0	"If P is T and Q is 0, then [P&Q] is 0"	[Ti8-7&R2]
Row 3)	F	0	F	"If P is F and Q is 0, then [P&Q] is F"	[Ti8-7&R3]
Row 4)	0	T	0	"If P is 0 and Q is T, then [P&Q] is 0"	[Ti8-7&R4]
Row 5)	T	T	T	"If P is T and Q is T, then [P&Q] is T"	[T8-703]
Row 6)	F	T	F	"If P is F and Q is T, then [P&Q] is F"	[Ti8-7&R6]
Row 7)	0	F	F	"If P is 0 and Q is F, then [P&Q] is F"	[Ti8-7&R7]
Row 8)	T	F	F	"If P is T and Q is F, then [P&Q] is F"	[Ti8-7&R8]
Row 9)	F	F	F	"If P is F and Q is F, then [P&Q] is F"	[T8-737]

Row	P	Q	(PvQ)	(The rows may be read as saying:)	I.e., the theorem:
Row 1)	0	0	0	"If P is 0 and Q is 0, then [PvQ] is 0"	[T8-759]
Row 2)	T	0	T	"If P is T and Q is 0, then [PvQ] is T"	[Ti8-7vR2]
Row 3)	F	0	0	"If P is F and Q is 0, then [PvQ] is 0"	[Ti8-7vR3]
Row 4)	0	T	T	"If P is 0 and Q is T, then [PvQ] is T"	[Ti8-7vR4]
Row 5)	T	T	T	"If P is T and Q is T, then [PvQ] is T"	[T8-738]
Row 6)	F	T	T	"If P is F and Q is T, then [PvQ] is T"	[Ti8-7vR6]
Row 7)	0	F	0	"If P is 0 and Q is F, then [PvQ] is 0"	[Ti8-7vR7]
Row 8)	T	F	T	"If P is T and Q is F, then [PvQ] is T"	[Ti8-7vR8]
Row 9)	F	F	F	"If P is F and Q is F, then [PvQ] is F"	[T8-723]

Row	P	Q	(P ⊃ Q)	(The rows may be read as saying:)	I.e., the theorem:
Row 1)	0	0	0	"If P is 0 and Q is 0, then [P ⊃ Q] is 0"	[T8-760]
Row 2)	T	0	0	"If P is T and Q is 0, then [P ⊃ Q] is 0"	[Ti8-7⊃ R2]
Row 3)	F	0	T	"If P is F and Q is 0, then [P ⊃ Q] is T"	[Ti8-7⊃ R3]
Row 4)	0	T	T	"If P is 0 and Q is T, then [P ⊃ Q] is T"	[Ti8-7⊃ R4]
Row 5)	T	T	T	"If P is T and Q is T, then [P ⊃ Q] is T"	[Ti8-7⊃ R5]
Row 6)	F	T	T	"If P is F and Q is T, then [P ⊃ Q] is T"	[T8-739]
Row 7)	0	F	0	"If P is 0 and Q is F, then [P ⊃ Q] is 0"	[Ti8-7⊃ R7]
Row 8)	T	F	F	"If P is T and Q is F, then [P ⊃ Q] is F"	[T8-740]
Row 9)	F	F	T	"If P is F and Q is F, then [P ⊃ Q] is T"	[Ti8-7⊃ R9]

Row	P	Q	(P ⇒ Q)	(The rows may be read as saying:)	I.e., the theorem:
Row 1)	0	0	0	"If P is 0 and Q is 0, then [P ⇒ Q] is 0"	[T8-831]
Row 2)	T	0	0	"If P is T and Q is 0, then [P ⇒ Q] is 0"	[Ti8 ⇒ r2]
Row 3)	F	0	0	"If P is F and Q is 0, then [P ⇒ Q] is 0"	[T8-832]
Row 4)	0	T	0	"If P is 0 and Q is T, then [P ⇒ Q] is 0"	[Ti8 ⇒ r4]
Row 5)	T	T	T	"If P is T and Q is T, then [P ⇒ Q] is T"	[T8-833]
Row 6)	F	T	0	"If P is F and Q is T, then [P ⇒ Q] is 0"	[Ti8 ⇒ r6]
Row 7)	0	F	0	"If P is 0 and Q is F, then [P ⇒ Q] is 0"	[Ti8 ⇒ r7]
Row 8)	T	F	F	"If P is T and Q is F, then [P ⇒ Q] is F"	[T8-834]
Row 9)	F	F	0	"If P is F and Q is F, then [P ⇒ Q] is 0"	[Ti8 ⇒ r9]

8.232332 Implications and Definitions

With definitions, where the definens is a disjunction, the truth of the definiendum follows validly from the truth of any disjunct. To apply such a definition often needs an A-implication to carry the proof through. This suggests a *de dicto* principle which is essential to logic. It may be expressed as the derived principle of inference, "If (T((R Syn$_{df}$ (PvQ)) and Q), then TR"

I.e., DR8-9a. Valid$_I$ [(T(R Syn$_{df}$ (PvQ)) & TQ) $\Rightarrow$ TR]
I.e., .: Valid$_I$ [(TR Syn$_{df}$ T(PvQ)) & TQ) $\Rightarrow$ TR]
I.e., .: Valid$_I$ [(TR Syn$_{df}$ (TPvTQ)) & TQ) $\Rightarrow$ TR]

The meaning of a predicate is explicated through conditional statements which state conditions of applicability. The antecedent of such a conditional describes a sufficient condition for applying the predicate; the consequent asserts, in effect, that when that condition is fulfilled the predicate may be applied. A full definition allows the interchangeability of definieindum and definiens.

For example, the dictionary tells us 'a sibling' means 'a brother or sister'. In logical notation: '<1>is a **sibling** of<2>' **Syn$_{df}$** '(<1>is a **brother** of<2>or <1>is a **sister** of <2>)'
or, using abbreviations, 'Sib<1,2>' Syn$_{df}$ '(Br<1,2> v Sis<1,2>)', which has the abstract form, 'R<1,2> Syn$_{df}$ (P<1,2> v Q<1,2>)' or 'R Syn$_{df}$ (P v Q)'. Given this definition, how do we show that 'Al is a sibling of Bobby' follows logically from 'Al is the brother of Bobby'?

Let 'P' stand for 'Al is a brother of Bobby', 'Q' for 'Al is a sister of Bobby" and 'R' for "Al is a sibling of Bobby'. The derivation is as follows:

1) TP	[Premiss]
2) R Syn$_{df}$ (P v Q)	[Definition]
3) TP Impl T(P v Q)]	[Ti7-83]
4) Not-Inc(TP & T(P v Q))	[Inspection]
5) Valid$_I$[TP $\Rightarrow$ T(P v Q)]	[4),5),DR8-6g]
6) T(P v Q)	[1),5),MP]
7) TR Syn T(P v Q)	[2),R7-1,MP]
8) TR	[2),6),SynSUB]
9) Valid$_I$ [TP $\Rightarrow$ TR]	[1) to 8),Cond.Pr.]

The derivation is doubly *de dicto*; it depends on the definition of a word, in Step 2), and it depends on the A-implication (which depends on the presupposition of Trivalence) in step 3).

The final Validity-theorem gives us no new information about the world, but it gives us a rule of translation which preserves truth by virtue of our semantic conventions.

Some predicates can be defined with only a conjunction of POS predicates in the definiens. In such cases the relation between the definiendum (the expression being defined) and any conjunct in the definiens (the defining condition of applicability) is that the latter is a necessary condition of the applicability of the definiendum. But many predicates are applicable under a variety of different sufficient conditions. In this case the definiens contains several alternative sufficient conditions, any one of which, if true, warrants the predication of the definiendum as true.

If the definiens of a predicate is a disjunction of other predicates (no one of which is inconsistent with the definiendum), an inference by use of A-implication does not introduce irrelevant elements. If the defininiens is a disjunction, then by a synthetic *a priori* linquistic convention, the truth of any one of its disjuncts warrants asserting the truth of the definiendum. The implicit premiss, which contains an occurrence of the antecedent in a Syn$_{df}$ statement, thus contains an occurrence of the consequent.

For example, it can be held that by legal definition,[19]

‘<1>is a citizen of the U.S.’ Syn_{df} ‘Either <1>is a person born in the U.S.
or <1> is a person who satisfied naturalization requirments’

Thus there are two ways to be a citizen. Citizenship is not defined by either one alone. Satisfying one of the two alternatives satisfies the disjunction as a whole, and thus the concept of being a citizen of the U.S.

The abstract logical form of this definition is (P Syn_{df} (Q v R)) and the inference under considerations has the form “(T((P Syn_{df} (QvR)) and Q), **.:** TP”. Hence, the Validity-theorem,

Valid [If (T(P Syn_{df} (Q v R)) & TQ) then TP]

<u>Proof</u>: 1)	(T(P Syn (Q v R) & TQ)	[Premiss]
2)	T(P Syn (Q v R))	[1),SIMP]
3)	TQ	[1),SIMP]
4)	If T(P Syn (Q v R)) then (TP Syn T(Q v R))	[R7-1,U-SUB]
5)	TP Syn T(Q v R)	[2),4),MP]
6)	TP Syn (TQ v TR)	[5),Ax.7-4]
7)	Not-Inc(TP & (TQ v TR))	[Inspection]
8)	Valid(TP $\Leftrightarrow$ (TQ v TR))	[6),7),DR8-6d]
9)	Valid((TQ v TR) $\Rightarrow$ TP)	[8), Df ‘$\Leftrightarrow$’,SIMP]
10)	Valid_I (TQ $\Rightarrow$ (TQ v TR))	[Ti8-784.U-SUB]
11)	Valid_I (TQ $\Rightarrow$ TP)	[10),6),SynSUB]
12)	TP	[3),11),MP]
13)	Valid_I [(T(P Syn_{df} (Q v R)) & TQ) $\Rightarrow$ TP]	[1) to 12), Cond. Pr.]

The use of Implication in step 11) does not involve a random addition of ‘TR’ into a disjunction. It is there because it is connected to the objective of establishing a definition of P. The principle of satisfying a definition by satisfying a disjunct may be stated generally as the derived rule,

DR8-6e. [If T(P Syn_{df} (QvR) & R) and not-Inc(P&R)) then Valid_I (TR $\Rightarrow$ TP)]

For another example, the definition of ‘Inc’ has five clauses in its disjunctive definiens. It has the form: ‘Inc<1>’ Syn_{df} ‘(P<1>. v Q<1> v R<1> v S<1> v T<1>)’. If we find that any one of these clauses is true, we can infer that the definiendum is true. Thus given any one of the clauses, we can infer INC[P]

Df ‘Inc’. ‘Inc[P]’ Syn_{df} ‘[(P Syn (Q&~R)) & (Q Cont R)
v (P Syn (Q&R) & Inc(R))
v (P Syn (QvR) & Inc(Q) & Inc(R))
v (P Syn (Q $\Rightarrow$ R) & Inc(Q&R)
v (P Syn Q & Inc(Q))]’

19. XIVth Amendment to the U.S.Constitution: “All persons born or naturalized in the United States and subject to the jurisdiction thereof, are citizens of the United States and the state wherein they reside.”

Any true conclusion that a certain expression is Inconsistent by this logic presupposes that this definition is implicitly present in the premisses, thus, unlike Addition, all expressions that occur in the conclusion also occur (at least implicitly) in the premisses.

8.232333 Uses of Implication In Reasoning About Possibilities of Fact

There are occasions when it might appear that Valid_I (TP $\Rightarrow$ T(P v Q)) is a necessary step in a valid *de re* deduction. Consider the following argument:

1) If either lemon juice or vinegar is combined with milk, the milk will curdle [Premiss]
2) Vinegar was combined with this milk [Premiss]
3) $\textit{Valid}_I$ [*If* vinegar was combined with milk *then* (either lemon juice was combined with milk or vinegar was combined with milk)] [T_I 8-783,U- SUB]
4) (Either vinegar was combined with mile or lemon juice was combined with milk) [2),3),MP]
5) This milk will curdle. [4),1),MP]

This argument has the following form:

1) (T(PvQ) $\Rightarrow$ TR	[Premiss]	
2) TQ	[Premiss]	
3) $\textit{Valid}_I$ [TQ $\Rightarrow$ T(P v Q)]	[Ti8-783]	(Addition)
4) T(PvQ)	[2),3),MP]	
5) TR	[4),1),MP]	

It may be thought that this requires Addition in Step 3), but this is not necessarily the case. There are legimate deductions *de re* which yield the same conclusion. Clearly the first premiss means that there are two ways to make milk curdle; these are expressed by: (i) "If lemon juice is combined with milk, then the milk will curdle", **and** (ii) "if vinegar is combined with milk, then the milk will curdle". In short, in this case, '((P or Q) $\Rightarrow$R)' means '((P $\Rightarrow$R) **and** (Q$\Rightarrow$ R)'. Understood in this way, there is no need for an A-implication in the argument above. Replacing the first premiss by ((TP$\Rightarrow$TR) **and** (TQ$\Rightarrow$TR), two arguments are valid without Addition and have the forms:

1) (TP$\Rightarrow$TR) & TQ$\Rightarrow$TR)	[Premiss]	1) (TP$\Rightarrow$TR) & TQ$\Rightarrow$TR)	[Premiss]
2) TQ	[Premiss]	2) TP	[Premiss]
3) [TQ $\Rightarrow$TR]	[1),Simp]	3) [TP $\Rightarrow$TR]	[1),Simp]
4) TR	[2),3),MP]	4) TR	[2),3),MP]

Similarly, in purely formal logic, when [P$\Rightarrow$R] and [Q$\Rightarrow$R] are both Valid, then [(PvQ$\Rightarrow$R] is Valid, For example, from the logical validity of [(P&R)$\Rightarrow$R] and [(Q&R)$\Rightarrow$R] we prove the logical validity of (((P&R) v (Q&R)) $\Rightarrow$R) as follows:

<u>Proof</u>:
1) VALID[(P&R) $\Rightarrow$R] [T6-137,U-SUB]
2) VALID[(Q&R)$\Rightarrow$R] [T6-137,U-SUB]
3) $\models$[(P&R) CONT R) & not-Inc (P&R&R) [1),Df'Valid']
4) $\models$[(Q&R) CONT R) & not-Inc (Q&R&R) [2),Df'Valid']
5) [(P&R) CONT R) & ((Q&R) CONT R)] [3),4),Simp]
6) If ((P&R) CONT R) & ((Q&R) CONT R) then ((P&R)v(Q&R)) Cont R) [DR1-23]
7) ((P&R)v(Q&R)) Cont R) [5),6),MP]

8) Not-Inc(P&R) & not-Inc(Q&R) [3),4),Simp]
9) Not-Inc((P&R)v(Q&R)) & R) [Inspection]
10) VALID[((P&R)v(Q&R)) $\Rightarrow$R)] [7),9),Df'Valid']

In ordinary language and science there are many cases where an inferential conditional statement of the form ((P v Q) $\Rightarrow$R) is based on first proving ((P$\Rightarrow$Q) & (Q$\Rightarrow$R)).

If Al is born in the U.S. **or** Al was naturalized in the U.S., **then** A is a citizen of the U.S.
If Jo is over 17 **or** accompanied by her parents **then** Jo is admissible to the movie theatre.
If (air pressure is normal and temperature is 100°C **or** the air pressure is .5 normal and the temperature is 81°C), **then** water boils
If (there is no oxygen **or** the match is wet) and the match is struck, **then** the match won't light.

Each of these is a contingent conditional with a disjunctive antecedent. They have the form ((PvQ)$\Rightarrow$R), but they are all understood to mean that both (P$\Rightarrow$R) and (Q$\Rightarrow$R) are valid in some extra-logical sense. These examples could be called 'empirically valid' conditionals; ('E-Valid' vs.'Logically Valid'). Empirically, they have true instances but no false instances. They will be dealt with in detail in Chapter 9. But although we can prove the rule,

$\models$[((Valid (TP$\Rightarrow$TQ) & Valid (TQ$\Rightarrow$TR)) $\Rightarrow$Valid (T(PvQ) $\Rightarrow$TR)],[20]

we can not prove the converse, '(Valid(T(PvQ) $\Rightarrow$TR) $\Rightarrow$(Valid(TP $\Rightarrow$TR) & Valid(TQ $\Rightarrow$TR)))', in analytic truth-logic.

But *in those cases* in which the validity of the inferential conditional (T(P v Q) $\Rightarrow$ TR) is derived from the validity (empirical or logical) of both (TP$\Rightarrow$ TR) and (TQ$\Rightarrow$TR), the rule, "[If (T(PvQ)$\Rightarrow$TR] is valid and P is true, then R is true]" appears to be a valid principle. If the instance of 'T(PvQ)' is one in which Q can be replaced by any expression which is irrelevant to TP, as it is if 'TP $\Rightarrow$T(PvQ)' is an A-implication, then this condition is not met. For there is no assurance that Q, which may be replaced by any expression, is such that (TQ $\Rightarrow$TR) is valid (logically or empirically),

8.232334 Rules of Inference as Valid *de dicto* Conditionals

All rules of of inference, primitive or derived, have been expressed in English as conditional statements.

All principles of inference behind these rules are about properties and relations of the meanings of linguistic expressions; thus they are essentially purely *de dicto* principles.

If a rule or principle of inference is presented, it is implicitly assumed that it purports to be valid. This can be made explicit, by predicating 'is Valid' of the rule, or in our convention of logical notation, by prefixing 'Valid' to a quasi-quotation of the rule.

The conditionals expressed in these principles are inferential conditionals and can be interpreted as C-conditionals. If interpreted as TF-conditionals anomalies ensue.

These conditionals are understood to be intended as saying, ***if and when*** the antecedent is **true**, ***then*** the consequent is **true**. Thus derived *principles of inference* (as distinct from rules warranted by them) in Chapters 1 through 8 may be translated and symbolized explicitly as statements of the form 'Valid[TP $\Rightarrow$TQ]' and included as theorems of analytic truth-logic in the broad sense. These principles are *a priori* statements about properties or relations of meanings of expressions.

---

20. Based on DR1-23. If [A CONT C] and [B CONT C], then [(AvB) CONT C] and the consistency of T(P&Q&~TL).

The following is a selection of the principles of inference we have used or proven. They are formalized with both the truth-operators and the claim of validity made explicit (hitherto the claim has been implicit only).

The first two *primitive* principles for SynSUB an U-SUB do not translate completely into symbols we have used, since they use expressions like 'is a component in', 'occurs in', 'is h-adic', which pertain to the visible characteristics of the physical symbols, rather than properties or relations of the ideas associated with them. Neverthess, the relationships of physical symbols are intended to be correlated with relations between the meanings assigned to the symbols. Whether their meanings are grasped or not depends on whether the explanations in ordinary English are clear and properly understood.

The next group of principles includes rules we have used in proving principles of inference. They are essentially detachment rules, telling us that when expressions have certain logical characterics, then another different expression may be taken as having, by itself, a specified logical characteristic.They are based on principles which are established in this and previous chapters, although in the case of the first one, Conditional Proof, we have not attempted to give a full proof. (If all principles of inference are formulated as C-conditionals, Conditional Proof becomes a form of Hypothetical Sorites.)

All of the rest of the principles are principles which have been proven from definitions of terms using priniples of inference of A-logic. In the broad sense, they are part of A-logic. But they are not part of purely formal logic, for their subject terms (the expressions they talk about) are not confined to expressions with syntax and syncategorematic words only. They are a special branch of A-logic (called "metalogic") which deals with the predicates that A-logic uses in its theorems. A separate treatise could present this branch of A-logic, but will not be attempted here.

Primitive Principles of Inference

R6-1. Valid[T($\models$P & Q is a component of P & $\models$(Q Syn R)) $\Rightarrow$T $\models$P(Q//R)] [SynSUB]

R6-2. Valid[T($\models$R & $P_i<t_1,\ldots,t_n>$ occurs in R
& Q is a h-adic wff, h > n
& Q has an occurrence of each numeral 1 to n
& no individual variable in Q occurs in R)
$\Rightarrow$ T($\models$(R($P_i<t_1,\ldots,t_n>$/Q)))] [U-SUB]

R6-3. Valid[T($\models P<t_1,\ldots,t_n>$) $\Rightarrow$T($\models P<t_1,\ldots,t_n>(t_i/a_j)$)] [INST]

R6-6. Valid[T(Valid(P $\Rightarrow$Q)) $\Leftrightarrow$T(Valid(P, **.:** Q))] [VC\VI]

Principles used in Proofs of Principles of Inference

CP. [T(A is a consistent ordered set of wffs, $<A_1, A_2,\ldots,A_n>$ $(1\leq i\leq n)$
& for each A_i,(i > 1), (T($\models A_i$) **v** ((g,h<i) & $A_h=(A_g \Rightarrow A_i)$ & Valid($A_g \Rightarrow A_i$)))
$\Rightarrow$T(Valid($A_1 \Rightarrow A_n$))] "Conditional Proof" [DR6-10]

MP. Valid [((TP $\Rightarrow$TQ) & TP) $\Rightarrow$TQ] "Modus Ponens" [T6-13,U-SUB]

ADJ. Valid [TP, TQ, **.:** T(P&Q)] "Adjunction" [From Ax.7-03]

SIMP. Valid [(TP & TQ), **.:** T(P)] "Simplification" [From T1-36]

Hyp-Syll. Valid[T(Valid(P$\Rightarrow$Q) & Valid (Q$\Rightarrow$R)) $\Rightarrow$T(Valid (P $\Rightarrow$R))]

Rules for deriving Syn- and Cont-theorems from Syn- and Cont-theorems

DR1-01. Valid [T(P SYN Q) $\Rightarrow$T(Q SYN P)]

DR1-02. Valid [T((P SYN Q) & (Q SYN R)) $\Rightarrow$T(P SYN R)]

DR1-11. Valid [T(P SYN Q) $\Rightarrow$T(P CONT Q)]

DR1-14. Valid [T(P SYN Q) $\Rightarrow$T((P CONT Q) & (Q CONT P))]

DR1-16. Valid [T((P SYN Q) & (R CONT S)) $\Rightarrow$T(R CONT S(P//Q))]

DR1-18. Valid [T((P CONT Q) & (Q CONT P)) ⇒ T(P SYN Q)]
DR1-19. Valid [T((P CONT Q) & (Q CONT R)) ⇒ T(P CONT R)]

Rules for deriving Inc- and TAUT-theorems from Syn- and Cont-theorems

From Ch 5. With Wffs of M-logic

DR5-5 Valid[(P Syn Q) & Inc(R)) ⇒ Inc(R(Q//P)]
DR5-5a Valid[(P Cont Q) ⇒ Inc(P & ~Q)]
DR5-5b Valid[(P Syn Q) ⇒ Inc(P & ~Q)]
DR5-5c Valid[(P Syn Q) ⇒ Inc(Q & ~P)]
DR5-5d Valid[(P Syn Q) ⇒ Inc(P&~Q) & Inc(Q&~P)]
DR5-5d' Valid[(P Syn Q) ⇒ Taut(P ≡ Q)]
DR5-5f Valid[(Inc(P) & Inc(~P & Q)) ⇒ Inc(Q)] [Inc-Detachment]
DR5-5f'. Valid[(Taut(P) & Taut(P ⊃ Q)) ⇒ Taut(Q)] [Taut-Detachment]

From Ch 6. With C-conditionals as components

DR6-5a .Valid[(P Cont Q) ⇒ Inc(P ⇒ ~Q)]
DR6-5a' Valid[(P Cont Q) ⇒ Taut(~(P ⇒ ~Q))]

From Ch7. With M-logic Wffs and T-operators

DR7-5a. Valid[(P Cont Q) ⇒ Inc(T(P & ~Q))]
DR7-5b. Valid[(P Syn Q) ⇒ IncT(P&~Q)]
DR7-5d. Valid[(P Syn Q) ⇒ Inc(T(P&~Q) & T(Q & ~P)]
DR7-5d'. Valid[(P Syn Q) ⇒ Taut~F(P ≡ Q)]

From Ch 8. With C-conditionals and T-wffs

DR8-5a. Valid[(P Cont Q) ⇒ Inc(T(P ⇒ ~Q))]
DR8-5a'. Valid[(P Cont Q) ⇒ Taut(~T(P ⇒ ~Q))]
DR8-5d. Valid[(P Impl Q) ⇒ Inc(T(P ⇒ ~Q)]
DR8-5d'. Valid[(P Impl Q) ⇒ Taut~T(P ⇒ ~Q)]
⊨ Valid[(P Cont Q) ⇒ Inc(TP ⇒ ~TQ))] [DR6-5a,U-SUB]
⊨ Valid[(P Cont Q) ⇒ Taut~(TP ⇒ ~TQ))] [DR6-5a',U-SUB]

Rules for deriving Valid Inference schemata from Syn- and Cont-theorems

From Ch.5. Valid Inference Schemata for Wffs of M-logic

DR5-6a .Valid[((P Cont Q) & ~Inc(P&Q)) ⇒ Valid (P, **.:**Q)]
DR5-6b . Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid (P, **.:**Q)]
DR5-6c . Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid (Q, **.:**P)
DR5-6d. Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid ((P, **.:**Q) & (Q, **.:**P)]

From Ch 6. With Wffs of M-logic and C-conditionals

R6-6 Valid[Valid(P ⇒ Q) ⇔ Valid(P, **.:**Q)] "VC\VI"
DR6-6a. Valid[((P CONT Q) & ~Inc(P&Q)) ⇒ VALID(P ⇒ Q)]
DR6-6b. Valid[((P SYN Q) & ~Inc(P&Q)) ⇒ VALID(P ⇒ Q)]
DR6-6c. Valid[((P SYN Q) & ~Inc(P&Q)) ⇒ VALID(Q ⇒ P)]
DR6-6d. Valid[((P Syn Q) & ~Inc(P & Q)), ⇒ Valid(P ⇔ Q)]
DR6-6e. Valid[Valid(P ⇒ Q) ⇒ Valid(P, **.:** Q)]
DR6-6f. Valid[Valid(P, **.:** Q) ⇒ Valid(P ⇒ Q)]

From Ch 7. With Wffs of M-logic and T-wffs (no C-conditionals)

DR_I7-4a. Valid[((P Cont Q) & (Q Impl R) & Not-(P Cont R)) ⇒ (P Impl R)] "CIISyll"
DR_I7-4b. Valid[((P Impl Q) & (Q Cont R) & Not-(P Cont R)) ⇒ (P Impl R)]"ICISyll"
DR7-6a. Valid[((P Cont Q) & ~Inc(P&Q)) ⇒ Valid (TP, **.:**TQ)]

DR7-6b. Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid (TP, .:TQ)]
DR7-6c. Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid (TQ, .:TP)]
DR7-6d. Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid ((TP, .:TQ) & TQ, .:TP))]
DR7-6g Valid[((P Impl Q) & ~Inc(P&Q)) ⇒ Valid_I (P, .:Q)]

From Ch 8. With C-conditionals with and T-wffs

DR8-6. Valid[Valid(TP, .:TQ) ⇒ Valid(TP ⇒ TQ)]
DR8-6a. Valid[((P Cont Q) & ~Inc(P&Q)) ⇒ Valid (TP ⇒ TQ)]
DR8-6b. Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid (TP ⇒ TQ)]
DR8-6c. Valid[((P Syn Q) & ~Inc(P&Q)) ⇒ Valid (TQ ⇒ TP)]
DR8-6d. Valid[((P Syn Q) & ~Inc(P & Q)) ⇒ Valid(TP ⇔ TQ)].
DR8-6g. Valid[((P Impl Q) & ~Inc(P&Q)) ⇒ Valid_I (P ⇒ Q)]

(Note: Derived Rules giving sufficient conditions for Inc- or TAUT-theorems have '-5' following the first number which indicates the chapter in which it was introduced; Rules for introducing 'Valid' have '-6', and rules for theorems with a truth-operator prefixed have '-7' in that position.)

8.24 Inc- and TAUT-theorems of Analytic Truth-logic

In this chapter we have C-conditionals with T-operators mixed in. There are two distinct ways in which a C-conditional is inconsistent or tautologous in truth-logic.

The first way is based on the presupposition of truth-logic DR7-5. [If Inc(P) then Inc(T(P)]— if an expression is inconsistent, then the assertion that it is true is inconsistent. This deals with inconsistency among *assertions that C-conditionals are true*. From this we get the derived rules DR8-5a and DR8-5a' which are unique to this chapter. For Impl-thoerems there are the similar rules DR8-5d and DR8-5d'.

DR8-5a. [If (P Cont Q) then IncT(P ⇒ ~Q)]
Proof: 1) If (P Cont Q) then Inc(T(P&~Q)) [DR7-5,U-SUB]
2) If (P Cont Q) then IncT(P⇒~Q) [1),Ax.8-01,SynSUB]

DR8-5a'. [If (P Cont Q) then Taut (~T(P ⇒~Q)))]
Proof: 1) If (P Cont Q) then IncT(P ⇒~Q) [DR8-5a]
2) If (P Cont Q) then Inc ~~T(P ⇒~Q) [1),DN,SynSUB(twice)]
3) If (P Cont Q) then Taut(~T(P ⇒~Q)) [3), Df 'Taut',SynSUB]

The second way is based on the fourth clause of Df 'Inc', in Chapter 6, that makes (P⇒Q) Inc if and only if (P & Q) is Inc. This deals with inconsistencies among *inferential* conditionals of truth-logic, a matter covered in Chapter 6 independently of truth-logic. It includes derivations using U-SUB and DR6-5a and DR6-5a'.

DR6-5a [If (P Cont Q), then Inc(P ⇒~Q)]
DR6-5a' [If (P Cont Q), then Taut(~(P ⇒~Q)]

Using U-SUBa, DR6-5a and DR6-5a' yield the derived rules,

|= [If (TP Cont TQ), then Inc(TP ⇒~TQ)] [By DR6-5a,U-SUB(P/TP,Q/TQ]
and |= [If (TP Cont TQ), then Taut(~(TP ⇒~TQ)] [By DR6-5a',U-SUB]
which yield new theorems from the Cont-, Syn- and Impl-theorems of Chapters 7 and 8, including

|=Inc[(T(P & Q) ⇒ ~ TP] [T7-136,DR6-5a, MP]
"[If P is true and Q is true, then P is not true]' is inconsistent".

|=Taut [~ (T(P & Q) ⇒ ~ TQ] [T7-137, T7-DR6-5a']
"[It is not the case that (if P and Q is true, then Q is not true)] is tautologous"

|=Inc[TP ⇒ ~ T(P v Q)] [Ti7-83,DR6-5a]
"[if P is true, then (P or Q) is not true] is inconsistent."

|=Inc[(TP & T(~ P v Q)) ⇒ ~ TQ] [T7-46,…,DR6-5a]
"[If P is true and (~ PvQ) is true, then Q is not true] is inconsistent."

|=Taut [~ (F(Pa) ⇒ T(∀x)Px)] [Ti8-90,…,DR6-5a']
"It is tautologous that [It is not the case that (If Pa is false, then it is true that for all x, Px]".
Among such derivations by DR6-5a and DR6-5a' are laws of logic related to trivalence.

|=Inc[TP ⇒ FP]	Three Laws of	[T7-13,DR6-5a,U-SUB]
|=Inc[TP ⇒ ~ TP]	Non-contradiction	[T7-11,DR6-5a,U-SUB]
|=Inc[FP ⇒ ~ FP]		[T7-12,DR6-5a,U-SUB]
|=Inc[0P ⇒ (TP v FP)]	The Law of	[T7-48,DR6-5a,U-SUB]
|=Taut[~ (0P ⇒ (TP v FP))]	Trivalence	[T7-48,DR6-5a',U-SUB]
|=Taut[~ (TP ⇒ FP)]		[T7-13,DR6-5a',U-SUB]
|=Taut[~ (TP ⇒ ~ TP)]	Three Basic	[T7-11,DR6-5a',U-SUB]
|=Taut[~ (FP ⇒ ~ FP)]	Tautologies	[T7-12,DR6-5a',U-SUB]

Note that the '⊃'-for-'⇒' analogues of all of these are neither inconsistent nor tautologous.[21]

From the truth of any C-conditional, it follows that the conditional with the same antecedent and the denial of its consequent is false. Validity theorems expressing this are,

T8-820. Valid[T(P ⇒ Q) ⇒ ~ T(P ⇒ ~ Q)]
T8-821. Valid[T(P ⇒ Q) ⇔ F(P ⇒ ~ Q)]

Some such principle has been attributed to Boethius.[22] His thesis can be formulated in several ways, among them as an Inc- or a Taut-theorem. Thus '**Inc**[T(P ⇒ Q) & T(P ⇒ ~ Q)]' and '**Taut**[T(P ⇒ Q) ⊃ ~ T(P ⇒ ~ Q)]' are derivable from T8-18 as follows:

---

21. By truth-tables all are contingent:

[TP ⊃ FP],	[TP ⊃~TP],	[FP ⊃~FP],	[(TP v FP) ⊃ 0P],	[~(0P ⊃ (TP v FP))],	[~(TP ⊃ FP)],
f 0 **T** f 0	f 0 **T** t f 0	f t **T** t f t	f 0 f f0 **T** t0	**T** t f f f 0 f f0	**F** f0 t f0
t t **F** f t	t t **F** f t t	f t **T** t f t	t t t f t **F** f t	**F** f t t t t t ft	**T** t t f f f
f f **T** t f	f f **T** f t f	t f **F** f t f	f f t t f **F** f f	**F** f f t f f t t f	**F** f f t t f

[~(TP ⊃~TP)],	[~(FP ⊃~FP)],	[~TTP ⊃~FTP],	[~TFP ⊃~FFP],	[~T0P ⊃~F0P]
F f0 T t f0	**F** f0 T t f0	T f f0 **F** f t f0	t f f0 **F** f t f0	t f f0 **T** f t f0
T t t F f t t	**F** f t T t f t	F t t t **T** t f t t	t f f t **F** f t f t	t f f f **F** f t f f
F f f T f t f	**T** t f F f t f	t f f f **F** f t f f	f t t f **T** t f t f	t f f f **F** f t f f

22. E.g., see Storrs McCall, "Connexive Implication", ***J of Symbolic Logic***, 1966, pp 415-433.

1) [T(P ⇒Q) Syn F(P ⇒~Q)] [T8-18]
2) [F(P ⇒~Q) Cont ~T(P ⇒ ~Q)] [T7-15,U-SUB]
3) [T(P ⇒ Q) Cont ~T(P ⇒ ~Q)] [1),2),HypSYLL]
4) Inc[T(P ⇒Q) & ~~T(P ⇒ ~Q)] [3),Df 'Inc']
➔ 5) **Inc[T(P ⇒Q) & T(P ⇒ ~Q)]** [4),DN,SynSUB]
➔ 6) **Inc[T((P ⇒Q) & (P ⇒ ~Q))]** [5),Ax.7-03,SynSUB]
7) Inc[~~(T(P ⇒Q) & ~~T(P ⇒~Q)] [5),DN,SynSUB]
8) Taut[~(T(P ⇒Q) & ~~T(P ⇒~Q)] [6),Df 'Taut']
➔ 9) **Taut[(T(P ⇒Q) ⊃ ~T(P ⇒~Q)]** [7),Df '⊃']

By various devices, from every Cont- Syn- and Impl-theorem and Validity-theorem in A-logic an Inconsistency and a Tautology can be be derived with the same antecedent but a contrary consequent. From every inference of M-logic which is valid according to A-logic, an Inc-theorem and a Taut-theorem is derivable in A-logic that has no inconsistent or tautologous analogue among the TF-conditionals of M-logic. A-logic has all of the tautologies of M-logic and vastly many more that M-logic can not express due to the lack of the C-conditional. However, listing tautologies is not the proper aim of logic, and the related inconsistencies are only of interest as candidates for elimination.

8.25 Logically Unfalsifiable and Unsatisfiable C-conditionals

There are derived rules for moving from valid inferential conditionals in the antecedent, to assertions of non-truth or non-falsity:

If any wff is Inconsistent it follows that it, and any instantiation of it is not-true. If any wff is Tautologous it follows that it, and any instantiation of it is not-false. These are the principles of inference expressed by R7-2.[If Inc(P) then ~T(P)] and DR7-2b.[If Taut(P) then ~F(P)].

In addition to the theorems of logical non-truth derivable by principles from tautologies of M-logic, we now have those derivable due to inconsistencies of C-conditionals in A-logic.

DR8-7.[If Valid(P⇒ Q) then ~T(P⇒ ~Q)]
Proof: 1) Valid(P ⇒ Q) ⇒ (P Cont Q) [Df 'Valid⇒ ']
2) If (P Cont Q) then ~T(P ⇒ ~Q) [DR6-5a]
3) If Valid(P ⇒ Q) then ~T(P⇒ ~Q) [1),2),HypSYLL]

DR8-7'.[If Valid(P⇒ Q) then ~F(P⇒ Q)]
Proof: 1) If Valid(P ⇒ Q) then ~T(P⇒ ~Q) [DR8-7]
2) If Valid(P ⇒ Q) then ~T(P&~Q) [1),Ax.8-01,SynSUB]
3) If Valid(P ⇒ Q) then ~F(P⇒ Q) [2),Ax,8-02,SynSUB]

These principles connect validity with *non*-falsehood and *non*-truth, but they do not infer truth from validity. They entail the unfalsifiability theorems which follow from Validity theorems or Taut-theorems of previous sections. Their duals establish unsatisfiability-theorems based on inconsistency:

Logically Unfalsifiable	Logically Unsatisfiable
|= ~F[~(TP ⇒ ~TP)]	|= ~T[TP ⇒ ~TP]
|= ~F[~(TP ⇒ FP)]	|= ~T[TP ⇒ FP]
|= ~F[~(FP ⇒ ~FP)]	|= ~T[FP ⇒ ~FP]
|= ~F[~(0P ⇒ (TP v FP))]	|= ~T[(~TP & ~FP) ⇒ 0P]
|= ~F[~(~TTP ⇒ ~FTP)]	|= ~T(~TTP ⇒ ~FTP)
|= ~F[~(~TFP ⇒ ~FFP)]	|= ~T(~TFP ⇒ ~FFP)
|= ~F[~(~T0P ⇒ ~F0P)]	|= ~T(~T0P ⇒ ~F0P)

From each Validity-theorem with a C-conditional the unfalsifiability of its C-conditional follows, as well as the unfalsifiability of its analogue with a TF-conditional (which could also have been derived by M-logic with T-operators). From each of these can be derived a referentially synonymous theorem asserting the logical non-truth of its opposing conditional which has the same antecedent and the denial of its consequent. Note that theorems of logical unfalsifiability and logical unsatisfiability are all 2nd-level T-wffs or higher.

8.26 *De dicto* Logically True and Logically False C-conditionals

To assert that an expression is logically not-false is not the same as asserting that it is logically true. The truths of logic are all ***de dicto*** truths, truths about expressions; e.g., true statements that two expressions are logically synonymous (as we have defined 'synonymous', etc.,) or that one logically contains another, or that an expression is inconsistent or tautologous, contingent, valid or invalid. All theorems of formal logic are implicitly presented as true statements about the meanings of logical operators, logical syntax, and the logical predicates of that branch of logic.

The sorts of theorems we have been talking about in the last three sections, might be called theorems of metalogic, or theorems of the logic of the predicates used in logic, i.e., the logic of logical theory. These theorems are distinct from the logic of mathematical theories, or the logic of physical theories, or the logic of biology, or all other logics of terms used in other areas of inquiry, large or small. The logic of any discipline or inquiry is simply the logic of the predicates which are investigated in efforts to describe the subject matter and the objectives of that inquiry. New discoveries and new objectives lead to new concepts and to new definitions of old terms and the logic of the new theories go on from there. Pure formal logic deals only with words used in every kind of inquiry.

We saw in Section 7.4224 that there are many 2nd-level T-wffs without C-conditionals that are *de dicto* logical truths in the sense that the final columns of their truth-tables have only T's in them. These included T7-71 T[0P v TP v FP] and

T7-72. T[~TP v TP]	T7-75. T[~(TP & ~TP)]	(Syn F[TP & ~TP])
T7-73. T[~TP v ~FP]	T7-76. T[~(TP & FP)]	(Syn F[TP & FP])
T7-74. T[~FP v FP]	T7-77. T[~(FP & ~FP)]	(Syn F[FP & ~FP])

It is possible to get endless numbers of merely *de dicto* logical truths with C-conditionals also. One need only find an inconsistent C-conditional and prefix 'T~T' to it and the truth-table will take only 'T's in the final column.

In M-logic, the concept of wffs which take only T's in their truth-tables, or the broader concept of sentences which are "true for all values of its variables", became a definitive characteristic of logical validity. In A-logic, logically true statements in this sense are never A-valid (since they are truth-assertions), and are trivial, merely *de dicto* expressions, which play no role at all in the application of logic to ordinary or scientific reasoning, Every Validity theorem in A-logic has instantiations ("values of its variables"), which are not-true, though none of them have false instantiations, so, though universally valid, none of them are universal logical *truths* in the sense of M-logic.

8.3 Miscellaneous

Two problems relate to the Traditional Law of Contraposition, and Squares of Oposition.

8.31 Transposition with C-conditionals

The strongest objection to the C-conditional is that it is incompatible with the traditional general law of contraposition (or transposition) i.e., the claim that it is universally true that,

'(If P then Q)' is true *iff* '(If not-Q then not-P)' is true. (Traditional Contraposition)

This objection is on target. If one accepts the C-conditional as the best interpretation of 'If…then', one must give up the universality of the law of contraposition as just stated, and settle for more limited versions. If one insists that the law of contraposition must hold in every legitimate interpretion of 'If…then', then one must reject the C-conditional. Hempel sticks with the universality of contraposition and rejects the C-conditional. We stick with the C-conditional and reject the traditional general law of contraposition.

Given the meaning of T(P ⇒ Q) in A-logic, the assertion that a C-conditional is true if and only its contrapositive is true is not only invalid, it is inconsistent. For, by Axioms 8-01 and 8-02 what makes (P ⇒ Q) true is T(P & Q) and what makes (~Q ⇒ ~P) true is T(~P&~Q). Thus if the first is true, the second is not and if the second is true the first is not. What makes a C-conditional true is that both antecedent and consequent are true, and what makes its contrapositive true is that both antecedent and consequent are false.

T(P ⇒ Q)	<=>	T(~Q ⇒ ~P)	T(P ⇒ ~Q)	<=>	T(Q ⇒ ~P)	T(~P ⇒ Q)	<=>	T(~Q ⇒ P)
F F F	000	F F F	F F F	000	F F F	F F F	000	F F F
F **T** F	0**F**0	F **F** F	F F **T**	00**F**	F F **F**	F F **F**	00**F**	F F **T**
F F **F**	00**F**	F F **T**	F **F** F	0**F**0	F **T** F	F **T** F	0**F**0	F **F** F

Hempel, assuming that contraposition must be a universal law of logic, argued that we must give up Nicod's "criterion of confirmation" (which is the same as that of the C-conditional).[23]

In Hempel's words, Nicod's criterion violated the "equivalence condition", i.e., that "whatever confirms (disconfirms) one of two equivalent sentences, also confirms (disconfirms) the other". But, of course, by "equivalence", Hempel meant "truth-functional equivalence" and the specific equivalence he had in mind was the truth-functional equivalence of the two truth-functional conditionals (P ⊃ Q) and (~Q ⊃ ~P). The "equivalence" of these two TF-conditionals is not at issue. In A-logic they are not only truth-functionally equivalent; they are synonymous. There is no change in the Laws of Contraposition if the conditional is the TF-conditional.[24]

23. Carl G. Hempel, "Studies in the Logic of Confirmation" *MIND* (1945), especially Section 3,"Nicod's Criterion of Confirmation and its Shortcomings".

24. In A-logic TAUT((P ⊃ Q)≡(~Q ⊃ ~P)) and |= [(P ⊃ Q) SYN (~Q ⊃ ~P)] since |= [(P ⊃ Q) SYN (~P v Q)] and |= [(~Q ⊃ ~P) SYN (~P v Q)].

The question of accepting Nicod's Criterion is not a question of violating Hempel's equivalence condition, but a question of whether the C-conditional, or the TF-conditional with General Contraposition, serves best the purposes of science including confirmation theory.

Laws of contraposition are generally formulated on the presupposition that the enterprise is one of truth-seeking. In other than truth-logics, the interchangability of contrapositives is frequently unacceptable and sometimes grammatically nonsensical.[25] However, it is only as laws of truth-logic that we are interested in them here. In analytic truth-logic, though Contraposition is not a universal law of truth-logic, there are several versions of contraposition which are valid and may be useful.

First, in A-logic there are valid laws of contraposition which hold with respect to falsehood and non-falsehood of conditionals.

T8-822. Valid:[F(P ⇒ Q) ⇔ F(~ Q ⇒ ~ P)]
"A C-conditional is false if and only if its contrapositive is false."[26]

Consequently, by DR4-1,

T8-823. Valid:[~ F(P ⇒ Q) ⇔ ~ F(~ Q ⇒ ~ P)]
"A C-conditional is not-false if and only if its contrapositive is not false."

From which we may conclude, by T8-713, U-SUBa, and Syll,

T8-824. Valid[T(P ⇒ Q) ⇒ ~ F(~ Q ⇒ ~ P)
"The truth of (P ⇒ Q) entails the unfalsifiability of (~ Q ⇒ ~ P),"

though the converse does not hold.

These formula are all valid inferential conditionals. This is related to the fact that the criterion of falsity for a conditional is the same in A-logic and PM-logic. In A-logic they are based on the following synonymy and containment theorems, derivable from T1-11.

	1) F(P ⇒ Q) Syn F(P ⇒ Q)	[T1-11]
	2) F(P ⇒ Q) Syn T(P& ~ Q)	[1), Ax.8-02]
	3) F(P ⇒ Q) Syn T(~ Q&P)	[2),&-Comm]
	4) F(P ⇒ Q) Syn T(~ Q& ~ ~ P)	[3),DN]
re: T8-822	5) **F(P ⇒ Q) Syn F(~ Q ⇒ ~ P)**	[4),Ax8-02]
re: T8-823	6) **~ F(P ⇒ Q) Syn ~ F(~ Q ⇒ ~ P)**	[5),R4-1]
	7) T(P ⇒ Q) Cont ~ F(P ⇒ Q)	[T7-15.U-SUB]
re: T8-824	8) **T(P ⇒ Q) Cont ~ F(~ Q ⇒ ~ P)**	[7),6),R1-1]

Trivalent truth-tables show that these laws are never false and sometimes true:

25. Intuitively, it is not necessary that if we accept, "If P is true, then do Y!" then we must accept "If Do Y! is rejected, then it is not the case that P is true". Again it does not seem intuitively clear that "If P is true, then Y is worth doing" entails "If Y is not worth doing, then P is not true". In the logic of questions, "If P is true, then is a F?" does not entail 'if not (is a F?) then it is not the case that P is true'; the latter is not even grammatically acceptable. Laws of contraposition seem particularly tied to truth-logic.

26. This is connected with the fact that 'F(P ⇒ Q)' Syn 'F(P ⊃ Q)'.

F(P ⇒ Q)	⇔	F(~Q ⇒ ~P)
F F F	000	F F F
F F T	00T	F F T
F F F	000	F F F

~F(P ⇒ Q)	⇔	~F(~Q ⇒ ~P)
T T T	TTT	T T T
T T F	TT0	T T F
T T T	TTT	T T T

T(P ⇒ Q)	⇒	~F(~Q ⇒ ~P)
F F F	000	T T T
F T F	0T0	T T F
F F F	000	T T T

The right-most truth table, for the wff in T8-824, is the closest a C-conditional comes to the traditional General Law of Contraposition, "T(If P then Q) if and only if T(If ~Q then ~P)". T8-824 says if a C-conditional is true, then its contrapositive can not be false.

With respect to purely inferential conditionals, two contraposed inferential conditionals are neutral with respect to each other, but either one validly entails the non-falsehood of the other, while being inconsistent with the assertion that the other is true. This is reflected in the truth-tables:

INVALID

(TP ⇒ TQ)	⇔	(FQ ⇒ FP)
0 0 0	000	0 0 F
F T F	000	0 0 F
0 0 0	000	0 0 T

VALID

(TP ⇒ TQ)	⇒	~F(FQ ⇒ FP)
0 0 0	000	T T F
F T F	0T0	T T F
0 0 0	000	T T T

INCONSISTENT

(TP ⇒ TQ)	⇒	T(FQ ⇒ FP)
0 0 0	000	F F F
0 T F	0F0	F F F
0 0 0	000	F F T

These results are intuitively understandable and provable in A-logic. For what makes either a C-Conditional or a TF-conditional false, is that the antecedent is true and the consequent false. When both are denied and the positions interchanged, we have that ~C true and ~A false, i.e., C false and A true, which is what makes conditionals false. Thus if an inferential conditional is valid, its contrapositive can't be false since that would make it false also thus not valid. But also its contrapositive can be true, since the sole case in which an inferential conditional is true, is one in which its contrapositive is false.

To infer from the non-Truth of a C-conditional the not-Truth of its contrapositive is invalid, and a contingent matter. Depending on content, some C-conditionals are not-True if their contrapositive is (e.g., when P is false and Q is true) but for other conditionals this would be false (if, for example both P and Q are true).

<u>INVALID</u>

~T(P ⇒ Q)	⇔	~T(~Q ⇒ ~P)
T T T	TTT	T T T
T **F** T	T**F**T	T **T** T
T T **T**	TT**F**	T T **F**

<u>Examples</u>:

~T(P ⇒ Q)	⇔	~T(~Q ⇒ ~P)
TF F 0 T	**T**	TF FT 0 TF

~T(P ⇒ Q)	⇔	~T(~Q ⇒ ~P)
FT T T T	**F**	TF FT 0 FT

The complexity of the laws of contraposition is somewhat meliorated by the fact that the chief use of the law of contraposition is to establish that the contrapositve of a valid or true conditional can't be false, and this principle is preserved in A-logic in the the two valid laws of contraposition,

T8-824 Valid[(T(P ⇒ Q) ⇒ ~F(~Q ⇒ ~P)]
and ⊨Valid[(TP ⇒ TQ) ⇒ ~F(~Q ⇒ ~P)]

8.32 Aristotelian Syllogistic and Squares of Opposition

The traditional logic accepted by most philosophers up to 1910 included the theory of the syllogism and the square of opposition which was developed from Aristotle's *Prior Analytics*. Sometimes described as Aristotle's greatest and most original achievement, it captured modes of inference that were recognized as valid and correct for two millenia.

Mathematical logic was enormously more powerful than Aristotle's logic. It gave rigorous clarification to vast areas of logical inference which traditional logic could not handle—in particular, where it applied to derivations in mathematical proofs.

Since both M-logic and A-logic differ in certain respects from Aristotelian logic, it is incumbent upon them to explain the differences, retaining what is right and explaining what is wrong in traditional logic.

In mathematical logic the first step was to re-define the traditional A, E, I and O propositions which are the four basic forms in Aristotelian logic. It first separated singulary from particular and general propositions, and then re-formulated A, E, I and O with quantifiers, variables and, for A and E, TF-conditionals. In modern logic elementary singular propositions have a predicate, one or more individual constants and are either negated or not: E.g., Pa, ~Pa, Qbc, ~Qbc, Rabc, ~Rabc,...etc. The most significant departure from Aristotelian logic is the recognition of polyadic predicates. Aristotle did not deal with them adequately. The general propositions A, E, I and O, are reformulated as follows:

A. 'All A's are B's' becomes '(∀x)(If Ax then Bx)'
E. 'No A's are B's' becomes '(∀x)(If Ax then ~Bx)'
I. 'Some A's are B's' becomes '(∃x))(Ax and Bx)'
O. 'Some A's are not B's' becomes '(∃x)(Ax and ~Bx)'

In M-logic, due to the truth-functional meaning assigned to "if...then" together with Quantifier Interchange and other principles, the Square of Opposition differs from the Aristotelian Square of Opposition; it lacks the relationships of implication, contrariety and sub-contrariety. In the theory of the syllogism, only 15 of the 24 syllogism valid in Aristotelian logic are valid in M-logic. The nine missing syllogisms are ones with two universal (A or E) premisses and a particular (I or 0) conclusion.

Aristotelian Square of Opposition

All A's are B's	<—Contraries—>	**No A's are B's**
	Con- ies	
	tra- or-	
Implies	dict-	Implies
	tra- or-	
	Con- ies	
Some A's are B's	<—Sub-Contraries—>	**Some A's are not B's**

M-logic Square of Opposition

$(\forall x)(Px \supset Qx)$		**$(\forall x)(Px \supset {\sim}Qx)$**
${\sim}(\exists x)(Px \,\&\, {\sim}Qx)$		${\sim}(\exists x)(Px \,\&\, Qx)$
	Con- ies	
	tra- or-	
	dict-	
	tra- or-	
	Con- ies	
$(\exists x)(Px \,\&\, Qx)$		**$(\exists x)(Px \,\&\, {\sim}Qx)$**

The standard explanation for these differences between M-logic and Aristotelian logic is by reference to "existential Import": it is said that A and E propositions in Aristotelian logic presuppose the existence of at least one entity to which the subject term applies, whereas in M-logic this is not assumed. But this explanation is unsatisfactory; it leads to inconsistencies. For one thing, '(∀x)Px' implies '(∃x)Px' in M-logic, this conflicts with any general claim that the universal quantifier lacks "existential import". Secondly, the effort by some proponents of M-logic to interpret Aristotle's A proposition with existential import as '((∃x)Sx & (∀x)(Sx ⊃ Px))' belies the claim that its contradictory, '∼((∃x)Sx & (∀x) (Sx ⊃ Px))', is an 0 proposition, for the latter is true if nothing is S, i.e., if O lacks existential import. A better explanation locates the difference in M-logic's truth-functional interpretation of "if…then".[27]

In A-logic, with the C-conditional, all the relations in the Aristotelian Square of Opposition can be shown to be valid provided the distinction is made between truth-assertions about conditionals and inferential conditionals. The full Aristotelian Square of Opposition does not work with **truth assertions** about C-conditionals because sub-contrariety fails, although the contrariety and implication relations are present:

A		E
(∀x)T(Px ⇒ Qx)	<——Contraries——>	**(∀x)T(Px ⇒∼Qx)**
(∀x)(TPx & TQx)		(∀x)(TPx & FQx)
|	Con- ies	|
|	tra- or-	|
Entails	dict-	Entails
|	tra- or-	|
↓	Con- ies	↓
(∃x)T(Px ⇒ Qx)		**(∃x)T(Px ⇒∼Qx)**
(∃x)(TPx & TQ)	< ——?????——>	(∃x)(TPx & FQx)
I	**O**	

The I and O propositions are neither contraries nor sub-contraries since they can be true together or false together. E.g., in a domain of two, {a,b}:

(∃x)T(Px ⇒ Qx)		**(∃x)T(Px ⇒ ∼Qx)**
(T(Pa ⇒ Qa) v T(Pb ⇒ Qb))		(T(Pa ⇒ ∼Qa) v T(Pb ⇒ ∼Qb))
Syn		Syn
((TPa & TQa) v (TPb & TQb))		((TPa & FQa) v (TPb & FQb))
TT F FF T TT T TT	<— BOTH TRUE —>	TT T TF T TT F FT
FF F TT F FF F FF	<—BOTH FALSE—>	FF F FT F FF F TF

But the Aristotelian Square of Opposition does hold in A-logic for inferential conditionals, provided "Some P's are Q's" means "There is some x such that if Px is true then Qx is true", as the next square of opposition shows.

27 For the inconsistencies see R.B.Angell, "Truth-functional Conditinals vs.Traditionally Syllogistic" in *MIND,* Vol XCV, No. 378 (April 1986) 210-223. The conditional logic offered as an alternative system there is not that of A-logic since it does not have a T-operator and takes transposition laws to be universal. But it treats 'If P then not-P' as inconsistent and brings out the other inconsistencies referred to above.

A		E
Valid: **(∀x)(TPx ⇒ TQx)**	<—Contraries—>	Valid: **(∀x)(TPx ⇒ FQx)**
|	Con- ies	|
|	tra- or-	|
Entails	dict-	Entails
|	tra- or-	|
↓	Con- ies	↓
(∃x)(TPx ⇒ TQx)	<—Sub-contraries—>	**(∃x)(TPx ⇒ FQx)**
I		0

The bottom two can be true together but are never false together. Whenever one is false, the other has to be true. They are sub-contraries. In a domain of two they can never be false together as the truth tables below show.

The two truth-tables below show how, in a domain of two entities, *a* and *b*, the two **I** and **O** T-wffs are sub-contraries because they can not both be false (although they can both be true).In larger domains the impossibility of both being false remains. For in every Boolean expansion of these inferential conditionals in any given domain the case in which all disjuncts are false (so that the whole disjunctive quantification is false) will the same case in which all disjuncts in the other disjunctive quantification will be true, so the whole quantification will be true.

(∃x)(TPx ⇒ TQx)								**(∃x)(TPx ⇒~ TQx)**						
((TPa	⇒	TQa)	v	(TPb	⇒	TQb))		((TPa	⇒	FQa)	v	(TPb	⇒	FQb))
T	T	T	**T**	T	T	T	↔	T	F	F	**F**	T	F	F
F	0	T	T	T	T	T		F	0	F	0	T	F	F
T	F	F	T	T	T	T		T	T	T	T	T	F	F
F	0	F	T	T	T	T		F	0	T	0	T	F	F
T	T	T	T	F	0	T		T	F	F	0	F	0	F
F	0	T	0	F	0	T		F	0	F	0	F	0	F
T	F	F	0	F	0	T		T	T	T	T	F	0	F
F	0	F	0	F	0	T		F	0	T	0	F	0	F
T	T	T	T	T	F	F		T	F	F	T	T	T	T
F	0	T	0	T	F	F		F	0	F	T	T	T	T
T	F	F	**F**	T	F	F	↔	T	T	T	**T**	T	T	T
F	0	F	0	T	F	F		F	0	T	T	T	T	T
T	T	T	0	F	0	F		T	F	F	0	F	0	T
F	0	T	0	F	0	F		F	0	F	0	F	0	T
T	F	F	T	F	0	F		T	T	T	T	F	0	T
F	0	F	0	F	0	F		F	0	T	0	F	0	T

But that "Some P's are Q's" should be taken to mean '(∃x)(TPx ⇒ TQx)' is counter to currently prevailing notions. A case can be made in its favor, but we will not be make it here.

Chapter 9

Inductive Logic—C-conditionals and Factual Truth

Differences between the *a priori* sciences and mathematics and logic and *a posteriori* empirical sciences have long been recognized. Methods of determining *logical* or mathematical facts and rules of inference differ from the methods that common sense and science use to determine facts and arrive at generalizations or probabilities about the actual world. Nevertheless, everyone agrees that logic can be, and should be, *used* both in the discovery of laws of nature, in the development of theories about the actual world, and in the use and application of these laws in predicting, retrodicting or controlling natural events.

The question addressed in this chapter is just *how* the facts and principles of logic are used to arrive at facts, generalities and probabilities about the actual world. This is not posed as a question for historians, psychologists, or the sociology of science, but as a question for logicians. Analytic truth-logic is a theory of logic which purports to present the basic facts and principles of logic. How well does it account for the ways in which intelligent people and natural scientists reason in moving from observations to reliable general laws, or from general laws and observations to sound predictions, retrodictions and plans to control events? The same questions should be asked about other systems of logic, including M-logic, and the answers for the different systems should be compared and evaluated. In this chapter we show how analytic truth-logic answers those questions, and compare it at various points with M-logic.

Truths about facts in any field of reference other than logic, including any portion of the actual, natural world, are not the business of logic. The business of logic is to help provide reliable *principles of inference,* and to show how to establish such principles in special fields. Principles of inference are not statements of fact; they are generalized conditional statements or predicates. It is not the *truth* of these principles that logic can help to establish; it is only their credibility or *validity* as rules of inference. The discipline of A-logic helps in arriving at or discovering empirically valid principles, in providing the criteria and grounds for accepting them as empirically valid, and in schematizing the forms of arguments by which truth asssertions can be validly be deduced from other truth-claims as premisses.

The first step is to make the distinction between assertions that a conditional expression is "empirically *valid*" and assertions that it is *true de re*.

9.1 "Empirical Validity" vs. Empirical Truth

The distinction between asserting that a conditional is true and holding that it is valid is central in A-logic's analysis of inductive logic. As we mentioned before, there is confusion in ordinary language about ascribing truth to a conditional. On the one hand what makes a conditional true, is the fact that the consequent is true when the antecedent is true; in other words both are true together. The difference between asserting that a conditional is true, and asserting that it is valid, is the difference between finding an inferential conditional true in fact, and finding it credible and "valid", but not necessarily true. We need a word, different than 'true', which can be used to preserve and sharpen this distinction. I will use the predicate 'is E-valid' (for "is empirically valid") for this purpose.[1]

One of the properties of a logically valid conditional in analytic truth-logic is that both

1) it is *logically possible* for it to be true,
and 2) it is *not logically possible* for it to be false
(though it is possible for it to be neither true nor false, hence not-true).

In the logic of the C-conditional, this applies to contrary-to-fact conditionals and subjunctive conditionals as well as to indicative conditionals. Indeed it applies to mere predicates; namely, to valid conditional predicates which are never, as mere predicates, either true or false.

In common sense and the natural sciences there are many contingent conditionals of the form "if P is true, then Q is true" which people claim to know and/or believe to have

1) some or many instances which are true *in fact*
and 2) no instances which are false *in fact*
(though in many instances, it is neither true nor false, hence not-true).

When an expression of this form meets these two conditions and it is believed that all findings of truth or non-falsehood offered in its support are based ultimately on sensory experiences, people often say the conditional is "true" in fact. But they also may say it is credible, or believable and/or useful for predictions, etc. Occasionally they may say such a conditional is "valid" in a sense which fits somewhere within ordinary usage. We want to separate truth from validity in this sense.

Although other terms may be closer to ordinary usage, I use "E-valid" or "Empirically valid" to emphasize the feature common to both the claim of *logically validity* for a C-conditional in truth-logic and the claim that a conditional is *sound empirically*, namely that both involve generality and are held to have true instances but no false instances. The broad common usage of the word 'valid' is flexible enough to allow 'empirical validity' as a sub-species of 'validity' along with other kinds, including logical validity.[2] In this chapter, however, we want to assign a much more precise meaning to "E-valid"; one which

---

1. The term "empirically valid" was used before in Section 8.232333.

2. The word 'valid' is given a very strict and precise meaning in theories of formal logic, including A-logic and M-logic. But in ordinary language it has a broader variety of meanings and is used much less rigorously. The entry in the Oxford English Dictionary reads:

> **"Valid...1** Good or adequate in law; legally binding or efficacious, **2.** Of arguments, assertions,etc**.:** Well founded and applicable; sound and to the point; against which no objection can be fairly brought. **3.** Of things: strong, powerful, Now *arch.* 1656. **4.** Of persons: Sound or robust in body; possessed of health and strength."

'Validity' in formal logics, and even "E-valid", falls under the second meaning given above as do the definitions in M-logic and A-logic. Some more recent dictionaries cite M-logic's definition of "valid".

focuses on the role of logic in both arriving at sound empirical generalizations and in solving problems by using them.

Obviously E-validity is not "*logical* validity" for it is logically possible for any contingent conditional to be false, whereas a logically valid conditional can not be false. And we do not mean "legally valid" or various other dictionary meanings. The concept we want to associate with "E-valid" is explained below. The choice of words is somewhat arbitrary and not important if the concept is the clear. But some word is needed and we have chosen "E-valid" to do the job.

To say a conditional is E-valid is a) to say that it is contingent, and b) to assert that as a matter of fact, it is true in some cases, c) it is never false in fact, and d) that both the truth-claim and the non-falsehood claim are grounded ultimately on actual empirical observations.

E-validity applies only to contingent conditionals, though not all of them are E-valid. Contingent conditionals are ones a) that are not inconsistent, and are such that b) the antecedent neither contains nor implies the consequent (thus they are never logically valid) and c) neither the antecedent not the consequent are tautologous. If an expression is a contingent conditional it is *logically* possible for both the antecedent, the consequent, and the conditional as a whole to be either true or false. This does not exclude all inconsistent or tautologous *components* from antecedent or consequent (e.g., [((~PvP) & Q) ⇒ ((~R&R) v S)] can be a contingent conditional wff) but, like logically A-valid conditionals in truth-logic, the conditionals must not as a whole be either inconconsistent or tautologous.

In relating logical validity to truth-tables we drew the distinction between using truth-tables to determine whether a particular complex sentence was true, and using a truth-table to determine whether a sentence was tautologous, or capable of being valid. The difference was between using one line of the truth-table to determine whether a particular statement was true or false, compared with surveying all possible cases in the truth-table as a whole, to see whether it was true in all cases (for tautology), or was false in none but true in some (a necessary condition of validity), etc. The difference between the *truth* and the *E-validity* of a conditional is somewhat similar. Factual truth is ascribed to a particular application of a predicate; E-validity refers to the set of all actual applications in the domain of reference.

People sometimes say that a conditional is true when either the antecedent or the consequent is not true. "If Roosevelt had drunk a cup of arsenic, he would have died" may evoke as a response, "That's *true*, but he didn't drink a cup of arsenic ever." Calling this conditional '*true*' is unfortunate; it confuses verifying, or establishing that a conditional is true, with accepting it as valid. Instead we should say "That's a *valid* (E-valid) statement, but he didn't drink a cup of arsenic."[3] In fact, according to A-logic, this conditional is *not true*; it is not true and not false because the antecedent was never fulfilled.

The credibility, or believability, or E-validity, of this conditional is based on a belief that there are many cases in which it is true that a person drank arsenic and died shortly thereafter (the antecedent and consequent were true together, making the conditional true), and the belief that there are no known cases where people drank a cup of arsenic and did not die shortly thereafter. (I.e., there are no cases where the antecedent was true and consequent was false). This justification appeals to past experience most of which we have heard second- or third-hand. The predicate, "If <1) is a human and <1> drinks a cup of arsenic, then <1> dies shortly thereafter" is an empirically valid predicate. It is *generally* E-valid. The attribution of this predicate to Roosevelt, or anyone else,—or to *all* humans in "All humans who drink a cup of arsenic die shortly thereafter"—is simply a case of instantiating a general predicate which we accept as *empirically valid*. If this predicate is *true* at all, it is only true of those cases in which

3. According to M-logic the "truth" of this conditional follows logically from the fact that Roosevelt didn't drink a cup of arsenic, or alternatively, from the fact that he died. This analysis exacerbates and reinforces the common confusion.

individuals actually drink a cup of arsenic and die therefrom. Its application to Roosevelt is neither true nor false, but it *is* E-valid because the predicate is E-valid.

The value of this analysis lies in its specific applications which are spelled out below. Our main question is: how is the empirical validity of C-conditionals established, supported, or confirmed by true descriptions of simple empirical matters of fact? We look first at the way the truth of conditionals can be derived from true descriptions of fact, and then at the grounds for asserting the E-validity of a conditional predicate or statement.

9.2 Truth, Falsehood, Non-truth and Non-falsehood of Conditionals

Determinations that a conditional has one or more *true* instances is a necessary step in arriving at a judgment of E-validity, although such a determination by itself is never sufficient to establish E-validity. Nevertheless, the basic initial step is from two or more simple assertions of truth or falsehood to an assertion that, based on them, a conditional is true, or false, or neither.

9.21 From True Atomic Sentences

At this moment[4] I select the room I am in and all of the objects in it as an objective field of reference. It serves as a microcosm for the actual world. There are more objects here than I can count: the computer at which I am working, hundreds of books in the book-cases around me, the chairs, the desk, my glasses, the sofa, the lamps, the rugs, my file cabinets, my clothes, my fingers, my nose, each of the pages in each book, each letter on each pages, each of sheet of paper in my files, the paperclips, the pens and pencils, each stitch in the threads of the curtains on my window… etc.

There is no limit to the number of simple sentences that can be formed in which the subject terms designate objects which exist in this room now, and the predicates ascribe properties or relations which are or are not the case in this room as I write. These basic, simple, "atomic" sentences are all negation-free and POS. Such sentences can be written and expressed without knowing which ones are true and which are false.

Having written such sentences, I can look around and determine by what I see in my room, which ones are true and which ones are false.

Having determined which ones are true, I can conjoin various pairs of non-synonymous true statements. If S1 and S2 are each true separately, I infer that their conjunction is true. That is, from T(S1) and T(S2) I infer T(S1 & S2). This is sanctioned by the rule of Adjunction in Truth-logic, which is supported by Axiom 7-03 of truth-logic, [(TP & TQ) Syn T(P&Q)].

Next, whenever we find, in a search for truth, that some sentence of the form (S1 & S2) is true, we can infer that (S1 $\Rightarrow$ S2) is true. I.e., from T(S1 & S2) we can infer T(S1 $\Rightarrow$ S2). This inference is warranted by Axiom 8-01 of analytic truth-logic, $\models$ [(T(P & Q) Syn T(P $\Rightarrow$ Q)].

These are the steps by which we pass from the *truth* of a pair of (non-synonymous) atomic statements to the *truth* of a particular conditional statement about those two statements. As particular truth-claims, these statements are not E-valid. Determining that a conditional is true is a one-time thing; it is a matter of finding that a particular application of the conditional predicate is true. That determination is complete when you find that both the antecedent and consequent are each true. To establish E-validity, we need to consider much more than one application of the sentence's predicate in order to give meaning to the concept that it has true instances but no false instances.

4. It is December 4th 1999, between 11 and 12 a.m. EST.

The possible conjunctions of two or more atomic truths yields many more true conditionals than we want or need in developing rules of inference. There are thousands of possible true atomic statements that I could make about the things in my room now. Provided any two statements are both true and neither entails the other, no matter how unrelated or irrelevant they are, the conjunction of these two statements entails that the conditional formed by taking one as antecedent and the other as consequent, is factually true. Thus for example, let S1 be "my glasses are on my nose", S2 be "the sofa is colored beige" and S3 be "my glasses are on the sofa". At this time S3 is false, but

since it is **true** that my glasses are on my nose at this time, TPab

and it is **true** that the sofa is colored beige, TQc

It follows by Adjunction, Ax7-03 and Axiom 6-02 (TPab & TQc), <u>T(Pab & Qc)</u>, <u>T(Qc & Pab)</u>

.: by Ax.8.01, It is **true** that (**if** my glasses are on my nose, **then** the sofa is beige) T(Pab ⇒ Qc)

.: by Ax.8.01, It is **true** that (**if** the sofa is beige, **then** my glasses are on my nose) T(Qc ⇒ Pab)

Innumerable results of this sort follow from the axioms of analytic truth-logic. This result is not peculiar to A-logic. M-logic is even more lavish in deriving assertions of the truth of conditional statements (TF-conditionals) from similar premisses:

By A-logic: (TP & TQ) <u>A-entails</u>	**and By M-logic**:(TP & TQ) <u>M-implies:</u>		
T(P ⇒ Q),	T(P ⊃ Q),	T(~P ⊃ Q),	T(~P ⊃ ~Q)
T(Q ⇒ P),	T(Q ⊃ P),	T(~Q ⊃ P),	T(~Q ⊃ ~P)
T(Q ⇔ P)	T(P ≡ Q),	T(~Q ≡ ~P)	

It may seem that this enormous proliferation of true conditionals is untenable. This is particularly the case if pronouncing a conditional *true* is confused with pronouncing it E-valid. For clearly the conditionals in the examples just given are not E-valid, even if they are true.

When you say that a conditional is *true*, you are not making a conditional statement, and you are not asserting its E-validity. You are making a categorical assertion of truth. According to A-logic you are saying nothing more or less than that the antecedent is true and the consequent is true. That is, to say 'If S1 then S2' is true, is to say nothing more than that both S1 and S2 are true. If you find that sentences S1 and S2 are each true, and then say '*therefore*, 'If S1 then S2' is true', you are still doing no more or less than saying that the antecedent and consequent are both true. (According to M-logic you *are* saying something more when you say a TF-conditional is true—namely, that either the antecedent is false or the consequent is true.) If you say a conditional is false, you say no more or less than that the antecedent is true and the consequent is false. These are what Axioms 8-01 and 8-02 say in asserting the referential Synonymy of their components. That a conditional is true, or false, does not describe its unique character as a conditional; it does not explain the nature of conditionality, and is not equivalent to establishing its E-validity.

A great value of the concept of truth lies in the concept that sentences, if once true, are always true. To satisfy this concept, each of the sentences above must be understood implicitly as recognizing ***when*** these statements are true or false, ***where***, geographically, the room is, and in the case of the first two statement ***who*** is the person whose glasses are being talked about. For example the sentence S3, "My glasses are on the sofa" should be understood as saying 'The glasses of Richard B.Angell between 11 and 12 a.m. on December 4, 1999 are (were) located on the sofa in his study at 150 Kendal Drive, Kennett Square, PA, USA." This sentence, which Quine would call an "eternal sentence", is in fact false, and will remain unchangeably false in the future. The truth is that my glasses are and have been on my nose throughout the current period. Thus the truth of sentence S1, "My glasses are on my nose" properly understood as expanded to specify the time and place of the writing is now and forevermore true.

Understood and expanded in a similar way, all possible simple atomic sentences can be understood to be true or not true once and for always. The question of how I *know* these statements are true, is a question for epistemology (theory of knowledge) and is not one that is answered or dealt with in logic. But from a common sense point of view everyone knows the answer: I or anyone who was not blind and was in the room at this time could *see*, by looking at the sofa, and at me and my glasses, that S1 was true and S3 was false, and so on with all other possible sentences about objects in this room. These would all be simple, empirically true (or false) statements.

When I finish writing this section, I will take off my glasses and put them on the sofa. At that time it *might* be said that S1 will be false and S3 will be true, and thus that T(Qc $\Rightarrow$ Pab) should be replaced by its contrary, F(Qc $\Rightarrow$ Pab). But this apparent switch in truth-value, is only the case if I have not interpreted the sentences as "eternal" sentences. If the difference in time is recognized, then Pab at time 1, ("My glasses are on my nose **from 11 to 12 a**.**m**. on December 4, 1999") is a different statement from Pab at time 2 ("My glasses are on my nose **from 12 to 1 p**.**m**. on December 4, 1999") and properly understood, the conditionals are do not conflict in truth-value, because they are talking about states of affairs at different times. This may be expressed symbolically by T(Qc & Pab$\mathbf{t_1}$) and F(Qc & Pab$\mathbf{t_2}$), which are not logical contraries. What remains constant is the predicate (Q<1> & P<2,3,4>) which has the two instantiations '(Qc & Pab$\mathbf{t_1}$)' and '(Qc & Pab$\mathbf{t_2}$)', one of which is true and the other of which is false. The false one precludes the *E-validity* of the predicate (Q<1> & P<2,3,4>) and of instantiations or quantifications of it.

There are other sentences which I can write which are "eternal sentences", but my grounds for saying I know or believe they are true are much more difficult to explain. For example, if I open my copy of Einstein's book, *Relativity*, I read on page vii, "Albert Einstein is the son of German-Jewish parents. He was born in 1879 in the town of Ulm, Wurtemberg, Germany." I accept this as true statements about an individual I never met but have read much about. Towards the end of the same book I read a report of photgraphs taken of seven stars during the eclipse of the sun of May 29, 1919. They are compared with photographs of the same stars taken a few months earlier. I accept this report as a true account of what was found with respect to the relevant photographs. There are innumerable statements which I read in the history books on my bookshelves, and believe to be true and part of human knowledge. There are many more statements about the location of rivers and cities, etc., that I will believe to be true by examining the World Atlas behind me.

The sentences referred to in the preceding paragraph are about events and objects far removed in space and time from my present small room and the time I write these words. The sensory data that ***I*** have is simply the words or maps in my books. But I assume that all such sentences are based on direct experiences of various people many times removed from me. The problem of how I do, or should, arrive at truth-claims is a matter for epistemology, not logic. For our present purpose, it is sufficient to assume that various descriptive sentences are asserted to be true or false based on direct human experiences, and that true and false conditionals can inferred from these truth-assertions. This is presupposed in the question, How can one move from accepting conditionals as true, to a judgement of empirical validity?

9.22 Differences in Truth-claims about Conditionals

Differences between asserting that a C-conditional statement is true, false, not true or not false, and not doing so, can be clarified by reference to truth-table principles.

For truth-logic four kinds of simple, unquantified conditional statements may occur:

a) Inferential C-conditionals in which no T-operators occur at all, E.g., [Pa ⇒ Qb]
b) Inferential C-conditionals with T-wffs as antecedent and consequent but with no T-operator prefixed to the conditional as a whole. E.g., [(TPa ⇒ TQb), (FPa ⇒ TQb),(∼TQb ⇒ ∼FPa), …
c) Truth-assertions about C-conditionals: withT-operators prefixed to the whole Inferential conditional. E.g., [T(Pa ⇒ Qb), ∼F(∼Pa ⇒ Qb), F(Qb ⇒ ∼Pa), … etc.
d) Truth-assertions about inferential C-conditionals with T-wffs as components. E.g., T(TPa ⇒ TQb), ∼F(FPa ⇒ TQb), F(∼TQb ⇒ ∼TPa),… etc.

By the principles of the trivalent truth-tables, an unquantified particular inferential C-conditional of the form [Pa ⇒ Qb] will be true in just one out of nine possible cases, false in just one out of nine, and neither-true-nor-false in the remaining seven cases in which either the antecedent is not true, or the consequent is neither true nor false. In six of these seven cases, the antecedent is not true (F or 0); the seventh case (in the second row) the consequent is neither true nor false. C-conditionals of the form [TPa ⇒ TQb] will be true in just one out of nine possible cases, false in two out of nine, and neither-true-nor-false in the remaining six cases.

When a T-operator is prefixed to these kinds of conditional statement as in c) or d) part of the meaning of the conditional is lost; at the same time the truth-operator adds a different kind of meaning to the assertion as a whole. Different T-operators are different in this respect.

Let us consider first how the meaning of conditionals with different kinds of T-operators prefixed, differs when components are, or are not, T-wffs. When 'T' is prefixed to C-conditionals in c) and d), they have the same truth-tables. T(Pa ⇒ Qb) and T(TPa ⇒ TQb) are synonymous and their denials are also synonymous.[5] But when F or ∼F is prefixed, the wffs F(Pa ⇒ Qb) and F(TPa ⇒ TQb) are not synonymous, and their truth-tables are different. The difference is found in the second rows of their truth-tables. The principle of the second row of the truth-table for F(Pa ⇒ Qb) says: (TPa & 0Qb) ⇒ F(Pa ⇒ Qb)), i.e., "If it is true that Pa and it is neither true nor false that Qb, then it is **False** that it is **False** that (if Pa then Qb)," whereas the principle of the second row of the truth-table for F(TPa ⇒ TQb) says: (TPa & 0Qb) ⇒ T(F(TPa ⇒ TQb)), i.e., "If it is true that Pa and it is neither true nor false that Qb, then it is **True** that it is **False** that if it is true that Pa then it is true that Qb."

	F(Pa ⇒ Qb)	F(TPa ⇒ TQb)
Row 1	F(0 0 0)	F(F0 0 F 0)
Row 2	**F**(T 0 0)	**T**(TT F F 0)
	. . . .	. . . . . .

The principle of the second row in the truth-table of F(TPa ⇒ TQb) makes possible a general logic which can sift out irrelevant cases. If Qb is an imperative or directive which is neither true nor false, then it is **true** that it is **false** to say that If Pa is true then Qb is true. Conditioned imperatives or directives which occur in the universe of discourse are not made true or false by the truth or falsity of components. If we are trying to find out whether an instance of "If it is true that <1> committed murder, then it is true that <1> was punished" is true, it would be false to say "If it is true that Al committed murder, then punish Al!' is true." We will return to this point shortly, in discussing how we arrive at the empirical validity of a inferential conditional of the form (∀x)(If TPx then TQx). On the other hand, to say "'If A committed murder, then punish Al!' is false", is false because a conditioned imperative is neither true nor false,though it may be accepted or rejected as useful, expedient, or morally right or wrong.

5. T8-17. [T(P ⇒ Q) Syn T(TP ⇒ TQ)

When 'It is true that…' Is prefixed to a conditional one feature of the conditional's meaning that is lost is the asymmetric aspect of conditionality. The concept of the consequent being conditioned on the antecedent is lost, since all that counts is correspondence of the two components with facts. This is what T8-17 conveys: T[Pa => Qb] means no more or less than [TPa & TQb]. But what establishes the truth of T(Pa => Qb) also establishes the truth of T(Qb => Pa), and vice versa. Thus the asymmetry of "If Pa then Qb" is lost when either 'T' or '~T' is prefixed to it. This entails asserting that many trivial unquantified C-conditionals are true (but less than the TF-conditional allows since it is also true whenever the antecedent is false).

But when 'it is False that…' is prefixed to a C-conditional, its asymmetry is preserved. By Ax.8-2 F(Pa => Qb) is synonymous with T(Pa & ~Qb) and F(Qb => Pa) is synonymous with T(Qb & ~Pa). But T(Qb & ~Pa) is certainly not synonymous with T(Pa & ~Qb), therefore F(Pa => Qb) and F(Qb => Pa) are not synonymous. The prefixing negation sign to 'F' does not affect the presence of absence of asymmetry; ~F(Pa => Qb) also preserves asymmetry.

TABLE 9-1

(Pa => Qb)			T(Pa => Qb)		F(Pa => Qb)		0(Pa => Qb) (~T(Pa => Qb) & ~F(Pa => Qb))			
0	0	0	F	0	F	0	TF	0	T TF	0
T	0	0	F	0	F	0	TF	0	T TF	0
F	0	0	F	0	F	0	TF	0	T TF	0
0	0	T	F	0	F	0	TF	0	T TF	0
T	**T**	T	**T**	T	**F**	T	**FT**	T	**F** TF	T
F	0	T	F	0	F	0	TF	0	T TF	0
0	0	F	F	0	F	0	TF	0	T TF	0
T	**F**	F	**F**	F	**T**	F	TF	F	**F** **F**T	F
F	0	F	F	0	F	0	TF	0	T TF	0

(TPa => TQb)			T(TPa => TQb)		F(TPa => TQb)		(~T(TPa => TQb) & ~F(TPa => TQb))			
F0	0	F0	F	0	F	0	TF	0	T TF	0
TT	**F**	F0	F	**F**	**T**	**F**	TF	**F**	**F** FT	**F**
FF	0	F0	F	0	F	0	TF	0	T TF	0
F0	0	TT	F	0	F	0	TF	0	T TF	0
TT	**T**	TT	**T**	**T**	F	**T**	**FT**	**T**	**F** TF	**T**
FF	0	TT	F	0	F	0	TF	0	T TF	0
F0	0	FF	F	0	F	0	TF	0	T TF	0
TT	**F**	FF	F	**F**	**T**	**F**	TF	**F**	**F** FT	**F**
FF	0	FF	F	0	F	0	TF	0	T TF	0

(Pa => Qb)	T(Pa => Qb)	F(Pa => Qb)	0(Pa => Qb) ((~T(Pa => Qb) & ~F(Pa => Qb))
0 0 0	F F F	F F F	T T T TT T T T T
0 **T** **F**	F **T** F	F F **T**	T **F** T **FF** T T **F** **T**
0 0 0	F F F	F F F	T T T TT T T T T

(TPa => TQb)	T(TPa => TQb)	F(TPa => TQb)	0(TPa => TQb) (~T(TPa => TQb) & ~F(TPa => TQb))
0 0 0	F F F	F F F	T T T TTT T T T
FT F	F **T** F	**T** F **T**	T **F** T **FFF** **F** T **F**
0 0 0	F F F	F F F	T T T TTT T T T

Thus '~T(P ⇒ Q)' and 'T(P ⇒ Q)' both lose the asymmetry of the C-conditional, and '~F(P ⇒ Q)' and 'F(P ⇒ Q)' both retain it if P and Q are not synonymous.

In asserting T(P ⇒ Q) the uniqueness of the condition which make (P ⇒ Q) **false** is lost. Instead of there being just one truth condition (represented by one row in the truth-table) which makes 'T(P ⇒ Q)' false, there are eight different rows, representing different situations which can make it false, whereas there is just one such row for (P ⇒ Q). The same is true for F(P ⇒ Q). The uniqueness of the condition which makes (P ⇒ Q) **false** is lost. Instead of there being just one truth condition, there are eight different rows, representing different situations which can make F(P ⇒ Q) false. Thus both the assertion that a conditional is true, and the assertion that it is false, can be false for many different reasons.

When the T-operator is negative, as in '~T' or '~F' or '0', the proliferation of false cases is converted into proliferation of true cases. But '~F(P ⇒ Q)' and ~T(P ⇒ Q) can be true in many different circumstances, only one of which is the case when '(P ⇒ Q)' is true. But both are false in only one case.

So much for the ways in which prefixing a T-operator to a C-conditional leads to a loss of the meaning—a loss of its asymmetry, or a loss of a unique condition for either the falsity of the whole, or the truth of the whole. In spite of these complications, these distinctions will be useful in clarifying how quantified C-conditionals are determined to be 'empirically valid' or not, by references to findings that particular conditionals are T, F or 0.[6]

Although prefixing a T-operator to a C-conditional loses key elements in the meaning of the C-conditional, it adds something to the meaning of the compound statement. The word 'true' entails the idea of facts in a field of reference of which a description could be true or not. Facts are conceived as being what they are regardless of ideas or meanings of the antecedent and consequent. The idea of an objective field of reference is not necessarily part of the idea of conditionality taken by itself. Prefixing 'It is true that…' and 'It is false that…' to a conditional, or to its components, injects the idea of correspondence or non-correspondence between the meanings of the conditional's components and facts in the field of reference. Whether a correspondence exists in fact can only be determined by the extra-logical activity of looking at the facts.

For these reasons statements of the forms, T(P ⇒ Q), F(P ⇒ Q), ~T(P ⇒ Q), ~F(P ⇒ Q), T(TP ⇒ TQ), F(TP ⇒ TQ), ~T(TP ⇒ TQ) and ~F(TP ⇒ TQ) which assert truth, falsity, non-truth or non-falsity of a conditional as a whole, mean both more and less than conditionals that do not lie in the scope of a T-operator such as [Pa ⇒ Qa], [TPa ⇒ TQb] or [~FPa ⇒ ~TQb].

9.23 From True Complex and Quantified Statements

Unquantified statements include descriptive statements built up by conjunctions, disjunctions and negation with very complex logical structures. Unquantified complex statements are found true, false or neither by the usual truth-table rules starting with determinations of truth, falsehood or neither for the atomic components.

---

6. In M-logic, ' is True' is considered interchangeable with ' is not-False'. The preceding remarks show that the semantics of A-logic is radically different. Because it has defined its conditional by what makes a conditional false, M-logic's TF-conditional preserves the asymmetric character of conditionality. When "it is not false that…" is prefixed to a TF-conditional the asymmetry is preserved, but the dependence of the truth of the consequent on the truth of both the antecedent and the consequent is given up, and many more grounds for its truth are injected.

Since a predicate can be very complex, a particular application of a predicate can be very complex, and the resulting true conditionals can be very complex. For example, looking at the first observation at time t1, from which Boyle arrived at Boyle's Law (see Section 9.333 for the discussion of Boyle's Law) we have a complex of observations at t1 in his experiment which are symbolized as a conjunction of true atomic sentences synbolized as follows:

T(G<a>) & **T**(O< t1>) & **T**(V< $\underline{a}$,12,t1>) & **T**(P< $\underline{a}$,29.5,t1>) & **T**(12 x 29.5 = 350($\pm$4)>)).

From this datum many true conditionals follow logically. Using the methods just described with re-ordering, we can derive the truth of many conditionals with '⇒' between sub-sets of the conjuncts in this conjunction of five atomic sentences:[7]

e.g.,**T** ((G<a> **⇒** (O<t1> & V<$\underline{a}$,12,t1> & P<$\underline{a}$,29.5,t1>) & (12 x 29.5 = 350($\pm$4) at t1)),
or **T** ((G<a> & O<t1>) **⇒** (V<$\underline{a}$,12,t1> & P<$\underline{a}$,29.5,t1>) & (12 x 29.5 = 350($\pm$4) at t1)),
or **T** ((O<t1>) **⇒** ((G<a>&V<$\underline{a}$,12,t1> & P<$\underline{a}$,29.5,t1>) & (12 x 29.5 = 350($\pm$4) at t1)),... etc.

However, for purposes of arriving at his law, Boyle in effect made use of just one, namely,

T(G<a> & O<t1> & V<$\underline{a}$,12,t1> & P<$\underline{a}$,29.5,t1>) ⇒ (12 x 29.5 = 350($\pm$4) at t1)),

which, with 24 other observations, provided 25 instances supporting the E-validity of his law.

Thus again we have an embarrassment of riches. The mere conjunction of atomic truths yields many more true conditionals than we want or need. The question we will ask is how we select just those, from the many possibilities, which will yield E-valid conditional statements.

When we assert the *truth* of a quantified generalized conditional, [T(∀x)(Px ⇒ Qx)],we are also asserting the truth of a complex particular conditional. For quantified conditionals abbreviate, in whatever domain is taken as its field of reference, a conjunction of all of the particular conditionals with its predicate in that domain:

E.g., |= [T(∀x)(Px ⇒ Qx) Syn T((Pa_1 ⇒ Qa_1) & (Pa_2 ⇒ Qa_2) & ... & (Pa_n ⇒ Qa_n))];
by Ax.7-03 |= [T(∀x)(Px ⇒ Qx) Syn (T(Pa_1 ⇒ Qa_1) & T(Pa_2 ⇒ Qa_2)& ... & T(Pa_n ⇒ Qa_n))];
by Ax.8-01 |= [T(∀x)(Px & Qx) Syn (T(Pa_1 & Qa_1) & T(Pa_2 & Qa_2)& ... & T(Pa_n & Qa_n))];
by Ax.7-03 |= [T(∀x)(Px & Qx) Syn T(Pa_1 & Qa_1 & Pa_2 & Qa_2 & ... & Pa_n & Qa_n)].
by Ax.7-03 |= [T(∀x)(Px & Qx) Syn (TPa_1 & TQa_1 & TPa_2 & TQa_2 & ... & TPa_n & TQa_n)].

But again, the data in the last line which the truth of the quantification purportedly abbreviates permits many other truths-claims about conditionals than those derivable from the quantified wff or statement.

---

7. Using all five different true-claims, (TP1 & TP2 & TP3 & TP4 & TP5), there are 30 non-synonymous binary conditionals which , after re-ordering, follow logically from the set including, e.g.,
T(P1 ⇒ (P2 & P3 & P4 & P5)), T(P2 ⇒ (P1 & P3 & P4 & P5)),
T((P1 & P2) ⇒ (P3 & P4 & P5)), T((P2 & P3) ⇒ (P1 & & P4 & P5)),
T((P1 & P2 & P3) ⇒ P4 & P5)), T((P2 & P3 & P4) ⇒ (P1 & P5)),
T((P1 & P2 & P3 & P4) ⇒ P5), T((P2 & P3 & P4 & P5) ⇒ P1) etc. ...,
plus many cojunctions of smaller conditionals, such as (T(P1 ⇒ P2) & T(P3 ⇒ P4) &T(P5 ⇒ P3))), ..., etc.

Which pairs of the atomic wffs on the right hand side of the last line should be conjoined in order to derive a true conditionals? If the true atomic statements include TPa_1, TQa_1, TPa_2, TQa_2 … TPa_n, TQa_n, we can conclude $(T(Pa_1 \Rightarrow Pa_n) \ \& \ T(Qa_n \Rightarrow Pa_2))$ hence, $(T((Pa_1 \Rightarrow Pa_n) \ \& \ (Qa_n \Rightarrow Pa_2))$ and so on, which say no more than $T(Pa_1 \ \& \ Pa_2 \ \& \ Pa_n \ \& \ Qa_n))$. From the same data which support T(∀x)(Px & Qx) assertions of truth for many many other conditionals can be derived. What distinguishes the first and second lines from the potential chaos of the last line, is the concept that a quantified conditional is an expression which abbreviates the application of the same predicate—in this case '(P<1> ⇒ Q<1>)' or '(P<1> & Q<1>)'—to all members of the domain.

9.24 Truth and Falsity of Quantified Conditionals

The great advantage of mathematical logic over Aristotle's theory lay in its analysis of general statements—statements about all, some or no members of some domain. In the Aristotelian analysis of "All P's are Q's", 'P' and 'Q' represent general nouns connected by the copula 'are'. In mathematical logic these generalizations are re-interpreted as quantified conditionals—conditionals with two predicates, each applying to a variable which is bound to a prefixed universal quantifier. Thus "All Ps' are Q's" becomes "For all x, if x is P then x is Q", or "No matter what x may be, if x is P then x is Q'. In symbols: '(∀x)(If Px then Qx)'. With multiple quantifiers and a theory of relations, this analysis made it possible to express and axiomatize mathematical relationships which could not be expressed or axiomatized in Aristotle's logic.

This basic analysis is retained in A-logic, but A-logic uses C-conditionals where M-logic had only TF-conditionals. Both '(∀x)(Px ⇒ Qx)' and '(∀x)(Px ⊃ Qx)' are well-formed formulae in A-logic, and the M-logic expressions remain with the same logical properties of tautologousness or inconsistency as they have in M-logic. But generalized TF-conditionals fail to give satisfactory accounts of induction and confirmation of general statements, causal statements, scientific laws, or statements of conditional probability. In subsequent sections we show that the C-conditional succeeds where the TF-conditional fails.

To clarify the distinction between the empirical *truth* of a quantified C-conditional, and the empirical *validity* of a C-conditional, let us investigate first how claims that "All P's are Q's" is true, or not true, or false, or not false, should be interpreted in A-logic?

The first seemingly simple answer is that " 'All ravens are black' is true " simply means

1) 'T(∀x)(Rx ⇒ Bx)' "It is true that (for all x, if x is a raven then x is black)",

or, synonymously,

2) ' (∀x)T(Rx ⇒ Bx)' "For all x, it is true that (if x is a raven then x is black)".

These expressions abbreviate a conjunction in which the predicate '(R<1> ⇒ B<1>)' is asserted to be true of each member of the field of reference.

This answer doesn't work in A-logic. In a world that has many kinds of things besides ravens, it is false if anything other than a raven is designated as the value of 'x'. Suppose there were just four things in the world—two ravens, a black pig and a green tree. Letting a and b stand for the ravens, c for the pig and d for the tree; the Boolean expansion of '(∀x)T(Rx ⇒ Bx)' is '(T(Ra ⇒ Ba) & T(Rb ⇒ Bb) & T(Rc ⇒ Bc) & T(Rd ⇒ Bd))'. Now if both ravens were black, then we would ordinarily say "All ravens are black" would be true in that world. If one or more were not black, "All ravens are black" would be false in that world. But according to A-logic, "*All* Ravens are black" can never to be true in that world (or in any world which has other things than ravens in it), because to be true, all instantiations of T(If R<1> then B<1>) must be true. Any instantiation on some a_i other than a raven would give us

F(Ra_i), hence 0(Ra_i ⇒ Ba_i), hence. ~T(Ra_i ⇒ Ba_i), hence ~T(∀x)(Rx ⇒ Bx).

T(∀x)(Rx ⇒ Bx) Syn ‘ (T(Ra ⇒ Ba) & T(Rb ⇒ Bb) & T(Rc ⇒ Bc) & T(Rd ⇒ Bd))’
F T T T T T T T T T F F F 0 T **F F** F 0 F

or putting T(R<1> & B<1>) for its synonym T(R<1> ⇒ B<1>),

T(∀x)(Rx & Bx) Syn ‘ (T(Ra & Ba) & T(Rb & Bb) & T(Rc & Bc) & T(Rd & Bd))’
F T T T T T T T T T **F** F F F T **F F** F F F

M-logic does not have this problem. With the TF-conditional the statement “All ravens are black” is true only if the predicate, ‘(<1> is a raven ⊃ <1> is black)’ is true of every object in the universe of discourse. With the TF-conditional, “All ravens are black” means “Everything is either a not a raven, or it is black”, and “All Ravens are black” turns out to be true if a) nothing is a raven, or b) everything is black, or c) there are ravens that are black and no ravens are not black. For A-logic, only the last case, E-validity, captures what we really want to say.

To untangle this mess ask, Are we trying to assert a truth about all things, or a truth only about ravens? The expression “For all x” suggests that it is about all things and M-logic satisfies this suggestion. Let us call this the “true-of-everything” thesis.

But the answer of A-logic is that ‘All ravens are black’ is about ravens only. To say it is true is only to say that ‘<1> is black’ applies to a member of the field of reference ***if*** it is a raven. The consequent is not asserted to apply to members of the domain of discourse which are not ravens. To say it is false, is to say there are members of the field that are ravens but are not black. The conditional is not made either true or false, if something other than a raven is denoted in the antecedent. This is expressed as the generalization of an inferential C-conditional,

3) (∀x)(TRx ⇒ TBx), “For all x, ***if*** it is true that x is a raven, ***then*** it is true that x is black”

This says that [(∀x)(If TRx then TBx)] correctly describes what is in a field of reference if and only if the predicate Q<1> truthfully applies to every member of the sub-set of entities to which P<1> truthfully applies.

Thus what we want to say is not captured by ‘It is *true* that (∀x)(TRx ⇒ TBx)’ i.e., by T(∀x)(TRx ⇒ TBx), but by a more complex expression. What we really mean with respect to any domain which has individuals that are not ravens in it, is that there are some things that are ravens and they are black, and that there are no things that are ravens and are not black. In other words, we believe “Some things are black ravens” is true, and “All ravens are black” is not false, i.e., “All ravens are black” is empirically valid. In this way we can make generalizations *about* different kinds of entities without holding that every generalization *is about* all things in the field of reference.

9.3 Empirical Validity of Conditionals

We now turn from *truth*-claims about conditionals to questions about the E-*validity* of inferential conditionals and how determinations of truth or falsity of conditionals play a role in establishing or confirming the acceptability of inferential conditionals. It is the acceptance of inferential conditionals as separated from particular truth or falsehood, which drives reasoning in *de re* problem solving. E-validity is never an argument about the truth of a particular statement, it always involves some degree of generality, of going beyond particulars. Generality is expressed in two ways, as a property of an abstract predicate, or of a quantified expression.

9.31 Empirically Valid Predicates

The conjunctive, or "universal", quantifier '(∀x)Px' is often read "*No matter what x may be*, x is P". This is usually construed extensionally as "a_1 is P & a_2 is P & a_3 is P & . . . & etc". But the phrase may be applied even more appropriately to an abstract predicate. *"No matter what <1> may be*, (If <1> is a man then <1> is mortal)" is a way of saying generally that every man is mortal. Indeed most everybody believes that *no matter what <1> may be* '(If <1> is a man, then <1> is mortal)' is never false and there are a very great many cases in which it is true. This is talking, not about the members of the class of men, but about the predicate, '(If <1> is a man, then <1> is mortal)'. It says that that predicate holds generally.

Such talk about this abstract property of a predicate is a kind of generalization. Instead of asserting that a *statement* of the form (∀x)((Px & Qx) ⇒ Qx) is logically valid, we can say simply that the *predicate* ((P<1> & Q<1>) ⇒ Q<1>) is logically valid, meaning that no matter what subject-terms are put for all occurrences of '1', the result will be a valid conditional. Indeed, it is the validity of the predicate which accounts for the validity of the quantified statement. If the predicate is logically valid, it will follow that both conjunctive ("universal") and disjunctive ("existential") generalizations of it will be logically valid. '(∀x)((Px & Qx) ⇒ Qx)' and '(∃x)((Px & Qx) ⇒ Qx)' abbreviate *statements* of indefinite length, depending on the number of individual objects in the intended field of reference. Dealing with predicates relieves us of the embarrassment of extensions beyond our reach. They are finite though abstract expressions which can serve as bearers of logical properties and relations.

Many conditional predicates can be asserted on good grounds to be empirically valid. We can say a predicate is empirically valid, meaning that it can be applied truly and no application will be false, without referring by quantification to all members of the domain including those which do not fulfill the antecedent. Such is the case with "(If <1> is a man then <1> is mortal)". It is even possible that the approach of treating E-validity and other logical properties and relations as properties of conditional predicates would render quantifiers and their Boolean extensions, unnecessary accoutrements of logic if developed completely. But we shall not pursue this in our present project. In this chapter we are dealing with *truth*-logic, attempting to cover the territory which M-logicians sought to cover. Only conditional *statements* can be true or false; conditional predicates can not. So we will not try at this point to develop a theory of formal logic dealing only with unsaturated predicates.

9.32 Empirically Valid Quantified Conditionals

The concept of *empirical validity* only enters the picture when we have a contingent, abstract, unsaturated conditional predicate and think of different applications of that predicate to different individual subjects. The *truth* of a C-conditional, is always an assertion that some simple or complex *statement* about a fixed set of particular entities, is true. Determining that a single conditional statement is *true* is a one-time thing; it is a matter of finding that an application of the conditional predicate to a particular set of entities is true.Assertions than an expression is E-valid are implicitly general, rather than particular, statements. But *E-validity* entails more than one instantiation of a conditional predicate,for it must give meaning to the concept that it has true instances but no false instances. It means that the expression will hold in cases beyond any particular cases we may have in mind. The grounds for asserting E-validity may be particular facts, but the import of the assertion is not limited to those facts.

9.321 Empirical Validity and Truth-determinations of Q-wffs

The predicate '<1>is E-valid', applied to a generalized inferential conditional of the form '$(\forall x)(Px \Rightarrow Qx)$', asserts that it is true that some P's are Q's, and there are no facts in the field of reference such that the antecedent is true and the consequent is false.

Df_1 'E-valid': $[\text{E-valid } (\forall x)(Px \Rightarrow Qx) \text{ Syn}_{df} (T(\exists x)(Px \Rightarrow Qx) \,\&\, {\sim}F(\forall x)(Px \Rightarrow Qx))]$

The definiens can be expressed rigorously in many logically synonymous ways. Alternative definiens include the following:

$Df_1.$	$[\text{E-valid}(\forall x)(Px \Rightarrow Qx) \text{ Syn}_{df} (\mathbf{T}(\exists x)(Px \Rightarrow Qx) \,\&\, {\sim}\mathbf{F}(\forall x)(Px \Rightarrow Qx))]$	[Initial Definition]
$Df_2.$	[" $\text{Syn}_{df} (\exists x)\mathbf{T}(Px \Rightarrow Qx) \,\&\, (\forall x){\sim}\mathbf{F}(Px \Rightarrow Qx)]$	$[Df_1$,T7-25 and T7-D26,SynSUB]
$Df_3.$	[" $\text{Syn}_{df} (\exists x)\mathbf{T}(Px \,\&\, Qx) \,\&\, {\sim}(\exists x)\mathbf{F}(Px \Rightarrow Qx)]$	$[Df_2$,T8-01,SynSUB]
$Df_4.$	[" $\text{Syn}_{df} [(\exists x)\mathbf{T}(Px \,\&\, Qx) \,\&\, {\sim}(\exists x)\mathbf{T}(Px \,\&\, {\sim}Qx)]$	$[Df_3$,T8-02,SynSUB]
$Df_5.$	[" $\text{Syn}_{df} [(\exists x)(\mathbf{T}Px \,\&\, \mathbf{T}Qx) \,\&\, {\sim}(\exists x)(\mathbf{T}Px \,\&\, \mathbf{T}{\sim}Qx)]$	$[Df_4$,T7-03 (twice),SynSUB]

Though the definiens in each of these definitions differ in logical structure, they all have the same referential meaning. For example, with 'P' for 'is a raven' and 'Q' for 'is black',

Version 1), $T(\exists x)(Px \Rightarrow Qx) \,\&\, {\sim}F(\forall x)(Px \Rightarrow Qx)$, may be read as:
"It is true that there is some x such that if x is a raven then that x is black and is it not false that for all x, if x is a raven, then x is black."

Version 5), $(\exists x)(TPx \,\&\, TQx) \,\&\, {\sim}(\exists x)(TPx \,\&\, T{\sim}Qx)$, may be read as:
"There is at least one x such that it is true that x is a raven and it is true that x is black and it is not the case that there is some x such that it is true that x is a raven and it is false that x is black."

Each of the five versions is logically synonymous to the others. All synonyms of the definiens reduce to truth-claims about particular cases. A statement, "E-valid$[(\forall x)(Px \Rightarrow Qx)]$" does not assert that $[(\forall x)(Px \Rightarrow Qx)]$ is true, but it does entail the truth-claims in its definiens. Prefixing non-falsehood in '${\sim}\mathbf{F}(\forall x)(Px \Rightarrow Qx)$' rather than truth in '$\mathbf{T}(\forall x)(Px \Rightarrow Qx)$' to the quantified conditional is essential. The latter is synonymous with '$\mathbf{T}(\forall x)(Px \,\&\, Qx)$' i.e., "It is true that everything is a black and is a raven" which is not what we are trying to express.[8] This synonymy is established as follows by T7-24, T8-01and T7-03:

8. In M-logic $T(\forall x)(Px \supset Qx)$ is synonymous with $(\forall x)T(Px \supset Qx)$ and with $(\forall x)(FPx \vee TQx)$, and with ${\sim}(\exists x)({\sim}FPx \,\&\, {\sim}TQx)$, which are also not what we are trying to express.

|= [T(∀x)(Px ⇒ Qx) Syn (∀x)(TPx & TQx)]

Proof: 1) (T(∀x)(Px ⇒ Qx) Syn (∀x)T(Px ⇒ Qx)] [T7-24,U-SUB]

2) (T(∀x)(Px ⇒ Qx) Syn (∀x)T(Px & Qx)] [1),T8-01,SynSUB]

3) (T(∀x)(Px ⇒ Qx) Syn (∀x)(TPx & TQx)] [2),T7-03,SynSUB]

No definiens of E-validity is synonymous with T(∀x)(Px ⇒ Qx). Of course if all individuals in a field of reference were black ravens, it would follow both that some entities in that field were black ravens, and that no ravens in that field were not black; so T(∀x)(Px ⇒ Qx) and E-valid(∀x)(Px ⇒ Qx), i.e. T(∃x)(Px ⇒ Qx) & ~F(∀x)(Px ⇒ Qx), would both be true in that domain. But if we chose only a set of black ravens for our field of reference, the factual *issue* (whether all ravens are black?) is destroyed by our choice of the field of reference. The significance of E-validity lies in choosing a field of reference in which it is logically possible for the conditional to be false.

The definiens for empirical validity selects within the field of reference just those cases which are relevant: the cases in which the antecedent of the conditional predicate is fulfilled and the consequent is either true or false. We are searching for truth and falsehood. What is neither true nor false is irrelevant. If the antecedent is not true (is false or neither true nor false) then the conditional is neither true nor false, so this helps confirm that the conditional is not false. If the antecedent is true and the conditional false, then this is conclusive evidence *against* the ‘not-false’ clause and refutes E-validity. If the consequent is neither true nor false, that application of the conditional predicate is also neither true nor false and can not be counted either for or against any judgment of E-validity.

Applied to the example we have been using, the statement “All ravens are black”, the sort of evidence for a judgment of its empirical validity can be found in a domain of 3 if there were just one raven and it was black, though the conditional predicate would not be true of all members of the reference set, since it is not true when its antecedent is false. Imagine then that c is a black raven in a domain which contains two other entities, one which is not a raven but is black and one which is neither black nor a raven.

“T(All ravens are black)” is *false*:

[T(∀x)(Px ⇒ Qx)]

[T((Pa ⇒ Qa) & (Pb ⇒ Qb) & (Pc ⇒ Qc))]

F F 0 F 0 F 0 T **0** T **T** T

^

but “(All ravens are black)” is *E-valid* (using definition 1):

[E-valid (∀x)(Px ⇒ Qx) (Df$_1$)]

[T(∃x)(Px ⇒ Qx) & ~F(∀x)(Px ⇒ Qx)]

[T[(Pa ⇒ Qa) v (Pb ⇒ Qb) v (Pc ⇒ Qc)) & ~F((Pa ⇒ Qa) & (Pb ⇒ Qb) & (Pc ⇒ Qc))]

T F 0 F **0** F 0 T **T** T **T** T **T** TF F 0 F 0 F 0 T **0** T **T** T

or, “(All ravens are black)” is *E-valid* (using Definition 5):

[E-valid(∃x)(Px ⇒ Qx) (Df$_5$)]

[(∃x)(TPx & TQx) & ~(∃x)(TPx & T~Qx)]

[(TPa&TQa) v (TPb&TQb) v (TPc&TQc)) & (~((TPa & T~Qa) v (TPb & T~Qb) v (TPc & T~Qc))]

TT FFF T FF FTT **T** TT **T**TT **T** **T** FF FTTF F FF FFTF **F** TT FFFT

If a raven, d, in some domain is found to be not black, then ‘Pd & ~Qd’ is *true*, i.e., T(Pd & ~Qd), hence (TPd & T~Qd), hence (∃x)(TPx & T~Qx). But this is the contradictory of ~(∃x)(TPx & T~Qx) in Df$_5$, which is synonymous with ‘~F(∀x)(Px ⇒ Qx)’ in Df$_1$. If all ravens in the domain were not black, both conjuncts would be false. If the field of reference is a domain that has no ravens in it, then the assertion that ‘(∀x)(Px ⇒ Qx)’ is E-valid, is neither true nor false of that domain.

While logical validity can not be altered by findings of empirical fact, empirical validity can be altered as new experiences are added. It was a common sense judgment before 1800 that the spoken words of humans could not be heard if the speaker was a hundred miles away from the hearer. This was a well-supported *empirically valid* judgment at that time. Though all people had heard human voices over shorter distances, no one had heard a human voice when the speaker was a hundred or more miles away. With the development of telephones, radio and television this is no longer empirically valid. On the other hand, a great many empirically valid judgments, including “All men are mortal”, “Oak trees grow only from acorns”, and “Dry untreated wood in the open air at sea level burns”, seem to be unchanged by new experiences of humans.

If T-wffs occur as antecedent and consequent in a quantified conditional, the assertion that it is true, is false if it is false without T-wffs as components, and it is E-valid, if the quantified conditional is E-valid without having T-wffs as components. For example, if we prefix a T to the atomic wffs in the example above, we get the following results as compared to the results on the previous page. Thus prefixing T’s to the components does not affect the outcome:

“T(All ravens are black)” is false:

[T(∀x)(TPx ⇒ TQx)]
[T((TPa ⇒ TQa) & (TPb ⇒ TQb) & (TPc ⇒ TQc))]
F FF 0 FF 0 FF 0 TT **0** TT **T** TT

But “(All ravens are black)” is empirically valid:

[E-valid(∀x)(TPx ⇒ TQx) (Df_1)]
[**T**(∃x)(TPx ⇒ TQx) & ∼**F**(∀x)(TPx ⇒ TQx)]
[T(TPa ⇒ TQa) v (TPb ⇒ TQb) v (TPc ⇒ TQc)) & (∼F((TPa ⇒ TQa) & (TPb ⇒ TQb) & (TPc ⇒ TQc))]
T FF 0 FF T FF 0 TT **T** TT **T** TT **T** TF FF 0 FF 0 FF 0 TT **0** TT T TT

In dealing with the concept of conditionals some quantified T-wffs play a central role and others do not. As we saw in Chapter 7, if the four truth-operators are prefixed to the two kinds of quantifiers, then with three values there are eight basically different quantifier functions. Their differences show up in Boolean-expansions in a domain of 2 = {a,b} and in all larger domains.[9]

9. Note that in two-valued M-logic, since falsehood is not distinguished from non-truth, nor truth from non-falsehood, there are only four basic quantifications: (∀x)Px, (∃x)Px, ∼(∃x)Px, and ∼(∀x)Px which have as their synonyms, respectively, ∼(∃x)∼Px, ∼(∀x)∼P x, (∀x)∼P x, and (∃x)∼P x. In the implicit two-valued semantics of M-logic all T’s are eliminated and each ‘F’ may be replaced by ‘-’. The result, with double negation, is M-logic wffs, with the same logical properties as before. Their truth-tables are contained in the trivalent truth-tables above; they are the 2X2 lower right quadrants.

1) or 8)	2) or 7)	3) or 6)	4) or 5)	[Syn 4)]	[Syn 3)]	[Syn 2)]	[Syn 1)]
T(∀x)Px	T(∃x)Px	F(∃x)Px	F(∀x)Px	∼T(∀x)Px	∼T(∃x)Px	∼F(∃x)Px	∼F(∀x)Px
(Pa&Pb)	(PavPb)	∼(PavPb)	∼(Pa&Pb)	∼(Pa&Pb)	∼(PavPb)	(PavPb)	(Pa&Pb)
T F	**T T**	F F	F **T**	F **T**	F F	**T T**	**T** F
F F	**T** F	F **T**	**T T**	**T T**	F **T**	**T** F	F F
(∀x)Px	(∃x)Px	∼(∃x)Px	∼(∀x)Px	∼(∀x)Px	∼(∃x)Px	(∃x)Px	(∀§x)Px

It is not the presence of a T-operator but the semantics of M-logic which makes the difference. Even if we keep the T-operators, if we treat (∼TP & ∼FP) as a contradiction, quantified TF-conditionals yield only four basic quantifiers. Thus two-valued logic can not make the distinctions needed for a C-conditional.

1)	2)	3)	4)	5)	6)	7)	8)
T(∀x)Px	T(∃x)Px	F(∃x)Px	F(∀x)Px	~T(∀x)Px	~T(∃x)Px	~F(∃x)Px	~F(∀x)Px
T(Pa&Pb)	T(PavPb)	F(PavPb)	F(Pa&Pb)	~T(Pa&Pb)	~T(PavPb)	~F(PavPb)	~T(Pa&Pb)
F F F	F **T** F	F F F	F F **T**	**T T T**	**T** F **T**	**T T T**	**T T** F
F **T** F	**T T T**	F F F	F F **T**	**T** F **T**	F F F	**T T T**	**T T** F
F F F	F **T** F	F F **T**	**T T T**	**T T T**	**T** F **T**	**T T** F	F F F

When the matrix is a C-conditional, the eight forms vary in usefulness for determining E-validity. Substituting ‘P<1> ⇒ Q<1>’ for ‘P<1>’ by U-SUB, we get,

1) T(∀x)(Px ⇒ Qx)	Syn (∀x)T(Px ⇒ Qx)	Syn (∀x)T(Px & Qx)	Not Useful
2) **T(∃x)(Px ⇒ Qx)**	Syn **(∃x)T(Px ⇒ Qx)**	Syn **(∃x)T(Px & Qx)**	Useful
3) F(∃x)(Px ⇒ Qx)	Syn (∀x)F(Px ⇒ Qx)	Syn (∀x)T(Px & ~Qx)	Not Useful
4) **F(∀x)(Px ⇒ Qx)**	Syn **(∃x)F(Px ⇒ Qx)**	Syn **(∃x)T(Px & ~Qx)**	Useful
5) ~T(∀x)(Px ⇒ Qx)	Syn (∃x)~T(Px ⇒ Qx)	Syn (∃x)~T(Px & Qx)	Not Useful
6) **~T(∃x)(Px ⇒ Qx)**	Syn **(∀x)~T(Px ⇒ Qx)**	Syn **(∀x)~T(Px & Qx)**	Useful
7) ~F(∃x)(Px ⇒ Qx)	Syn (∃x)~F(Px ⇒ Qx)	Syn (∃x)~T(Px & ~Qx)	Not Useful
8) **~F(∀x)(Px ⇒ Qx)**	Syn **(∀x)~F(Px ⇒ Qx)**	Syn **(∀x)~T(Px & ~Qx)**	Useful

Four of the eight forms, numbered 1), 3), 5), and 7), are “not useful” or relevant in determining E-validity. Cases 1) and 3) say that the predicate of the quantification is *true of all* members of the domain. Thus both antecedent and consequent must apply to everything in the domain; i.e., T(P<1>& Q<1>) and T(P<1> & ~Q<1>) must hold of every member of the domain. Cases 5) and 7) say that it is not the case that these predicates are true of all members of the domain. They simply say that some things are not black ravens and some things are not non-black ravens. Wffs 1) and 3) are proven false and 5) and 7) are proven true if there is any entity in the domain of which the antecedent is not true, (e.g., if anything in the domain is not a raven). They do not contribute to determinations of E-validity because they let a single false case of the antecedent determine the truth or falsity of their quantified C-conditionals. Thus the quantified C-conditionals 1), 3), 5) and 7) are not relevant to the question of whether “all ravens are black” is empirically valid.

Wffs 1) & 3) are equivalent to conjunctions of wffs:

Wff 1) T(∀x)(Px ⇒ Qx) Syn (∀x)T(Px ⇒ Qx) Syn (∀x)T(Px & Qx) Syn (∀x)(TPx & TQx)
Wff 3) F(∃x)(Px ⇒ Qx) Syn (∀x)F(Px ⇒ Qx) Syn (∀x)T(Px & ~Qx) Syn (∀x)(TPx & FQx)

Conjunctions of POS wffs can be proven false by the falsehood of any conjunct. Finding Pa false, will make (∀x)T(Px & Qx) and (∀x)T(Px & ~Qx) false. Hence their synonyms, T(∀x)(Px ⇒ Qx) and F(∃x)(Px ⇒ Qx) will be false. But we don’t want the falsehood of T(∀x)(Px ⇒ Qx) or F(∃x)(Px ⇒ Qx) to follow logically from the falsehood of some lone Pa. This is akin to anomalies of the TF-conditional we wanted to avoid.

Wffs 5) and 7) are equivalent to disjunctions of wffs,

Wff 5) ~T(∀x)(Px ⇒ Qx) Syn (∃x)~T(Px ⇒ Qx) Syn (∃x)~T(Px & Qx) Syn (∃x)(~TPx v ~TQx)
Wff 7) ~F(∃x)(Px ⇒ Qx) Syn (∃x)~F(Px ⇒ Qx) Syn (∃x)~T(Px & ~Qx)Syn (∃x)(~TPx v ~FQx)

Disjunctions of wffs can be proven true by one not-true instantiation of a disjunctive component. Finding Pa false or not-true will make 5) ~T(∀x)(Px ⇒ Qx) and 7) ~F(∃x)(Px ⇒ Qx), true. But we don't want the truth of "E-valid(∀x)(Px ⇒ Qx) or ~F(∃x)(Px ⇒ Qx)" to follow logically from the falsehood of a single 'Pa'. Again, this is akin to anomalies of the TF-conditional we are trying to avoid.

By contrast 2) and 8) are relevant for determinations of empirical validity and 4) or 6) would suffice to prove empirical invalidity. By definition, if (∀x)(Px ⇒ Qx) is E-valid, both 2) and 8) must hold.

Wff 2) **T(∃x)(Px ⇒ Qx)** Syn **(∃x)T(Px ⇒ Qx)** Syn **(∃x)T(Px & Qx)** Syn **(∃x)(TPx & TQx)**
Wff 8) **~F(∀x)(Px ⇒ Qx)** Syn **(∀x)~F(Px ⇒ Qx)** Syn **(∀x)~T(Px & ~Qx)** Syn **(∀x)(~TPx v ~FQx)**

Statements of the form 2) **T(∃x)(Px ⇒ Qx)** are not established as true by finding any single Pa_i false. They are true only if we find some a_i such that (TPa_i & TQa_i). And *only* a finding that there is some a_i, such that (TPa_i & FQa_i) can establish that 8) **~F(∀x)(Px ⇒ Qx)** is false; otherwise it holds. This is as it should be. Presumably everyone would agree that acceptance of both 2) and 8) entails the acceptance of an empirical generalization or natural law connecting P and Q.

Wffs 6) **~T(∃x)(Px ⇒ Qx)** and 4) **F(∀x)(Px ⇒ Qx)** and their synonyms are the contradictories of 2) and 8) respectively, so that "**E-valid[(∀x)(Px ⇒ Qx)]**" will be false if there is any a_i such that (TPa_i & FQa_i) is the case or if there is no a_i such that (TPa_i & TQa_i) is the case. Neither of these can be proven true by finding some single Pa_i false. This also is as it should be, and these results are among the *plausible* results M-logic requires for quantified TF-conditionals.

In the preceding discussion, the main reason for dismissing forms 1), 3), 5) and 7) was to avoid the use of a false antecedent to assert the truth of conditional. But there is another point to be made. We have been dealing with generalized *inferential* conditionals with T-wffs as components, as in (∀x)(TPx ⇒ TQx), "For all x, if it is true that Px then it is true that Qx)". Suppose we have a generalized conditional in which the consequent is neither true nor false, as in "If any one commits murder, then they ought to be punished", or "If any one here is under 18, raise your hands!". Expressed in logic,

> "For all x, if it is true that x commits murder, then <u>it ought to be that</u> x is punished",
> "For all x, if x is here and is under 18, then <u>raise your hands</u>!"

are clearly different than

> "For all x, if it is true that x commits murder, then <u>it is true that</u> x is punished".
> "For all x, if it is true that x is here and is under 18, then <u>it is true that</u> x raises his hands!"

Expressed with C-conditionals these have the forms '(∀x)(TPx ⇒ **O**(Qx)' and '(∀x)(TPx ⇒ Qx!)' respectively. The grounds, or evidence, which would count for or against one of these does not necessarily support or refute the other. Thus we would not want to include the grounds or premmisses which support the statements "All murderer ought to be punished" as grounds or evidence either for or against the generalization, "All murderers are punished".

What we are dealing with here is the second row of the truth-table for C-conditionals of the form (TPa ⇒ TQb), for this is the row in which the antecedent is true and the consequent is neither true nor false. Since the grounds or reasons for asserting Qx! or "it ought to be that Qx" are not what is found to exist whether we want it to or not, the actual world as our objective field of reference can not determine the truth or falsehood of these condtitionals. It can't establish that Pa is true and Qb! is false or that both antecedent and consequent are true. Hence even if the antecedent is fulfilled, the conditional is neither true nor false if the consequent is an ought-statement or a command.

Transitivity of the C-conditional and the principle of the "strengthened antecedent" do not preserve E-validity or truth. They preserve logical validity in the conclusion when all premisses are logically valid, i.e., "[L-valid(P ⇒ R) & L-valid(R ⇒ Q) **⇒** L-valid(P ⇒ Q)]" is valid and from this we can deduce that if [P ⇒ R] is L-valid, then [(P&Q) ⇒ R] is L-valid. But this does not entail that "if [P ⇒ R] is E-valid, then [(P&Q) ⇒ R] is E-valid" is valid, or that "⊨ L-valid[(P&Q) ⇒ P] and E-valid [P ⇒ R], therefore [(P&Q) ⇒ R] is E-valid," is valid.

For example, 1)	If ***a*** is a raven and ***a*** is not black, then ***a*** is a raven.	[Logically valid by T6-136]
2)	If ***a*** is a raven, then ***a*** is black	[E-valid,from general law]
.: 3)	If ***a*** is a raven and ***a*** is not black,then ***a*** is black	[1),2), hypothetical Syll]

In this argument, the conclusion is inconsistent since the conjunction of antecedent and consequent is inconsistent; a conjunction of conclusion and premisses is inconsistent and the argument is not A-valid.

9.322 Quantified Conditionals and T-operators in A-logic and M-logic Compared

In M-logic universal generalizations are interpreted as generalizations of the TF-conditional, (∀x)(Px ⊃ Qx), and therefore as being true-of-every entity in the domain of reference. Up to a point this works, because the falsehood of the antecedent will make the conditional as a whole true of each thing in the domain which is not a raven. This is the same feature which is responsible for so-called "paradoxes of material and strict implication". The predicate, "<1>is a raven ⊃ <1>is black", which is synonymous with "either <1> is not a raven or <1>is black" is true of stones, trees, books and other non-ravens and of all black objects. Imagine that the field of reference is a set of 3 objects: a white dove, a black piece of coal and a black raven. "All ravens in this set are black" is true; but in what sense? Let 'a' be the dove, 'b' be the coal and 'c' be the raven:

T(∀x)(Px ⊃ Qx) Syn T((Pa ⊃ Qa) & (Pb ⊃ Qb) & (Pc ⊃ Qc))
Syn (T(Pa ⊃ Qa) & T(Pb ⊃ Qb) & T(Pc ⊃ Qc))
T **F** T F T T **F** T T T T T T T

The TF-conditional works in the sense that nothing can make the generalization false except finding a raven that is not black. If the TF-conditional '⊃' were to replace '⇒' in the definiens of "E-valid", yielding ((∃x)T(Px ⊃ Qx) & (∀x)~F(TPx ⊃ TQx))', the latter would come out true, because of the falsity of the antecedent:

((∃x)T(Px ⊃ Qx) & (∀x)~F(TPx ⊃ TQx))
(T[T(Pa ⊃ Qa)v T(Pb ⊃ Qb)v T(Pc ⊃ Qc)) & (~F(TPa ⊃ TQa) & ~F(TPb ⊃ TQb) & ~F(TPc ⊃ TQc))
(T F F T F **T** F F T T **T** T T T T <u>**T**</u> TF FF T FF T TF FF T TT **T** TF TT **T** TT

But the TF-conditional proves too much. Every domain which has no ravens at all becomes "evidence" which "confirms" by the falsity of the antecedent both of the conclusions above.[10] Only the

10. Interpreting "All ravens are black" with a TF-conditional as having the form '(∀x)(Rx ⊃ Bx)' and asking how we confirm the claim that it is true, Hempel writes,

> "…we have to recognize as confirming … any object which is neither black nor a raven. Consequently, any red pencil, any green leaf, any yellow cow, etc., becomes confirming evidence for the hypothesis that all ravens are black."

Carl G. Hempel, "Studies in the Logic of Confirmation", MIND (1945) reprinted in B.A.Brody and R.E.Grandy, *Readings in the Philosophy of Science*, 1989, p.264

requirement for E-validity with the C-conditional in '$(\exists x)T(Rx \Rightarrow Bx)$', satisfies the ordinary view that empirical confirmation of "All ravens are black" requires finding ravens that are black. What is odd is the facts that are said to make it true. The generalization cannot be false and thus in two-valued M-logic must be true in any domain which has no ravens, exactly because it has no ravens; it also cannot be false and must be true of any domain consisting entirely of black objects, whether or not they are ravens.

The M-logic interpretation of "if...then" has the further disadvantage in quantification theory of implying if a predicate P is false of everything in a domain, then no matter what predicate Q may be or what it is applied to, 'If P<1> then Q<2>' applies truthfully to every member; or, if a predicate, P, is true of every thing in a domain, then no matter what predicate, Q, you may put in the antecedent, it is a valid inference that everything is such that (if Q then P).

In principles at issue here are (with Q representing any sentence whatever):

$$\text{TAUT}[T(\forall x){\sim}Px \supset (\forall x)(TPx \supset Q)]$$
$$\text{TAUT}[T(\forall x)Px \supset (\forall x)(Q \supset TPx)]$$

The problem is partly one of interpreting '⊃' as "if...then". So interpreted, these read as:

"If it is true that nothing is P, then everything is such that if P then Q"
"If it is true that everything is P, then everything is such that if Q then P."

Goodman uses the first principle to present the problem of the counterfactual conditional. You have a piece of butter which is eaten without ever having been heated. It follows by M-logic that for all times, if that piece of butter is heated to 150 degree (or to 2,000 degrees), then it does not melt at that time. Given this premiss that the butter is never heated to 150 degrees, $(\forall t){\sim}M(a,t)$ is true; from this it follows that for every moment in time if that piece of butter, a, is heated to 150 degrees, then it does not melt,... or then it does melt, ... or then it turns into a flower,... or then it changes to water, or...

Carnap used the same principle to show that dispositional predicates could not be defined using M-logic. His example was a piece of wood that is destroyed by burning it without ever having been put in water. Since 'x is soluble' is defined as '$(\forall x)(\forall t)$(if x is put in water at time t, then x dissolves at time t', this definition makes the wood soluble in water. See section 10.334.

The C-conditional does not entail such irrelevant and contradictory conditionals. In A-logic these problems are avoided a) by abandoning the interpretation of "If P then Q" as '$\sim(P \ \& \ {\sim}Q)$' and using C-conditionals instead, and b) by distinguishing valid implications as *de dicto* theorems rather than theorems suitable for *de re* reasoning.

Making these changes we never get a '⇒'-for-'⊃' analogue of the M-logic theorems either in TAUT-theorems or in Validity-theorems. I.e.,

Not-Taut$[T(\forall x){\sim}Px \Rightarrow (\forall x)(TPx \Rightarrow Q)]$	Not-Valid$[T(\forall x){\sim}Px \;.\!:\; (\forall x)(TPx \Rightarrow Q)]$
Not-TautT$[T(\forall x)Px \Rightarrow (\forall x)(Q \Rightarrow TPx)]$	Not-Valid$[T(\forall x)Px \;.\!:\; (\forall x)(Q \Rightarrow TPx)]$

Instead we get, at best, the following kinds of implication-theorems, with all of the cautions about misusing implications to deal with *de re* questions:

Ti8-786. $\text{Valid}_I\ [(\forall x)TQx \Rightarrow (\forall x)(0Px \supset TQx)]$ [Ti7-86,DR8-6g]
Syn $\text{Valid}_I\ [(\forall x)TQx \Rightarrow (\forall x)({\sim}0Px \vee TQx)]$
Ti8-787. $\text{Valid}_I\ [(\forall x)TQx \Rightarrow (\forall x)(TPx \supset TQx)]$ [Ti7-87,DR8-6g]
Syn $\text{Valid}_I\ [(\forall x)TQx \Rightarrow (\forall x)({\sim}TPx \vee TQx)]$

Ti8-788. $\text{Valid}_I\ [(\forall x)TQx \Rightarrow (\forall x)(FPx \supset TQx)]$ [Ti7-88,DR8-6g]
  Syn $\text{Valid}_I\ [(\forall x)TQx \Rightarrow (\forall x)(\sim FPx \vee TQx)]$
Ti8-789. $\text{Valid}_I\ [(\forall x)\sim TPx \Rightarrow (\forall x)(TPx \supset FQx)]$ [Ti7-89,DR8-6g]
  Syn $\text{Valid}_I\ [(\forall x)\sim TPx \Rightarrow (\forall x)(\sim TPx \vee FQx)]$
Ti8-790. $\text{Valid}_I\ [(\forall x)FPx \Rightarrow (\forall x)(TPx \supset FQx)]$ [Ti7-90,DR8-6g]
  Syn $\text{Valid}_I\ [(\forall x)FPx \Rightarrow (\forall x)(\sim TPx \vee FQx)]$
Ti8-791. $\text{Valid}_I\ [(\forall x)TPx \Rightarrow (\forall x)(\sim TPx \supset TQx)]$ [Ti7-91,DR8-6g]
  Syn $\text{Valid}_I\ [(\forall x)TPx \Rightarrow (\forall x)(TPx \vee TQx)]$

If a quantified conditional has an inconsistent antecedent, as in $(\forall x)((Px\&\sim Px) \Rightarrow Qx)$, no instance can be true, because in no case can $((P<1>\&\sim P<1>)\&Q<1>)$ apply truthfully to any object. Cases in which $(\forall x)(Px \Rightarrow Qx)$ is true because $(P<1> \Rightarrow Q<1>)$ is true of every object in the domain are rare special cases, and may or may not be empirically significant.

Generalized C-conditionals are not precluded by logic from being true of all elements in a domain; they are just not required (as in M-logic) to have that feature. Even generalized conditionals about the whole universe are not precluded. For example, metaphysical statements about what is the case in Ultimate or Universal Reality may make assertions about all entities in the universe; e.g., "All states of affairs in the real world are brought about solely by the laws of physics", "All entities in the universe are created by God". These statements convey the idea of associating God or physical laws causally with every entity or event.[11] Neither one is inconsistent by itself, though they can be defined in such a way that to hold both of them true would be inconsistemt.

But when the field of reference is broader than the entities referred to in the antecedent—especially if the subject is a sub-set of the entities in the actual world, like ravens—applications of conditional predicates are neither true nor false when the antecedent does not apply.

Thus we reject any general requirement that the predicate of a universal generalization, "If $P<1>$ then $Q<1>$", must be true of every individual in the domain. The same principles apply in mathematics and other fields; '$(\forall x)(\forall y)((x \text{ is odd } \& \ y=x) \Rightarrow y^2 \text{ is odd})$' ("The square of every odd number is odd") is true of some numbers (odd numbers and their squares) and is not false of any number. But it is not a truth about all numbers, much less about all entities. The meaning of claiming that a universal generalization is factually "valid" (though not logically valid) in truth-talk, is a claim that a generalized inferential conditional is not-false and sometimes true—not a claim that some predicate is true of everything.[12]

9.33 Generalizations about Finite and Non-finite Domains

The concept of a generalized C-conditional covers a great many variations. We will discuss first simple monadic generalizations, as they apply to finite domains. Then we discuss generalizations about non-finite domains. Then we discuss generalizations with polyadic predicates, employing more that one

11. According to many theories of knowledge, whether such statements are true or false of ultimate reality is unknowable, although ideas thus expressed often support motivations and may be used to guide to action. A-logic is compatible with, but does not entail such a theory of knowlege.

12. Universal generalizations in ethics and in directives are similar in that they too are not about everything, but only about what is specified in the antecedent. However, they have meanings not reducible to truth-talk. "Everyone ought to keep their promises" has the form '$(\forall x)(T(Px) \Rightarrow O(Qx))$' with 'O' for 'it ought to be that'; "If it is true that a house is on fire, call the fire department!" has the form '$(\forall x)(T(Px) \Rightarrow \text{do } Q!)$'.

variable in its quantifiers. This will lead to more specific types of general statements, including functional statements, causal statements and probability statements.

9.331 Monadic Generalizations

Q-wffs with only one quantifier are monadic quantifications. If there is more than one quantifier and the total expression is reducible (by "rules of passage") to synonymous expressions in which no quantifier has an individual variable other than its own within its scope, this too is a monadic quantification.

Monadic quantification stands in contrast to polyadic quantification, in which variables that are bound to different quantifiers occur together in atomic wffs with polyadic predicates so that there is no way to reduce the expression to monadic components.

9.3311 Monadic Generalizations About Finite Domains

Let us call '(∀x)Px' a "*definite* universal generalization" if it is a generalization about a field of reference which contains a definite, finite number of individuals.

For a great many questions the answer sought is a definite universal generalizations: Have all members paid their dues? Are all signers of the petition registered voters? Are all members of the Commission members of the Republican Party? Are all bottles in this container clear glass? Are all tiles on all space-ships now in use by NASA able to withstand the heat of re-entry?

The logical moves from data to *definite* universal generalizations differs in certain respects from logical moves from data to indefinite generalizations, so we take them up in turn. Both, however may be conceived as moves from data to "E-valid" judgments.

With the C-conditional in universal quantifications it is possible to distinguish cases relevant to what we are talking about from those that are irrelevant, and also to provide formal algorithms for determining frequencies which provide the basis for statistics and probability statements.[13]

EXAMPLE 1 illustrates how a definite empirically valid generalization follows logically from actual particular data, how another generalization is proven to be empirically invalid by the same data, and how frequencies of definite universal generalizations follow logically from the data. TABLE 9-2 on Page 498 shows:

(i) in columns 1 and 2, how observations of the facts in EXAMPLE 1 are the basis for determining whether truth, non-truth, falsity or non-falsity apply to applications of 'P<1>' and Q<1> in each box (each instantiation) of the quantified conditional,

(ii) how these in turn become the basis for assigning '~F' or 'T' to applications of the quantified C-conditional predicate (P<1> ⇒ Q<1>) or (Q<1> ⇒ P(1>) at each instantiation in columns 3), and column 4).

(iii) how the conjunction of findings in column 3 shows whether '(∀x)~F(Px ⇒ Qx)' or '(∀x)~F(Qx ⇒ Px)' as a whole are true or false, and how the disjunction of findings in column 4) determines whether (∃x)T(Px ⇒ Qx) or (∃x)T(Qx ⇒ Px) as a whole are true or false. On these results the judgement is made that in this example **(∀x)(Px ⇒ Qx)** is E-valid and **(∀x)(Qx ⇒ Px)** is not.

---

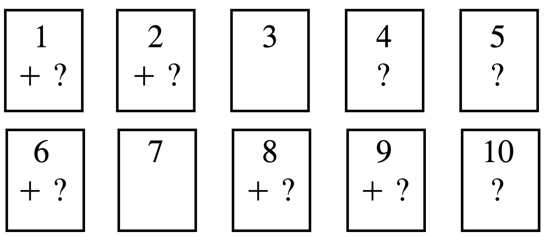
13. See Section 9.353 for the contrast between the conformity of the probability of a C-conditional with the concept of conditional probability in standard probability theory, and the disconnect between the probability of a TF-conditional and conditional probability.

EXAMPLE 1

Consider the following statements about the little boxes in Figure 1:

(1) "Every box with a **p**lus sign in it has a **q**uestion mark in it" (∀x)(TPx ⇒ TQx)
(2) "Every box with a **q**uestion mark in it has a **p**lus sign in it" (∀x)(TQx ⇒ TPx)

Figure 1



Using ' P<1>' for 'Box <1> has a **p**lus sign in it'
and 'Q<1>' for 'Box <1> has a **q**uestion mark in it"
(1) and (2) may be symbolized as:
(1) (∀x)(TPx ⇒ TQx) (2) (∀x)(TQx ⇒ TPx)
Which is empirically valid?
Which is not empirically valid?
and if it is not empirically valid, how can we
indicate that it is partly so?

In addition, the example shows (iv) how, in a finite domain we can count up the cases in which the predicates (P<1> ⇒ Q<1>) and (Q<1> ⇒ P<1>) have true applications and compare them with the total number of cases in which they are either true or false (i.e., instances in which the antecedent is true and the consequent either true or false). On the basis of these numbers we establish frequencies—e.g. in EXAMPLE 1, that (P<1> ⇒ Q<1>) applies truthfully in 100% of the instances in which the antecedent is true, while (Q<1> ⇒ P<1>) only applies truthfully in 62.5% of those instances. This sort of computation is the basis of statistics and probability theories.

In TABLE 9-2 on the next page, we count the number of T's in column 4, and divide it by that number plus the number of F's in column 3 (which gives the total number of cases that are either T or F to get the frequency in the reference class of true instantiations over the number of relevant instantiations. In general, if [(∀x)(Px ⇒ Qx)] is applied to a domain of n individuals, and has k cases in which the antecedent and consequent are both true, and j cases in which the antecedent is true and the consequent false, then the frequency of true cases is k/(j+k) in that domain. Probability theory is based on this kind of ratio. The number of cases in which (Pa_i ⇒ Qa_i) is neither true nor false is (n-(j+k)).

The process of combining logic and observation to establish the empirical validity of the generalized inferential conditional is the business of inductive logic. Through it the C-conditional allows a new approach to inductive logic.

Statement (1), (∀x)~F(Px ⇒ Qx), is true; in all five cases in which the antecedent is true, the consequent is also, and when Pa_i is false, the conditional is neither true nor false,thus not false:

((T(P1 ⇒ Q1) & T(P2 ⇒ Q2) & T(P6 ⇒ Q6) & T(P8 ⇒ Q8) & T(P9 ⇒ Q9))
T T T T T T T T T T T T T T T T T T T T T T T T T

TABLE 9-2

Box	Ante-cedent	Conse-quent	**(1) E-valid(∀x)(Px ⇒ Qx)** (∀x)~F(Px⇒Qx)	&	(∃x)T(Px⇒Qx)	M-logic: (∀x)T(Px ⊃ Qx)
1	T[P1]	T[Q1]	(~F[P1 ⇒ Q1]	&	(T[P1 ⇒ Q1]	(T[P1 ⊃ Q1]
2	T[P2]	T[Q2]	& ~F[P2 ⇒ Q2]		v T[P2 ⇒ Q2]	& T[P2 ⊃ Q2]
3	F[P3]	F[Q3]	& ~F[P3 ⇒ Q3]		v ~T[P3 ⇒ Q3]	& T[P3 ⊃ Q3]
4	F[P4]	T[Q4]	& ~F[P4 ⇒ Q4]		v ~T[P4 ⇒ Q4]	& T[P4 ⊃ Q4]
5	F[P5]	T[Q5]	& ~F[P5 ⇒ Q5]		v ~T[P5 ⇒ Q5]	& T[P5 ⊃ Q5]
6	T[P6]	T[Q6]	& ~F[P6 ⇒ Q6]		v T[P6 ⇒ Q6]	& T[P6 ⊃ Q6]
7	F[P7]	F[Q7]	& ~F[P7 ⇒ Q7]		v ~T[P7 ⇒ Q7]	& T[P7 ⊃ Q7]
8	T[P8]	T[Q8]	& ~F[P8 ⇒ Q8]		v T[P8 ⇒ Q8]	& T[P8 ⊃ Q8]
9	T[P9]	T[Q9]	& ~F[P9 ⇒ Q9]		v T[P9 ⇒ Q9]	& T[P9 ⊃ Q9]
10	F[P10]	T[Q10]	& ~F[P10 ⇒ Q10])		v ~T[P10 ⇒ Q10])	& T[P10 ⊃ Q10]
	T(P<1>)	T(Q<1>)	(TP<1> ⇒TQ<1>)	&	(TP<1> ⇒TQ<1>)	(P<1>⊃Q<1>)

(2) Not E-valid(∀x)(Qx ⇒ Px)

Box	Ante-cedent	Conse-quent	but, (2) Freq(∀x)(Qx⇒Px) = .625 (∀x)~F(Qx⇒Px)	&	(∃x)T(Qx⇒Px)	M-logic: (∀x)T(Px ⊃ Qx)
1	T[Q1]	T[P1]	(T[Q1 ⇒ P1]	&	(T[Q1 ⇒ P1]	(T(Q1 ⊃ P1]
2	T[Q2]	T[P2]	& T[Q2 ⇒ P2]		v T[Q2 ⇒ P2]	& T[Q2 ⊃ P2]
3	F[Q3]	F[P3]	&~F[Q3 ⇒ P3]		v~T[Q3 ⇒ P3]	& T[Q3 ⊃ P3]
4	T[Q4]	F[P4]	& F[Q4 ⇒ P4]		v F[Q4 ⇒ P4]	& F[Q4 ⊃ P4]
5	T[Q5]	F[P5]	& F[Q5 ⇒ P5]		v F[Q5 ⇒ P5]	& F[Q5 ⊃ P5]
6	T[Q6]	T[P6]	& T[Q6 ⇒ P6]		v T[Q6 ⇒ P6]	& T[Q6 ⊃ P6]
7	F[Q7]	F[P7]	&~F[Q7 ⇒ P7]		v~T[Q7 ⇒ P7]	& T[Q7 ⊃ P7]
8	T[Q8]	T[P8]	& T[Q8 ⇒ P8]		v T[Q8 ⇒ P8]	& T[Q8 ⊃ P8]
9	T[Q9]	T[P9]	& T[Q9 ⇒ P9]		v T[Q9 ⇒ P9]	& T[Q9 ⊃ P9]
10	T[Q10]	F[P10]	& F[Q10 ⇒ P10])		v F[Q10 ⇒ P10])	& F[Q10⊃P10])

<u>Conclusions</u>: (1) "E-Valid(∀x)(Px ⇒ Qx)" is true: "If P<1> is true, then Q<1> is true" is true in several cases (boxes 1,2,6,8,9,10), and is never-false.

(2) "E-valid(∀x)(Qx ⇒ Px)" is false; in two boxes (4 and 5) P<1> is T but Q<1> is F, thus it is not never-false.

However, (2) "Fr(∀x)(Qx ⇒ Px) = .625" is true. I.e., (Q<1> ⇒ P<1>) is true 62.5% of the time—in 5 of the 8 boxes in which (P<1> ⇒ Q<1>) is either T or F.

Statement (2), (∀x)~F(Qx ⇒ Px), is false, because instances 4 and 5 and 10 are false. Hence it is not E-valid. But it is true in five out of eight cases; i.e., 62.5% of the time.

In the same field of reference, both (∃x)T(Px ⇒ Qx) and (∃x)T(Qx ⇒ Px) are true. All that is needed is one true instantiation, and both have five (the same five).

Note that for statement (1) instances 3,4,5,7,and 10 are neither true nor false—they are irrelevant because the antecedent P<1> does not apply. For statement (2) only instances 3 and 7 are irrelevant; the antecedent Q<1> applies in the other eight cases.

The range of the quantified variable is definite and finite in these examples. The field of reference (the boxes in EXAMPLE 1) has a finite number of entities, namely 10. The number of entites in that field to which the antecedent's predicate, '<1>is a box in EXAMPLE 1 with a plus sign' applies is a specific finite number, namely 5. Thus we prove conclusively from empirical data that "all boxes in EXAMPLE 1 with plus signs in them have question marks in them" is E-valid as a matter of fact. '(P(<1> ⇒ Q<1>)' is true of some individuals and not-false of all individuals in the field of reference.

It is also possible to prove conclusively from the same data, that "All boxes with a question mark in them, have a plus sign" is not empirically valid, since there are three cases in which the antecedent is true and the consequent is false (see underlined cases below). The application of '(Q<1> ⇒ P<1>)' is true in 62.5% of the cases to which the antecedent applies. This is expressed as "The frequency of T(Q<1> ⇒ P<1>) = .625" or "(∀x)FrT(Qx ⇒ Px) = .625" which describes the fact that the 62.5% or 5/8 of the relevant (doubly underlined) conjuncts in the conjunction represented by '(∀x)(Qx ⇒ Px)', are true; i.e.,

$$((\underline{Q1}\Rightarrow P1)\&(\underline{Q2}\Rightarrow P2)\&(Q3\Rightarrow P3)\&(\underline{Q4}\Rightarrow P4)\&(\underline{Q5}\Rightarrow P5)\&(\underline{Q6}\Rightarrow P6)\&(Q7\Rightarrow P7)\&(\underline{Q8}\Rightarrow P8)\&(\underline{Q9}\Rightarrow P9)\&(\underline{Q10}\Rightarrow P10))$$

t T t   t T t   f 0 t   t F f   t F f   t T t   f 0 f   t T t   t T t   t F f

The logical moves illustrated in this example are easily extended to cover all of the many varieties of establishing the E-validity and frequencies of definite universal generalizations.

9.3312 Monadic Generalizations About Non-finite Domains

Many empirical generalizations of common sense and natural science are intended as "indefinite universal generalizations"—generalizations about some class of entities for which the number of members is not specified or is deliberately left unspecified. To be indefinite, is not necessarily to be infinite. "All men are mortal" is presumably about the class of all humans. Is the number of entities in that class (i.e, humans) finite? Will humans become extinct, never to re-occur like, perhaps. the dinosaurs?) or will there always be humans?—is the class infinite?). These are questions we do not need to ask. We do have to recognize, however, that the statement "All men are mortal" is not intended as a statement about a definite, finite number of men—men who have lived up to this time perhaps. As a law of nature, it is intended to refer to all future men, and we deliberately do not assume either a specific finite number, or that the number must be some finite number. In some cases we want to assert positively that the domain has more than one member and is not finite—as in mathematics when we are dealing with the positive integers, or with irrational numbers.[14]

When the universal generalization is indefinite, its empirical validity can not be established once and for all by any finite number of cases. But it can still be *asserted* that it is empirically valid, i.e., that there are no cases in which it is false, and there are some cases in which it is true. It can also be found not-valid, but true in some cases and false in others, and the frequency of true cases relative to those that are either true or false among those that have been empirically determined can be established and used as a basis for probability judgements about future cases.

---

14. To say a domain is not finite is not synonymous with saying that it is "actually infinite". But this raises an issue that need not be addressed here.

A disjunctive ("existential") generalization, can be conclusively proven true, if one or more particular instantiations are found true. It does not matter whether the domain is definite and finite, or indefinite but finite, or an infinite; if there is some field of reference in which both antecedent and consequent apply in fact to one or more particular entities, the disjunctive generalization is established as true in fact for that field of reference.

A generalized inferential conditional, of the form $(\forall x)\sim F(Px \Rightarrow Qx)$, is true only if all *relevant* instantiations are true. Relevant instantiations are those in which the antecedent is true and the consequent is either true or false. In a *definite* universal generalization, the range of the variable is both specific and finite, so the generalization can be proven true by finding that every particular instantiation in the finite set of relevant instantiation is true. But the range of an *indefinite* universal generalization is not specific and the number of relevant situations in the range of its variables is not any definite finite number. It follows that such a generalization can not be conclusively proven to be factually *true* by examination of any finite number of data, since by definition there are always additional cases which may be relevant and could be false. The conjunction abbreviated by the quantifier is not complete although the predicate form of all conjuncts, past, present and future is known. Thus indefinite universal generalizations can be proven false but can not be proven true by facts in the supposed field of reference. The most that can be proved is that this conjunction has not been shown false to date, though all relevant observed case are true; thus there is no reason to stop holding it or acting upon it.

There are two sorts of reasons one might have to reject an indefinite empirical generalization: 1) no positive, empirical true instantiations have been found, and 2) at least one empirically false instantiation has been found.

The first reason does not entail that the generalization is false, but it makes the generalization merely theoretical and speculative. It becomes unspeculative, though not conclusively proved, when an instance of it is found to be true. Einstein's general theory of relativity entailed that all light rays which passed through the gravitational field of the sun would be curved. This was purely a speculative hypotheses until 1919 when Eddington and others confirmed that the antecedent and consequent were both true during the eclipse of the sun in 1919. In this example almost all of the supposedly infinite number of occasions when this conditional was true were inaccessible to observation and human measurement; the one case that was measureable turned out to be true and no accessible cases were false. Supporters of Einstein's theory hold that subsequent observations have supported the empirical validity of generalizations entailed by Einstein's theory; other empirical observations predicted by the theory have also turned out, upon observation, to be true.

The second reason for rejecting an indefinite universal generalization is by finding a single instantiation which is false. If found, the generalization is conclusively proven false. However, often centrally important features of an empirical generalization which has worked in a great many cases but fails in other newly discovered cases, can be preserved by adding a restriction in the antecedent which confines the relevant cases to a more precisely defined set. If the rephrased generalization has some true empirical instantiations and no empirically false instantiations, it remains empirically valid in its revised form.

The concept of empirical validity may be applied whenever the two conjuncts in the definiens apply—1) there are no known false cases of the conditional in the matrix, and 2) there are one or more true cases of the conditionals. Thus it can apply equally to cases in which the antecedent is satisfied in only a finite number of cases in the field of reference, and to cases which the number of cases which satisfy the antecedent is indefinite or not-finite. The generalizations or laws of natural science, since they are intended to be used in prediction and control of future events, fall in the latter category. To distinguish the two cases we might call the former cases of definite empirical validity and the latter cases of indefinite empirical validity.

9.333 Polyadic Quantification

So far we have dealt with monadic predicates and single quantifiers. Russell rightly pointed out that the really great advance made by mathematical logic was its logical theory of relations—i.e, the logic of polyadic predicates. It is this which made it possible to schematize the complex theorems in mathematics and show how they could be derived from axioms and postulates expressible in the language of logic.

In M-logic, the move from monadic quantification theory to quantification theory with polyadic relations and multiple quantifiers, makes it possible to make general statements involving atomic relational predicates—dyadic predicates like '<1>loves<2>' or '<1>is greater than<1>' for describing a relationship between two entities (schematized as 'P<1,2>'), triadic predicates for describing relationships between three entities '<1>is between<2> and<3>' and generally n-adic predicates symbolized by R<1,…,n>.

Extending polyadic quantification to wffs with C-conditionals, it is possible to express the many "natural laws" recognized by common sense, as well as very complicated "laws of natural science", and show how they are derived or confirmed by observation of facts without getting into the difficulties that M-logic encounters. To hold that a quantified conditional is a natural law, is to hold that it is empirically valid. (Many quantified conditionals do not qualify).

One of the "natural laws" long recognized by common sense and science, is the genetic law that offspring of any biological species are always are members of the same species as their parents; e.g., the chldren of humans are humans, the offspring of horses are horses, the offspring of oak trees are oak trees. None of these statements are analytic truths. One can imagine without contradiction, that an egg from human ovaries, fertilized by a human sperm, might develop into a dog or an elephant. But the empirical facts are otherwise. Ignoring malfunctioned births, there are no known cases of a human couple giving birth to a healthy dog or some other animal.

Let 'P<1>' be any biological-species predicate (e.g., '<1>is a horse') and let 'R<1,2>' be '<1>is the offspring of<2>', then the genetic law of reproduction may be expressed as

$$\text{E-valid}(\forall x)(\forall y)((Px \,\&\, R<y,x>) \Rightarrow Py)$$

The instantiation, "Every offspring of a horse is a horse" is E-valid; or, "For all x and all y if x is a horse and y is an offspring of x, then y is a horse" is empirically valid.

Another empirical generalization with two variables but different quantifiers is "Every human being has a human father"—in logical symbolism, (∀x)(T(Hx) ⇒ T(∃y)(Hy & Fyx)) or synonymously, (∀x)(∃y)(T(Hx) ⇒ T(Hy & Fyx)). I.e., (For all x, there is some y, such that if it is true that x is a human then it is true that y is human and y is the father of x). Some persons who call themselves Christians believe in the "virgin birth"; they believe there is one human, Jesus, that did not have a human father. But presumably they believe that with this one exception this proposition is empirically valid. Those who do not believe in the virgin birth may accept the statement as empirically valid without exception, though the empirical data is indirect rather than direct. With the advent of human cloning it will no longer be E-valid, but maybe "every human has a human mother" will remain E-valid.

Note that the order of quantifiers makes a difference. "(∃y)(∀x)(THx ⇒ T(Hy & Fyx)" says "there is some human who is the father of all humans". No rational person would consider this empirically valid.

Determinations of truth, non-falsehood, empirical validity, etc. of this sort of law, though it involves two variables, do not differ in most respects from determinations for the monadic generalizations previously discussed. But binary predicates can be reflexive or irreflexive or non-reflexive. These and other properties (being symmetric, asymmetric or anti-symmetric, transitive, etc.), are properties which monadic predicates do not have. Consequently, there are problems in the logic of polyadic predicates which do not arise when dealing with monadic predicates only.

Most laws in the physical sciences associate mathematical functions with measurable properties of physical entities. Boyle's Law is a simple early example; it states that the volume of a gas is inversely proportional to the pressure exerted on it. Let us analyze the mathematical function and its relation to observed facts, with particular attention to the logic required to move from the facts to the law.

When Boyle did his first experiments in 1661 he measured in successive moments a) the volume, V, of a bit of gas enclosed in the lower end of J-shaped tube and b) increases in the pressure, P, placed on that gas, as measured by the number of ounces of mercury added in the other side of the J-shaped tube. The conclusion that he reached, based on these observations, was that the measure of the volume of the gas, times the measure of the pressure, V X P, was always the same number (or almost the same)—a constant. From (V X P) = k, it follows in mathematics that V = k/P; the numerical measure of volume is a function of a constant divided by the numerical measure of the pressure. (The numerical values depend on the units of measure used).

Boyle's account of the experiments was as follows. He had a J-shaped tube with some air trapped in the enclosed lower end and he poured mercury in the other end, making a measurement after each of 25 additions of mercury. He used the space occupied by the air to measure the volume V, and the space occupied by the mercury at the other end of the tube to measure the amount of pressure, P, exerted on the air. The first and last two pairs of measurements were as follows:

Observation	Observed volume of the gas	Measured pressure on the gas[15]	Constant k V X P = k = 350 (± 3.5)
t1:	V = 12"	P = 29 1/2"	V_1 X P_1 = 346 1/2
t2:	V = 11 1/2"	P = 30 9/16"	V_2 X P_2 = 351 1/2
	....		
t24:	V = 3 1/4"	P = 107 13/16"	V_{24} X P_{24} = 350 1/4
t25:	V = 3"	P = 117 9/19"	V_{25} X P_{25} = 352 3/4

The important point is that the product V x P hovered around a constant number, k. Note that the specific number for k, 350(± 4), is due to the use of inches, rather than millimeters or some other units. It does not matter whether the units are inches, centimeters, or millimeters, etc., provided the same system of measurement is used throughout.[16] What is important is that the product of the two magnitudes is the same when one quantity increases while the other decreases.

Boyle's law is usually expressed loosely as "The volume of a gas varies inversely with the pressure exerted upon it"; i.e., as the pressure goes up the volume goes down. To show more precisely the logical relations between 1) the descriptions of empirical data which were asserted (implicitly) to be true, and 2) the generally law which was claimed to supported by them (to be empirically valid), we analyze the situation and the relevant expressions in logical language as follows:

---

15. Boyle assumed throughout that the air-pressure in the rooms where he did his experiments was (as Torricelli had shown in 1643) equivalent to the pressure of 29 1/2 inches of mercury. Thus this was added to the observed heights of the mercury column above its bottom level.

16. If all measurements were in millimeters, the constant result of V X P would be 227,516(± 2,600) mm. If V were measure in millimeters and P in inches, the constant product wold be 8890(± 102)mm. If measurements were made in cm's, the constant would be about 2300 ± 25 .

First, underlying his experiments are many factors which are presupposed and unexpressed, including the presupposion of the stability and constancy of Boyle's equipment and its environment throughout the experiments, the presupposition of the invariablity of units of measurements (inches and ounces) in the measurement systems used, the presupposition of the reliability of arithmetic algorithms for multiplying numbers, etc. The law of how volume of a gas varies with pressure, and the empirical basis of it, is expressed without mention of these presupposed stable factord, without which the outcome might have been different.

Presupposing these constancies, the experiment is expressed in terms of the following abstract predicate, which has argument positions for, on the one hand, by a constant number and a constant amount of gas which is enclosed in the short end of the J-shaped tube, and on the other for varying singular terms for times, numbers measuring volume of the gas, and numbers measuring pressures on the gas.

The predicate:	Abbreviated Symbolically:
If <1>is a constant positive number and	(K<1>
and <2>is a certain bit of gas	& G<2>
and it is true that(<3> is an occasion (or time)	&(T(O<3>
and <4> is the volume of <2> on occasion <3>	& V<4,2,3>
and <5> is the pressure on <2> on occasion <3>	& P<5,2,3>)
then <4> times <5> on occasion <3> equals <1>.	⇒ (R<4,5,3,1>))

Thus the complex predicate used in the law has the form,
((K<1> & G<2> & (O<2> & V<4,3,2> & P<4,3,2>) ⇒ (R<4,5,2,1>)))

Leaving the presuppositions about relevant conditions unexpressed, it is clear how the inferences from observations to the law proceeded, according to A-logic:
The first observation, at t1: V = 12" P = 29½" 12 X 29½ = 346½
asserts true atomic statements, which when conjoined, instantiate the predicate above as follows, letting 'a' designate the particular piece of gas that Boyle had in the short end of his J-shaped tube:

T (K(350 ± 4) & **T**(G<a>)
& **T** (O<t1>) & **T**(V<a,12,t1>) & **T**(P<a,29 1/2,t1>)
& **T** (R<12,29.5, 350(± 4), t1>)).

By Ax.8-01, Ax7-03 &-ORD, and SynSUB the T-operators can be redistributed as follows

T(K(350 ± 4) & (G<a>) & ((O<t1> & V<a,12,t1> & P<a,29½,t1>) & (R<12,29.5, 350(± 4),t1>)**)**.

From this, by &-ORD, Axiom 8.01 and SynSUB the first expression with a C-conditional asserted true follows:

T(K(350 ± 4) & (G<a)> & ((O<t1> & V<a,12,t1> & P<a,29½,t1>) ⇒ (R<12,29.5,350 ± 4,t1>)**)**

In like manner each measurement from t1 to t25 yields a true conjunction with a predicate of the same form but with different numbers in positions <4> and <5> form:

T(K(350 ± 4) & **T**(G<a)> &
& **T** ((O<t1> & V<a,12.00,t1> & P<a, 29.1250, t1>) **⇒** (R<12.00, 29.1250,346.50, t1>))
T ((O<t2> & V<a,11.50,t2> & P<a, 30.5625, t2>) **⇒** (R<11.52, 30.5625,351.50, t2>))
....
T ((O<t24> & V<a, 3.25,t24> & P<a,107.8125,t24>) **⇒** (R< 3.25,107.5625,350.25,t24>))
T ((O<t25> & V<a, 3.00,t25> & P<a,117.5625,t25>) **⇒** (R< 3,00,117.5625,352.75,t25>))

Actually, the number for the constant k, did not come out exactly the same on each trial. The produce of volume and pressure ranged from 346 to 353.75, i.e., fell within the range of 350 ± 4. Nevertheless this was a variation of only about 1%; and quite close to a fixed single number. To represent the empirical data truthfully, though, we should use '350 ± 4', rather than a single number. Using this number, and EG (disjunctive generalization), the 25 trials yield:

(∃x)(∃y)(∃z)T(K(350 ± 4) & (G<a)> & **(**((Ox & V<a,y,x> & P<a,x,z>) ⇒ TR(y,z,350 ± 4,x)

Also, for the finite field of reference consisting of these 25 occasions, we can conclude definitely that for the particular bit of gas a with volume measured in inches and pressure in ounces,

(∀x)(∀y)(∀z) ~F((G<a)> & K(350+4) & (Ox & V<a,y,x> & P<a,x,z>))) ⇒ R(y,z,350 ± 4,x))

and from these results, it follows that for this piece of gas in these 25 trials, the following statement is empirically valid: For all x, y and z, if x is a trial in which the volume of a is y and the pressure on a is z, then the product of x and y lies in the constant 350 ± 4.

This remains true as we move to a broader field of reference; the (∃x)T(Px ⇒ Qx) clause and the (∀x)~F(Px ⇒ Qx) clause continue to hold. But it is only *proven* for this small set of 25 trials. However, if other scientists repeat the experiment and get similar results, the empirical data base expands; the generalization remains empirically valid not only for the piece of gas that Boyle had in his J-shaped tube at that time, but for other pieces and kinds of gas tested at other times, in other places by other people. The generalization is expanded to cover any kind of gas, and any methods of measuring volume and pressure, and found empirically valid by further experiment.

In fact of course the variations above and below 350 suggested there were other variables to be considered. When the temperature is varied during the trials the deviation from the average number increased. Indeed variations with temperature were so great as provide counter-examples to the law as stated by Boyle, showing that the law as initially stated does not hold in general, but only when temperature is held constant. Charles' law showed that volumes of gases varied not only with pressure but with temperature. This requires that Boyle's law be qualified; it holds provided the temperature does not change. But constancy of temperature may also be viewed simply as one of the relevant, or necessary, conditions on which Boyle's Law rests.

When M-logic tries to deal with the empirical validity of empirically justified mathematical functions, it has the same problems as when it tries to show the logical process of moving from descriptions of fact to simpler generalizations. The rules of M-logic permit inferences from the factual descriptions which would be the opposite of the conditionals we are trying to establish, and the initial step directly from a conjunction of atomic truths to a TF-conditional are indirect and unnatural ones which make the antecedent irrelevant, and proceed only through the truth of the consequent and the *de dicto* principle of Addition.

The move from a conjunction of true descriptions.

$$\mathbf{T}(K(350\pm 4)\&\mathbf{T}G<a>\&\mathbf{T}O<t1>\&\mathbf{T}(V<\underline{a},12.00,t1>)\&\mathbf{T}(P<\underline{a},29\tfrac{1}{2},t1>)\&\mathbf{T}(R<12,29.5,350(\pm 4),t1>)).$$

to a true C-conditional:

$$\mathbf{T}(K(350\pm 4)\&G<a>\&O<t1>\&V<\underline{a},12.00,t1>\&P<\underline{a},29.1250,t1>)\Rightarrow(R<12.00,29.1250,346>)).$$

is much restricted by Axiom 8-01. Had a TF-conditional been used instead of a C-conditional, this would have followed from all sorts of false data. In fact by allowing the falsehood of readings to imply the truth of the TF-conditional, we could establish all sorts of false laws such as the law that the volume of a gas doubles with increases of pressure. We simply make the pressure reading false, thus the whole antecedent false, thus the TF-conditional true, and let the consequent say whatever we want, including that the volume of the gas equals its volume multiplied by 2.

In a sense, Boyle's law may be viewed as causal law; increased pressure causes the volume of a gas (as opposed that of a liquid or a solid) to decrease. But its importance lies in the precision with which it specifies exactly how much the volume changes with specified amounts of change in pressure. In this respect it is like most non-probabilistic laws of modern physical science, though the latter usually involve much more complicated mathematical functions than Boyle's law does.

9.34 Causal Statements

What is the logic by which people move from observations of facts to conclusions about causal connections?

We begin by focusing on two classes of causal statements which everybody understands, and about which there is general agreement with respect to truth or falsity. Then we will compare the ways M-logic and A-logic would account for the move from the observational evidence to accepting a conclusion that C causes E.

The first class of causal statements is typified by the following examples:

"My turning the-first-light-switch-at-the-entry-of-my-house up,
was the cause of the overhead-light-in-the-entry-way's going on."
"My pressing the key on the keyboard of my computer with 'E' on it
was the cause of the letter 'e's appearing on the screen."
(and similar statements, for every letter on the keyboard, as well
as for various manipulations of function keys with their effects)
"My pulling the handle of the water faucet in my kitchen sink down and forward
caused the water to stop running."
"My pushing my foot on the brake *caused* my car to stop"

Each of these singular causal statements describes an event which actually occurred in the 24 hours prior to my first writing them. In each case, the cause ascribed was some physical act I performed with a physical object. The effect was a physical event which I could not have brought about without the physical machinery which made it possible for my action to cause the effect. But it was my physical act, not the machine, which I correctly said was the cause of the effect. I assume that my reader can produce many similar true first-person causal statements about events in the 24 hours before they read these words.

The causal statements above are related to conditional statements. Let '*a*' and '*b*' stand for events; each of these statements has the form '(*a* caused *b*)'. For illustrative purposes I shall usually use the first statement, where *a* is *my-turning-the-first-light-switch-at-the-entry-of-my-house-up* and *b* is *the-overhead-*

light-in-the-entry-way's-going-on. My assertion that each one of the statements above was true, entails at least two things in each case. Roughly stated they are:

	In Symbols
(i) it is true that *a* occurred and *b* occurred, and	True(O*a* & O*b*)
(ii) if *a* had not occurred, then *b* would not have occurred.	E-valid(If ~O*a* then ~O*b*)

If I reject (i), then I am rejecting that *a actually* caused *b*, i.e., that "*a* caused *b*" is ***true***. If I reject (ii), then I reject the claim that *a* was the *cause* of *b*; I do not deny (i) that *a* occurred and *b* occurred, but by rejecting (ii) I intimate that if *a* had not occurred *b* might have occurred anyway—or that something other than *a,* if anything, was the cause of *b*'s occurring. Thus both (i) and (ii) are part of what I mean in asserting that "*a* caused *b*"is true.

This is not all that I mean by asserting that in fact "*a* caused *b*". I also mean that *a* occurred before *b* occurred (or, certainly not after *b* occurred) and I preuppose, but don't mention, many other conditions that were present and were necessary to make *a* cause *b*. These other parts what I mean by "*a* caused *b*" will only be investigated as far as they impact on the conditionals. Our focus is on the conditional statements entailed or implied by causal statements, and on the conjunctive statements of facts that support them. In each case we study we will assume that the other components of meaning entailed by "*a* caused *b*" are satisfied, so that "*a* caused *b*" is accepted as true or valid, if and only if the conditional clauses we are investigating are also accepted as true or valid.

Regardless of the other components of meaning, (ii) is a contrary-to-fact, subjunctive conditional. By statement (i) *a* occurred and statement (ii) says, given the way things were at the moment of causation, if *a* had *not* occurred, then *b* would *not* have occurred. In other words, the occurrence of *a* was a necessary condition of the occurrence of *b*. What is the status of this contrary-to-fact conditional with respect to truth and falsehood**?**

It depends on how we interpret the 'if…then'.

If 'if ~O*a* then ~O*b*' is a truth-functional conditional the answer is that the contrary-to fact conditional is true. This follows in M-logic by the falsity of its antecedent, which is entailed by the fact that the event *a* did occur. This latter fact makes 'if ~O*a* then ~O*b*' contrary-to-fact.

At first all of this may appear reassuring. For now we can say that the truth of '(*a* caused *b)*' in each case can be logically established in the following way:

1) we start with a definiton
'(*a* caused *b)*' syn_{df} ((i) '*a* occurred and *b* occurred, ((O*a* & O*b*)
& (ii) if *a* had not occurred then *b* would not have occurred) & (~O*a* ⊃ ~O*b*)
& (iii) etc. (the other parts of the meaning)) & P_3)
2) thus '*a* caused *b*' can be judged ***true***, because (along with other things)
(1) it is ***true*** that *a* occurred and *b* occurred, and (True (O*a* & O*b*)
(2) it is ***true*** that (if *a* does not occur, then *b* does not occur). & True (~O*a* ⊃ ~O*b*)
(3) the other parts of the meaning are **true** (by hypothesis) &True P_3)
Hence, by (1), (2) and (3) the definition of '*a* caused *b*' is satisfied and
(4) "*a* caused *b*" is ***true***. **.:** True (*a* caused *b)*

But do we really want to use this line of argument? The only grounds in M-logic for asserting 2) that "(~O*a* ⊃ ~O*b*)" is true, is that O*a* is true in fact and thus by the falsity of its antecedent, "(~O*a* ⊃ ~O*b*)" is true. Using this same line of argument, since

(i) it is false that *a* does not occur (it is false that the the light switch is not turned up), F(~O*a*)
therefore (ii) it is true that *b* does occur (it is true that the light is on), T(O*b*)

In fact from the data under consideration, we can derive four different TF-conditionals: F($\sim$O*a*), M-implies the truth of two TF-conditionals by falsity of the antecedent:

1) "If the light-switch is not-Up, then the light does go ON" is true T($\sim$O*a* $\supset$ O*b*)
2) "If the light-switch is not-Up, then the light does not go ON"is true T($\sim$O*a* $\supset\sim$O*b*)

and T(O*b)*, M-implies the truth of two TF-conditionals, by the truth of the consequent:

3) "If the light-switch is Up, then the light goes ON" is true T(O*a* $\supset$ O*b*)
4) "If the light-switch is not-Up, then the light goes ON" is true) T($\sim$O*a* $\supset$ O*b*)

The conditionals 2) and 3) seem to be compatible with what *"a* caused *b*" means. But 4), which is the same as 1), is not. It supports "If I do not turn the light switch up, then the light goes on". This suggests that in the given situation, turning the light switch up is not necessary to make the light to go on, and that, had I not turned the switch up, the light might have gone on anyway—despite my assertion that my turning the switch up did *cause* the light to go on.

Of course the TF-conditionals only *seem* to have these anomalous consequences to people who don't understand what 'if…then' really means in M-logic. But the conditionals *seem* that way because that is the way we would ordinarily use conditionals, regardless of how M-logic uses them. When the four TF-conditionals are replaced by logically equivalent expressions with 'or' in them, we see what they really mean in ordinary language, namely, T(O*a* or O*b*), T(O*a* or $\sim$O*b*), T($\sim$O*a* or O*b*) and T(O*a* or O*b*). These disjunctions seem to have little to do with the meaning of "*a* caused *b"*.

How are things the same or different if the 'If…then' is interpreted as a C-conditional? In that case F($\sim$O*a*) and T(O*b*) follow logically from premiss T(O*a* & O*b*) as in M-logic, but they do not entail or imply T($\sim$O*a* $\Rightarrow$ O*b*) or T($\sim$O*a* $\Rightarrow\sim$O*b*). So we are not in the awkward position of having to accept that if not-*a* is the case, then *b* occurs anyway. According to A-logic, since the antecedent is not true ($\sim$O*a* $\Rightarrow$ O*b*) and ($\sim$O*a* $\Rightarrow\sim$O*b*) would not be true and would not be false. Nevertheless, an ordinary language conditional of the form '(If $\sim$O*a* then $\sim$O*b*)' is certainly acceptable in some important sense for each statement *"a* caused *b*". In our example, the statement, "If I had not turned the light-switch up, then the entry-light would not have gone on" is an *empirically valid* conditional given the way things have been (and still are, as I write) at my house, even though I don't call such a C-conditionals *true* because the antecedent is false.

Let us shift the analysis to a second, somewhat different, set of statements, closely related to the first set. We replace the form '*a* cause**d** *b*' with the form '*a* cause**s** *b*'. The resulting statements are ones I would have made without hestitation, and would make now, before or after the causal events described in the first set. These statements express ideas about how things work. I carry them around with me in my head when I am away from my house, or don't know what is actually happening, or want to tell someone what to do to bring about or avoid the mentioned effects.

"Turning the-first-light-switch-at-the-entry-of-my-house up,
causes the overhead-light-in-the-entry-way's going on."
"Pressing the key on the keyboard of my computer with 'E' on it
causes the letter 'e's appearing on the screen."
(and similar statements, for every letter on the keyboard)
"Pushing the handle of the water faucet in my kitchen sink up and to the left
causes the hot water's starting to come out of the faucet"
"Pushing one's foot on the brake ***causes*** my car to stop"

These statements have a kind of generality. They are no longer first person statements about single events. They represent principles which are useful to me, but could be useful to anyone who uses my house or car. This is not absolute or universal generality, for they are about the very local conditions at my particular house, and their reliability depends upon "things staying the way they are (or were)" in all relevant respects in my house and with my car. But normally, within that context, they are dependable, empirically valid statements that can be relied on from day to day.

What is meant by these causal statements in each case is partially expressed by two conditional statements, without any assertion of the actual truth of antecedent or consequent. Both may be viewed as subjunctive or counterfactual. Where *a* is the cause and *b* is the effect,

"*a* causes *b*" cont_{df}	(i) if *a* occurs then *b* occurs, and	((If O*a* then O*b*)
	(ii) if *a* does not occur, then *b* does not occur.	& (If ~O*a* then ~O*b*))

To include the idea that a cause can not come after its effect, we treat 'O' as a dyadic predicate, introducing 't_i' and 't_{i+}' to get $(Oat_i \Rightarrow Obt_{i+})$ and 't_j' and 't_{j+}' to get $({\sim}Oat_j \Rightarrow {\sim}Obt_{j+})$. Thus '$(Oat_1 \Rightarrow Obt_{1+})$' abbreviates '(if ($t_1$ is a time & *a* occurs at t_1) then (t_{1+} is a time at or after t_1 & *b* occurs at t_{1+}))'. In other words, "if *a* occurs at time t, *b* occurs at a specifed later time t+." According to A-logic, $(Oa \Rightarrow Ob)$ and $({\sim}Oa \Rightarrow {\sim}Ob)$ can not both be true of the same objects at the same time, for their conjunction would be synonymous (by Ax.8-01.Ax.7-03 and Df 'F') with (TO*a* & TO*b* & FO*a* & FO*b*). But $(Oat_i \Rightarrow Obt_{i+})$ and $({\sim}Oat_j \Rightarrow {\sim}Obt_{j+})$ can both be true without contradiction if t_i is not the same time as t_j, even though *a* and *b* retain their identities as event-types. For example, "Pushing-the-first-light-switch-in-the-entryway-of-my-house -**up**" is a unique, individual event-type, which <u>can not</u> occur at the same time as the unique event-type "Pushing-the- first-light-switch-in-the-entryway-of-my-house-**down**"occurs, but these event-types, with their consequences, can occur at different times repeatedly. (By contrast, M-logic's $(Oat_i \supset Obt_{i+})$ and $({\sim}Oat_j \supset {\sim}Obt_{j+})$ <u>can</u> be true at the same time, if j = i. For both are implied if either $(Oat_i \ \&\ Obt_{i+})$ or $({\sim}Oat_i \ \&\ {\sim}Obt_{i+})$ is true, by the falsity of the antecedent or the truth of the consquequent.)

Thus we partially re-define the conditionals involved in the idea that "*a* causes *b*" as follows:

['*a* causes *b'* Cont_{df} '(If Oat_i then Obt_{i+} & If ${\sim}Oat_j$ then ${\sim}Obt_{j+}$))'],

An important advantage of introducing 't' and 't_+', is that it forestalls deriving the truth or validity of both "*a* causes *b*" and "*b* causes *a*" from $T(Oat_i \ \&\ Obt_{i+})$ and $T({\sim}Oat_j \ \&\ {\sim}Obt_{j+})$). For in A-logic and M-logic the truth of (P & Q) entails or implies that both "If P then Q" and "If Q then P" are true; in A-logic by Ax.8-01, [T(P ⇒ Q) Syn T(P & Q)], and &-COMM, and in M-logic, where sentences are implicitly asumed to be truth-assertions, (P & Q) implies both (P ⊃ Q) and (Q ⊃ P) by falsity of antecedent or truth of consequent. The new definition indicates, in both conditionals (i) and (ii), that *b* must not occur before *a*. From $T(Oat_1 \ \&\ Obt_{1+})$, both $T(Oat_i \Rightarrow Obt_{i+})$ and $T(Obt_{i+} \Rightarrow Oat_i)$ both still follow of course. But only the first of these two conditionals satisfies conditional (i) because only the first conditional preserves the temporal priority of the antecedent cause. Evidence for the E-validity of the converse causation, "*b* causes *a*" requires $T(Obt_i \ \&\ Oat_{i+})$ and $({\sim}Obt_j \ \&\ {\sim}Oat_{j+})$ from which the appropriate conditionals (i) and (ii) follow.

We next examine the problem of analyzing causal statements in M-logic in more detail. Then we will present A-logic's account of causal statements and the logic by which their empirical-validity can be confirmed or disconformied.

9.341 The Problem of Causal Statements in M-logic

M-logic can not give a coherent account of the inference from true observations to the conditionals in empirically valid causal statements.

We assume the connection between pre-analytic 'if…then' statements and statements of the form '*a* causes *b*' remains constant. But in M-logic the meaning of 'if…then' and the logical connection between conjunctive and conditional truth-claims differs. Interpreting 'if…then' as a truth-functional conditional, we have

"*a* causes *b*" cont_{df} (i) if *a* occurs then *b* occurs, and $(\ (Oat_1 \supset Obt_{1+})$
and (ii) if *a* does not occur, then *b* does not occur. $\&\ (\sim Oat_2 \supset \sim Obt_{2+}))$

If we also assume, as Hume and Mill did, that the evidence for inferring that '*a* causes *b*' consists, on the one hand, in finding that *a* and *b* occur close together, i.e., $T(Oat_1\ \&\ Obt_{1+})$, and on the other hand, that *b* does not occur if *a* does not occur, i.e., $T(\sim Oat_2\ \&\ Obt_{2+})$, then interpreting causal statements by means of truth-functional conditionals leads to chaos.

The premisses from which conclusions about cause and effect are inferred—conjunctions which describe the occurrences of cause and effect together, or the absence of the effect with the absence of cause—imply too many TF-conditionals. The data according to M-logic would support causal relations which are the opposite of what rational people normally infer from that data.

In A-logic, finding that *a* occurs at t_1 and *b* occurs at t_{1+} entails just one C-conditional. The same datum implies three TF-conditionals in M-logic, by falsity of the antecedent or the truth of the consequent:

In A-logic $T(Oat_1\&\ Obt_{1+})$ Entails (i) $T\ (Oat_1 \Rightarrow Obt_{1+})$
In M-logic $T(Oat_1\&\ Obt_{1+})$ Implies $T\ (Oat_1 \supset Obt_{1+})$ By truth of the consequent
and $\mathbf{T\ (\sim Oat_1 \supset Obt_{1+})}$ or falsity of the antecedent.
and $T\ (\sim Oat_1 \supset \sim Obt_{1+}))$ By falsity of the antecedent.

Likewise, in A-logic, finding that *a* **does not** occur at t_2 and *b* **does not** occur at t_{2+} entails just one C-conditional, while in M-logic the same observation implies three different conditionals.

In A-logic $T(\sim Oat_2\ \&\ \sim Obt_{2+})$ Entails $T\ (\sim Oat_2 \Rightarrow \sim Obt_{2+})$
In M-logic $T(\sim Oat_2\ \&\ \sim Obt_{2+})$ Implies $T\ (\sim Oat_2 \supset \sim Obt_{2+})$ By truth of the consequent
and $\mathbf{T\ (Oat_2 \supset \sim Obt_{2+})}$ or falsity of the antecedent.
and $T\ (Oat_2 \supset Obt_{2+}))$ By falsity of the antecedent

Thus the *same evidence,* $T(Oat_1\&\ Obt_{1+})$ which, in A-logic supports only

(i) "It is true that if *a* occurs at t_1 *then* *b* occurs at t_{1+}" i.e., $T((Oat_1) \Rightarrow Obt_{1+})$,
according to M-logic also supports (by truth of the consequent only) the conditional

"It is true that if *a* does **not occur** at t_1 then *b* **occurs** at t_{1+}". i.e., $T(\sim Oat_{)} \supset Obt_{1+})$,
and the *same evidence*, namely $T(\sim Oat_2\ \&\ \sim Obt_{2+})$, which, in A-logic supports only
(ii) "It is true that if *a* **does not** occur at t_2 then *b* **does not** occur at t_{2+}", i.e., $T(\sim Oat_2 \Rightarrow \sim Obt_{2+})$,
according to M-logic also supports (by falsity of the antecedent only) the Tf-conditional

"It is true that if *a* occurs at t_2 then *b* **does not** occur at t_{2+}". i.e., $T(Oat_2 \supset \sim Obt_{2+})$

These extra conditionals which are implied in M-logic by the data are the ones which we associate pre-analytically with the opposite of *"a* causes *b",* namely with "*a* causes ***not***-*b*". They are the conditionals (i) If *a* occurs, then *b* does not occur, and (ii) if *a* does not occur, then *b* occurs.

Further, it is customarily said that when all other necessary conditions are present, if *a* is observed to occur, but is not followed by the occurrence of *b*, and if *b* is observed to occur without *a*, such observations would disprove that *a* causes *b*. A-logic supports this insight. It has:

$$\text{T }(\text{O}at_1 \ \& \ {\sim}\text{O}bt_{1+}) \text{ entails (i) } (\text{O}at_1 \Rightarrow \text{O}bt_{1+}) \text{ is false,} \qquad \text{[by Ax.8-02]}$$
$$\text{and T }({\sim}\text{O}at_2 \ \& \ \text{O}bt_{2+}) \text{ entails (ii) } ({\sim}\text{O}at_2 \Rightarrow {\sim}\text{O}bt_{2+}) \text{ is false,} \qquad \text{[by Ax.8-02,DN]}$$

Thus according to A-logic, in these cases the data entails the falsehood of conditionals entailed by "*a* causes *b*", and thus establishes that "*a* causes *b*" is false. But according to M-logic, the same two bits of evidence imply that the following two conditionals, as TF-conditionals, are both true. In pre-analytic ordinary language, the first two conditionals would confirm that "*a* causes *b*" is true.

$$\text{T }(\text{O}at_1 \& \ {\sim}\text{O}bt_{1+}) \text{ implies (ii) } ({\sim}\text{O}at_1 \supset {\sim}\text{O}bt_{1+}) \text{ is true,} \quad \text{[by truth of its consequent, } {\sim}Obt_{1+}]$$
$$\text{and T }({\sim}\text{O}at_2 \& \ \text{O}bt_{2+}) \text{ implies (i) } (\text{O}at_2 \supset \text{O}bt_{2+}) \text{ is true,} \quad \text{[by truth of its consequent, O}bt_{2+}],$$

But the same data, according to M-logic, also supports TF-conditionals for "*a* causes not-*b*":

$$\text{T }(\text{O}at_1 \& \ {\sim}\text{O}bt_{1+}) \text{ implies (ii) } (\text{O}at_1 \supset {\sim}\text{O}bt_{1+}) \text{ is true,} \quad \text{[by truth of its consequent, } {\sim}Obt_{1+}]$$
$$\text{and T }({\sim}\text{O}at_2 \& \ \text{O}bt_{2+}) \text{ implies (i) } ({\sim}\text{O}at_2 \supset \text{O}bt_{2+}) \text{ is true,} \quad \text{[by truth of its consequent, O}bt_{2+}],$$

So the effort to show how causal statements are supported or established by deriving their implicit conditional statements from observed truths, ends in a choas of opposing results if the conditionals are construed as TF-conditionals. M-logic derives too many TF-conditionals from too sparse facts. Neither the truth of a consequent by itself, nor the falsehood of an antecedent by itself is sufficient to establish the truth of a conditional in any ordinary sense. When the contrary view is adopted, the isolated facts produce too many conditionals.

9.342 Causal Statements Analyzed with C-conditionals

A-logic, with C-conditionals, does not have these results. Both antecedent and consequent must be true, if a C-conditional is to be true. The antecedent must be true and the consequent false, if the conditional is to be false.

In A-logic's account the key lies in Axioms.8-01 and 8-02. By U-SUB these yield,

$\text{T}(\ \text{O}at_1 \ \& \ \text{O}bt_{1+}) \text{ Syn T}(\text{O}at_1 \Rightarrow \text{O}bt_{1+})$ $\qquad$ $\text{T}(\ \text{O}at_1 \ \& \ {\sim}\text{O}bt_{1+}) \text{ Syn F}(\ \text{O}at_1 \Rightarrow {\sim}\text{O}bt_{1+})$
$\text{T}({\sim}\text{O}at_2 \ \& {\sim}\text{O}bt_{2+}) \text{ Syn T}({\sim}\text{O}at_2 \Rightarrow {\sim}\text{O}b_{2+})$ $\qquad$ $\text{T}({\sim}\text{O}at_2 \ \& \ {\sim}\text{O}b_{2+}) \text{ Syn F}({\sim}\text{O}at_2 \Rightarrow \text{O}b_{2+})$
$\text{T}(\ \text{O}at_3 \ \& {\sim}\text{O}bt_{3+}) \text{ Syn T}(\text{O}at_3 \Rightarrow {\sim}\text{O}b_{3+})$ $\qquad$ $\text{T}(\ \text{O}at_3 \ \& \ {\sim}\text{O}b_{3+}) \text{ Syn F}(\ \text{O}at_3 \Rightarrow \text{O}b_{3+})$
$\text{T}({\sim}\text{O}at_4 \ \& \ \text{O}bt_{4+}) \text{ Syn T}({\sim}\text{O}at_4 \Rightarrow \text{O}b_{4+})$ $\qquad$ $\text{T}({\sim}\text{O}at_4 \ \& \ {\sim}\text{O}b_{4+}) \text{ Syn F}({\sim}\text{O}at_4 \Rightarrow {\sim}\text{O}b_{4+})$

Whether "*a* causes *b*" is *empirically valid* or not depends on finding instances in which conditionals (i) and (ii) are *true*. The evidence that makes them true is expressed by conjunctive statements of the forms $\text{T}(\text{O}at_1 \ \& \ \text{O}bt_{1+})$, $\text{T}({\sim}\text{O}at_2 \ \& {\sim}\text{O}bt_{2\ +})$, etc. The truth or falsehood of C-conditionals then follows by SynSUB from the Syn-theorems above.

However, the picture presented so far is too simple. The full meaning of "*a* causes *b*" has at least three other meaning components that require a more complicated account of the conditionals (i) and (ii). First, there is the generality aspect of causal statements. Secondly, we must consider relationships of causal statements and conditionals to empirical validity and truth. Finally, there is a *ceteris paribus* aspect when talking about actual, empirically supported causal principles.

1. Statements of the form "*a* causes *b*" involve generality. To say "*a* causes *b*" is to say that at all times, *a* causes *b,* provided that all necessary conditions for *a* to cause *b* are present. Ignoring the last proviso for the moment, to generalize from particular cases we must quantify over times. We drop the subscripts *i* and *j* and use 't' and 't_+' to get, $(\forall t)((Oat \Rightarrow Obt_+)$, $(\exists t)(Oat \Rightarrow Obt_+)$ or $(\forall t)(\sim Oat \Rightarrow \sim Obt_+)$, etc. The Boolean expansion of $(\forall t)(Oat \Rightarrow Obt_+)$ is a conjunction of particular instances, in each of which the effect *b* comes after the cause *a*:

$$(\forall t)(Oat \Rightarrow Obt_+) \text{ Syn } ((Oat_1 \Rightarrow Obt_{1+}) \;\&\; (Oat_2 \Rightarrow Obt_{2+}) \;\&\; (Oat_3 \Rightarrow Obt_{3+}) \text{ v } \ldots) .$$

The Boolean expansion of $(\exists t)(Oat \Rightarrow Obt_+)$ is a disjunction of particular instances with the effect *b* comeing after the cause *a* in each instance:

$$(\exists t)(Oat \Rightarrow Obt_+) \text{ Syn } ((Oat_1 \Rightarrow Obt_{1+}) \text{ v } (Oat_2 \Rightarrow Obt_{2+}) \text{ v } (Oat_3 \Rightarrow Obt_{3+}) \text{ v } \ldots) .$$

2. Our central concern is in the relation of assertions of causal connections to *truth*; i.e., the logical steps from *true* observation statements to an *empirically valid* (or "true") causal statement.

In ordinary discourse we tend to say that "*a* causes *b*" is *true*, and thus each of the conditional statements (i) and (ii) must be *true*. But we use the term *E-valid* instead of *true*, since we must distinguish finding a conditional *true* from concluding that a quantified C-conditional is *empirically valid* on the basis of instances like $T(Oat_i \;\& Obt_{i+})$ or $T(\sim Oat_j \;\& \sim Obt_{j+})$.

To find a conditional $(Oat_i \Rightarrow Obt_{i+})$ *true* is to find Oat_i *true* and Obt_{i+} *true*. This is what Axiom.8-01, $[T(P \Rightarrow Q) \text{ Syn } T(P \;\&\; Q)]$ says. Since both conditionals (i) and (ii) can not be true at the same time, at least one of them must be neither true nor false (hence not-true) at any given time. Further, at least one of them must be a contrary-to-fact conditional; sometimes both are. In asserting *a causes b* we often don't know (and it does not matter) whether the antecedent, or the consequent, or the whole conditional, is *true* at that moment. Nevertheless we may have very good reasons to accept both (i) and (ii) as E-valid; this is essential if we are to find "*a* causes *b*" E-valid.

We also avoid ascribing *truth* to a general causal statement because to establish "*a* causes *b*" is *true,* we would have to prove that (i) $(\forall t)T(Oat \Rightarrow Obt_+)$ is *true* and (ii) $(\forall t)T(\sim Oat \Rightarrow \sim Obt_+)$ is *true*. According to A-logic, both of these statements are false in practically every case. For the Boolean expansion of $T(\forall t)(Oat \Rightarrow Obt_+)$ is

$T(Oat_1 \Rightarrow Obt_{1+}) \;\&\; T(Oat_2 \Rightarrow Obt_{2+}) \;\&\; T(Oat_3 \Rightarrow Obt_{3+}) \;\&\; \ldots)$

which is synonymous by Ax.8-01 and Ax.7-03 to

$(TOat_1 \;\&\; TObt_{1+} \;\&\; TOat_2 \;\&TObt_{2+} \;\&\; TOat_3 \;\&TObt_{3+} \;\&\; \ldots)$, which asserts falsely that *a* occurs at every moment and that *b* occurs at a specified time at or after that moment.

Thus we will treat sound causal statements as '*empirically valid*' rather than as '*true*'.

The concept of the *empirical validity* of a causal statement requires that we distinguish the meaning of "*a* causes *b*"—mere ideas of causation—from findings of actual causation based on empirical observation. When the field of reference is the actual world so that 'T' stands for truths about events in the actual world, there is a difference between causal statements *per se*, and *empirically valid* causal statments. In the story of Aladdin, if Aladdin rubs his lamp, the genii appears immediately, and if he does not rub the lamp the genii does not appear. This satisfies the causal concept; Aladdin's rubbing that lamp *causes* the genii to appear. But this is not accepted as a true or valid causal statement about the actual world. I can form and express the idea of a causal connection between any kind of activity in my house and the entry-way-light's-turning-on: "waving my hand causes it to go on", "saying "abracadabra" causes it to go on", "*willing* that it go on, causes it to go on", etc, All of these conform to the idea of "*a* causes *b*". Their meaning is clear. But none of them are *empirically valid* because there is a lack of sufficient relevant instances in which '$(\mathrm{O}at_i \,\&\, \mathrm{O}bt_{i+})$' and '$(\sim\mathrm{O}at_j \,\&\, \sim\mathrm{O}bt_{j+})$' are *true*. The expressions "*a* causes *b*" and " '*a* causes *b'* is E-valid (or true)" do not mean the same thing.

Empirical validity is based on truths. To see how it is based on truths, we note first that to say that (*a* causes *b*) is E-valid entails that both of the conditionals, (i) and (ii), are E-valid.

$$\begin{aligned}&[\text{E-valid } (a \text{ causes } b) \text{ Cont} \quad \text{(i)} \;\; (\text{ E-valid } (\forall t)(\mathrm{O}at \Rightarrow \mathrm{O}bt_+)\\ &\qquad\qquad\qquad\qquad\qquad\qquad \text{(ii)} \;\; \& \text{ E-valid } (\forall t)(\sim\mathrm{O}at \Rightarrow \sim\mathrm{O}bt_+))]\end{aligned}$$

Then, applying the definition of E-validity,

$$\mathrm{Df}_1 \text{ 'E-Valid': } [\text{E-Valid}(\forall x)(Px \Rightarrow Qx) \;\mathrm{Syn}_{df}\; (T(\exists x)(Px \Rightarrow Qx) \;\&\; \sim F(\forall x)(Px \Rightarrow Qx))]$$ [17]

to both (i) and (ii), we get four statements by substitution which must be supported by true observation statements, if "*a* causes *b*" is to be E-valid:

$$\begin{aligned}&\text{(i) (E-valid } (\forall t)(\mathrm{O}at \Rightarrow \mathrm{O}bt_+) && \text{Syn} \quad \text{(i,a)} \quad ((\;\; T(\exists t)(\mathrm{O}at \Rightarrow \mathrm{O}bt_+)\\ &&& \qquad\quad\; \text{(i,b)} \quad \& \sim F(\forall t)(\mathrm{O}at \Rightarrow \mathrm{O}bt_+))\\ &\text{(ii) \& E-valid } (\forall t)(\sim\mathrm{O}at \Rightarrow \sim\mathrm{O}bt_+)) && \text{Syn} \quad \text{(ii,a)} \quad \& ((T(\exists t)(\sim\mathrm{O}at \Rightarrow \sim\mathrm{O}bt_+)\\ &&& \qquad\quad\; \text{(ii,b)} \quad \& \sim F(\forall t)(\sim\mathrm{O}at \Rightarrow \sim\mathrm{O}bt_+)))\end{aligned}$$

We will discuss these and their relationships to observation statements shortly. But first, to complete this account of causal statements, we must introduce a "*ceteris paribus* clause".

3. If any particular actual instantiation of "*a* causes *b*" is *empirically valid*, it presupposes an indefinitely defined set of *necessary* conditions which are actually present, and upon which the causal connection depends. These conditions are not part of the *meaning* of the instantiations of *'a'* or *'b'* or "*a* causes *b*". They are part of the actual situation or factual context that the speaker refers to when asserting some instantiation of "*a* actually causes *b*" or " '*a* causes *b*' is empirically valid".

The presupposed states of affairs are sometimes referred to by the phrase "given the way things are" or "*ceteris paribus*", which means "other things being equal". But in this context it doesn't mean that *all* other things must be the same on each occasion. I can move the furniture around in my house, and this will not affect the *empirical validity* of "pushing-the-handle-of-the- water-faucet-in-my-kitchen-sink-up-and-to-the-left *causes* the-hot-water's-starting-to-come-out-of- the-faucet". Rather, in each case there is an indefinitely defined group of unique *necessary conditions* which have to be present for this particular *a* to cause this particular *b*. These include things we know about as well as things beyond our control or knowledge. For example, my assertion that,

"Turning the-first-light-switch-at-the-entry-of-my-house up,
causes the overhead-light-in-the-entry-way's going on,"

---

17. Cf. this book, page 488.

holds on the presuppositions that the power is on, that the light bulb is not burned out, that the wiring is not broken at some point. And the claim that

"Pushing ones foot on the brake *causes* my car to stop"

presupposes a situation in which the brake pedal is properly connected with the braking system, that the brake shoes on the wheels are not worn, etc. Each different empirically valid statement of the form "*a* causes *b*" will have an implicit '*ceteris paribus* clause' which stands for a different set of necessary conditions some of which I know about. But there are always more than I can mention.

Let SC_1 stand for the full state of affairs which is actually both necessary and sufficient *at this time* to cause E, the-turning-on-of-the-light-in-the entryway-in-my-house. SC_1 is a conjunction of necessary conditions, (NC_1 & NC_2 & NC_3 & NC_4 &....). The conjunction of all these necessary conditions together constitute SC_1 . To say SC_1 causes E is to say that E occurs if and only if all of the necessary conditions exist. To represent the way things actually are at my house (and have been in the six years I have lived here), let 'NC_1'stand for "the electric power's being on", 'NC_2' for "the wiring's existing from the switch to the light-bulb in the entry-light", 'NC_3'for "the light-bulb's being in working order", 'NC_4' for "the switch's being up". The dots, '....', at the end stand for indefinitely many other necessary conditions—conditions in the absence of which, SC_1 would not be sufficient to cause the light to turn on. The wire must have a certain conductivity, it must have a certain chemical structure, every segment of the wire between the power station and the light bulb must be connected to the next segment, it must be insulated, the various parts of machines in the power station must be in place... the earth on which my house is located must not be too close to the sun, or too far away from it,... etc. The complete description of SC_1 is impossible or impractical to list. SC_1 refers to an complex, incompletely described, *actual* condition that is both necessary and sufficient to cause the-turning-on-of-the-light-in-the entryway-of-my-house, given the ways things are at this time.

Relative to the effect E (my overhead entry-light's being on), *theoretically* there are many different sufficient conditions which might cause E. If my house had been wired differently, the entry light might have been connected to the second switch as I enter the front door. Then my turning the first switch up would not be either necessary or sufficient in any sense to make the entry light turn on. Or if the entry light were run on a battery independent of any electric utility company, the utility's-power's-being-on, would not be necessary, etc. Thus the particular total actual state of affairs which is sufficient to make the entry light go on in my house at this time is not an absolutely necessary state of affairs; if things were not the way they are in fact, many other arrangements might have been, and might be in the future, sufficient to cause E.

Actually, normally from day-to-day, I assume that, except for NC_4 (the switch's-being-up), all the other particular necessary conditions (NC_1 & NC_2 & NC_3 &___ & ....) of SC_1 which obtain at this time in my house, exist. When I say "my pushing-the-light-switch-up *causes* the-entry- light's- going-on", abbreviated as 'NC_4 causes E', I mean that, assuming all necessary conditions in SC_1 *except for* NC_4 exist, the event which makes NC_4 become true, (my pushing the light switch up) is what causes E (the entry-light's turning on); **and** assuming all conjuncts in SC_1 *except for* NC_4 are true, if NC_4 is not made true, E will not become true.

Let 'A_{ONCE}' stand for '**A**ll **o**ther **a**ctually **n**ecessary **c**onditions for producing E", so that '(A_{ONCE} & NC_4)' is the same as SC_1, and '(A_{ONCE} & $\sim NC_4$)' is the same as SC_1 *without* NC_4. Then a more complete account of the two conditionals implicit in '*a* causes *b*' requires the inclusion of a *ceteris paribus* clause, A_{ONCE}, in the antecedents of conditionals (i) and (ii). In general,

"*a* causes *b*" cont_{df} "$((\forall t)((A_{ONCE} \,\&\, Oat_i\,) \Rightarrow Obt_+\,) \,\&\, (\forall t)((A_{ONCE} \,\&\, \sim Oat\,) \Rightarrow \sim Obt_+)))$

Thus incorporating the *ceteris paribus* clause into the meaning of the claim that "*a* causes *b*" is empirically valid, we have,

"E-valid(*a* causes *b)*" cont_{df} "(i) (E-valid $(\forall t)((A_{ONCE}\ \&\ Oat_i) \Rightarrow Obt_+)$
(ii) & E-valid $(\forall t)((A_{ONCE}\ \&\ {\sim}Oat) \Rightarrow {\sim}Obt_+))$

and the two conditionals entail the conjunction of four quantified C-conditionals:

(i) (E-valid $(\forall t)T((A_{ONCE}\ \&\ Oat) \Rightarrow Obt_+)$ Syn (i,a) (($T(\exists t)((A_{ONCE}\ \&\ Oat) \Rightarrow Obt_+)$
(i,b) & $\sim F(\forall t)((A_{ONCE}\ \&\ Oat) \Rightarrow Obt_+))$
(ii) & E-valid $(\forall t)T((A_{ONCE}\ \&{\sim}Oat) \Rightarrow{\sim}Obt_+))$ Syn (ii,a) &(($T(\exists t)((A_{ONCE}\ \&{\sim}Oat) \Rightarrow{\sim}Obt_+)$
(ii,b) & $\sim F(\forall t)((A_{ONCE}\ \&{\sim}Oat) \Rightarrow{\sim}Obt_+)))$

The proof or confirmation that "*a* causes *b*" is *E-valid,* entails the truth or non-falsehood of the four conditionals (i,a), (i,b), (ii,a), and (ii,b).

This formulation allows for the possibility that on other non-normal occasions a different cause will be assigned. For example, I might say, correctly, that *the electric-power's-being-on* causes *the-entry-way-light's-going-on* in some situations. Imagine that all other necessary conditions in SC_1, including NC_4 (the-switch-is-up) exist, except NC_1—the electric power keeps going on and off. In this situation A_{ONCE} (i.e., all other necessary conditions of the set-up, SC_1), is '(___& NC_2 & NC_3 & NC_4 &....)' so that (A_{ONCE} & NC_1) is the same as SC_1. The *E-validity* of "NC_1 causes E" is established by the *E-validity* of "(E-valid $(\forall t)((A_{ONCE}\ \&O(NC_1,t) \Rightarrow O(E,t_+))$ & E-valid $(\forall t)((A_{ONCE}\ \&{\sim}O(NC_1,t) \Rightarrow {\sim}O(E,t_+))$. Instead of my normally E-valid statement that turning the light switch up (NC_4) causes the light to go on, we would have an E-valid statement that, *the-electric-power's-being on* (NC_1) causes *the light-in-the-entry-to-go-on* (E)—as long as that situation persists.

This completes our analysis of the meaning of "'*a* causes *b*' is *empirically valid*" as it applies to the narrowly defined kinds of local causal statements illustrated by the examples above.

We pause to introduce some remarks on Necessary and Sufficient Conditions.

J. L. Mackie has said that when we assert that C actually caused E in a particular case, C is best understood as an "INUS" condition—that is, C is an **i**nsufficient but **n**ecessary condition of an **u**n-necessary, but **s**ufficient condition.[18]

At first this seems to fit pretty well the analysis we have given. The condition NC_4 is **i**nsufficient to cause E by itself without the other necessary conditions, but *is* a **n**ecessary condition of a sufficient condition SC_1. The complex state of affairs SC_1 is objectively **u**n-necessary for E, since we might have had a different wiring system, or a different power source, etc., but it is a **s**ufficient condition for E (turning on the light).

However, there are several ambiguous terms in Mackie's account. There is a sense in which, *given the way things are at my house now*, NC_4 (turning the light switch up) is *sufficient* to turn on the entry-light. It is also necessary; there is no other way to turn that light on. Thus NC_4 is *actually* both sufficient and necessary to turn the entry-light on, rather than being **i**nsufficient and necessary. This means we are using 'sufficient' in a different sense than the sense in which Mackie says NC_4 is **i**nsufficient. Also given the way things actually are at my house now, since there is no other way to turn the entry light on, it is

18. See J. L. Mackie, "Causes and Conditions" in *Causations and Conditionals*, edited by Ernest Sosa, Oxford University Press, 1975, p 16.

necessary that the whole set of conditions (NC_1 & NC_2 & NC_3 & NC_4 &….) which constitute SC_1 co-exist if the light is to go on; i.e., SC_1 is *actually* a necessary condition of the entry-light's being on rather than being **u**n-necessary as well as being sufficient. This is clearly a different sense of 'necessary' than the sense in which Mackie thinks of SC_1 as **u**n-necessary.

These ambiguities can be resolved by distinquishing what is "actually necessary" in the actual spatio-temporal situation at my house now, from what is "theoretically necessary". Both *truth* and *empirical validity* are based on what is *actually* the case. Thus NC_4 is *actually* sufficient *given the way things are at my house now*, but *theoretically* would be insufficient if any one or more of the other necessary conditions NC_1, NC_2, or NC_3 *had not been present*. Again, SC_1 is *actually* necessary in the sense that, *given the way things are at my house now*, if SC_1 (the complex conjunction of all necessary components as a whole) does not exist, then the effect E, does not take place. But *theoretically* SC is not necessary, since *had things been different than they are in fact*, another complex state of affairs would be have been sufficient to turn the entry light on.

Thus if we call NC_4 an INUS condition of E, we are saying that NC_4, (my-pushing-the-first- light-switch-up) is *theoretically **i**nsufficient* (since if the other necessary conditions were not met, it would not be sufficient, although it is *actually* sufficient since other necessary conditions are met) but *actually a **n**ecessary condition of SC_1* given the way things are at my house now (though it is theoretically not necessary since there might have been another wiring arrangement), and SC_1 *is theoretically **u**nnecessary* (since the effect could be gotten by other arrangements though actually, the way things are at my house it is a necessary condition for producing E) *though SC_1 is actually sufficient to cause E (*as well as being theoretically sufficient—if we could spell out all the other necessary conditions).

Note that the *theoretically* necessary or sufficient conditions above are described by contrary-to-fact conditionals. In these paragraphs "theoretically" means "hypothetically, with a contrary-to-fact C-conditional". To establish that a condition is *actually* sufficient or necessary, we must establish the *truth* of conditionals $T(X \Rightarrow E)$ or $T({\sim}X \Rightarrow {\sim}E)$, but to establish the E-validity of statements that condition is *theoretically* sufficient or necessary, we need only the establish the validity of inferential conditionals $(TX \Rightarrow TE)$ or $(T{\sim}X \Rightarrow T{\sim}E)$, which depend on defined or derived synonymy or containment theorems, without finding that these conditionals are *true*.

We turn finally to the logical steps, according to A-logic, by which we can move from observational *truths* to a conclusion that "*a* causes *b*" is *empirically valid*.

To clarify the process, we first reduce the four conditionals that the evidence must support to synonymous non-conditional forms which are directly related to assertions of particular truths. Largely because of Ax. 8-01 and Ax. 8-02, each of the four quantified conditionals wffs (i,a), (i,b), (ii,a) and (ii,b) are reducible to truth-assertions about simple conjunctions:

For (i,a): $[\ T(\exists t)((A_{ONCE} \,\&\, Oat) \Rightarrow Obt_+)$ Syn $(\exists t)T(A_{ONCE} \,\&\, Oat \,\&\, Obt_+)]$
For (ii,a): $[\ T(\exists t)((A_{ONCE} \,\&\, {\sim}Oat) \Rightarrow {\sim}Obt_+)$ Syn $(\exists t)T(A_{ONCE} \,\&\, {\sim}Oat \,\&\, {\sim}Obt_+)]$
For (i,b): $[{\sim}F(\forall t)((A_{ONCE} \,\&\, Oat) \Rightarrow Obt_+))$ Syn $(\forall t){\sim}T(A_{ONCE} \,\&\, Oat \,\&\, {\sim}Obt_+)]$
For (ii,b): $[{\sim}F(\forall t)((A_{ONCE} \,\&\, {\sim}Oat) \Rightarrow {\sim}Obt_+)))$ Syn $(\forall t){\sim}T(A_{ONCE} \,\&\, {\sim}Oat \,\&\, Obt_+)]$

The proofs of these four synonymies are simple:

For (i,a): $T(\exists t)((A_{ONCE} \,\&\, Oat) \Rightarrow Obt_+)$ Syn $(\exists t)T(A_{ONCE} \,\&\, Oat \,\&\, Obt_+)$

Proof: 1) $T(\exists t)((A_{ONCE} \,\&\, Oat) \Rightarrow Obt_+)$ Syn $(\exists t)T((A_{ONCE} \,\&\, Oat) \Rightarrow Obt_+)$
[T7-25. [$T(\exists x)Px$ Syn $(\exists x)TPx$], U-SUB]

2) $T(\exists t)((A_{ONCE} \,\&\, Oat) \Rightarrow Obt_+)$ Syn $(\exists t)T((A_{ONCE} \,\&\, Oat) \,\&\, Obt_+)$ [1),T8-01, SynSUB]

For (ii,a):$T(\exists t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})$ Syn $(\exists t)(T(A_{ONCE} \& \sim Oat \& \sim Obt_{+})$
Proof: 1) $T(\exists t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})$ Syn $(\exists t)T((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})$
[T7-25. [T(∃x)Px Syn (∃x)TPx], U-SUB]
2) $T(\exists t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})$ Syn $(\exists t)T(A_{ONCE} \& \sim Oat \& \sim Obt_{+})$ [1),T8-01, SynSUB]

For (i,b): $\sim F(\forall t)((A_{ONCE} \& Oat) \Rightarrow Obt_{+}))$ Syn $\sim(\exists t)(T((A_{ONCE} \& Oat) \& \sim Obt_{+}))$
Proof: 1) $\sim F(\forall t)((A_{ONCE} \& Oat) \Rightarrow Obt_{+}))$ Syn $(\forall t)\sim F((A_{ONCE} \& Oat) \Rightarrow Obt_{+}))$
[T7-26. [~F(∀x)Px Syn (∀x)~FPx], U-SUB]
2) $\sim F(\forall t)((A_{ONCE} \& Oat) \Rightarrow Obt_{+}))$ Syn $(\forall t)\sim T((A_{ONCE} \& Oat) \& \sim Obt_{+}))$
[1), Ax.8-02,SynSUB]

For (ii,b): $\sim F(\forall t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+}))$ Syn $\sim(\exists t)T(A_{ONCE} \& Oat \& \sim Obt_{+})$
Proof: 1) $\sim F(\forall t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+}))$ Syn $(\forall t)\sim F((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+}))$
[T7-26. [~F(∀x)Px Syn (∀x)~FPx], U-SUB]
2) $\sim F(\forall t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+}))$ Syn $(\forall t)\sim T((A_{ONCE} \& \sim Oat) \& Obt_{+}))$
[1), Ax.8-02, SynSUB]
3) $\sim F(\forall t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+}))$ Syn $\sim(\exists t)\ T(A_{ONCE} \& \sim Oat \& Obt_{+}))$
[2),Q-Exch,SynSUB]

The E-validity of a causal statement "*a* causes *b*" is confirmed or disconfirmed by observations that *a* occurs or not when *b* occurs or not, in the following ways:

First, (i,a) $T(\exists t)((A_{ONCE} \& Oat) \Rightarrow Obt_{+})$ is established if one or more disjuncts in the Boolean expansion of its synonym $(\exists t)T(A_{ONCE} \& Oat \& Obt_{+})$ is true. Its Boolean expansion begins: $(T(A_{ONCE} \& Oat_1 \& Obt_{1+})$ v $T(A_{ONCE} \& Oat_2 \& Obt_{2+})$ v $T(A_{ONCE} \& Oat_3 \& Obt_{++})$ v ....) If at any time all the other necessary conditions exist (i.e., A_{ONCE} is true), and the light-switch is up, and the entry light is on, then one of the disjuncts of this Boolean expansion is true at that time, and thus the disjunctively ∃-quantified expansion, and its synonymous ∃-quantified conditional is true. The falsity or non-truth of other disjuncts does not affect the logical validity of the inference from particular observational truths to the truth of the ∃-quantified conditional.

In the same manner, (ii,a) $T(\exists t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})$ is established if one or more disjuncts in the Boolean expansion of its synonym $(\exists t)T(A_{ONCE} \& Oat \& Obt_{+})$ are found true. $(T(A_{ONCE} \& \sim Oat_1 \& \sim Obt_{1+})$ v $T(A_{ONCE} \& \sim Oat_2 \& \sim Obt_{2+})$ v $T(A_{ONCE} \& \sim Oat_3 \& \sim Obt_{+})$ v ....) is true bcause of the many times when the other necessary conditions existed (A_{ONCE} was true), and the light-switch was **not-up**, and the entry light was **not-on**. Here too the inference from observed truth, to the truth of conditional (ii,a) is a logically valid one.

For these two disjunctively quantified conditional expressions, observations of present or past individual events are sufficient to establish their objective truth.

Establishing the objective non-falsehood of conjunctively quantified conditionals is a different matter. It can be established empirically, that **so far as we know**, the conditionals in (i,b) $\sim F(\forall t)((A_{ONCE} \& Oat) \Rightarrow Obt_{+}))$ and (ii,b) $\sim F(\forall t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})))$ are never false, but since the quantifier (∀t) is intended to cover future times, as well as past or present events which take place unobserved, knowledge that they are **never** false objectively is deliberately beyond our reach. We may consistently *believe* that they are objectively never false if we know of no cases in which they are false. Such a belief may be useful as well as credible. But we can not prove it *true*.

However, (i,b) and (ii,b) are synonymous with statements that can be used to conclusively disprove the causal statements which entail them. By Ax. 8.02, the wff (i,b), $\sim F(\forall t)((A_{ONCE} \& Oat) \Rightarrow Obt_{+})$ is synonymous with $(\forall t)\sim T(A_{ONCE} \& Oat \& \sim Obt_{+})$ and (ii,b), $\sim F(\forall t)((A_{ONCE} \& \sim Oat) \Rightarrow \sim Obt_{+})$

is synonymous with $(\forall t){\sim}T(A_{ONCE}\ \&\ {\sim}Oat\ \&\ Obt_{+})$. Therefore, if at any time, t_j, we find that all other necessary conditions are present (i.e, $T(A_{ONCE})$) and either the cause occurs at t_j without the effect at t_{j+}, or the effect occurs without the cause, this finding will prove that '*a* causes *b*' is *false* by disproving a quantified conditional that it entails. Logically, this can be seen by considering the Boolean expansions of the synonyms.

For example (i,b), ${\sim}F(\forall t)((A_{ONCE}\ \&\ Oat) \Rightarrow Obt_{+})$ Syn $(\forall t){\sim}T(A_{ONCE}\ \&\ {\sim}Oat\ \&\ Obt_{+})$, which is expanded as

$$({\sim}T(A_{ONCE}\ \&\ {\sim}Oat_1\ \&\ Obt_{1+})\ \&\ {\sim}T(A_{ONCE}\ \&\ {\sim}Oat_2\ \&\ Obt_{2+})\ \&\ {\sim}T(A_{ONCE}\ \&\ {\sim}Oat_3\ \&\ Obt_{3+})\ \&\ldots)$$

To find at some time t_j, that $T(A_{ONCE}\ \&\ {\sim}Oat_j\ \&\ Obt_{j+})$, i.e., A_{ONCE} and ${\sim}Oat_j$ and Obt_{j+} are all true, is to disprove a conjunct of this conjunction and thus the whole conjunction. It follows logically that *a* causes *b* is not true, because the synonym of (ii,b), ${\sim}F(\forall t)((A_{ONCE}\ \&{\sim}Oat) \Rightarrow {\sim}Obt_{+})))$, which is entailed by '*a* causes *b*', is not true. Similarly, $(\forall t){\sim}T(A_{ONCE}\ \&\ Oat\ \&\ {\sim}Obt_{+})$, is expanded as

$$({\sim}T(A_{ONCE}\ \&\ Oat_1\ \&\ {\sim}Obt_{1+})\ \&\ {\sim}T(A_{ONCE}\ \&\ Oat_2\ \&\ {\sim}Obt_{2+})\ \&\ {\sim}T(A_{ONCE}\ \&\ Oat_3\ \&\ {\sim}Obt_{3+})\ \&\ldots)$$

and any contradictory, $T(A_{ONCE}\ \&\ Oat_j\ \&\ {\sim}Obt_{j+})$, of a conjunct would make the quantification false, and thus its synonym (ii,a), ${\sim}F(\forall t)((A_{ONCE}\ \&\ Oat) \Rightarrow bt_{+})$ false, and thus "*a* causes *b*" not true.

On the other hand finding instances of '$(A_{ONCE}\ \&{\sim}Oat_i\ \&\ Obt_{i+})$' and '$(A_{ONCE}\ \&\ Oat_i\ \&{\sim}Obt_{i+})$' not-true or false, with no counter instances, adds *confirmation* for (ii,a) and (ii,b), which in turn adds *confirmational* support for '*a* causes *b*'.

Often we discover that a state of affairs we did not know about is a necessary condition and that one of the (dotted) parts of the set-up we had presupposed in holding that "*a* causes *b*" is E-valid. And sometimes we discover a necessary condition which is a disjunction of two possible states of affairs.

For example, when I first enter the entry hall of my house, there is a second switch. When I turned this switch up, a light in the living room went on. Let us call this effect, E_2. When the second switch was not up, the living room light was not on. For quite a while I assumed that turning the second switch up was the single normal cause of E_2. But there is a third switch on the wall in the living room, and I did not know until later that it had to be in the right position for the 2nd switch to cause E_2. It turned out that provided all other necessary conditions exist, that

1) If the 2nd switch is **up** and the 3rd wall switch is **down**, E_2 occurs.
((Sw2**Up** & Sw3**Down**) $\Rightarrow OE_2$)
2) If the 2nd switch is **down** and the 3rd wall switch is **up**, E_2 occurs.
((Sw2**Down** & Sw3**Up**) $\Rightarrow OE_2$)
3) If the 2nd switch is **up** and the 3rd wall switch is **up**, E_2 does not occur.
((Sw2**Up** & Sw3**Up**) $\Rightarrow {\sim}O(E_2)$)
4) If the 2nd switch is **down** and the 3rd wall switch is **down**, E_2 does not occur.
((Sw2**Down** & Sw3**Down**) $\Rightarrow {\sim}O(E_2)$)

Under these conditions, the wiring from switch #2 to the living room light, had to have been more complex than I supposed.

What I discovered was that among the conditions unkown to me (represented by 'A_{ONCE}') was a disjunctive condition NC_5 = ((Sw2**Down**t_i & Sw3**Up**t_i **)** v (Sw2**Up**t_i & Sw3**Down**t_i)) which was a necessary condition for E2. Thus the *actual* full sufficient condition for E_2, SC_2, could be formulated as:

(A_{ONCE2} & NC_5) where the other necessary conditions in SC_2 were similar to those in SC_1 except that it was a for a different light bulb, involved different switches, and had disjunctive necessary condition where I had thought there was a single causal procedure with switches. The logical analysis of this and other related situations can become very complex. But we will not pursue it here. Our point is simply that there can be necessary condtions that are disjunctive as well as conjunctive.

The account of causal statements thus far falls far short of being a complete account. The examples are local and particular, dealing with human/machine interaction. We have not dealt with general causal laws of physics, chemistry, or social sciences. The first examples were particularly simple causal connections in which just one event-type was normally the cause and the last example was the barest beginning of an exploration of cases in which two or more event-types, or sequences of events, might cause the same effect.

Though the examples above are very simple, the interrelationship of the conditional statements (i) and (ii) with causal statements and observational truths as described above is a part of the meaning of causal statements at all levels of generality and complexity.

Critical attacks on various efforts to relate causal statements to conditionals, are filled with arguments which fallaciously rely on M-logic and its presuppositions.

Our theory, expressed with uninterpreted 'if…then's, satisfies the requirements most logicians lay down for any analysis of truth-conditions of "*C* causes *E*" as a contingent statement, though based on a lawlike, expression. Chief among these are,

(i,a) '($\forall$t) If $A_{ONCE}t$ and Ct, then Et+', must not be logically valid or tautologous, but must be an empirical (E-valid) law.
(ii,a) '($\forall$t) If $A_{ONCE}t$ and ~Ct, then ~Et+', must not be logically valid or tautologous, but must be an empirical (E-valid) law.
(($A_{ONCE}t_i$ & Ct_i & ($\forall$t)(If ($A_{ONCE}t$ & Ct) then Et+)) logically entails Et_{i+} (by UI & MP).
(($A_{ONCE}t_j$ & ~Ct_j & ($\forall$t)(If ($A_{ONCE}t$ & ~Ct) then ~Et+)) logically entails ~Et_{j+} (by UI & MP).
But ($A_{ONCE}t_i$ & Ct_i) does not *logically* entail or imply Et_{i+}.
($A_{ONCE}t_j$ & ~Ct_j) does not *logically* entail or imply ~Et_{j+}.
($A_{ONCE}t_i$ & ($\forall$t)(If $A_{ONCE}t$ and Ct, then Et+) does not *logically* imply Et_{i+}.
($A_{ONCE}t_j$ & ($\forall$t)(If $A_{ONCE}t$ and ~Ct, then ~Et+) does not *logically* imply ~Et_{j+}.[19]

One kind of attack on all efforts to explicate "*a* causes *b*" by means of conditionals, is the argument that if it is true that C is the cause of E, then it will follow logically that any other event or state of affairs that is actual, is also the cause of E, according to this analysis. This attack is valid if the conditionals in the definiens are construed as TF-conditionals, but it is not valid if they are C-conditionals.

Actually, if the conditionals (i) and (ii) in our definiens of "*a* causes *b*" are construed as TF-conditionals, every bit of evidence that supports "Ct causes Et+" will support "Xt causes Et" whether X occurs or does not occur in that data. The bits of evidence are of the forms

(1) T($A_{ONCE}t_1$ & Ct_1 & Xt_1 & Et_{1+}) or (2) T($A_{ONCE}t_1$ & Ct_1 & ~Xt_1 & Et_{1+})
or (3) T($A_{ONCE}t_2$ & ~Ct_2 & Xt_2 & ~Et_{2+}) or (4) T($A_{ONCE}t_2$ & ~Ct_2 & ~Xt_2 & ~Et_{2+})

19. Compare to Ernest Sosa, in the Introduction to *Causation and Conditionals*, Edited by Ernest Sosa, Oxford University Press, 1975, pages 1-2.

(1) $T(A_{ONCE}t_1\ \&\ Ct_1\ \&\ Xt_1\ \&\ Et_{1+})$ entails $T(A_{ONCE}t_1\ \&\ Xt_1\ \&\ Et_{1+})$ which implies
(i,a) $\sim F((A_{ONCE}t_1\ \&\ Xt_1) \supset Et_{1+})$ & (i,b) $(\exists t)T((A_{ONCE}t\ \&\ Xt) \supset Et+)$
(2) $T(A_{ONCE}t_1\ \&\ Ct_1\ \&\sim Xt_1\ \&\ Et_{1+})$ entails $T(A_{ONCE}t_1\ \&\sim Xt_1\ \&\ Et_{1+})$ which implies
(i,a) $\sim F((A_{ONCE}t_1\ \&\ Xt_1) \supset Et_{1+})$ & (i,b) $(\exists t)T((A_{ONCE}t\ \&\ Xt) \supset Et+)$
(3) $T(A_{ONCE}t_2\ \&\sim Ct_2\ \&\ Xt_2\ \&\ \sim Et_{2+})$ entails $T(A_{ONCE}t_1\ \&\ Xt_1\ \&\sim Et_{1+})$ which implies
(ii,a) $\sim F((A_{ONCE}t_2\ \&\sim Xt_2) \supset\sim Et_{2+})$ & (ii,b) $(\exists t)T((A_{ONCE}t\ \&\sim Xt) \supset\sim Et+)$
(4) $T(A_{ONCE}t_2\ \&\sim Ct_2\ \&\sim Xt_2\ \&\ \sim Et_{2+})$ entails $T(A_{ONCE}t_1\ \&\sim Xt_1\ \&\sim Et_{1+})$ which implies
(ii,a) $\sim F((A_{ONCE}t_2\ \&\sim Xt_2) \supset\sim Et_{2+})$ & (ii,b) $(\exists t)T((A_{ONCE}t\ \&\sim Xt) \supset\sim Et+)$

Thus, all data which supports the E-validity of "*C* causes *E*" will support the E-validity of "*X* causes *E*", if the conditionals entailed by "*C* causes *E*" are TF-conditionals.

The implications here are A-implications as well as M-implications. This can be seen by reducing '$P \supset Q$' to '$\sim P \vee Q$'. For example in 5), $T(A_{ONCE}t_1\ \&\ Ct_1\ \&\ Xt_1\ \&\ Et_{1+})$ entails $T(Et_{1+})$, and this implies both (i,a) $\sim F(\sim(A_{ONCE}t_1\ \&\ Xt_1) \vee Et_{1+})$ and (i,b) $(\exists t)T(\sim(A_{ONCE}t\ \&\ Xt) \vee Et+)$ by Addition. There is no problem of the validity of this derivation. The problem is in interpreting the results as conditionals. Viewed as "conditionals" these implications are all based on the fallacy that the truth of a conditional follows if its consequent, by itself, is true.

Let us see whether "*X* causes *E*" follows in A-logic from the four kinds of evidence (1), (2), (3), and (4), when the conditionals (i) and (ii) are construed as C-conditionals.

(1) $T(A_{ONCE}t_1\ \&\ Ct_1\&\ Xt_1\ \&\ Et_{1+})$ entails $T((A_{ONCE}t_1\ \&\ Xt_1)\ \&\ Et_{1+})$ which entails $T((A_{ONCE}t_1\ \&\ Xt_1) \Rightarrow Et_{1+})$ by Ax.8-01. From this $\sim F(A_{ONCE}t\ \&\ Xt) \Rightarrow Et+)$ and (i,b) $(\exists t)T((A_{ONCE}t\ \&\ Xt) \Rightarrow Et+)$ follow. But we can not infer X causes E, because that also requires (i) E-valid $((\forall t)((A_{ONCE}t\ \&\ Xt) \Rightarrow Et+)$ and (ii) E-valid $((\forall t)((A_{ONCE}t\ \&\sim Xt) \Rightarrow\sim Et+)$ which entail, by defintion of 'E-valid':

$$\text{(i,a)}\ ((\forall t)\sim F((A_{ONCE}t\&\ Xt) \Rightarrow Et+)\quad \&\quad \text{(i,b)}\ (\exists t)T((A_{ONCE}t\ \&\ Xt) \Rightarrow Et+)$$
$$\&\ \text{(ii,a)}\ ((\forall t)\sim F((A_{ONCE}t\&\sim Xt) \Rightarrow\sim Et+)\quad \&\quad \text{(ii,b)}\ (\exists t)T((A_{ONCE}t\&\sim Xt) \Rightarrow\sim Et+)$$

Only (i,b) and one instantiation of (i,a), namely $\sim F(A_{ONCE}t_1\ \&\ Xt_1) \Rightarrow Et_{1+})$, are derivable from (1).

(2) $T(A_{ONCE}t_1\ \&\ Ct_1\&\sim Xt_1\ \&\ Et_{1+})$ entails $T(A_{ONCE}t_1\ \&\sim Xt_1\ \&\sim\sim Et_{1+})$ which entails $\mathbf{F}((A_{ONCE}t_1\ \&\sim Xt_1) \Rightarrow\sim Et_{1+})$; which implies that (ii,a) $((\forall t)\sim F(A_{ONCE}t\ \&\ Xt) \Rightarrow Et+)$ is false and therefore that "X causes E" is false.

(3) $T(A_{ONCE}t_2\ \&\sim Ct_2\ \&\ Xt_2\ \&\sim Et_{2+})$ entails $T(A_{ONCE}t_1\ \&\ Xt_1\ \&\sim Et_{1+})$ which entails $F((A_{ONCE}t_2\ \&\ Xt_2) \Rightarrow Et_{2+})$, which implies that (i)a $((\forall t)\sim F(A_{ONCE}t\ \&\ Xt) \Rightarrow Et+)$ is false and therefore that "X causes E" is false.

(4) $T(A_{ONCE}t_2\ \&\sim Ct_2\ \&\sim Xt_2\ \&\sim Et_{2+})$ entails $T(A_{ONCE}t_1\ \&\sim Xt_1\ \&\sim Et_{1+})$ which entails $\sim F((A_{ONCE}t_2\ \&\sim Xt_2) \Rightarrow\sim Et_{2+})$ and (ii,b) $(\exists t)T((A_{ONCE}t\ \&\sim Xt) \Rightarrow\sim Et+)$ but we can not infer that X causes E, because that requires the presence of evidence like (1), and the absence of evidence like (2) and (3) which prove that X does not cause E in the supposed circumstances.

Thus the data required to support the empirical validity (truth) of "C causes E" do not support the empirical validity of "X causes E" if conditionals (i) and (ii) are C-conditionals.

Other arguments against our analysis of "*a* causes *b*" succeed by appealing to other presuppositions from M-logic which do not hold in A-logic. For example, Davidson argues that causes can not be represented by sentences, and causal statements can not be translated into a compound sentence with a non-truth-functional connective between such sentences. From a purportedly non-truth-functional translation of the true statement, "The short circuit caused the fire", he derives the absurd statement "The fact that there was a short circuit *caused it to be the case that* Nero fiddled," by substituting terms of tautologous or factually true TF-biconditionals.[20] But such substitutions do not yield *de re* inferences which are valid in A-logic.[21] Again, Jaegwon Kim argues that there are difficulties in a proposal to salvage Mackie's account of causation, because "sentences do not generally have unique disjunctive normal forms."[22] This is true of normal forms in M-logic but is not true of the "basic normal forms" derived by logically synonymy in A-logic.

There is much more to be said. But this section provides preliminary grounds for holding that A-logic with its C-conditionals can handle aspects of causal statements that M-logic can not handle.

9.35 Frequencies and Probability

In the logic of probability A-logic succeeds in one area where M-logic fails. It provides a coherent probability logic in which conditional probabilities are probabilities of C-conditionals whereas, as all agree, conditional probabilities —the probability that Q is true if P is true —cannot be the probabilities of TF-conditionals.

The mathematical concept of a frequency is basic to, but not the same as, the concept of the concept of a probability. However, the logic of mathematical frequencies is basic to any logic of probability. To sidestep philosophical controversies over the concept of probability, we will focus on the A-logic of relative frequencies.[23] By '<1> is a relative frequency' we shall mean "<1> is a rational number, m/n, which is the ratio between the frequency (number of members) in a class which has n members and the frequency of a sub-class of that class with m members". Where 'f(P)' means "the frequency (number of instances) in which P<1> is true" and R<1> is '<1> is a member of the reference class", we call these relative frequencies "1st-level relative frequencies" (abbr. 'rf_1') as in,

$$rf_1(P) = \frac{f(P)}{(f\,(P) + f\,(\sim P))} = \frac{f(P)}{f(R)}$$

There are also "2nd-level relative frequencies" (abbreviated by 'rf_2'), which are ratios of a 1st-level frequency over another 1st-level frequency which is equal to or greater than it, as we shall explain.

Standard Probability Theory is a theory about truth-claims, and that it how we treat it here.[24] In standard probability theory with M-logic, propositional variables stand for expressions purporting to be true and every proposition is either true or false in fact. To make explicit the fact that probabilities have to do with truth-claims, we let 'f' stand for 'the number of cases that...' and 'TP' for 'P is true' or 'it is true that P' so that 'f(P) = x ' becomes 'fT(P) = x' for "the frequency that P is true = x", while '$rf_1T(P)$' is the "the frequency that it is true that P, over the frequency (number of members) of the

20. Donald Davidson, "Causal Relations", *Journal of Philosophy*, 64 (1967), p. 694

21. See Section 10. 1223, p. 566

22. Jaegwon Kim, "Causes and Events: Mackie on Causation" in *Causation and Conditionals*, Edited by Ernest Sosa, Oxford University Press, 1975, page 55.

23. The mathematical concept of a frequency and its logic, should not be confused with the controversial "frequency theory of probability" which proposes to equate probability with frequencies in empirical fact.

24. This does not preclude the possibility of a coherent theory of probability about value-claims—e.g., the claim that probably if Pa were true, then the results (Pa & R & S) would be *good*. There is no inconsistency in talking of the probability that some state of affairs would be morally right, or aesthetically good even when the probability of its being true is clearly zero. But the standard probability theory we have been discussing is implicitly a sub-section of truth-logic; it is probability theory in the context of a truth-search, focused on probabilities of truth or falsehood. We defer extending probability theory beyond this to another time.

reference class:

$$rf_1T(P) = \frac{fT(P)}{(fT(P) + fT(\sim P))} = \frac{fT(P)}{fT(R)}$$

Putting 'F(P)' for 'T(~ P)' the lst-level relative frequency that a predicate ~ P is true in a field of reference is the ratio of the number of cases (frequency) ~ P is true, over the number of cases P is either true or false in that field. This agrees with probability theory using M-logic where every atomic proposition is either true or false, so

$$(fT(P)+fF(P)) = f(TPv\ FP) = fT(R) \text{ and } rf_1T(R) = \frac{fT(R)}{fT(R)} = 1.00.$$

The concept of a relative frequency can apply to sub-classes of a finite class of actual entities which is used as a sample for statistical projections. The relative frequencies or proportions found in a sample can be projected (properly or improperly) onto other similar groups of indefinite size, as in actuarial figures of the insurance industry. They can also be projected to sub-classes of infinite size in an infinite class; e.g., the frequency within the infinite class of all positive integers, of integers with either 2 or 5 as a prime factor, is .60 or 6/10.

Philosophical theories of probability are constructed to explicate common sense and/or scientific probability judgments, such as "It will probably not rain tomorrow", "The chances of death in a heart transplant operation is 5%", "The odds are 3 to 2 that the Yankees will beat the Braves" tomorrow. Theories differ as to whether "It is probable that…" should be interpreted as (a) a projection of frequencies which occur in fact, (b) frequencies determined *a priori* by enumeration of possible outcomes, (c) subjective guesses with respect to chances or odds of various outcomes, etc. For present purposes, we need not get involved in these issues, or with problems of how to make proper projections from samples. If there is a logic of probability statements, it consists in a set of principles which determines how one probability statement can follow logically from one or more other probability statements; this must be in accordance with the logic of relative frequencies.

The 1st-level relative frequency of a C-conditional predicate is the frequency that [P<1> => Q<1>] is true relative to the Reference Class. Our thesis is that a conditional probability can be defined as the '2nd-level relative frequency' of the truth of a C-conditional: the number of cases in which the the 1st level frequency that [P<1> => Q<1>] is true, over the total cases in which it is either true or false, which is the same as the lst-level frequency that the antecedent P<1> is true. The thesis depends on Axioms 8-01 and 8-02 in Analytic Truth logic.

9.351 The Logic of Mathematical Frequencies and Probabilities

Despite different interpretations of 'probable' there is widespread acceptance of the mathematical axioms for standard probability theory. The following set of axioms is a "classical" one which presupposes M-logic.[25] We will propose a system which is complete with respect to it.

Ax.1 $Pr(A) \geq 0$
Ax.2 $Pr(\sim A) = (1 - Pr(A))$
Ax.3 $Pr(AvB) = Pr(A) + Pr(B) - Pr(A\&B)$
Ax.4 If $\vdash (A)$, then $Pr(A) = 1$

25. This axiomatization is in accord with the "classical" axiomatization of probability theory presented in A. N. Kolmogorov's, *Foundations of the Theory of Probability*, New York,1950.

Ax.5 If $\vdash (A \equiv B)$, then $Pr(A) = Pr(B)$

Ax.6 If $Pr(A) \neq 0$, then $Pr(B|A) = \frac{Pr(A \& B)}{Pr(A)}$

To avoid philosophical issues concerning probability, we replace ‘Pr (P)’ for ‘the probability of P’ by ‘rf (P)’ for ‘the relative frequency of P’ and based on Chapter 5, we replace ‘$\vdash$ P’ by ‘TAUT(P)’. The resulting system will be called RF. Thus we restate the axioms as,

RF-Ax.1 $rf(A) \geq 0$	A mathematical Axiom
RF-Ax.2 $rf({\sim}A) = (1 - rf(A))$	A mathematical Axiom
RF-Ax.3 $rf(AvB) = rf(A) + rf(B) - rf(A\&B)$	Axiom using connectives of M-logic
RF-Ax.4 If A is TAUT, then $rf(A) = 1$	Axiom to use M-logic Taut-Theorems
RF-Ax.5 If $TAUT(A \equiv B)$, then $rf(A) = rf(B)$	Axiom to use M-logic Taut-Theorems
RF-Ax.6 If $rf(A) \neq 0$, then $rf(B|A) = \frac{rf(A \& B)}{rf(A)}$	Conditional Probability Axiom

In addition, three inference rules are used in deriving theorems of this system from these axioms:

RK-1. Substitutivity of Logical Equivalents. If (i) A is a theorem of Probability Logic and (ii) P is any wff of M-logic which occurs in A, and (iii) Q is truth-functionally equivalent to P and (iv) A′ is the result of substituting Q at some occurrence of P in A, then A′ is also a theorem.

RK-2. U-SUB$_{\mathbf{Prob.}}$ If A is an axiom or theorem of Probability Logic which contains a sentential variable P, and Q is any wff of M-logic, then the result of substituting Q at all occurrences of P in A is also a theorem.

RK-3. SUB=s. (Substitutability of arithmetic equals). If A = B is a theorem and C = D is a theorem, and A’ is the result of replacing an occurrence of C in A by D, then (A′ = B) is a theorem.

All theorems derivable from these axioms are either statements of a mathematical relation (‘=’, or ‘≤’, or ‘>’) between two terms describing relative frequencies, or conditional statements about such statements. RFAx.1 is a categorical mathematical statement with no logical terms thrown in; it asserts that all relative frequencies are equal to or greater than zero. RFAx.2 says the negation of a wff A has a relative frequency of 1 minus the relative frequency of A. Thus any sub- class of the reference class can be expressed with any desired degree of accuracy by numbers that are not less than 0 or more than 1. RFAx.3 is a categorical equality-statement schema for determining the relative frequency of a disjunction of any two statements connected by ‘v’, provided we have the relative frequencies of the two propositions separately and of their conjunction by ‘&’. RFAx.4 and RFAx.5 are conditional theorems. They say that *if* an expression is a logical tautology (i.e., a theorem of M-logic), *then* certain equalities of relative frequencies will hold of it. RFAx.6 defines “the relative frequency of A, given B” in terms of the mathematical ratio of two 1st-level relative frequencies, *provided* the relative frequency of the antecedent is greater than 0.

RFAx.6 is of special logical interest because it does not define conditional probabiity as the probability of a conditional. Since M-logic deals only with conjunction, disjunction, negation (defining a truth-functional “conditional” only in those terms), the relationship to the language of M-logic breaks down at this point. We shall see why in the section 9.353.

The changes above in the standard axioms are only in terminology. They do not, so far, affect either the logical conditions which are tied to relative frequencies, or the mathematical equations derivable from the axioms. They are made to help clarify the problem and its solution.

9.352 The "General Problem of Conditional Probability"

In standard probability theory the expression 'Pr(B|A)' in K5 is read as " the probability of B, given A", and 'Pr(B|A)' defined as 'Pr(A&B)/PrA' and is called "the conditional probability of B, given A". The question arises, why not say this is the probability of a conditional; i.e., the probability that B is true *if* A is true, or, the probability that [*if* A *then* B] is true relative to the times it is either true or false? Since the logic of probabilities deals with the probabilities of propositions being true, this seems a natural and reasonable suggestion which should be pursued.

As we shall see in the next section, if the "if…then" is a TF-conditional, then Pr(A ⊃ B) diverges sharply from Pr(B|A). This is the special problem of conditional probability for M-logic.

But there is a more general problem of conditional probability: the question of whether ***any*** conditional could be formulated so that conditional probability is the probability of that conditional. I.e., is there any system of logic with a conditional, '➔', such that the expression '(B|A)' in axiom RFAx.6, can be replaced by '(A ➔ B)'? This is a problem of the nature of conditionals and the logic of conditionals and how they operate in probability theory. We will show how the (second- level) relative frequency of an inferential C-conditional, [A ⇒ B], works out exactly as the mathematical concept of conditional probability requires, and show why arguments against this do not hold.

To make the general problem of conditional probability clearer, we will use a 10 X 10 arrangement of squares to illustrate the logic of relative frequencies in a specific, actual case. The model in TABLE 9-3 represents a set of 100 tiles, like the squares in a Scrabble game, such that each tile either does or does not have one or more of the following letters printed on it: 'p', 'q', 'r', 's', 'x', 'y'. That a tile (or square) has a 'p' on it, is a positive property of that tile. The entire set of tiles is the Reference Class. Obviously, there are many distinct sub-classes and sub-sub-classes of the Reference Class.

Each tile or square is a distinct individual. If we wished to, we could give them names and print the names on their backs: 't_1', 't_2', …, 't_{98}', 't_{99}', 't_{100}'. But we are not interested in information about particular individuals here, so we disregard who they are, shuffle them about and look only at what letters they have on their frontside. Let 'P<1>' stand for "<1>has a 'p' on it", 'Q<1>' stand for the predicate, "<1> has 'q' on it", and so on. Since there are 100 tiles, the individual tiles can be arranged in trillions (actually 99!) of different kinds of 10X10 arrangements, just two of which are pictured as Arrangements #1 and #2. Or they could all be picked up and jumbled into a bag or pot, like the tiles in a scrabble game. The manner of arrangement does not affect in any way the frequencies or relative frequence we are investigating, but arrangement #1 helps to make clearer the meaning of statements about relative frequencies.

The important thing to note is that there are two kinds of relative frequencies that are relevant to whether [Q|P] or [If P<1> then Q<1>] is true these statements. To get the first kind we simply count the number of tiles of which the predicate is true, then divide by 100 (the number of members in the Reference Class). These "1st-level" relative frequencies represent proportions of the Reference class. For example, there are 40 squares with 'p' in them, thus rf_1 (P) = $40/100^{ths}$ or $4/10^{ths}$ or $2/5^{ths}$ = .40 of the reference class. All 23 statements are determined this way.

The last 10 rf_2-statements are statements of the second kind—assertions of conditional probability or conditional frequency. They stand for ratios between two 1st-level frequencies. They are second-level frequencies, with the 1st-level relative frequency of the antecedent (not the Reference class itself) as the denominator. We will show that a C-conditional's 2nd-level relative frequency is the frequency that it is true relative to the times is it either true or false. In a finite model, we can get the value of this frequency directly simply by counting the number of tiles of which some predicate (e.g.,'Q<1>') is true, then counting, <u>among those tiles</u>, the number of tiles or entities of which some other predicate, (e.g., 'S<1>')

is true and presenting the ratio of the latter number over the former number. For example, the number of tiles with ‘q’ on them is 12. The number of <u>those</u> tiles with ‘s’ on them, (i.e., the number of which ‘(Q<1> & S<1>)’ is true) is 2. Thus the relative frequency that ‘S<1>’ is true, <u>given that</u> ‘Q<1>’ is true, is 2/12 = .1/6 = .1667. This is established without reference to the Reference Class or its 100 members. This ratio does not represent a proportion of the Reference class; rather it represents a proportion of the subclass which is determined by the antecedent. However, if we wish to view it as a function of 1st-level frequencies, we can view it as a 2nd-level ratio of ratios:

$$\frac{\frac{f\,T(Qa_i \,\&\, Sa_i)}{fT(R)}}{\frac{f\,T(Qa_i)}{fT(R)}} = \frac{\frac{2}{100}}{\frac{12}{100}} = \frac{1}{6} = \frac{16666..}{100000} = .167$$

In some cases, *second-level* frequencies can be expressed as rational fractions with 100 as the denominator. Forexample, $rf_2(Q|P) = 30/100$. But the occurrence of ‘100’ in such cases does not represent the number of members of the Reference Class as was the case in our first 23 equations; and 30/100 does not represent a proportion (3/10ths) of the Reference class. Rather it stands for 12/40ths or 3/10ths of the subclass of 40 individuals of which the predicate ‘P<1>’ is true. The fact that relative frequencies are all expressible as decimal numbers to a base of some power of ten, hides the fundamental difference between ratios that represent **lst-level frequencies** relative to the reference class, and conditional probabilities i.e., **2nd-level frequencies** which are ratios of one of these 1st-level frequencies to another one.

If the Reference Class were to consist of five individuals $\{a_1, a_2, a_3, a_4, a_5,\}$ and P<1> were true in three instances, and of these (P ⇒ Q) were true in two instances, then the *1st-level* frequency of T(P ⇒ Q) would be equal to the 1st-level frequency of T(P & Q) which is .40 or 2/5:

$$\mathbf{rf_1}T(P \,\&\, Q) = \frac{2}{5} = \mathbf{rf_1}T(P \Rightarrow Q)$$

$(Pa_1 \,\&\, Qa_1) \,\&\, (Pa_2 \,\&\, Qa_2) \,\&\, (Pa_3 \,\&\, Qa_3) \,\&\, (Pa_4 \,\&\, Qa_4) \,\&\, (Pa_5 \,\&\, Qa_5)$ $\quad \frac{fT(P\&\,Q)}{fT(R)} = \frac{2}{5} = \frac{fT(P \Rightarrow Q)}{fT(R)}$

T T T   F 0 T   T F F   F 0 F   T T T

To get the *2nd-level* frequency of cases in which (P ⇒ Q) is true we divide the 1st-level relative frequency of T(P ⇒ Q) by the sum of the 1st-level frequencies in which it is either true or false. The latter is the same as the number of cases in which the antecedent is true. In the example of five cases, the second-level frequency is $\mathbf{rf_2}$ T(P ⇒ Q) = .667 or 2/3.

$(Pa_1 \Rightarrow Qa_1) \,\&\, (Pa_2 \Rightarrow Qa_2) \,\&\, (Pa_3 \Rightarrow Qa_3) \,\&\, (Pa_4 \Rightarrow Qa_4) \,\&\, (Pa_5 \Rightarrow Qa_5)$

T T T   F 0 T   **T F F**   F 0 F   **T T T**

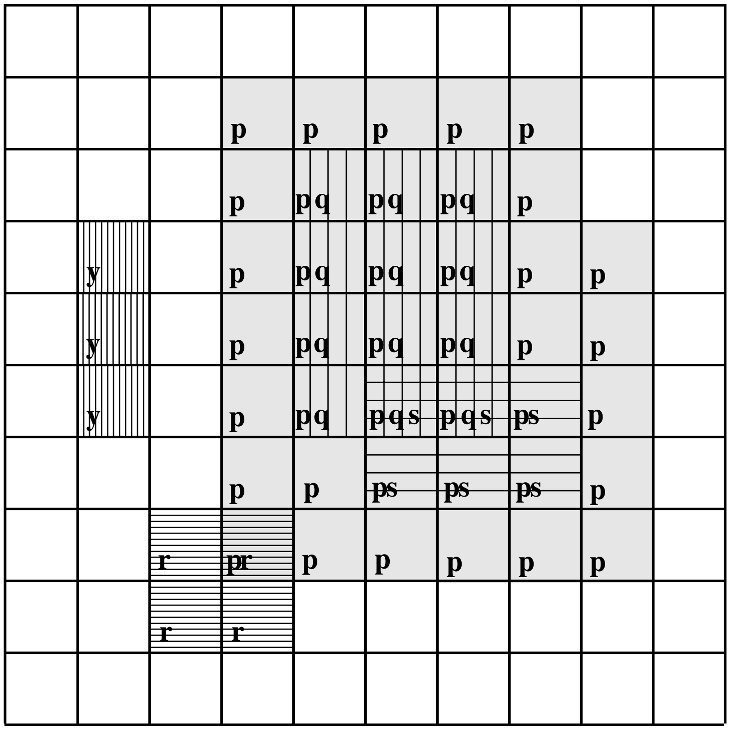
$$\frac{\mathbf{rf_1}\,T(P \Rightarrow Q)}{\mathbf{rf_1}T(P \Rightarrow Q) + \mathbf{rf_1}\,F(P \Rightarrow Q)} = \frac{2}{2+1} = \frac{2}{3} = \frac{\mathbf{rf_1}\,T(P \Rightarrow Q)}{\mathbf{rf_1}\,\mathbf{T}(P)}$$

Thus a 2nd-level frequency is the ratio of the lst-level relative frequency that a C-conditional is true in the reference class over the lst-level relative frequency that its antecedent is true in the reference class. In the example above, the 1st-level relative frequency that (P ⇒ Q) is true is 2/5. This is the numerator and the 1st-level frequency that P is true. The ratio 3/5 is the denominator and it represents the lst-level relative frequency in which P is

$$\mathbf{rf_2}T(P \Rightarrow Q) = \frac{\mathbf{rf_1}T(P \Rightarrow Q)}{\mathbf{rf_1}T(P)} = \frac{\frac{fT(P \Rightarrow Q)}{fT(R)}}{\frac{fT(P)}{fT(R)}} = \frac{\frac{2}{5}}{\frac{3}{5}} = \frac{2}{3} = .667$$

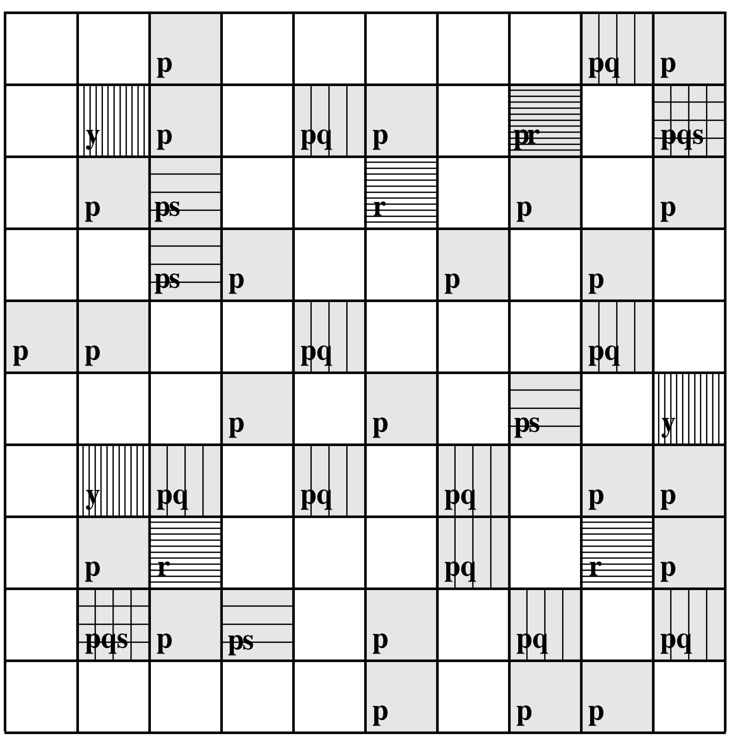
TABLE 9-3

Arrangement #1



The positioning in Arrangement #1 is not relevant to these frequencies. The relative frequencies are the same in Arrangements #1 and #2, or if all tiles are jumbled in a bag.

Arrangement #2



1st-Level Relative Frequencies
In a Reference Class of 100

rf_1 (P) = 40/100 = .40
rf_1(Q) = 12/100 = .12
rf_1 (R) = 4/100 = .04
rf_1 (S) = 6/100 = .06
rf_1 (∼P) = (100–40)/100 = .60
rf_1 (∼Q) = (100–12)/100 = .88
rf_1 (∼R) = (100–4)/100 = .96
rf_1 (∼S) = (100– 6)/100 = .94
rf_1 (P&Q) = 12/100 = .12
rf_1 (P&∼Q) = 28/100 = .28
rf_1 (P&S) = 6/100 = .06
rf_1 (P&∼S) = 34/100 = .34
rf_1 (P&R) = 1/100 = .01
rf_1 (∼P&R) = 3/100 = .03
rf_1 (Q&S) = 2/100 = .02
rf_1 (Q&∼S) = 10 /100 = .10
rf_1 (∼Q&∼S) = 84/100 = .84
rf_1 (Q&R) = 0/100 = .00
rf_1 (Q&∼R) = 12/100 = .12
rf_1 ∼(P&∼Q) = (100–28)/100 = .72
rf_1 ∼(Q&∼R) = (100–12)/100 = .88
rf_1 ∼(P&∼R) = (100–39)/100 = .61
rf_1 ∼(∼P&R) = (100–03)/100 = .97
rf_1 ∼(Q&∼S) = (100–10)/100 = .90
rf_1 ∼(∼Q&∼S) = (100–84)/100 = .16

2nd-Level Conditional Probabilities:

$rf_2(Q|P) = \frac{rf_1\ (P\&Q)}{rf_1\ (P)} = \frac{.12}{.40} = .30$

$rf_2(P|Q) = \frac{rf_1\ (Q\&P)}{rf_1\ (Q)} = \frac{.12}{.12} = 1.00$

$rf_2(\sim Q|P) = \frac{rf_1\ (P\&\sim Q)}{rf_1\ (P)} = \frac{.28}{.40} = .70$

$rf_2(R|\sim P) = \frac{rf_1\ (\sim P\&R)}{rf_1\ (\sim P)} = \frac{.03}{.60} = .05$

$rf_2(R|P) = \frac{rf_1\ (P\&R)}{rf_1\ (P)} = \frac{.01}{.40} = .025$

$rf_2(\sim R|Q) = \frac{rf_1(Q\&\sim R)}{rf_1\ (Q)} = \frac{.12}{.12} = 1.00$

$rf_2(\ S|Q) = \frac{rf_1\ (Q\&S)}{rf_1\ (Q)} = \frac{.02}{.12} = .166$

$rf_2(S|Q) = \frac{rf_1\ (Q\&S)}{rf_1\ (Q)} = \frac{.02}{.12} = .166$

$rf_2(S|R) = \frac{rf_1\ (R\&S)}{rf_1\ (R)} = \frac{.00}{.01} = .00$

$rf_2(Q|(P\&S))$
$= \frac{rf_1(P\&S\&Q)}{rf_1(P\&S)} = \frac{.02}{.06} = .333$

$rf_2((S\&Q)|P)$
$= \frac{rf_1(P\&S\&Q)}{rf_1(P)} = \frac{.02}{.40} = .05$

true, as well as the sum of the lst-level relative frequencies that (P ⇒ Q) is true and that (P ⇒ Q) is false.

Next, let us analyze the mathematical processes of adding and subtracting, as they occur in the logic of probabilities. We first look at how **1st-level frequencies** are added to or subtracted from one another in accordance with the first two axioms of probability theory, to get the frequencies for a negation, conjunction or disjunction of the predicates involved.

The relative frequency of a <u>negated</u> predicate or statement refers to the number of individuals of which the predicate '~P' is true, relative to the whole Reference Class. This is expresssed in the theorem of probability logic which says "$rf_1(\sim P) = 1 - rf_1(P)$"; i.e., to get $rf_1(\sim P)$, one subtracts the relative frequency of cases in which P is true, from 1, which stands for the ratio of the number of members in the reference class to itself. $\frac{f(\sim P)}{f(R)} = \frac{f(R)}{f(R)} - \frac{f(P)}{f(R)}$.

In TABLE 9-3, since $rf_1(P) = 40/100$, this means that $rf_1(\sim P) = (100/100 - 40/100) = 60/100$, i.e., $(1 - .40) = .60$. Thus, in the actual example of TABLE 9-3, $rf_1(\sim P)$ denotes the ratio between the 60 distinct tiles or squares which do not having a 'p' in them, to the whose set of 100.

The 1st-level relative frequency of a *disjunctive* predicate or statement, has to do with the number of times at least one of the disjuncts applies truly to different tiles in the reference class, relative to the number of members in the Reference Class. Since both predicates might be true of the same squares, we must subtract the cases, if any, in which (Q<1> & S<1>) is true of a given square. This is the meaning of Axiom RFAx.3, $[rf_1(QvS) = (rf_1(Q) + rf_1(S) - rf_1(Q \& S)]$; you count the number of individuals of which Q<1> is true, add the number of which S<1> is true, and subtract the number of which both Q<1> and S<1> are true to get the number of cases in which either Q<1> or S<1> (or both) are true. In TABLE 9-3, this gives us $rf_1(Q \text{ v } S) = 16/100 = .16$, since $rf_1(Q) = 12/100$, $rf_1(S) = 6/100$ and $rf_1(Q \& S) = 2/100$ and $12 + 6 - 2 = 16$.

The general formula for the 1st-level relative frequency of a *conjunctive* predicate or statement is gotten by the purely mathematical procedure of changing '+' into '–' and moving a number across the equality sign in RFAx.3; i.e., $[rf_1(Q\&S) = (rf_1(Q) + rf_1(S) - rf_1(Q \text{ v } S)]$. Thus if you already know the 1st-level relative frequency of (Q v S), as well as $rf_1(Q)$ and $rf_1(S)$, then you just add $rf_1(S)$ to $rf_1(Q)$ and subtract $rf_1(QvS)$ to get $rf_1(Q\&S)$. But in TABLE 9-3, the simplest way to determine the 1st-level frequency of a conjunctive expression is to count the tiles of which every conjunct in the conjunction is true and divide the result by the number of individuals in the field of reference. The 1st-level relative frequency of a *conjunctive* predicate or statement is the number of times all its conjuncts are true of the same individual divided by the number of individuals in the Reference Class. Thus the relative frequency of (P & S) is 6/100, since there are just six individual tiles among the 100 squares which have both a 'p' and an 's' in them. This method does not involve adding or subtracting other frequencies.

Now let us see if these methods of adding and substracting 1st-level frequencies can be successfully applied to the **2nd-level frequencies** representing conditional probabilities. It quickly becomes apparent that these processes of addition and subtraction do not apply in the same way.

If $rf_1(\sim P)$ is a 1st-level relative frequency greater than 0 it represents a distinct set of individuals in the reference class. In TABLE 9-3 $rf_1(\sim P) = .60$, $rf_1(\sim(Q\&\sim S)) = .90$, and $rf_1(\sim(Q\&S)) = .98$. These stand respectively for distinct sets of 60, 90 and 98 individual tiles in our model. As relative frequencies they stand for proportions of the whole field, 3/5, 9/10 and 98/100 and these will be their proportion in any field, larger or smaller, onto which they are projected. But what proportion of the Reference class is represented by '$\sim(S|Q)$'? It does not stand for any. We have seen what $rf_2(S|Q) = .167$ (or 2/12) signifies; the number of members of the reference class that are both S and Q (i.e.,2) divided by the number which are Q (12). If we try to apply the formula which works on negations of 1st-level frequencies, we get a result that does not represent a sub-class of the reference class, or any proportion of the members of the Reference class:

$$rf_2{\sim}(S|Q) = (1 - rf_2(S|Q)) = (1\text{-}2/12) = (1 - .1667) = .8333.$$

Neither $rf_2(S|Q) = .1667$ nor $({\sim}(S|Q)) = .8333$ represent sets of 16.17 or 83.33 tiles in our model, nor do they represent proportionate parts of the Reference class of 100 nor proportions of any references class, larger or small, on which we might want to project these ratios. What they represent are proportions of the sub-class of R of which Q<1> is true, in which S<1> or ~S<1> is true.

Clearly one thing that is needed is a distinction between talk about 1st-level relative frequencies which represent proportions of the Reference class, and 2nd-level relative frequencies which represent proportions of various sub-classes of the Reference class.

Substituting the expression (Q|P) for a propositional variable in a theorem of Probability Logic, is equally problematic if the variable occurs in a conjunction or disjunction. The expression '$rf_2(Q|P)$' stands for a number, not a sentence or predicate. Axiom RFAx.3 makes sense when we substitute any wff of M-logic for the sentential or predicate variables. But if the expression '(Q, given P)' replaces a sentence letters in a 1st-level relative frequency the result is baffling. What do '$rf_1(rf_2(Q|P) \vee R)$' or '$rf_1(rf_2(\text{If P then Q}) \vee R)$' mean? And what could "the 1st-level relative frequency of both (Q,given P) and R" mean? In our 100 member model, Q stands for the sub-class of 12 squares which have a 'q' in them; P stands for the sub-class of 40 squares which have a 'p' in them, and (P&Q) stands for the sub-class of 12 squares that have both 'p' and 'q' in them, etc. But $rf_2(Q|P)$ stands for the rational number, $rf_1(P\&Q)/rf_1(P)$, i.e., $12/40 = 30/100$. It does not stand for any sub-class of the Reference class. $Rf_1((Q/P) \& R)$ does not stand for a sub-class of squares that has 'q|p' and 'r' in them. They do not stand for any sub-class of R in the model. It stands only for a relation between two sub-classes.

If the ambiguity in the prefix 'rf ' (or 'Prob') with respect to lst- or 2nd-level frequencies, is not recognized, false equations can be derived from mathematically correct theorems. Such a derivation has been used to discredit the possibility of defining conditional probability in terms of the probability of a conditional. A detailed account of this derivation will be provided in Section 9.354 which has a notation capable of preserving the difference between the two meanings.

Using a clear distinction between 1st-level and 2nd-level relative frequencies, and with stricter rules of substitution and analytic truth-logic for C-conditionals, the difficulties we have just seen can all be resolved. But before we explain this, we will show why the TF-conditional cannot be used to express conditional probabilities.

9.353 Why the Probability of the TF-conditional is Not Conditional Probability

Why can't "the frequency of (S, given Q)" be interpreted as "the frequency of (If Q then S)", using '$(Q \supset S)$' for 'If Q then S' as is done in M-logic? Why isn't $rf_2(S|Q)$ the same as $rf_2(Q \supset S)$?

To find the answer we look at the set of tiles in TABLE 9-3. There are 12 tiles of which 'Q<1>' is true, and 6 tiles of which 'S<1>' is true. We look at the number of tiles to which both 'Q<1>' and 'S<1>' truthfully apply. There are just two. The answer to the question 'What is $rf_2(S|Q)$?' obviously is:

$$rf_2(S\ |Q_i) = \frac{2/100}{12/100} = \frac{2}{12} = \frac{1}{6} = .16666. \quad \text{or} \quad rf_2\ (S|Q) = \frac{rf_1\ (Q\&S)}{rf_1\ (Q)} = \frac{2/100}{12/100} = \frac{2}{12} = \frac{1}{6} = .1666$$

This is what we get if we use the definition of conditional probability in RFAx.6.

What is the probability of a truth-functional conditional? Since 'If Q then S' means '$(Q \supset S)$' and this is SYN to '$({\sim}Q \vee S)$', the 1st-level relative frequency of the TF-conditional, $rf_1(Q \supset S) = rf_1({\sim}Q \vee S)$, which is **.90** rather than .1667. This follows from Axiom RFAx.3 and the synonymy and logical equivalence of $(Q \supset S)$ and $(\sim Q \vee S)$ in accordance with M-logic:

$$\mathbf{rf_1(Q \supset S) = rf_1(\sim Q v S)} = (rf_1(\sim Q) + rf_1(S) - rf_1(\sim Q\&S)) = rf_1(\sim QvS) = rf_1 \sim(Q\&\sim S)$$

$$\frac{88}{100} + \frac{6}{100} - \frac{4}{100} = \frac{90}{100} = .90$$

In contrast, as we have seen, $rf_2(S|Q) = .1667$. So the conditional probability $rf_2(Q|P)$ can not be the same as the probability of the TF-conditional, $rf_1T(P \supset Q)$. (And the idea of a 2nd-level relative frequency, $rf_2(Q \supset S)$ as $\frac{rf_1(Q \supset S)}{rf_1(Q)} = \frac{.90}{.12} = 7.50$, is unacceptable; it would violate Axiom K1.)

TABLE 9-4

Relative Frequencies of TF-conditionals (P ⊃ Q)
and Conditional Probabilities (Q/P),
and The Discrepancies (in bold) Between Them,
as Functions of rf(P) and rf(P & ~Q)

rf(P)	rf(p&~q): .00	.20	.40	.50	.60	.80	1.00
1.00	⊃ 1.00 ➔1.00 .00	⊃ .80 ➔.80 .00	⊃ .60 ➔.60 .00	⊃ .50 ➔.50 .00	⊃ .40 ➔.40 .00	⊃ .20 ➔.20 .00	⊃ .00 ➔ .00 .00
0.8	⊃1.00 ➔1.00 .00	⊃ .80 ➔.75 .05	⊃ .60 ➔.50 .10	⊃ .50 ➔.375 .125	⊃ .40 ➔.25 .15	⊃ .20 ➔.00 .20	no cases (Inc)
0.6	⊃1.00 ➔1.00 .00	⊃ .80 ➔.667 .133	⊃.60 ➔.333 .267	⊃.50 ➔.167 .333	⊃.40 ➔.00 .40	no cases (Inc)	no cases (Inc)
0.5	⊃1.00 ➔1.00 .00	⊃.80 ➔.60 .20	⊃.60 ➔.20 .40	⊃.50 ➔ .00 .50	no cases (Inc)	no cases (Inc)	no cases (Inc)
0.4	⊃1.00 ➔1.00 .00	⊃.80 ➔.5 .30	⊃.60 ➔.00 .60	no cases (Inc)	no cases (Inc)	no cases (Inc)	no cases (Inc)
0.2	⊃1.00 ➔1.00 .00	⊃.80 ➔.00 .80	no cases (Inc)	no cases (Inc)	no cases (Inc)	no cases (Inc)	no cases (Inc)
.00	⊃ 1.00 ➔ Un- defined	no cases (Inc)	no cases (Inc)	no cases (Inc)	no cases (Inc)	no cases (Inc)	no cases

The discrepancies between conditional probability and the probability of the truth-functional conditional are shown in each box.

The upper figures in each box, preceded by '⊃', are

$rf_1(P \supset Q) = rf_1 \sim(P\&\sim Q)$
$= rf_1 (1 - rf_1(P\&\sim Q))$.

The middle figures in each box is the conditional probability, or relative frequency of $fr_2(Q|P)$, indicated by the arrow, '➔',

$= \frac{rf_1(P\&Q)}{rf_1(P)} = \frac{rf_1(P) - rf_1(P\&\sim Q))}{rf_1(P)}$.

The third figure in each box, is the discrepancy between the first and the second figures. This discrepancy is

$(rf_1(P \supset Q) - rf_2(Q|P))$, i.e.,
$= (1 - rf_1(P\&\sim Q)) - \frac{(rf_1(P)}{rf_1(P)} - \frac{rf_1(P\&\sim Q))}{rf_1(P)}$

$rf_2(Q|P) = rf_1(P \supset Q)$ iff $rf_1(P) > 0$ and
1) $rf_1(P\&\sim Q) = 0$
or 2) $rf_1(\sim P) = rf_1(P \supset Q)$
or 3) $rf_1(P) = rf_1(P\&\sim Q)$

Suppose the letter, 'z', is not on any tile. Then the relative frequency assigned to "It is true that if Z<1>, then P<1>" with a TF-conditional is 1.0, due to the falsity of the antecedent. Since there are no 'z's on any tile, $(\sim Z)$ is true of every tile, hence $rf(\sim Z) = 1.0$.

$rf_1(\sim Z\&Q) = rf_1(Q) = .12$, so by RFAx.3, $rf_1(Z \supset Q) = rf_1(\sim Z v Q) = (rf_1(\sim Z) + rf_1(Q) - rf_1(\sim Z\&Q))$
$= (\ \ 1.00 + .12 - .12\ \ ) = 1.00$

In contrast, the conditional probability, rf (Q|Z), gives no frequency at all; because the proviso "If $rf_1(A) > 0, \ldots$" in RFAx.6 is not met, the conditional probability, of "Q, given Z" is *undefined*. Like the C-conditional, if the antecedent has no true cases

TABLE 9-4 shows that, given a fixed conditional probability, $rf_2(Q|P)$, the discrepancy between it and the probability of the truth-functional conditional, $rf_1(P \supset Q)$, increases as the 1st-level probability of the antecedent, P, decreases.

Consider, for example, the cases diagrammed in TABLE 9-5 with reference classes of 25 individual squares. The top set of three diagrams, shows what we mean by '$rf_2(Q|P)$', the 2nd-level relative frequency of "Q, given P". This ratio remains 1/5, or .20 in all three cases. It is the ratio of the number of squares with both a 'p' and a 'q', to the number of all squares that have a 'p' in them.

The bottom set of three diagrams shows what we mean in the same three possible cases, by $rf_1(P \supset Q)$,—1st-level relative frequency of the truth-functional conditional. The $rf_1(P \supset Q)$, or $rf_1 \sim(P\&\sim Q)$, is the ratio of squares that *do not have ('p's and no q's),* (i.e., the squares with diagonal lines in them), to *all* of the squares (the whole Reference class). This is what the lst-level relative frequency of "$(P \supset Q)$" means. Obviously this is not what the 2nd-level rf of "Q, given P" means.

In the top three boxes, with the 2nd-level relative frequency of (Q|P) is fixed at .20, as the 1st-level relative frequency of (P) decreases from 1.00 to .60 to .20. In the bottom three boxes, the 1st-level relative frequency of the truth-functional conditional increases from .20 to .65 to .84. Comparing them, and the discrepancy between $rf_2(Q|P)$ and $rf_1(P \supset Q)$ increases from .00 to .45 to .64.

TABLE 9-5

$rf_2(Q|P)$ = 5.25 = **.20** | $rf_2(Q|P)$ = 3/15 = **.20** | $rf_2(Q|P)$ = 1/5 = **.20**

pq	**pq**	**pq**	**pq**	**pq**	**pq**	**pq**	**pq**	**p ~ q**	**p ~ q**	**pq**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**					
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**					
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**										
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**										

$rf_2(Q|P)$ = $rf_2T(P \Rightarrow Q)$ = 5/25 = **.20** | $rf_2(Q|P)$ = $rf_2T(P \Rightarrow Q)$ = 3/15 = **.20** | $rf_2(Q|P)$ = $rf_2T(P \Rightarrow Q)$ = 1/5 = **.20**

rf(P) = 25/25 = 1.00 | **rf(P) = 15/25 = .60** | **rf(P) = 5/25 = .20**

Discrepancy: = **.00** | **Discrepancy**: = **.45** | **Discrepancy**: = **.64**

$rf_1(P \supset Q)$ = $rf_1 \sim(P\&\sim Q)$ = 5/25 = **.20** | $rf_1(P \supset Q)$ = $rf_1 \sim(P\&\sim Q)$ = 12/25-**.65** | $rf_1(P \supset Q)$ = $rf_1 \sim(P\&\sim Q)$ = 21/25 = **.84**

pq	**pq**	**pq**	**pq**	**pq**	**pq**	**pq**	**pq**	**p ~ q**	**p ~ q**	**pq**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**					
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**					
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**										
p ~ q	**p ~ q**	**p ~ q**	**p ~ q**	**p ~ q**										

The relative frequency in each of the six sets of 25 squares, is the ratio of the number of squares with diagonals, over the number squares that are shaded.The shaded squares represent the denominator of each fraction. In the top three sets the conditional probability—the 2nd-level relative frequency of (Q|P)—is the ratio of those with both 'p' and 'q' (diagonals) over all those that have **p**'s (shaded). In the bottom three sets the probability of (P ⊃ Q) or ~(P&~Q) is the ratio of squares which do not have both 'p' and '~q' (diagonals), to all 25 squares in the total field (shaded).

In summary, 1) If the antecedent has a 1st-level frequency of 1.00, rf_2(Q|P) and rf_1(P ⊃ Q) agree since both are equal to rf_1(P & Q). 2) If the 1st level relative frequency of (P&Q) is equal to that of P (i.e., rf_1(P&~Q) = 0) there is no discrepancy. 3) As the 1st-level relative frequency of P decreases the discrepancy increases. 4) For any given 1st-level relative frequency of P, the discrepancy increases as the 1st-level frequency of (P&Q) approaches 0 (or as frequency of P and ~Q approaches 1.00). 5) If the lst-level relative frequency of the antecedent P is 0, then the probability of the TF-conditional is 1.0, but there is no probability ratio at all for the conditional probability. 6) As the 1st-level relative frequencies of both P and Q approach 0, the discrepancy approaches 1, i.e., an outright contradiction between the two results.

These discrepancies are consequences of the facts that (i) each instantiation of the TF-conditional of standard logic, (A ⊃ B), must be assigned either T or F exclusively, and (ii) the TF-conditional is true whenever its antecedent is not true or its consequent is true.

9.354 Solution to the "Problem of Conditional Probability"

The solution to the "problem of conditional probability" given below equates the conditional probability of "Q, given P" (i.e., rf(Q|P)), with the 2nd-level relative frequency of the truth of the corresponding C-conditional, i.e., with $\mathbf{rf_2}$T(P ⇒ Q). This solution retains all of the mathematical results of the classical axiomatization of probability, but is grounded in A-logic rather than M-logic.

In 1976 David Lewis held that "there is no way to interpret a conditional connective so that, with sufficient generality, the probabilities of conditionals will equal the appropriate conditional probabilities."[26] The crucial argument for this conclusion was the purported results of substituting any proposed conditional '(A ➔ C)' for the sentence letter D in "the familiar expansion by cases" (i.e., the Law of Compound Probabilities). This law, converted into our notation with 'rf ' for 'P', is:

Step (11) rf(D) = (rf(D|C) x rf(C)) + (rf(D|~C) x rf(~C)), where rf D > 0.

Lewis claims that if '(A ➔ C)' was substituted for 'D' throughout step 11), then using several other generally acceptable principles, the conclusion would follows that

Step (12) rf(C|A) = rf(C).

Clearly this can not be generally true. It was pointed out that with a fair die and 'A' for 'the die comes up even' and 'C' for 'the die comes up 6', 1/3 = 1/6, would result. More generally it was held that given any three pairwise incompatible wffs, each with a probablility between 0 and 1, rf(A ➔ C) = rf(C) would always have false instances.

The solution which follows is based on an critique and revision of the substitution rules used in this derivation of 'rf(C|A) = rf(C)' from the Law of Compound Probabilities.[27]

---

26. David Lewis, "Probabilities of Conditionals and Conditional Probabilities", *The Philosophical Review*, LXXXV, 3 (July 1976) , page 298. The crucial argument appears on page 300.

27. For an important clue to this part of the solution I am indebted to Ernest W. Adams' *A Primer of Probability Logic,* CSLI Publications, Stanford, CA . 1998. Holding that a "way out" of Lewis's argument is to challenge one or more of his assumptions, Adams says on page 260 that Lewis's Argument "assumes that

It begins by insisting on recognizing and preserving the distinction between 1st-level and 2nd-level relative frequencies throughout all substitutions. The Law of Compound Probabilities holds generally only if revised to include this distinction as follows:

$$\models [\mathbf{rf_1}(D) = (\mathbf{rf_2}(D|C) \text{ x } \mathbf{rf_1}(C) + \mathbf{rf_2}(D|{\sim}C) \text{ x } \mathbf{rf_1}({\sim}C))]$$

The proper meaning of the law of Compound Probabilities is indicated by the following method of proving it:

<u>Proof</u>: 1) $[rf_1(D) = rf_1(D\&C) + rf_1(D\&{\sim}C)$ [cf. RF-T11]
2) $[rf_1(D) = rf_1(C\&D) + rf_1({\sim}C\&D)$ [1),&-COMM,SynSUB]
3) $[rf_1(D) = \frac{rf_1(C\&D)}{rf_1(C)} \text{ x } rf_1(C)) + \frac{rf_1({\sim}C\&D)}{rf_1({\sim}C)} \text{ x } rf_1({\sim}C)]$ [2),Arithmetic; a = (a.c)/c]
4) $[rf_1(D) = rf_2(D|C) \text{ x } rf_1(C)) + rf_2(D|{\sim}C)\ rf_1({\sim}C)]$ [3),Df '(A|C)',SynSUB]

Of-course what is substituted for C or D must be the same at all occurrences. The symbol '(D|C)' stands for a 2nd-level relative frequency involving two terms. 2nd-level relative frequencies are characteristic only of conditional probabilities, or as we shall show, of C-conditionals. They are always equivalent to the ratio of some lst-level relative frequency over another lst-level relative frequency of equal or greater value. Wffs built up from atomic wffs using only the operators '&' and '~' can have only 1st-level relative frequencies.

Secondly, the change from 'f (P)' to 'fT(P)' throughout probability logic, is essential if the C-conditional is to be substituted for propositional variables. For 'f(P ⇒ Q)' i.e., "the frequency of the inferential conditional (P ⇒ Q)" has no clear meaning. Inferential conditionals may be valid logically or empirically, but they are not true or false in themselves. The difference between inferential conditionals (P ⇒ Q) or (P ⇒ ~Q) and asserting their truth in T(P ⇒ Q) or T(P ⇒ ~Q) is that the latter are reducible by Axiom 8-01 to synonymous statements about *de re* conjunctions of facts T(P&Q) and T(P&~Q) whereas inferential conditionals are not so reducible. Standard probability theory is about the probability that statements are, will be, or were, *true or false*. This applies to all statements including conditionals. By embedding the truth-operator 'T' in the operator 'fT(...)' for "the frequency that (...)is true", it is guaranteed that whatever is in the scope of 'fT(...)' is part of a truth-claim. Thus if '(Q ⇒ R)' is substituted for 'P' in any expression which lies in the scope of 'fT(...)', then T(...(Q ⇒ R)...) becomes a T(...T(Q ⇒ R)...) and then T(...T(Q&R)...) by Axioms 7-03,7-04, then 8-01 and 8-02.

The final step in the solution, is based on the fact that T(P ⇒ Q) always has both a meaningful 1st-level frequency relative to the Reference Class, and a different, meaningful 2nd-level relative frequency within the same Reference Class. Thus '$\mathbf{rf_1}$T(P ⇒ Q)' is synonymous with '$\mathbf{rf_1}$T(P & Q)', by Axiom 8-01 and SynSUB. This is different than the expression $\mathbf{rf_2}$T(P ⇒ Q) which replaces the conditional probability symbol, Pr (Q|P), by the probability to a C-conditional.

---

nonmaterial conditionals can be embedded in larger contexts, like '~(A ⇒ B)' and '(B & (A ⇒ B))' and moreover that thus embedded they satisfy all of the standard laws and probability functions." I interpreted this to mean that any substitution rule that permits such conditionals to be substituted for sentence letters in probability theorems, needs a restriction or revision of some sort.

For example: In TABLE 9-3 (P) had a frequency of 40, T(P&Q) had a frequency of 12. Therefore,

$$\mathbf{rf_1T(P \Rightarrow Q)} \text{ Syn } \mathbf{rf_1}\text{T(P \& Q)}, \mathbf{rf_1}\text{T(P)} = \frac{\text{fT(P)}}{\text{fT(R)}} = \frac{40}{100} = .40 \text{ and } \mathbf{rf_1}\text{T(P \& Q)} = \frac{\text{f T(P \& Q)}}{\text{fT(R)}} = \frac{12}{100} = .12.$$

$$\text{and therefore, } \mathbf{rf_2}\ (\text{P} \Rightarrow \text{Q}) = \frac{rf_1T(P\&Q)}{rf_1T(P)} = rf_2(Q|P) = \frac{.12}{.40} = .30.$$

Standard probalility logic makes no provision for a lst-level relative frequency of "Q, given P" as distinct from the defined 2nd-level relative frequency, $rf_2(Q|P)$. Indeed, the prefix 'Pr' does not recognize a difference between the kind of relative frequency intended by Pr(Q|P) and the kind relative frequency intended by Pr(Q&P), though they are always clearly different in kind.

The difficulties and absurd results of substitutions made in Section 9.352 and in Lewis's article are due to two errors: (1) substituting the 2nd-level relative frequency $\mathbf{rf_2}$(C|A) for the 1st-level relative frequency $\mathbf{rf_1}$(D) on the left-hand side of the equation in the Law of Compound Probabilities, and (2) using the principle SynSUB (or the principle of substituting truth-functional equivalents) for the substitution of 1st-level synonyms in a 2nd-level relative frequency.

The first error is simply one of ill-formedness. Substituting '$rf_2(A \Rightarrow C)$' or '$rf_2(C|A)$' for '$fr_1(D)$' in Step (11) the Law of Compound Probabilities, produces the ill-formed wff

$$rf(C|A) = ((\ rf((C|A)|C) \text{ x. } rf(C)) + (\ rf((C|A)|\sim C) \text{ x } rf(\sim C)))$$

In our notation, with all 2nd-level relative frequencies replaced by their definitions, this is expressed as

$$\frac{rf_1T(A \,\&\, C)}{rf_1T(A)} = ((\ \frac{rf_1T((C \,\&\, \frac{rf_1T(A \,\&\, C)}{rf_1T(A)})}{rf_1T(C)} \text{ x } rf_1(C)) + (\ \frac{rf_1T(\sim C \,\&\, \frac{rf_1T(A \,\&\, C)}{rf_1T(A)})}{rf_1T(\sim C)} \text{ x } rf_1(\sim C)))$$

The expressions above '$rf_1T(C)$' and '$rf_1T(\sim C)$' are ill formed because they conjoin a sentence letter with a singular term representing a mathematical ratio; e.g., '(P & 20/100)' is ill-formed. As was said earlier, the meaning is not clear.

If the difference between $\mathbf{rf_2}$T(P ⇒ Q) and $\mathbf{rf_1}$T(P ⇒ Q) is ignored false equations can be derived from mathematically correct premisses. Lewis's derivation of (12) rf(C|A) = rf(C) from Step (11), the Law of Compound Probabilities is a case in point. Substitutions for C and A in (12) yield false instantiations in all cases except when (rf(A)= 1 or rf(C) = 0 and $0 < rf(A) \leq 1$). Lewis's derivation of (12) was essentially as follows:

(11) $rf(D) = rf(D|C) \text{ x } rf(C) + rf(D|\sim C) \text{ x } rf(\sim C)$.
(11.1) $rf(C|A) = ((\ rf((C|A)|C) \text{ x. } rf(C)) + (\ rf((C|A)|\sim C) \text{ x } rf(\sim C)))$ [U-SUB,'(C|A)'for'D']
(11.2) $rf(C|A) = ((\ rf(C|C\&A) \text{ x } rf(C)) + (\ rf(C|\sim C\&A)) \text{ x } rf(\sim C)))$ [1), "Importation"]
(11.3) $\frac{rf(A\&C)}{rf(A)} = ((\ \frac{rf(C\&A\&C)}{rf(C\&A)} \text{ x } rf(C)) + ((\frac{rf(\sim C\&A \,\&\, C)}{rf(\sim C\&A)} \text{ x } rf(\sim C)))$ [2),Df 'C|D',SynSUB]

The right-hand expression '+ (...)', has the value + 0.0, since its numerator is inconsistent. Therefore we need only consider the remaining purported equality,

(11.4) $\frac{rf(A\&C)}{rf(A)} = (\frac{rf(C\&A \,\&C\ )}{rf(C\&A)}) \text{ x } rf(C)$

but, since the right side = 1 x rf(C), therefore

(12) rf(C|A) = rf(C).

Translating this derivation into our notation, the *non-uniformity of the substitutions* from Step (11) to Step (11.l) becomes very clear:

(11') $\mathbf{rf_1}$T(D) = $\mathbf{rf_2}$T(C ⇒ D) x $\mathbf{rf_1}$T(C) + $\mathbf{rf_2}$T(~C ⇒ D) x $\mathbf{rf_1}$T(~C).
(11.1') $\mathbf{rf_2}$T(A ⇒ C) = ($\mathbf{rf_2}$T((C ⇒ (A ⇒ C)) x $\mathbf{rf_1}$T(C)) + ($\mathbf{rf_2}$(~C ⇒ (A ⇒ C)) x rf_1T(~C)))

Replacing each $\mathbf{rf_2}$T(...) by its definiens, we get

(11.1'') $\dfrac{rf_1T(A\ \&\ C)}{rf_1T(A)} = \dfrac{(\mathbf{rf_1}T((C\ \&\ (A \Rightarrow C))}{\mathbf{rf_1}T(C)}$ x $\mathbf{rf_1}$T(C)) + $\dfrac{(\mathbf{rf_1}(\sim C\ \&\ (A \Rightarrow C))}{\mathbf{rf_1}T(\sim C)}$ x $\mathbf{rf_1}$T(~C)))

Whereas the two occurrences of 'D' on the right hand side of (11.1'') are replaced by '*(A ⇒ C)*' the substituent for 'D' on the lefthand side is not '(A ⇒ C)'. Rather, the whole expression 'rf_1T(D)' is replaced by the complex expressions '$\dfrac{rf_1T(A\ \&\ C)}{rf_1T(A)}$'. If we stop the parallel with Lewis's derivation here additional steps yield $\mathbf{rf_2}$T(A ⇒ C) = $\mathbf{rf_1}$(A&C), i.e., $\dfrac{\mathbf{rf_1}T(A\ \&\ C)}{\mathbf{rf_1}T(A)}$ = $\mathbf{rf_1}$(A&C) which obviously has false instantiations.

Lewis's step from to (11.1) from (11.2), involves another error,—of the second type. However once it is made it is reducible to the false general equation, (12') $\mathbf{rf_2}$T(A ⇒ C) = $\mathbf{rf_1}$(C) i.e., $\dfrac{\mathbf{rf_1}T(A\ \&\ C)}{\mathbf{rf_1}T(A)}$ = rf_1(C) which also has obviously false instantiations.

The nature of the second error is as follows. If one of two synonymous wffs of A-logic is substituted for a synonym (by SynSUB) which occurs in a 1st-level relative frequency, $\mathbf{rf_1}$T(...), then logical theoremhood and all relations of arithmetic equality, greater-than and less-than will be preserved. For example, using the Syn-theorems of analytic truth-logic, Ax.8-01 T(P ⇒ Q) Syn T(P & Q), and T8-16. T(P ⇒ Q) Syn T(Q ⇒ P), theoremhood—as well as truth-table sameness—is preserved by SynSUB in the following $\mathbf{rf_1}$T(...) contexts:

a) ⊨ [rf_1T(P ⇒ Q) Syn rf_1T(Q ⇒ P)]	[T8-16,SynSUB]
b) ⊨ [rf_1T(P ⇒ Q) Syn rf_1T(Q & P)]	[Ax.8-01,Ax.4-02,SynSUB]
c) ⊨ [rf_1T(P ⇒ Q) = rf_1T(Q ⇒ P)]	[T8-16,SynSUB]
d) ⊨ [rf_1T(P ⇒ Q) = rf_1T(Q & P)]	[T8-01,Ax4-02,SynSUB]

But substitution of these same 1st-level synonyms in contexts of the form $\mathbf{rf_2}$T(...) do not always preserve theoremhood or equality. Most instantiations will be false. For example, using SynSUB with T8-16 to replace T(P ⇒ Q) by T(Q ⇒ P) in the theorem, 9 $\mathbf{rf_2}$T(P ⇒ Q) Syn $\mathbf{rf_2}$T(P ⇒ Q), we get,

$\mathbf{rf_2}$T(P ⇒ Q) Syn $\mathbf{rf_2}$T(Q ⇒ P) hence, $\dfrac{rf_1T(P\ \&\ Q)}{rf_1T(P)} = \dfrac{rf_1T(Q\ \&\ P)}{rf_1T(Q)}$, (Instantiated in TABLE 9-3 yields $\dfrac{.12}{.40}$ = .30 = $\dfrac{.12}{.12}$ = 1.00)

Thus, although [$\mathbf{rf_1}$T(C ⇒ (A ⇒ C)) Syn $\mathbf{rf_1}$T((C & A) ⇒ C)) is a theorem, $\mathbf{rf_2}$T(C ⇒ (A ⇒ C)) is not synonymous with $\mathbf{rf_2}$T((C & A) ⇒ C)). Their denominators are different. In our notation Lewis's move from rf(C|(C|A)) to rf (C|(CA)) in Step (11.2) (with the relevant $\mathbf{rf_2}$(...) context replaced by its definiens in bold), is the move from

$$\text{(11.1')}\quad [\mathbf{rf_2}\mathrm{T}(A \Rightarrow C) = ((\frac{\mathbf{rf_1 T(C \,\&(A\&C))}}{\mathbf{rf_1 T(C)}} \times \mathrm{rf_1}(C)) + (\mathrm{rf_2}({\sim}C \Rightarrow (A \Rightarrow C)) \times \mathrm{rf_1}({\sim}C)))]$$

$$\text{to (11.2')}\quad [\mathbf{rf_2}\mathrm{T}(A \Rightarrow C) = ((\frac{\mathbf{rf_1 T((C\&A) \,\&C)}}{\mathbf{rf_1 T(C\&A)}} \times \mathrm{rf_1}(C)) + (\mathrm{rf_2}(({\sim}C\&A) \Rightarrow C) \times \mathrm{rf_1}({\sim}C)))]$$

This reduces to $\mathbf{rf_1}$T(A & C) = $\mathbf{rf_1}$T(C) which obviously has false instantiations.

Thus to produce a logic of probabilities in which conditional probability is the probability of the C-conditional, we need 1) the differentiation of 1st and 2nd-level relative frequencies throughout the system, and 2) a constraint on the substitutivity of synonyms (or truth-functional equivalents) in 2nd-level statements of relative frequency.

We can retain the rule of U-SUB intact, provided it is clear that this is a rule for substituting predicates or sentences (saturated predicates) for predicate or sentence letters. It does not sanction replacing predicate or sentence letters by terms for numbers, frequencies, or relative frequencies.

RF-R2. **U-SUB$_{\mathbf{Prob}}$**
If A is a theorem of RF
and a predicate or sentence letter has one or more occurrences in A,
and Q is any well-formed wff of basic A-logic
then the result of substituting 'Q' at all occurrences of P in A is also a theorem of RF.

The substitution of arithmetic terms is handled by the rule of substitution of equals for equals. This is a rule which operates only on the terms in mathematical equations. Mathematical equations are basically dyadic atomic sentences of the form [a=b]. Thus we have

RF-R3. **SUB = s**. (Substitutability of equals).
If [A = B] is a mathematical theorem, and [C = D] is a mathematical theorem,
and E is the result of replacing an occurrence of C in A by D,
then E is a theorem.

The special constraints apply only to the rule of substitution of synonyms. Since all synonyms in A-logic are truth-functional equivalents as well, the constraints also restrict any rule for substituting truth-functional equivalents.

RF-R1. **SynSUB$_{\mathbf{Prob}}$**. (Substitutability of synonyms in Probability Theory)
If A is a theorem of RF
and P is a wff of A-logic
and |= [P Syn Q]
and P occurs in a component of A *which has 1st-level relative frequencies only*,
and A′ is the result of substituting Q for P at that occurrence,
then A′ is a theorem of RF.

Under these rules, Errors 1 and 2 are avoided, and Lewis's argument about three pairwise incompatible wffs each with a probablility between 0 and 1 fails.[28]

This restricted rule for substituting synonyms does not violate the general principle that synonyms can be substituted in any expression without affecting truth-values. It simply requires that the expressions into which the substitution is made be expressed in terms sufficiently close to primitive notation. An analogy with the ancestor relation, may be helpful. The relations between "rf_1", "rf_2", the undifferentiated "rf " and "f" (for "frequency") is akin to the relations between "lst-generation ancestor", "2nd-generation ancestor", the undifferentiated "ancestor", and "parent". "Ancestor" is defined in terms of the "parent relation", just as "relative frequency" (rf) is defined in terms of "frequency "(f). But to be an ancestor is not the same as being a parent, just as being a relative frequency is not the same as being a frequency. A 1st-generation ancestor of an individual is a parent of that individual, but a 2nd-generation ancestor (grandparent) of an individual is not a parent of that individual, just as a 2nd-level relative frequency of a kind of individual, relative to another kind, is not a first-level frequency of that kind of individual.

Synonyms can be substituted for synonyms in every expression, but they must be substituted at the right level of notation and meaning. Every theorem of RF is subject to the substitutivity of synonyms because every one is reducible to a statement about 1st-level relative frequencies only. Once so reduced, all components are subject to $SynSUB_{Prob}$.

With these changes, $rf_2T(P \Rightarrow Q)$ corresponds exactly to the conditional probability, $rf_2(Q/P)$, as stated in standard probability axioms and as widely understood. The proviso "If $PrT(P) \neq 0$..." remains. For $rfT(P) = 0$ if and only if P is inconsistent or is not true of anything in the Reference Class. In either case the definiendum of $rf_2T(P \Rightarrow Q)$, namely, $\frac{rf_1(T(P \& Q))}{rf_1(T(P))}$, has zero or 0 as its denominator and that is not arithmetically acceptable.

We restate the axioms and substitution rules of RF with A-logic and the C-conditional:

RF-Ax.1 $rf_1T(A) \geq 0$
RF-Ax.2 $rf_1T(\sim A) = (1 - rf_1T(A))$
RF-Ax.3 $rf_1T(AvB) = rf_1T(A) + rf_1T(B) - rf_1T(A\&B)$
RF-Ax.4 If A is TAUT, then $rf_1T(A) = 1$
RF-Ax.5 If TAUT($A \equiv B$), then $rf_1T(A) = rf_1T(B)$
RF-Ax.6 If $rf_1T(A) \neq 0$, then $rf_2T(A \Rightarrow B) = \frac{rf_1T(A \& B)}{rf_1T(A)}$

and add the Rules of Inference above,

RF-R1. $SynSUB_{Prob}$. (Substitutability of synonyms).
RF-R2. $U\text{-}SUB_{Prob}$
RF-R3. SUB=s. (Substitutability of equals).

---

28. These rules preserve theorem hood for any three pairwise incompatible wffs. In TABLE 9-3 let $C = rf_1(P\&\sim Q) = .28$, $D = rf_1(Q\&\sim R) = .12$ and $E = rf_1(\sim P\&R) = .03$. These are pairwise incompatible. Let A = CvD; then $rf_1T(A) = .40$. Since we can not make either Error 1 or Error 2, we can not get either $\mathbf{rf_2}T(A \Rightarrow C = \mathbf{rf_1}T(C)$ or $\mathbf{rf_1}T(A \Rightarrow C) = \mathbf{rf_1}T(C)$. But we can apply the revised versiuon of the Law of Counpound Probabilities: RF-T28. If $1 > rfT(C) > 0$, then

$$\models [rf_1T(A \Rightarrow C) = ((rf_2T(C \Rightarrow (A \Rightarrow C)) \times rf_1T(C)) + (rf_2T(\sim C \Rightarrow (A \Rightarrow C)) \times rf_1T(\sim CB))]$$
$$[\quad .28 \quad = (\quad 1.00 \quad \times \quad .28 \quad) + (\quad 0 \quad \times \quad .72 \quad)]$$

From this base, we get a different probability logic, since it includes T-operators and C-conditionals and distinguishes lst- and 2nd-level frequencies. But with A-logic and the ordinary rules of arithmetic, all of the standard theorems of Probability Theory are all derivable. In addition there are some theorems with $rf_1T(A \Rightarrow C)$, all of which reduce to $rf_1T(A \ \& \ C)$, which have no analogues in standard theory. The following are some of the usual theorems, with proofs in some cases:

RF-T1. If INC(A), then $rf_1T(A)=0$
RF-T2 $rf_1T(\sim A) = (1 - rf_1T(A))$.
RF-T3. If $rf_1T(A)=1$, then $rf_1T(\sim A)=0$
RF-T4. If TAUT(A) then $rf_1T(\sim A)=0$
RF-T5. If INC(A), then $rf_1T(AvB) = rf_1T(B)$
RF-T6. If $rf_1T(A \_ B) = 1$, then $rf_1T(A) \leq rf_1T(B)$
RF-T7. If $rf_1T(A\&B)=0$ then $rf_1T(AvB) = rf_1T(A) + rf_1T(B)$
RF-T8. $rf_1T(A\&B) = rf_1T(A) + rf_1T(B) - rf_1T(AvB)$
RF-T9. $rf_1T(A) = rf_1T((A\&B)v(A\&\sim B))$
RF-T10. $rf_1T((A\&B)v(A\&\sim B) = rf_1T(A\&B) + rf_1T(A\&\sim B)$
RF-T11. $rf_1T(A) = rf_1T(A\&B) + rf_1T(A\&\sim B))$

RF-T12. $rf_1T(A) = rf_1T(A \Rightarrow B) + rf_1 F(A \Rightarrow B))$
Proof: 1) $rf_1T(A) = rf_1T(A\&B) + rf_1T(A\&\sim B))$ [RF-T11]
2) $rf_1T(A) = rf_1T(A \Rightarrow B) + rf_1T(A\&\sim B))$ [1),Ax.8-01,$SynSUB_{Prob}$]
3) $rf_1T(A) = rf_1T(A \Rightarrow B) + rf_1F(A \Rightarrow B))$ [2),Ax.8-02,$SynSUB_{Prob}$]

RF-T13. $rf_1T(AvB) = rf_1T(A\&B) + rf_1T(A\&\sim B) + rf_1T(\sim A\&B)$

RF-T14. $rf_1T(A) = rf_1T(A\&(AvB))$
Proof: 1) $rf_1T(A \ \& \ (AvC)) = (rf_1T(A) + rf_1T(AvC) - rf_1T(A \ v \ (Av \ C))$ [RF-Ax2, $U\text{-}SUB_{Prob}$]
2) $rf_1T(A \ \& \ (AvC)) = (rf_1T(A) + rf_1T(AvC) - rf_1T(A \ v \ C))$ [1),v-ORD,$SynSUB_{Prob}$]
3) $rf_1T(A \ \& \ (AvC)) = (rf_1T(A)$ [2),Arithmetic]

RF-T15. $rf_1T(A) = rf_1T(A \Rightarrow (AvC))$
Proof: 1) $rf_1T(A) = rf_1T(A \ \& \ (AvC))$ [RF-T14]
2) $rf_1T(A) = rf_1T(A \Rightarrow (AvC))$ [1),Ax.8-01,$SynSUB_{Prob}$]
RF-T16. $rf_1T(A\&B) = rf_1T(B\&(\sim BvA))$
RF-T17. $rf_1T(A) = rf_1T((Bv\sim B)\&A)$
RF-T18. If INC($A\&\sim B$), then $rf_1T(A \supset B) = 1$
RF-T19. If TAUT(A), then $rf_1T(A\&B) = rf_1T(B)$
RF-T20. If $rf_1T(A\&B) > 0$ then $rf_1T(A) > 0$
RF-T21. If TAUT($A1\&\ldots\&An) \supset B$, then $rf_1T(\sim B) \leq (rf_1T(\sim A1)+\ldots+rf_1T(\sim An))$
RF-T22. If TAUT(A), then $rf_2T(A \Rightarrow B) = rf_1T(B)$

RF-T23. $rf_2T(A \Rightarrow B) + rf_2F(A \Rightarrow B)) = 1.00$
Proof: 1) $rf_1T(A\&B) + rf_1T(A\&\sim B)) = rf_1T(A)$ [RF-T11(converse)]
2) $\frac{rf_1T(A \Rightarrow B)}{rf_1T(A)} + \frac{rf_1T(A \Rightarrow \sim B)}{rf_1T(A)} = \frac{rf_1T(A)}{rf_1T(A)}$ [1),Arithmetic, If a+b=c then a/d +b/d = c/d)]
3) $rf_2T(A \Rightarrow B)+rf_2T(A \Rightarrow \sim B)) = \frac{rf_1T(A)}{rf_1T(A)}$ [2), Df '$rf_2T \Rightarrow$ ',$SynSUB_{Prob}$]
4) $rf_2T(A\&B) + rf_2T(A\&\sim B)) = 1.00$ [3), Arithmetic; a/a = 1.00]

RF-T24. $rf_2T(A \Rightarrow B) = rf_1T(B)$ ***iff*** $rf_2T(B \Rightarrow A) = rf_1T(A)]$

Proof: 1) $rf_2T(A \Rightarrow B) = rf_1T(B)$ [Premiss]

2) $\frac{rf_1T(A\&B)}{rf_1T(A)} = rf_1T(B)$ [1, Df 'rf₂T ⇒ ',SynSUB$_{Prob}$]

3) $\frac{rf_1T(B\&A)}{rf_1T(A)} = rf_1T(B)$ [2), &-COMM,SynSUB$_{Prob}$]

4) $\frac{rf_1T(B\&A)}{rf_1T(B)} = rf_1T(A)$ [3), Arithmetic: if a/b=c,then a/c=b)]

5) $rf_2T(B \Rightarrow A) = rf_1T(A)$ [4), Df 'rf₂T ⇒ ',SynSUB$_{Prob}$]

6) If $rf_2T(A \Rightarrow B) = rf_1T(B)$ then $rf_2T(B \Rightarrow A) = rf_1T(A)$ [1) to 5), Cond. Proof]

7) If $rf_2T(B \Rightarrow A) = rf_1T(A)$ then $rf_2T(A \Rightarrow B) = rf_1T(B)$ [6), Re-lettering]

The Special Conjunction Rule:

RF-T25. If $rf_2T(A \Rightarrow B) = rf_1T(B)$ then $(rf_1T(A\&B) = rf_1T(A) \times rf_1T(B)]$

RF-T26. If $rf_1T(A) > 0$, then $rf_1T(A\&B) = rf_2T(A \Rightarrow B) \times rfT(A)$

RF-T27. $rf_1T(A \Rightarrow B) = (rf_2F(A \Rightarrow B) \times rf_1T(A))$

Proof: 1) $\frac{rf_1T(A\&B)}{rf_1T(A)} = rf_2T(A \Rightarrow B))$ [Df 'rf₂T ⇒ ']

2) $rf_1T(A\&B) = (rf_2T(A\&B) \times rf_1T(A))$ [1),Arithmetic: If a/b =c then a= (c.b)]

The Law of Compound Probability:

RF-T28. If $1 > rfT(B) > 0$, then $\models [rf_1T(A) = ((rf_2T(B \Rightarrow A) \times rf_1T(B)) + (rf_2T(\sim B \Rightarrow A) \times rf_1T(\sim B))]$

Proof: 1) $[rf_1T(A) = rf_1T(A\&B) + rf_1T(A\&\sim B)$ [RF-T12]

2) $[rf_1T(A) = rf_1T(B\&A) + rf_1T(\sim B\&A)$ [1),&-COMM,SynSUB$_{Prob}$]

3) $[rf_1T(A) = \frac{rf_1T(B\&A)}{rf_1T(B)} \times rf_1T(B)) + \frac{rf_1T(\sim B\&A)}{rf_1T(\sim B)} \times rf_1T(\sim B)]$ [2),Arithmetic]

4) $[rf_1T(A) = \frac{rf_1T(B \Rightarrow A)}{rf_1T(B)} \times rf_1T(B)) + \frac{rf_1T(\sim B \Rightarrow A)}{rf_1T(\sim B)} \times rf_1T(\sim B)]$ [3, Ax.8-01,SynSUB$_{Prob}$]

5) $[rf_1T(A) = rf_2T(B \Rightarrow A) \times rf_1T(B)) + rf_2T(\sim B \Rightarrow A)\, rf_1T(\sim B)]$ [3),Df 'rf₂T ⇒',SynSUB$_{Prob}$]

Bayes Theorem (Jeffrey's version)**:**

RF-T29. ***If*** $rf_1T(A) > 0$ ***and*** $rf_1T(B) > 0$ ***then*** $rf_2T(A \Rightarrow B) = \frac{rf_2T(B \Rightarrow A) \times rf_1T(B)}{rf_1T(A)}$

Proof: 1) $rf_1T(A) > 0 \,\&\, rf_1T(B) > 0$ [Hypothesis]

2) $rf_1T(A \,\&\, B) = rf_1T(A \,\&\, B)$ [Arithmetic, a=a]

3) $rf_1T(A \,\&\, B) = rf_1T(B \,\&\, A)$ [1),Ax.7-04, SynSUB$_{Prob}$]

4) $\frac{rf_1T(A \,\&\, B)}{rf_1T(B)} = \frac{rf_1T(B \,\&\, A)}{rf_1T(B)}$ [3),1) $rf_1T(B) > 0$ Arithmetic: if a=b, then a/c = b/c]

5) $rf_1T(A \,\&\, B) = \frac{rf_1T(B \,\&\, A)}{rf_1T(B)} \times rf_1T(B)$ [4),Arithmetic: a = $\underline{a \times b}$]

6) $rf_1T(A \,\&\, B) = rf_2T(B \Rightarrow A) \times rf_1T(B)$ [5),Df 'rf₂ ⇒ ']

7) $\frac{rf_1T(A \,\&\, B)}{rf_1T(A)} = \frac{rf_2T(B \Rightarrow A) \times rf_1T(B)}{rf_1T(A)}$ [6),1) $rf_1T(A) > 0$,Arithmetic: if a=b, then a/c = b/c]

8) $rf_2T(A \Rightarrow B) = \frac{rf_2T(B \Rightarrow A) \times rf_1T(B)}{rf_1T(A)}$ [7) Df 'rf₂ T ⇒ ']

9) If $rf_1T(A) > 0$ and $rf_1T(B) > 0$ then $rf_2T(A \Rightarrow B) = \frac{rf_2T(B \Rightarrow A)}{rf_1T(A)} \times rf_1T(B)$ [1) to 8),Cond.Proof]

RF-T30. If $rf_1T(A) > 0$ and $INC(A\&{\sim}B)$, then $rf_2T(A \Rightarrow B) = 1$
RF-T31. If $rf_1T(A) > 0$ and $INC(A\&{\sim}B)$, then $rf_2T(A \Rightarrow B) = 1$
RF-T32. If $rf_1T(A) > 0$ and $TAUT(A \supset {\sim}B)$, then $rf_2T\ (A \Rightarrow TB) = 0$
RF-T33. If $rf_1T(A) > 0$, then $rf_1T(A \supset B) = rf_1T(A) \text{ x } rf_2T(A \Rightarrow B) + rf_1T({\sim}A)$
RF-T34. If $rf_1T(A) > 0$, then $rf_2T(A \Rightarrow A) = 1$
RF-T35. If $rf_1T(A) > 0$, then $rf_2T(A \Rightarrow ((A\&B)v(A\&{\sim}B)) = 1$

The Multiplicative law:
RF-T36. If $rf_1T(A\&B) > 0$, then $rf_2T(A \Rightarrow (B\&C)) = (rf_2T(A \Rightarrow B) \text{ X } rf_2T((A\&B) \Rightarrow C))$

Proof: 1) $T(A \& (B \& C))$ Syn $T((A \& B) \& C)$ [A-logic, T7-404]
2) $rf_1T(A \Rightarrow (B \& C))$ Syn $rf_1T((A \& B) \Rightarrow C)$ [1),Ax.8-01, A-logic, $SynSUB_{Prob}$]
3) $\dfrac{rf_1T(A \& (B \& C))}{rf_1T(A)} = \dfrac{rf_1T((A \& B) \& C)}{rf_1T(A)}$ [2),Arithmetic $(a = b) \Rightarrow (a/c = b/c)$]
4) $\dfrac{rf_1T(A \& (B\&C))}{rf_1T(A)} = \left(\dfrac{rf_1T(A \& B)}{rf_1T(A)} \text{ x } \dfrac{rf_1T((A\&B) \&C)}{rf_1T(A \&B)}\right)$ [3),Arithmetic, $b/a = (c/a \text{ x } b/c)$]
5) $rf_2T(A \Rightarrow (B\&C)) = (rf_2T(A \Rightarrow B) \text{ x } rf_2T((A\&B) \Rightarrow C))$ [4),Df '$rf_2T \Rightarrow$ '.$SynSUB_{Prob}$]

RF-T37. If $rf_2T(A \Rightarrow B) = 1$ then $rf_2T(A \Rightarrow B) = 0$
RF-T38. If $rf_2T(A \Rightarrow B) > 0$ then $rf_2T(A \Rightarrow {\sim}B) = 1 - rf_2T(A \Rightarrow TB)$
RF-T39. $rf_1T(A \Rightarrow B) = rf_1T(A \& B)$ (New theorem)
RF-T40. $rf_2T(A \Rightarrow B) = rf_2F(A \Rightarrow {\sim}B)$ (New theorem)

This solution to the "problem of conditional probability" ties defines conditional probability in as the probability of a C-conditional in formal analytic truth-logic, instead of treating it as a strange added concept unrelated to connectives of logic.

But it does more than that. It is possible to make the probability of a C-conditional the central primitive concept in the mathematics of probability theory. For all probabilities are basically probabilities of a C-conditional. That is, all relative frequencies may be construed as frequencies that some conditional is true. All 1st-level relative frequencies and probabilities are answers to the question, "what is the relative frequency that an entity is P, **if** it is member of the reference Class R"? The 1st-level frequency of T(P) is the same as 1st-level-frequency of $T(R \Rightarrow P)$:

$$rf_1T(P) = \frac{fT(R \Rightarrow P)}{f\,T(R)} = \frac{fT(R \& P)}{f\,T(R)} = rf_1T(R \Rightarrow P).$$

Given any field of reference, R, there will be one predicate, 'R<1>' which applies to just the individuals in the reference class. It will be a predicate which is not logically necessary but is true of all members of the reference class and false of none of them. If the reference class is a finite group of individual entities for which there is no common property, the predicate '<1> is a member of R' has as its extension just the reference class, however determined, chosen as the basis for judgement of relative frequencies.

$$rf_1(R) = \frac{fT(R)}{fT(R) + f(T{\sim}R)} = \frac{fT(R)}{fT(R) + 0} = \frac{fT(R)}{fT(R)} = 1.0$$

The axioms of probability theory could be re-formulated such that every axiom and theorem was either a C-conditional statement about conditional probabilities, or an assertion concerning conditional probabilities (i.e., probabilities that some C-conditional is true) and the term 'Valid' (which applies only to C-conditionals) could replace 'TAUT'. However, this will not be pursued in this book.[29]

In this section we have relied heavily on the distinction between 'lst- and 2nd-level' frequencies. This seemed a good way to make our points. However, the same line of argument can be made in a semantically plausible way, by replacing letting PrT(P ⇒ Q) replace 'rf_1T(P ⇒ Q)' and 'rf(TP ⇒ TQ)' replace 'rf_2 T(P ⇒ Q)'. Obviously since ⊨ [T(P ⇒ Q) Syn T(P&Q)] by Ax. 8-01, Pr(T(P ⇒ Q)) = Pr(T(P&Q)). In other words, the probability that (If P then Q) is true in the Reference Class is the same as the probability that (P and Q) is true in the Reference Class. But the probability that Q is true, *if* P is true, i.e., Pr (TP ⇒ TQ) is a different matter. Here we have the probability of an inferential conditional, not of the truth of a conditional. '(TP ⇒ TQ)' says "If P is true, then Q is true" which means, "In those *cases in which* P is true, then Q is also true" or "In those *cases in which* P is true, (P&Q) is true". We want to know what is the proportion of those cases in which P is true, to those in which Q is *also* true (i.e., both P and Q are true). The answer is obviously, the number of cases in which both (P & Q) is true, divided by the number of cases in which P itself is true. This is precisely the conditional probability, Pr (Q|P) of standard probability theory.

$$\mathrm{Pr}(\mathrm{TP} \Rightarrow \mathrm{TQ}) = \mathrm{Pr}(\mathrm{Q}|\mathrm{P}) = \frac{\mathrm{PrT(P\&Q)}}{\mathrm{PrT(P)}}$$

We will leave the re-writing of the logic of probabilities along these lines to another time, though this will be referred to again in Section 10.

29. The identification of conditional probability with the probability of C-conditionals, was proposed in in my papers "Conditionals and Probabilities" (Wayne State University, Philosophy Department Colloquium, March 1, 1986), and "Conditional Probability and Nicod's Conditional" (Society for Exact Philosophy, Memorial University, St John's, Newfoundland, August 28, 1987). But the defense of this proposal against the arguments in Lewis's article, by focusing on the rule of substitution for variables as the "way around" Lewis's article was motivated by Adams' book (see Fn. 247) only in 1999. In addition, I was struck by some of Adams' diagrams with rectangles overlapping rectangles, and probabilities assigned to the resulting regions. These diagrams seemed vague to me, and I wanted to get down to showing how properties of individuals in a sample led to a conclusions about relative frequencies. The result was the "scrabble-tile" type model in TABLEs 9-3 and 9-5.

Chapter 10

Problems of Mathematical Logic and Their Solutions in A-logic

"Logic is concerned with the principles of valid inference"
William and Martha Kneale[1]

In this chapter we summarize problems confronting Mathematical Logic and show how they are eliminated or solved in A-logic.

The main source of problems for Mathematical Logic is its flawed definition of logical validity. The flaw lies in defining logical validity in terms of truth and falsehood. Specifically, it lies in holding that an inference is logically valid if and only if it is impossible for its premisses to be true and its consequent false, and in identifying logical validity with universal truth. The main source of the problems of M-logic is not the truth-functional interpretation of 'If…then'; for the interpretation of "Not both P and not-Q" as "If P then Q" is not essential to Mathematical Logic's formal system of "valid theorems". To be sure, the truth-functional interpretation of the conditional fits neatly with M-logic's concept of validity and reinforces M-logic's claim to be a universal logic. Without including some account of conditionals M-logic could not claim to be a complete theory of logic. However, the failure of the truth-functional conditional to do jobs which conditionals are expected to do in science and ordinary discourse poses additional serious problems for M-logic.

In the following sections the connection of the problems of Mathematic Logic to its account of validity is emphasized first. We consider problems arising directly from the concept of validity in M-logic (Section 10.1). Following this are somewhat more subtle problems about inferences by "Addition" (e .g., TP **.:** T(PvQ)) when engaged in *de re* reasoning; our claim that these inferences are problematic, contrasts with M-logic's unqualified dogma that (P v Q) is a validly deducible from P even when Q has no connection of any sort with P (Section 10.2). Finally there are more familiar problems that arise when various important uses of "if P then Q" in ordinary discourse and science are interpreted with a truth-functional conditional (Section 10.3).

1. William and Martha Kneale, *The Development of Logic*, Oxford, 1964, p.1, The first sentence.

10.1 Problems Due to the Concept of Validity in M-logic

First, we consider the problem posed by the "Paradox of the Liar" (Section 10.11). Here the problem is that while Truth is supposed to be what logic is all about, the predicate 'is true' can not be admitted in the object language of logic without leading, by derivations that are valid in M-logic, to a contradiction.

Next are at least three kinds of arguments which are "valid" in M-logic, but are *non sequiturs* in ordinary discourse and A-logic (Section 10.12). First are the non-sequiturs of Strict Implication, then follow the *non sequiturs* by Substitutivity of Truth-Functional Equivalents, and then the *non sequiturs* by Substitutivity of Factual Equivalents.

A-logic's definition of logical validity renders the first problem—the paradox—non-paradoxical by virtue of its consistency requirement, and it eliminates the various *non sequiturs* by its containment requirement.

10.11 The "Paradox of the Liar" and Its Purported Consequences

> "The liar is one of many puzzles connected with the notions of truth and falsehood … These are of great importance to logic, because the fundamental notion of logic is validity and this is definable in terms of truth and falsehood."
>
> William and Martha Kneale[2]

We agree that the fundamental notion that logic is concern with is validity, but deny that this is definable in terms of truth-and falsehood. Among other consequences, the effort to so define it leads to a paradox. Tarski held that 'is true' can not be admitted as a predicate in the object language of logic due to the "Paradox of the Liar".[3] Tarski, like the Kneales, accepts the definition of 'validity' in M-logic.

In A-logic the predicate '<1>is true' and the T-operator are admitted without paradox, since validity is not defined in terms of truth and falsehood. The "paradox of the liar" is of little importance because it is not a true logical paradox for A-logic. To be a true logical paradox one must produce a contradiction using only the rules for producing a valid inference as defined in the given system of logic.

A simple version of how the predicate '<1>is true' leads to paradox in M-logic is: Let us give the sentence, 'S_1 is not true' the name 'S_1';

1) S_1 = 'S_1 is not true'. [Premiss; this is factually true, by the act of naming]
2) [S_1 is true $\equiv$ S_1 is true] [Theorem of M-logic; T1-11 in A-Logic]
3) [S_1 is true $\equiv$ 'S_1 is not true' is true] [2),1), Substitution of Identicals)]
4) ['S_1 is not true' is true $\equiv$ S_1 is not true] [Tarski's Convention T]
5) [S_1 is true $\equiv$ S_1 is not true] [3),4),Substitution of TF-Equivalents]
6) [(S_1 is true $\equiv$ not (S_1 is true)) $\equiv$ (S_1 is true & not(S_1 is true)] [Theorem of M-logic]
T F F T T T F F T
F F T F T F F T F
7) S_1 is true & S_1 is not true [5),6),Substitution of TF-equivalents]

Clearly the premiss is true and the conclusion follows logically in accordance with M-logic. The proof begins by assigning a meaning to the symbol 'S_1'. Step 1) in the proof is a factual truth based on act of naming that was a result of human decision. Step 2) is a Tautologous Biconditional, a theorem

---

2. William and Martha Kneale, Opus Cit., p.16.

3. This has been attributed to Eubulides, 4th Century B.C. (in Diogenes Laertius, ii. 108). St Paul in his *Epistles to Titus* wrote of a Cretan prophet (Epimenides) who said, "All Cretans are liars". Cicero expressed it as: "A man says he is lying. Is what he says true or false?" (*De Divinatione*,ii. 11, Academica,ii. 96)

of M-logic. Step 3) follows from steps 1) and 2) by Substitution of Identicals (or Co-extensive terms). Step 4) is a substitution instance of Tarski's Convention T which is generally accepted in M-logic. Step 5) is gotten from 3) by a sort of biconditional syllogism. Step 6) is a general theorem of M-logic establishing truth-functional equivalence of 5) and 7), so that 7) follows from 5) by the principle of the substitutivity of TF-equivalents. Hence this is an valid argument in M-logic that leads by an M-valid argument from a true premiss to a false, self-contradictory conclusion.

This is a paradox for M-logic because it is M-valid from one point of view and not M-valid from another. The conclusion follows by M-logic's rules of transformation from a premiss which is true. But the conclusion is a contradiction and hence is not true. Thus the argument can not be M-valid, because to be M-valid means to have no case in which the premisses are true and the conclusion is false. Thus it is both M-valid and not-M-valid. This is the paradox. If the predicate '<1>is true' is allowed, the concept of M-validity would be inconsistent. This result would be doubly disturbing for M-logic, because every statement "follows logically" from a contradiction according to M-logic. Thus if '<1>is true' were allowed to apply to S_1, every sentence in M-logic's object language could be proved true from Step 5) as premiss so that M-logic would be rendered both inconsistent and completely pointless.

To avoid these results one could

1) reject or place restrictions on the rules for naming sentences, or
2) reject or place restrictions on the Rule of Substitution of Identicals, or
3) reject or place qualificatons on the Logical Theorems in Steps 2) and 4), or
4) reject or place restrictions on the Rule of Substitutivity of TF-equivalents, or
5) exclude the predicate 'is true' from the object language of logic, or
6) define "logically valid" differently.

Tarski and mathematical logicians generally have adopted the fifth course. A-logic avoids paradox by the sixth path. The Liar is not a paradox for A-logic, because it does not conflict with its concept of validity. In A-logic "A therefore C" and "If A then C" are logically valid if and only if both (i) A Cont C (explicitly or implicitly) and (ii) (A & C) is satisfiable. What always renders the derivations in the Liar A-invalid is not a violation of (i) but a violation of (ii), that the premisses and conclusion must be satisfiable together. In contrast, M-logic does not require (A & C) to be consistent if "A therefore C" is to be valid. The inferences, '(P & ∼P), therefore Q' and '(P & ∼P & Q), therefore Q' are both M-valid; but neither is A-valid since the conjunctions '((P& ∼P) & Q)'and '((P& ∼P)& Q) & Q' are not satisfiable. Also the first fails the containment requirement (though the second does not). Thus the fact that an inconsistency can be derived from an expression A means that A can not be the premiss of a valid argument in A-logic. By contrast, in M-logic such an A is always a premiss of a valid argument to any and every possible conclusion.

In A-logic as well as M-logic, the premiss in the Liar leads to contradiction. But leading to a contradiction is not a paradox. In both logics inconsistency can not be true and the denial of an inconsistency can not be false, but the existence or production of an inconsistency is not in itself a paradox. In both logics one is free to write down inconsistencies, introduce inconsistent expressions in valid statements in accord with the (unrestricted) rule of U-SUB, or lay down premisses which lead only to inconsistencies. The contradiction is brought about by a bad choice of naming 'S_1 is not true'. Logicians can counsel people to avoid such naming if they wish to be avoid contradictions. But this, like banning '<1>is true' from M-logic's object-language, doesn't touch the source of the paradox—M-logic's definition of validity.

The contradiction derived in the "Liar Paradox" can not be avoided by A-logic. But that does not mean it is a logical paradox.

It might be thought that using the Substitutivity of Synonyms in place of the Substitutivity of Identicals, would avoid inconsistencies. One could replace [S_1 = 'S_1 is not true'] in Step 1) by the definition, [S_1 Syn S_1 is not true]. But then the inconsistencies of the Liar reappear by substitution of synonyms.

It might be thought that the derivation of the contradiction could be avoided by A-logic's rejection of Tarski's Convention T. But this is not the case. A rather different derivation of the contradiction using some of the notation and principles of A-logic, but without Tarski's convention, begins as Tarski did with a sentence as follows:

The sentence on page 544, line 11, of this book is not true.

Let 's_1' stand for the subject term of this sentence, 'The sentence on page 544, line 11, of this book'. Thus, 's_1' Syn 'the sentence on page 544, line 11 of this book' and

1) s_1 = the sentence on page 544, line 11 of this book

is true; for this says that what is designated by 's_1' is the same thing that is designated by the description 'the sentence on page 544, line 11 of this book'.

But also 'Ts_1' means "The sentence written on page 544, line 11 of this book is true", so that '$\sim Ts_1$' means "The sentence written on page 544, line 11 of this book is not true", and *this meaning of* '$\sim Ts_1$' is exactly the same as *the meaning of* s_1 (the sentence which actually appears above on page 544, line 11 of this book). And since these two sentences mean the same thing, the one is true if and only if the other is true:

2) [$Ts_1 \equiv T\sim Ts_1$] In English: "The sentence on page 544, line 11 of this book is true, if and only if it is true that the sentence on page 544, line 11 of this book is not true."
3) [$Ts_1 \equiv FTs_1$] [2),Df 'F', SynSUB]
4) [$Ts_1 \equiv \sim Ts_1$] [3), Ax.7-02, SynSUB]
5) [$(Ts_1 \equiv \sim Ts_1) \equiv (Ts_1 \ \& \sim Ts_1)$] [a TF-equivalence, by truth table tests of either logic]
6) [$Ts_1 \ \& \sim Ts_1$] [4),5),Substitution of TF-equivalents]

It might also be thought that with three values in analytic truth-logic it could be shown that the initial sentence, and/or its denial is neither true nor false, and thus that there is no contradiction. But though A-logic rejects the law of Bivalence, $\vdash$ [TP v FP], the availability of the third value and the Law of Trivalence can not prevent the derivation of inconsistencies. All of the following derivations proceed by Rules of A-logic: Letting 'S_1' stand for 'the sentence on page 544, line 11 of his book is not true'

Premiss:	1) S_1 Syn $\sim T(S_1)$	1) S_1 Syn $\sim T(S_1)$	1) S_1 Syn $\sim T(S_1)$	**1) S_1 Syn $\sim T(S_1)$**
Assume:	S_1 is True:	S_1 is not-true:	S_1 is false:	S_1 is not false:
Premiss:	2) $T(S_1)$	2) $\sim T(S_1)$	2) $T\sim(S_1)$	2) $\sim F(S_1)$
	3) $T(\sim T(S_1))$	3) $\sim T(\sim T(S_1))$	3) $T\sim(\sim T(S_1))$	3) $\sim T\sim(S_1)$
	4) $FT(S_1)$	4) $\sim FT(S_1)$	4) $TT(S_1)$ [DN]	4) $\sim T\sim(\sim T(S_1))$ T7-5]
	5) $\sim TS_1$	5) $T(S_1)$	5) $T(S_1)$	5) $\sim TT(S_1)$
	6) $(T(S_1)\&\sim T(S_1)$	6) $(\sim T(S_1)\&T(S_1)$	6) $(\sim T(S_1)\ \&\ T(S_1))$	6) $\sim T(S_1)$
	\\	INC - A-Invalid	/	7) $\sim T(S_1)\ \&\ \sim F(S_1)$
				8) **$O(S_1)$ not-Inc :** Valid

Premiss:	1) S_1 Syn $T\sim(S_1)$	**1) S_1 Syn $T\sim(S_1)$**	1) S_1 Syn $T\sim(S_1)$	1) S_1 Syn $T\sim(S_1)$
Assume:	S_1 is true:	S_1 is not-true:	S_1 is false:	S is not false:
Premiss:	2) $T(S_1)$	2) $\sim T(S_1)$	2) $T\sim(S_1)$	2) $\sim F(S_1)$
	3) $T(T\sim(S_1))$	3) $\sim T(T\sim(S_1))$	3) $T\sim(T\sim(S_1))$	3) $\sim T\sim(T\sim(S_1))$
	4) $TF(S_1)$	4) $\sim T\sim(S_1)$	4) $\sim T\sim(S_1)$	4) $T\sim(S_1))$ [T7-5]
	5) FS_1	5) $\sim F(S_1)$	5) $\sim F(S_1)$	5) $F(S_1)$
	6) $(T(S_1)\&F(S_1)$	6) $(\sim T(S_1)\&\sim F(S_1))$	6) $(F(S_1)\ \&\ \sim F(S_1))$	6) $(F(S_1)\ \&\ \sim F(S_1)$
		7) **$0(S_1)$**		
	\ INC,A-invalid /	\ **not-inc**: Valid /	\ INC, A-invalid	/

But even though two of these derivations come to a point where '$0(S_1)$' is not inconsistent, and has been correctly derived, the derivation from the initial step can be continued to a contradiction:

Assume:	S is not-true and not false:		S is not-true and not-false:	
Premiss:	1) S_1 Syn $T\sim(S_1)$		1) S_1 Syn $\sim T(S_1)$	
Premiss:	2) $0(S_1)$		2) $0(S_1)$	
	3) $0(T\sim(S_1))$	[1),2),SynSUB]	3) $0(\sim T(S_1))$	[1),2),SynSUB]
	4) $0(FS_1)$	[3),Df 'F']	4) $0(TS_1)$	[3),T7-52]
	5) $\sim FS_1\ \&\ FS_1$	[4),T7-62]	5) $\sim TS_1\ \&\ TS_1$	[4),T7-61]

It might be thought that the C-conditional, being non-truth-functional, could avoid reaching the contradiction by replacing '$\equiv$' by '$\Leftrightarrow$' in Step 4). But this is not the case. In A-logic, $T(S_1 \equiv \sim S_1)$ Syn $T(S_1\ \&\sim S_1)$ and $T(S_1 \Leftrightarrow \sim S_1)$ Syn $T(S_1\ \&\sim S_1)$, though derived in different ways than in M-logic.

It might be thought that by replacing the Substitutivity of TF-equivalents by SynSUB, A-logic could avoid the paradox. But this is not so. The paradox re-appears with SynSUB. Replacing [S_1 is true $\equiv$ S_1 is true] by [S_1 is true SYN S_1 is true] doesn't help. The inconsistent Syn-statement. [S_1 is true Syn S_1 is not true] follows by A-logic's rules:

Let 'S_1' be the name of the sentence, 'S_1 is not true'.

2) [S_1 is true Syn S_1 is true]	[$T(S_1)$Syn $T(S_1)$]	[T1-11,U-SUB]
3) [S_1 is true Syn S_1 is not true is true]	[$T(S_1)$ Syn $T(\sim T(S_1))$	[2), 1), SynSUB]
4) [S_1 is true Syn S_1 is not true]	[$T(S_1)$ Syn $\sim T(S_1)$]	[3), SynSUB]

The conclusion is inconsistent, hence the argument can not be A-valid. In analytic truth-logic, as in M-logic, a valid argument can not lead from true premisses to a false or non-True conclusion. In any case in which 'S_1' is the name of the sentence, 'S_1 is not true', or, of 'S_1 is false', an inconsistent synonymy statement will follow logically from the true premiss using inference rules of A-logic.

Finally, the contradiction can not be eliminated by restrictions on self-reference. In the examples above the problem sentence is one whose subject term is the name of the problem sentence itself, so the sentence predicates non-truth of its subject, which is itself. It might be thought that elimination of self-reference—prohibiting the subject term's denoting the sentence in which it occurs—would prevent the problem. This would unjustly eliminate some perfectly good, self-evidently true sentences, such as "This sentence has exactly six words" or "This sentence is not written in the French language". But more important, elimination of self-reference would not remove the problem. For the problem can arise with two or more sentences, neither of which names itself. Imagine two sentences: the first says the second is true, the second says the first is not true. Each points by description to another independent sentence; neither mentions itself and neither is logically inconsistent.

<u>The two-sentence case</u>: Assume: S_1 = '$T(S_2)$' and S_2 = '$\sim T(S_1)$'

1) $T(S_1)$ Syn $T(T(S_2))$	Premiss	
2) $T(S_2)$ Syn $T(\sim T(S_1))$	Premiss	
3) $T(S_1)$	Premiss	
4) $T(T(S_2))$	[3),1),SynSUB]	
5) $T(T\sim T(S_1))$	[4),2),SynSUB]	
6) $T\sim T(S_1))$	[5),T7-20,SynSUB]	
7) $\sim T(S_1)$	[6),Df 'F',T7-02,SynSUB]	
8) $(T(S_1) \& \sim T(S_1))$	[3),7),ADJ]	Contradiction !

Approaching the "Paradox of the Liar" from another angle, the intuitive conditionals which come to mind upon hearing that 'S_1' is the name of the sentence '$\sim T(S_1)$' are:

1) If S_1 is true then S_1 is not true $(TS_1 \Rightarrow \sim TS_1)$
and 2) If S_1 is not true, then S_1 is true. $(\sim TS_1 \Rightarrow TS_1)$

Intuitively 1) and 2) both a) follow logically from the definition of 'S_1', and b) appear self-contradictory. In M-logic, with TF-conditionals, they are not contradictory despite appearances. 1) and 2) become

1') $(TS_1 \supset \sim TS_1)$ which is TF-equivalent and Syn to $\sim TS_1$
2') $(\sim TS_1 \supset TS_1)$ which is TF-equivalent and Syn to TS_1

But the conjunction of 1') and 2') is equivalent to 3') $(TS_1 \& \sim TS_1)$ which is a contradiction. Thus by M-logic a contradiction follows from '$((TS_1 \supset\sim TS_1) \& (\sim TS_1 \supset TS_1))$ i.e., $(TS_1 \equiv \sim TS_1)$, though neither 1) nor 2) are contradictory.

In A-logic, 1) and 2) are translated as $(TS_1 \Rightarrow \sim TS_1)$ and $(\sim TS_1 \Rightarrow TS_1)$. Both are inconsistent by Df 'Inc' (Clause iv) and both are derivable from the definition of 'S_1'. If 'S_1' is the name of the sentence 'S_1 is not true' then $[T(S_1)$ Contains $T(\sim T(S_1))]$ and $T(\sim T(S_1))$ may be said to follow from and be derivable from $T(S_1)$ *given that definition*. But containments can be inconsistent, and such is the case here. In A-logic $(TS_1 \Rightarrow\sim TS_1)$ and $(\sim TS_1 \Rightarrow TS_1)$ can be derived from these containments but they are invalid. They are self-contradictory though the M-logic analogues $(TS_1 \supset\sim TS_1)$ and $(\sim TS_1 \supset TS_1)$ are not.

Thus the contradiction involved in the Liar can not be avoided by A-logic. What is avoided is the judgment that the derivation of the contradiction is *valid*. Since the derivations are not A-valid, there is no paradox for A-logic. The inconsistent conclusions are correctly derivable from the premisses, but the derivations are not <u>valid</u>, because the conjunction of premisses and conclusion is not satisfiable.[4]

---

4. If the conjunction of premiss and conclusions must be consistent for an argument to be valid, how does this impact on Reductio ad Absurdum proofs? A Reductio ad Absurdum proof consists of two steps. The first step is to prove the premisses as a whole are inconsistent. This is done by proving that the conjunction of the premisses Contains an inconsistent conjunct. The second step involves removing or revising one or more premisses to establish consistency and then prove that the final step follows validly from the new premisses, and thus the argument and its derived conditional is valid. The proof that an expression is inconsistent, e.g., that $\models$ Inc$(P\&\sim P)$ is not a proof that the inference from a premiss to an inconsistency, e.g., $(P\&\sim P)$, is valid. A sound and valid proof that a wff is inconsistent, differs from a sound and valid proof that a given conditional wff, or an inference, is valid, e.g., $\models$ Valid$((P\&Q) \Rightarrow P)$.

In the puzzle of the Liar linguistic facts are created—certain words and sentences are written and named or referred to in certain ways as a result of human decisions. Then applying other principles used in logic they lead to contradictions. Such facts have no more significance for A-logic than the fact that any one at any time can write down a sentence of the form '(A & ~ A)'. That anybody can write or produce inconsistent statements is a factual truth. This is not part of logic and logic can not and should not try to prevent them or deny the existence of inconsistent expressions. Puzzles like the problem of the Liar are fun, with many angles to be explored. The Liar *paradoxes* only result from M-logic's concepts of the nature of logical validity. To avoid that *paradox*, what is needed is a different system of logic.

Still logical reasoning requires the absence of inconsistent statements. For those who wish to think logically, logic can counsel avoidance of acts and definitions which lead to inconsistency, and it can provide rules of definition and transformation which will not lead to inconsistencies.

10.12 Anomalies of "Valid" Non Sequiturs in M-logic

In both M-logic and A-logic the meaning of "validity" and "following from" are defined more rigorously than in traditional logic. But they are very different. To keep them separate we use the terms 'M-valid' vs. 'A-valid' and 'follows M-logically' vs. 'follows A-logically'.

In M-logic C is said to <u>follows M-logically from</u> A if and only if it is impossible that A be true and C false. *Non-sequiturs* in M-logic presumably include all purported inferences such that it is possible for the premisses to be true and the conclusion false. From the point of view of A-logic.— and ordinary usage—this is inadequate; many argument forms M-logic calls valid are also *non sequiturs*.

In A-logic C <u>follows A-logically from</u> A if and only if A logically Contains C. Thus 'A therefore C' is a *non-sequitur* if and only if A does not contain C. If an inference is a *non-sequitur* it is not valid. The containment need not be made explicit. Enthymemes (arguments in which additional premisses needed to contain the conclusion are presupposed but not expressed) like the A-implications of analytic truth-logic, are not *non-sequiturs* if the implicit containment can be made explicit. Conclusions may be consistent or inconsistent, and both kinds can be contained in, and thus follow from, inconsistent premisses. But the inference from premiss A to conclusion C will not be valid if (A&C) is inconsistent. Thus being invalid, is different from being a *non-sequitur*. C may be contained in A and still not be valid. But it can not be valid if it is a *non-sequitur*.

According to A-logic inferences in which the conclusion is not contained in the premisses, are *non sequiturs* even in cases where it could not be that the premisses were true and the conclusion false. That the such cases should be called valid is an anomaly. The M-logic anomalies in this class are independent of the question whether '⊃' should be interpreted as "If…then". All of them can be presented without using truth-functional conditionals. To keep questions about the truth-functional conditional separate, we will translate premisses and conclusions into expressions with conjunction, alternation and denial, and use 'therefore' as the connective placed between premisses and conclusion.

As an explication of "follows M-logically", consider Quine's use of "follows logically" in *Methods of Logic*. Quine begins Section 7, "Implication", in a traditional way:

> Logic is largely concerned with devising techniques for showing that a given statement does or does not "follow logically" from another.[5]

5. W.V.Quine, *Methods of Logic*, (4th Ed.) Harvard, 1982, p.45.

In illustrating the concept of "follows from" Quine ties it to M-implication. "The chief importance of logic lies in implication" he says, "which therefore, will be the main theme of this book."[6]

> From the point of view of logical theory, the fact that 'Cassius is not both lean and hungry' *follows from* 'Cassius is not hungry' is conveniently analyzed into the two circumstances: (a) the two statements have the respective logical forms '~(p&q)' and '~q'; and (b) *there are no two statements which, put respectively for 'p' and 'q' make '~p' true and '~(p&q)' false*. Circumstance (b) will hereafter be phrased in this way, '~q' implies '~(p&q)'[7]

Here '<1>M-implies<2>' is the converse of '<2>M-follows from<1>'. Both are dyadic predicates, used to describe the same relation between two sentences. In this segment we focus only on "following M-logically".[8] The general concept of "C follows M-logically from A", or "A M-implies C", then, is the following:

> C follows M-logically from A [or, A M-implies C]
> if and only if
> No two statements, A′ and C′, gotten by uniform joint substitution for any atomic component in A and C respectively, could be such that A′ is true and C′ is false.

According to this concept, if the logical form of either A or ~C is inconsistent, or if (A&~C) is inconsistent then [TA and T~C], i.e., [TA & FC] can never be the case. Thus C "follows M-logically" from A. Thus there are six possible cases in which inconsistency (which can not be true) or tautology (which in M-logic can only be true) makes C follow M-logically from A, since it can not be that both A is true and C is false:

(i) A is inconsistent and C is consistent,
or (ii) A is inconsistent and C is inconsistent,
or (iii) A is inconsistent and ~C is inconsistent (i.e., C is TAUT),
or (iv) A is contingent and C is Taut,
or (v) A is Taut and C is Taut.
or (vi) A is contingent and C is contingent, but (A&C) is inconsistent.

Each of these sanction describing some arguments as cases of "C follows M-logically from A", although C does not follow A-logically from A, and thus these arguments are non-sequiturs for A-logic and, I think, for ordinary language as well. In cases (i), (ii) and (iii) the inferences, though M-valid, are never A-valid because they can not be consistent. Cases (iv), (v) and (vi) include some cases which are

---

6. Ibid., p. 4.

7. Ibid. pp 45-6, my italics. On page 46 he also says "In a word, *implication is the validity of the conditional*." This connects Quine's concepts of "implication" and "following from" with the meaning of the TF-conditional (P ⊃ Q). See Section 10.31.

8. From the traditional point of view, it would seem natural to say that if A implies C, then C must follow logically from A, and any argument of the form, [A, therefore C], and any conditional of the form (If A then C) would be valid. However, Quine's use of the word 'valid', as well as other aspects of his use of 'implication' pose anomalies which must be dealt with.

The distinction between A-logic and M-logic on '<1>is valid' was discussed in Sections 5.5, 6.4 , and 7.424; the distinction between '<1>implies<2>' in A-logic and M-logic was discussed in Section 7.423.

M-valid but not A-valid only because there is no containment relation between A and C; in these cases C does not "follow logically" according to the definition of A-logic.

The *non sequiturs* can be divided into *non sequturs* of Strict inference (or Strict implication), and the *non seqiturs* via *Salve Veritate*.

The *non sequiturs* of Strict inference are found in the first five cases. None of the first three cases and only certain limited sub-classes in cases (iv) and (v) can be A-valid.

Additional kinds of *non sequiturs* are to be found in case (vi). These are inferences from contingent premisses to a contingent conclusion. That the premisses can not be true and the conclusion false, follows from the truth-functions or truth-values of A or C, rather than from logical inconsistency or tautology of A, or C. The *non sequiturs* of case (vi) will be discussed in Section 10.122.

10.121 Non-sequiturs of "Strict Inference"

The first class of non-sequiturs are the anomalies of "strict inference" in cases (i) to (v).

The name "paradox of strict implication" has frequently been applied to statement forms which are not really paradoxes. These are *statements* of forms like [(P & ~ P) ⊃ Q] and [Q ⊃ (P v ~ P)] that by logic can never be false. As tautologies or theorems of M-logic these are not at issue. They are best understood by their synonyms, e.g., ~ (P & ~ P & ~ Q) and ~ (Q & ~ P & P), which are obviously tautologies—denials of inconsistencies.

It is not these tautologies which are *non sequiturs*, but instantiations of the principles of inference which M-logic draws from them. The principles which are anomalous are: 1) that every statement "follows M-logically" from any inconsistency, and 2) that any statement which is always true "follows M-logically" from any statement whatever. For example, contrary to common sense,

Joe died and Joe did not die,	Snow is always red
therefore	therefore
All birds grow fish scales.	The cube root of 4913 is 17.

are arguments in which the conclusion follows M-logically from the premisses. In A-logic such arguments are provably invalid. A-logic does not have these anomalies.

The anomalies of Strict Inference, covered by cases (i) – (v), are cases in which "A therefore C" is M-valid by virtue of either the inconsistency of A or the tautologousness of C regardless of what the other may be.

The particularly egregious sub-class of M-inferences are those in which C and A have no components in common, but are said to be such that C "follows logically from" A because A is inconsistent (and always false), or because C is Tautologous (and always true), or both. These are eliminated in A-logic by the requirement that 'C follows A-logically from A' only if either C is contained in A, or A implies C. This is a necessary, but not sufficient, condition for A-validity.

The following argument schemata all represent forms of arguments in which the conclusion is said in M-logic (and on Quine's account of "following from") to "follow logically" from the premiss even when no word or sentence in the conclusion is logically contained in the premiss.

(i) [(P& ~ P) **.:** Q],
(ii) [(P& ~ P) **.:** (Q& ~ Q)],
(iii) [(P& ~ P) **.:** ~ (Q& ~ Q)],
(iv) [P **.:** ~ (Q& ~ Q)] and
(v) [~ (P& ~ P) **.:** ~ (Q& ~ Q)]

None of them are A-valid. All five of them violate the consistency requirement of A-validity. They all violate the containment requirement (1) as well.

These results are completely consistent with the observation that the following Tautologous schemata display forms of statements which, by virtue of their logical form, can never be false.

Theorems in M-logic	Equivalents in A-logic	
(a) [(P&∼P) ⊃ Q],	TAUT[(P&∼P) ⊃ Q],	(Syn INC[(P&∼P) &∼Q])
(b) [(P&∼P) ⊃ (Q&∼Q)],	TAUT[(P&∼P ⊃ (Q&∼Q)]	(Syn INC[(P&∼P &∼(Q&∼Q)])
(c) [(P&∼P) ⊃∼(Q&∼Q)],	TAUT[(P&∼P ⊃∼(Q&∼Q)]	(Syn INC[(P&∼P & Q&∼Q)])
(d) [P ⊃∼(Q&∼Q)] and	TAUT[(P ⊃∼(Q&∼Q)]	(Syn INC[P & (Q&∼Q)])
(e) [∼(P&∼P) ⊃∼(Q&∼Q)]	TAUT[[∼(P&∼P ⊃∼(Q&∼Q)]	(Syn INC[∼(P&∼P)&(Q&∼Q)])

By certain substitutions the lack of containment may be overcome. In the substitution instances of classes (i) to (iii) below the conclusion is logically Contained in the premiss thus they are not *non sequiturs* but they are still invalid in A-logic because they violated the consistency requirement.

(i') [(P&∼P) **.:** P],
(ii') [(P&∼P) **.:** (P&∼P)],
(iii') [(P&∼P) **.:** ∼(P&∼P)],

Substitutions of 'P' for 'Q' in (iv) yields 'P **.:** ∼(P&∼P))' which satisfies the requirements for A-valid implication, but not for entailment. Substitutions of 'P' for 'Q' in (v) yields '(∼(P&∼P) **.:** ∼(P&∼P))' which satisfies both requirements for A-validity.

In summary, many forms of "strict inference" in which the conclusion follows M-logically from the premisses and are M-valid, are ones that A-logic and ordinary discourse would call *non sequiturs*. In the cases in which 'following M-logically' and 'following A-logically' both apply, they apply for very different reasons.

C.I. Lewis argued that paradoxes of strict inference could not be avoided if one accepted that

1) it is valid to move, by simplification from (P&∼P) to P and to ∼P,
2) by addition from P to (PvQ),
3) by disjunctive syllogism, from ∼P and (PvQ) to Q, and
4) thence by conditional proof to (If (P&∼P) then Q).[9]

C.I. Lewis's argument does not hold in A-logic; the first step is not valid since the conjunction of premiss and conclusion is inconsistent; thus the conditional-proof step at the end is not A-valid.[10]

---

9. Clarence Irving Lewis and Cooper Harold Langford, *Symbolic Logic,* Century, 1932, p 250.

10. Anderson and Belnap, in *Entailment*, argue against Lewis's argument by rejecting the disjunctive syllogism in step 3). They say, p 299, (i) It is not true that (A & (∼AvB)) –> B. (With '–>' as their symbol for Entailment) and (ii) it is not true that there is a proof that (A and (∼AvB)) entails B. We reject the disjunctive syllogism without T-operators as invalid, but accept it with T-operators—see Sect.7.42124, T7-44 to T7-47.

10.122 Non-sequiturs via Salve Veritate

The non-sequiturs of M-logic are not limited to inferences with inconsistent premisses or tautologous conclusions. In case (vi) both premisses and conclusion are contingent. M-logic produces non-sequiturs in this case also (e.g., P **:.** (P & (Qv ~ Q))).

Logically valid arguments from contingent premisses to contingent conclusions are called deductions. As Quine says, this is the area in which formal logic has a practical importance for science and common sense.

> Logic has its practical use in inference from premisses which are not logical truths to conclusions which are not logical truths. Logic countenances such inference when the conditional sentence 'If... then...' connecting premiss with conclusion is itself logically true ... and it is in this way that logical truth links up with extra-logical conclusions.[11]

In ordinary language this seems a remarkably clear and concise statement of the role logic plays in science and practical use; and if we replace 'logically true' by 'valid' (which presumably Quine would allow) we would have a statement with which both M-logic and A-logic would appear to agree. This apparent agreement indicates a common source of data from ordinary language; but it obscures deep disagreements due to the very different meanings given to 'If...then' and 'valid' in M-logic and A-logic. To sort out genuine agreements and substantial differences is sometimes difficult.

Translated into his metalanguage about M-logic, Quine's statement means that M-logic "countenances an inference" from a non-logical truth A to a non-logical truth C, when the statement [A ⊃ C] would remain *true* no matter what sentences were substituted for its atomic sentences. Thus such inferences are *salve veritate*—they preserve truth. We assume that if M-logic "countenances an inference" that means the inference is considered a valid one, and this means that the conclusion "follows logically" from the premisses according to M-logic.

Now in M-logic '[A ⊃ C] is logically true' and '[A ⊃ C] is M-valid' and '[A M-implies C]' are all true just in case [C follows M-logically from A] is true. And as we saw,

> 'C follows M-logically from A' *means*
> 'No two statements, A' and C', gotten by uniform joint substitution for any atomic component in A and C respectively, could be such that A is true and C is false.'

while in A-logic 'following logically' requires a containment relations and the mutual consistency of premisses and conclusion, neither of which is required for validity in M-logic.

The primary substitution rule in analytic *truth*-logic is SynSUB: [From T(R) and T(P Syn Q) infer T(R(P//Q))]. It is based on the principle that if two expressions mean the same thing, then the substitution of one for the other does not change the meaning and hence does not change the truth.[12]

---

11. W.V.Quine, *Mathematical Logic*. Harvard, 1983, p. 7 (from Introduction)

12. It is universally agreed that the substitution of one synonym for another in a linguistic expression would preserve whatever truth, falsehood or nullity, consistency, inconsistency, tautology, validity and invalidity might belong to that expression. In particular, if C "follows logically" from A, replacing a component by one of its synonyms in A or C can not alter the relation of "following logically". Some, including Quine, question whether synonymy can be rigorously defined; but presumably they would agree that if two expressions were synonymous then the statement above would hold (and not just because they hold that the antecedent of this sentence is false). We, of course, hold that the concept of logical synonymity as defined in this book is rigorous and sufficient.

Two substitution rules in M-logic appear similar to the rule of SynSUB with [P ≡ Q] in place of [P Syn Q], but they are quite different. The first is the substitutivity of truth-functional equivalents:

Tf-SUB. [From T(R) and TAUT (P≡Q), infer T(R(P//Q))],

The second is the substitutivity of factual equivalents (statements with the same truth-value):

T-SUB. [From T(R) and T(P≡Q), infer T(R(P//Q))].

As purported rules of logically valid inference sanctioned in M-logic, these rules say that the conclusion to be inferred, will *follow M-logically* from the premisses. If the antecedent and consequent are satisfied, the inference can not have true premisses and a false conclusion.

The first rule is based on the principle that if R is contingently true *and* (P ≡ Q) *is TAUT* (i.e., is a theorem of M-logic) and Q is substituted for one or more occurrences of P in R, then the result of that substitution, [R(P//Q)], will have the same truth value as R. The second rule is based on the principle that if R is true *and* (P ≡ Q) *is true in fact* (i.e., if in fact P and Q are both true, or, in fact they are both false) and Q is substituted for any occurrence of P in R, then the result of that substitution, T[R(P//Q)], will have the same truth-value as R.

It is true that whenever the conditions stated in the premisses of these rules are satisfied, the conclusion allowed by these rules will have the same truth value as the statement put for R in the premisses, and thus the premisses cannot be true and the conclusion false and thus the conclusion follows M-validily from the premises.

It is equally clear that when only truth-values are concerned, the conclusion may have a very different meaning from the premisses, and that the meaning of the conclusion does not follow from the meaning of the premisses. Preservation of truth-values does not necessarily preserve meanings. The basic issue is whether we want to define "follows-logically" in terms of relations of truth-values, or in terms of containment of meanings, and particularly whether we really want to say that if it cannot be that A is true and C false then C "follows logically" from A.

10.1221 Non-sequiturs by Substitution of TF-Equivalents

A simple example of a non-sequitur by the first rule is the following:

1) P	[Premiss]
2) TAUT[P ≡ (P&(Qv ~ Q))]	[By truth-tables] [13]
3) (P & (Qv ~ Q))	[1),2), Tf-SUB]

Here neither a conjunction of the premisses, nor the conclusion, are inconsistent or tautologous, hence both premiss and the conclusion are contingent. And each side of the TF-biconditional in the second premiss is contingent. But clearly the conclusion in step 3) is not logically contained in the premiss, as containment is defined in A-logic. Thus this is a *non sequitur* for A-logic.

By simplification we get an irrelevant tautology in one more step:

4) (Qv ~ Q)	[3), SIMP]

13. The second step is synonymous by definitions of '≡' and '⊃', with 2) 'TAUT(~ (P & ~ (P&(Rv ~ R)))) & ~ ((P&(Rv ~ R)) & ~ P))'.

According to A-logic, this rule yields non-sequiturs in both steps 3) and 4), since in both cases the conclusion is not contained in the premisses, nor does the premiss A-imply the conclusion.[14] Putting 'Tito died' for both 'R' and 'P' in Tf-SUB, and 'K2 erupted' for 'Q' we get:

1) T(Tito died)	
2) Taut[Tito died ≡ (T(Tito died & (K2 erupted v K2 did not erupt)))	[Tautology]
3) T(Tito died & (K2 erupted v K2 did not erupt))	[By 1),2),Tf-SUB]
4) T(K2 erupted v K2 did not erupt)	[By 3),SIMP]

A logical deduction entitles one to skip intermediate logical principles and derived premisses and say the conclusion follows from the primary premisses. But it is difficult to imagine that any rational person would say that 4) "T(K2 erupted v K2 did not erupt)" follows logically from 1) "Tito died".

Tf-SUB is much more powerful than this simple example suggests. Any contingent wffs or statements can be put in place of P and any tautology or logical truth can be put in place of '(Qv ∼ Q)'. The principle of ***salve veritate*** holds. The replacement of any occurrence of 'P' in any wff S, by 'P&(Qv ∼ Q)', preserves truth-values.

In A-logic Tf-SUB doesn't even preserve truth-values. That (P≡Q) is Taut doesn't guarantee that Q is true if P is. For In the trivalent truth-tables of A-logic, there are cases when P is true and (P & (∼ RvR)) is not true, and in analytic truth-logic 'TP' can be true and 'T(P&(∼ RvR))' false, namely when R is neither true nor false:

P ≡ (P & (R v ∼R))	T P ≡ T(P & (R v ∼R))
T 0 T **0** 0 0 0 0	TT **F** F T 0 0 0 0 0

Thus although '(P ≡ (P & (Rv ∼ R))' is TAUT in A-logic and has no F's in its truth-table it does not maintain sameness of truth-values. 'TP≡T(P & (Rv ∼ R))' is not even Taut since it can be false.[15]

But even if it did preserve truth-values, the definitions of "following logically" and "valid" in A-logic would not be satisfied. The basic point is that for A-logic, to define the concept of "following logically" in terms of truth-values misses the essential characteristic of logical inference—containment of meanings.

10.1222 Non-sequiturs by Substitution of Material Equivalents

In the early literature of M-logic, if P and Q happened to have the same truth-values, they were said to be "materially equivalent". T-SUB, the second substitution rule mentioned is: If T(P≡Q) and T(R), then T(R(P//Q)). It says that if (P ≡ Q) is *true* (in other words if P and Q happen to have the same truth-value—if they are "materially equivalent") and Q is substituted for any occurrence of P in some statement R, then R(P//Q)—the result of substituting Q for P at one or more places in R—will have the same truth-value as R. Obviously, if R and R(P//Q) have the same truth-value, it can not be that R is true and R(P//Q) false. Thus if P and Q happen to be both true or both false, then T(R(P//Q)) follows M-logically from T(R), i.e., [TR **.:** T(R(P//Q))] is M-valid.

---

14. (∼ (Rv ∼ R) does not contain ∼ P, and ∼ (P & (R v ∼ R) does not contain ∼ P, as is required if P impl (P & (R v ∼ R)).

15. For 'TP ⊃ T(P & (Rv ∼ R))' is Syn to 'TP ⊃ (TP & (TR v FR))' by Ax.7-03 and Ax.**7**-04, and '(TR v FR)' is F when R is 0. Therefore, it doesn't follow from P's being true and [P ≡ (P & (R v ∼ R))]'s being Taut, that (R v ∼ R) is true, or that (P & (R v ∼ R)) is true. The only thing one can conclude from the fact that [P ≡ (P & (R v ∼ R))] is TAUT is that its negation would be inconsistent and unsatisfiable.

Though attention is seldom drawn to the consequences of this rule, this principle was recognized by mathematical logicians from the beginning. Gottlob Frege wrote,

> Now if our view is correct, the truth value of a sentence containing another as part must remain unchanged when the part is replaced by another sentence having the same truth value.[16]

As an example, Frege said that '$(2^2 = 4) = (2 > 1)$' is a "correct equation" since '$2^2 = 4$' and '$2 > 1$' stand for the same thing, truth.[17] In other words, '$2 > 1$' and '$2^2 = 4$' are both *true*—have the same truth-value; therefore substituting the former for an occurrence of the latter in the *true* statement '$(2^2 = 4) = (2^2 = 4)$' to yield '$(2^2 = 4) = (2 > 1)$' does not change the truth-value. This (as Frege admitted) may seem a bit arbitrary. But if we replace the middle '=' by "if and only if" the point remains that since '$2 > 1$' is *true*, and '$2^2 = 4$' is *true*, substituting '$2 > 1$' for '$2^2 = 4$' in any statement which has the latter as a component—e.g., '$((2^2 = 4)$ iff $(2^2 = 4))$'—will result in another statement '$((2^2 = 4)$ iff $(2 > 1))$' which has the same truth-value. Viewed this way, Frege was using the rule of T-SUB, and holding that since '$2 > 1$' and '$2^2 = 4$' have the same *truth-value,* the inference to '$((2^2 = 4)$ iff $(2 > 1))$' is M-valid.

In *Mathematical Logic* Quine calls this same principle the "substitutivity principle" describing it as follows:

> if ϕ and ϕ' are statements agreeing in truth value, then ϕ' can be substituted for any occurrence of ϕ in any statement ψ without affecting the truth value of ψ."[18]

In *Mathematical Logic* Quine does not use this unrestricted substitutivity principle for his *proofs*, but elsewhere he uses it in arguments against intensionalism and essentialism.

T-SUB is basically the same as Frege's principle and Quine's "substitutivity principle". If the truth-values of the containing sentence is unchanged, then it is impossible that before the replacement that sentence is true and after replacement it is false, so the inference from the one to the other must be M-valid. In other words, ***if*** P and Q have the same truth-value, ***then*** [R **.:** R(P//Q)] is M-valid, which is the same as T-SUB. We assume here that "truth" entails fixity; it is impossible that $(P \equiv Q)$ and R be true and R(P//Q) false because if unambiguous statements of fact are once true, they are always true.[19]

Now the point of our discussion is not that T-SUB is false or falsifiable. On the contrary T-SUB can be expressed and proven to be an unfalsifiable principle in analytic truth-logic. As a principle for determining the truth-value of one statement from the truth-values of other statements it is valid *de dicto*. This is not same as being valid *de re*. Valid_I [TP $\Rightarrow$ TQ] is not the same as Valid[P $\Rightarrow$ Q].

---

16. Gottlob Frege, "On Sense and Reference", see Peter Geach and Max Black, *Translations From the Philosophical Writings of Gottlob Frege* Basil Blackwell, 1970, p. 65.

17. Gottlob Frege, Opus Cit. p. 29. in "Function and Concept".

18. W.V.Quine, *Mathematical Logic*, 1983, Section 18, "Substitutivity of the Biconditional". In proofs of theorems of quantification theory and set theory, Quine does not use the pure "substitutivity principle". Rather, he employs a more restricted, advanced version of Tf-SUB, Metatheorem *123, which allows the move from a TAUT-theorem $\vdash (\phi \equiv \phi')$ to a TAUT-theorem $\vdash (\psi \equiv \psi')$ where ψ' is formed from ψ by replacing one or more occurrences of ϕ by ϕ'. This is derived with some difficulty in *Mathematical Logic* from Quine's sole primitive rule of inference, TF-*modus ponens,*and his axioms of quantification . It is a rule for moving from M-theorem to M-theorem (from TAUT to TAUT) rather than from contingent truth to contingent truth via factually true biconditionals by the substitutivity principle.

19. We may mistakenly say something is true when it is not, but if a sentence is true (or false) in fact it is impossible for it be otherwise provided its meaning stays the same.

Our main point is that T-SUB is not a general principle which establishes logical validity *per se* in A-logic. As it is used in M-logic, where assertions that a declarative sentence is true are not distinguished from the sentence itself, it produces an abundance of *non sequiturs*. In A-logic, though T-SUB determines that the truth-value of R(P//Q) is the same as that of R, it does not establish that R(P//Q) itself (not its truth-value) follows validly from R itself (not its truth-value).

That T-SUB is a rule for M-valid inference in M-logic is clear by reference to the principles of truth-tables. Take any compound factual statement expressible in the language of PM. Analyze it into components. Whether it is true or false in fact is determined by the truth-values in fact of its different components. The actual state of affairs of which it is true or false is represented by just one of the rows in the table. For example, suppose S, P, and Q all stand for statements which are true in fact; then the complex statement, '((S v ~(P & (Q v ~S)) ⊃ (P & ~Q))' is false in fact, and this is shown in Row 1.

Row 1: T T F T T T T F T **F** T **F** F T

In this situation, replacing any occurrence of any component by another expression which has the same truth-value will not alter the truth value of ((S v ~(P & (Qv ~S)) ⊃ (P & ~Q)). For example let U be any false statement. Since U and (P & ~Q) both are false in fact, (U ≡ (P& ~Q)) is true in fact, and U can be substituted by T-SUB for (P& ~Q) thoughout the statement, getting ((S v ~(P & (Q v ~S)) ⊃ U)

T **T** F T T T T F T **F F**

which is also false. Continuing, one can put P for all occurrences of S, since both have the same truth value. The result, ((P v ~(P & (Q v ~P)) ⊃ U), is also false:Or, one can take any other sentence, W, that

T **T** F T T T T F T **F F**

happens to be true, and since W and the component '(P v ~(P & (Q v ~P)))' are true, one can replace the latter by W getting (W ⊃ U) which is still false. Thus the Rule of T-SUB allows a complete

T **F** F

transformation of one sentence into another with a completely different meaning, while ensuring that the truth-value of the expression within which the substitutions are made remain unchanged.

In Analytic Truth-logic, T-SUB and Tf-SUB can be expressed and proven as the theorems,

T-SUB. Valid_I [(T(P ≡ Q) & TR) ⇒ TR(P//Q)]

and Tf-SUB. Valid_I [(Taut(P ≡ Q) & TR) ⇒ TR(P//Q)]

Substitutions by these principles never lead from antecedents that are true to consequents that are false. They can be proven in Analytic truth-logic from the principles of analytic truth-tables. But they are always merely valid *de dicto*, and support rules only for *de dicto* inferences about properties (truth or falsehood) of linguistic entities. They are not valid *de re* because, in most cases the antecedent by itself does not Logically Contain the conclusion . Thus without the subscript '$_I$',

'Valid [(Taut(P ≡ Q) & TR) ⇒ TR(P//Q)]' is not generally true.

and 'Valid [(T(P ≡ Q) & TR) ⇒ TR(P//Q)]' is not generally true,

They are valid *de dicto* only in *truth*-logic; they follow from the meaning of 'is true' and the presuppositions of truth-logic. Without the T-operator they are not valid; thus

"Valid_I[((P ≡ Q) & R) ⇒ R(P//Q)]" is false,

and "Valid_I [(Taut(P ≡ Q) & R) ⇒ R(P//Q)]" is false.

By contrast, in A-logic "Valid [((P Syn Q) & R) ⇒ R(P//Q)]" is true,

and "Valid [(T(P Syn Q) & TR) ⇒ TR(P//Q)]" is true, and both are valid *de re*.

The latter 'Syn'-for-'≡' analogue is much stricter than T-SUB or TF-SUB and holds for a much smaller class of instantiations. In those instantiations in which T(P Syn Q) is true,

Valid_I [(Taut(P ≡ Q) & TR) ⇒ TR(P//Q)]

and Valid_I [(T(P≡ Q) & TR) ⇒ TR(P//Q)] will also be true.

But these are exceptions. In all other cases T-SUB and Tf-SUB have no correlation with any 'Syn'-for-'≡' analogues that are valid *de re*.

To see how extensively T-SUB sanctions *non sequiturs* we begin with some examples from proponents of M-logic. Following Frege's approach, Church showed how, by similarity of meanings and substituting co-extensive terms, one can move through a succession of statements that have the same truth-value (true) from "Sir Walter Scott is the author of Waverley" to "The number of counties in Utah is twenty-nine".[20]

1) Sir Walter Scott is the author of Waverley [Factual truth]
2) Sir Walter Scott is the man who wrote twenty-nine Waverley novels altogether.
3) The number, such that Sir Walter Scott is the man who wrote that many Waverley novels = twenty-nine.
4) Twenty-nine = the number of Counties in Utah. [Factual truth]
5) The number, such that Sir Walter Scott is the man who wrote that many Waverley novels = the number of Counties in Utah. [4),5), SUB of =s]
6) Hence, the number of counties in Utah is twenty-nine. [From 3) and 5)]

Steps 1) and 4) are taken to be true in fact. The moves from 1) to 2) to 3), and from 4) to 5), are based on similarity of meaning, while the move from 3) and 4) to 5) and 6) are connected by the substitution of coextensive terms. Looking for some common feature of 1) through 6), Church finds only that they and all intermediate step were *true*.

By T-SUB we must conclude that

7) Sir Walter Scott is the author of Waverley **.:** the number of counties in Utah is twenty-nine.

is an M-valid deduction and that

8) If Sir Walter Scott is the author of Waverley, the number of counties in Utah is twenty-nine.

is an M-valid conditional theorem (with a TF-conditional).

This is an instance of the principle T-SUB: (TP & TQ & TR) **.:** TR(P//Q): Let P stand for sentence 1), Q for sentence 4), R for the true statement, (Either ~P or P). R(P//Q) will therefore be 7) T(Either ~Por Q), or in M-logic "If P then Q". Or the conclusion could have been "T(Q ⊃ P)" for "If Utah has 29 counties then Scott is the author of Waverley."

It seems clear, that the inference in 7) would ordinarily be considered a *non sequitur*, and the conditional in 8) would not ordinarily be consider a *logically valid* conditional.

Note that (TP & TQ & T(~ P or Q)) does not logically Contain T(~Q v P), since '~Q' does not occur premisses. Thus this is not valid *de re*. However it is valid *de dicto*, since TP A-implies T(~Qv P).

For Church an important part of the derivation was the substitution of co-extensive terms:

Step 1) The number of Waverley Novels = 29
Step 4) The number of counties in Utah = 29
.: 5) the number of counties in Utah = the number of Waverley novels
.: 7) Scott was the author of the Waverley novels if and only if the number of counties in Utah is 29.

As Church said, the *sense* of 'the number of counties in Utah' is not the same as the *sense* of 'the number of Waverley novels'. Thus, it is not true that

---

20. Alonzo Church, *Introduction to Mathematical Logic,* Vol. 1, Princeton, 1956, p. 24

‘the number of counties in Utah’ *Syn* ‘the number of Waverley novels’.

Thus SynSUB would not sanctioned the move from 5) to 7) because 5) signifies only sameness of denotation, not sameness of meaning. Different definite descriptions may have the same denotation, but that doesn’t entail that they have the same sense or meaning. Unless the two predicates in the descriptions are the same, or related in certain ways by definitions or semantic axioms, the descriptions can not be related by the semantic containment or synonymy required for SynSUB.

Church, like other proponents of M-logic, grounds his logic on the concepts of truth and falsehood, hence on extensions and denotations. A-logic is grounded on concepts of semantic synonymy and containment; it is based on *Salve Sens,* not *Salve Veritate*. The concept of truth belongs to that special branch of logic, truth-logic—not to general logic.

Davidson used T-SUB to argue that a causal statement—“The short circuit caused the fire”—can not be translated into a compound sentence with a non-truth-functional connective between sentences such as “***The fact that*** there was a short circuit ***caused it to be the case that*** there was a fire”. From the latter, using T-SUB and Tf-SUB he derived the absurd statement “*The fact that* there was a short circuit *caused it to be the case that* Nero fiddled.”[21] Having agreed that such a connective can not be truth-functional, although this purportedly showed it must be truth-functional, Davidson rejected the hypothesis that causal statements could be expressed with any sentential connective. His argument for the truth-functionality of his causal connective is (somewhat reconstructed) basically as follows:

Let ‘P’ represent ‘there was a fire’; Let ‘Q’ represent ‘Nero Fiddled’;
Assume Q is materially equivalent to P; i.e., ‘P ≡ Q’ is True.
Let ‘(S **C** P)’ represent
“*The fact that* there was a short circuit *caused it to be the case that* there was a fire.”

To be proved: If P and Q have the same truth-value and (S **C** P) is true, then (S **C** Q) is true.
I.e., then “*The fact that* there was a short circuit *caused it to be the case that* Nero fiddled.” is true.

Proof: 1) T(P ≡ Q) (I.e. P and Q have the same truth-value) [Premiss]
2) T(S **C** P)(for R) [Premiss]
3) TAUT[(x̂(x=x & P) = x̂ (x=x)) ≡ P] [M-logic w/Set theory]
4) TAUT[(x̂ (x=x & Q) = x̂ (x=x) ≡ Q] [M-logic w/Set theory]
5) T(S **C** (x̂ (x=x & P) = x̂ (x=x))) [2),3),Tf-SUB]
6a) TAUT[(P ≡ Q) ≡ (P ≡ Q)] [M-logic Theorem]
6) TAUT[(P≡Q) ≡ (x̂ (x=x & P) = x̂ (x=x)) ≡ (x̂ (x=x & Q) = x̂ (x=x)) [6a),3),4),Tf-SUB]
7) T(((x̂ (x=x & P) = x̂ (x=x)) ≡ (x̂ (x=x & Q) = x̂ (x=x))) [1),6),T-SUB]
8) T(S **C** (x̂ (x=x & Q) = x̂ (x=x)) [5),7),Sub of=s]
9) T(S **C** Q) is true. (for R(P//Q) [8),4),Tf-SUB]
Hence,10) T-SUB: If T(P≡Q) & T(S **C** P), then T(S **C** Q). [1)&2) to 8), Conditional Proof]

This result is supposed to show that the connective “the fact that … caused it to be the case that—” must be truth-functional, contrary to our wishes.[22] Thus analyses of causal statements like the one we make with the C-conditional (see Section 9.34) must be rejected. But, as we shall see, this argument has no force against the C-conditionals of A-logic.

---

21. Donald Davidson, “Causal Relations”, *Journal of Philosophy*, 64 (1967), p. 694

22. Though Davidson said it was tempting to tamper with the principles of substitution, he thought the preferable way out was to reject the idea that “… caused—” is translatable into a sentential connective.

The absurd results from Church and Davidson are the tip of the tip of an iceberge. Students in universities that take courses in Mathematical Logic are not asked to investigate the *non sequiturs* that are 'valid' according to the criterion of 'validity' in M-logic. Illustrative examples and exercises are chosen to show how traditional patterns of valid inference turn out to be 'valid' in M-logic. Indeed, all or almost all traditional patterns, and all arguments and conditionals that are A-valid in A-logic, are M-valid in M-logic. The problem is not that valid inferences that are omitted from M-logic; it is the vast number of inferences that are included as valid according to M-logic's criterion, though they are not valid in any ordinary sense. Standard textbooks do not display these easily recognizable arguments, though some textbooks of M-logic mention that every sentence is logically implied by any contradiction, and/or that every logical truth is implied by any propostion (intimating that these do not disqualify the logic). But in addition to those *non sequiturs*, and to the less dramatic ones from TF-SUB, there are the myriads of *non sequiturs* that could be presented as M-valid arguments by T-SUB, but are ignored.

Consider all statements which an ordinary individual would say he or she *knew* to be true or false. Add all those which he or she would accept as true or false after looking them up in books and libraries. Then consider all the thousands of compound statements that might have any statement, P_i, that is true or false as a component. From any of these compound statements, the result of replacing Pi by one of the millions of other statements with the same truth-value follows M-validly. Indeed, by T-SUB, every true statement (simple or complex), follows validly from every other true statement and every false statement follows validly from every other false statement. Such lack of discrimination makes 'valid' almost meaningless. A rule for determining truth-values from other truth-values, is confused with a rule for determining whether on sentence follows validly from another.

The argument Davidson used is very similar in form to an earlier argument used by Quine in support of a general policy of extensionality in mathematics and logic.[23] This argument is in effect an argument in support of T-SUB. Quine defined the policy of extensionality as follows:

> In mathematical logic…a policy of extensionality is widely expoused: a policy of admitting statements within statements truth-functionally only (apart of course from such contexts as quotation which are referentially opaque).[24]

The phrase, "admitting statements within statements truth-functionally only", means restricting substitutions to T-SUB (his "principle of substitutivity") and its sub-principle, Tf-SUB:

> An occurrence of a statement as part of a longer statement is called truth-functional if, whenever we supplant the contained statement by another statement having the same truth-value, the containing statement remains unchanged in truth-value.[25]

Quine argued that it is hard to avoid this principle in logic:

---

23. P. 161, W.V.Quine, *Ways of Paradox*, Random House, New York, 1966; in "Three Grades of Modal Involvement", (pp 157-174) reprinted from *Proceedings of the XIth International Congress of Philosophy*, Brussels, l953, Volume 14(Amsterdam: North-Holland Publishing Co.).

24. Ibid. p. 160

25. Ibid, p. 159

> Genuine violation of the extensionality policy, by admitting non-truth-functional occurrences of statements within statements … is less easy than one at first supposes. Extensionality does not merely recommend itself on the score of simplicity and convenience; it rests on more compelling grounds, as the following argument will reveal.[26]

The argument Quine presented (slightly simplified at step 8)) is essentially as follows:

Let 'P' represent any statement simple or compound,
Let '(… P…)' represents a true statement which contains the statement p as a well-formed part, such that the context represented by '(…_…)'
(i) is not referentially opaque, and
(ii) "logical equivalents are interchangeable within it *salve veritate*", (I.e., Tf-SUB holds)
Let 'Q' represent <u>any</u> statement which has the same truth-value as 'P';
<u>To be proved</u>: [(… Q …)] is true.

<u>Proof</u>:
1) [P≡Q] is true. (I.e., P & Q are both T or both F -- have the same truth-value) [Premiss]
2) [(… P…)] is true. [Premiss]
3) [$\hat{x}$(x = Λ) = ιΛ] is always true [by Quine's definition,(ιζ $=_{df}$ $\hat{\alpha}$(α = ζ))][27]
4) If P is true, '$\hat{x}$(x=Λ & P) = $\hat{x}$(x=Λ) = ιΛ' is true; if P is false, '$\hat{x}$(x=Λ & P) =Λ' is true.
Hence, TAUT[($\hat{x}$ (x=Λ & P) = ιΛ) ≡ P] [3),M-logic w/Set theory]
5) If Q is true, '$\hat{x}$(x=Λ & Q) = ιΛ' is true; if Q is false, '$\hat{x}$(x=Λ & Q) = Λ' is true.
Hence, TAUT[($\hat{x}$ (x=Λ & Q) = ιΛ) ≡ Q] [3),M-logic w/Set theory]
6) (…$\hat{x}$ (x=Λ & P) = ιΛ …) is true. (19) [2),4) Tf-SUB]
7) (($\hat{x}$ (x=Λ & P) = $\hat{x}$ (x=Λ& P)) [(x)x=x, instantiated]
8) ($\hat{x}$ (x=Λ & P) = $\hat{x}$(x=Λ & Q)) is true. (20) [7),1),T-SUB][28]
9) (… $\hat{x}$ (x=Λ & Q) = ιΛ …) is true. [6),8),SUB of identicals]
10) (… Q …) is true. [9),5),Tf-SUB]

Hence11) If [P≡Q] is true & (… P …) is true, then (… Q …) is true. [1)&2) to10), Conditional Proof]

---

26. Ibid, Page 161. The dots mark the omission of the modifying clause "without referential opacity". The relation of this concept to non-truthfumctionality and intensionality in A-logic will be discussed a few pages on.

27. <u>Note</u>: Definion D20 in Quine's *Mathematical Logic*. 'Λ' stands for the null class, 'ιΛ' for the unit class which has the null class as it sole member. Note also, whether (… P…) is true or false is irrelevant. '[… P…] is true' is replacable by '[… P …] is false', i.e., '[~(… P …)] is true', and nothing is changed. However, the truth of [P ≡ Q] is indispensable—P and Q must have the same truth values. The rule Tf-SUB is needed to get steps 5), 7) and 9), and the rule of substitutivity of co-extensive terms is needed to get step 9). T-SUB is used only at Step 7).

28. Quine got to 8) by a more round about route: "Since 'P' and 'Q' are alike in truth value, …" (Step 1))— this implicitly indicates that the next move is by T-SUB—and (assumption) each is either true or false, '(($\hat{x}$ (x=Λ & P)=Λ & $\hat{x}$(x=Λ&Q)=Λ) **v** ($\hat{x}$ (x=Λ & P)=ιΛ & $\hat{x}$(x=Λ& Q)=ιΛ)))' is true. Therefore, Since equals of equals are equal, '($\hat{x}$(x=Λ & P) = $\hat{x}$(x=Λ& Q)) **v** ($\hat{x}$ (x=Λ & P) = $\hat{x}$(x=Λ& Q))' is true. Hence by v-Idempotence, 8) (($\hat{x}$ (x=Λ & P) = $\hat{x}$(x=Λ& Q)) is true.

The conclusion, 11), is another way of stating T-SUB.[29] But it has no analogue with SynSUB that could make the conclusion valid *de re*. If 'Syn' replaces '≡' in 4) or 5) the result is false. For 'P' does not Contain '($\hat{x}$ (x=Λ & P) = ιΛ)' and 'Q' does not Contain '($\hat{x}$ (x=Λ & Q) = ιΛ)'. We can replace '=' with 'Syn' in 3), since it is gotten by substitution in a definition. But when '$\hat{x}$(x=Λ & P) = $\hat{x}$(x=Λ)' occurs, a result of replacing the main '=' by 'Syn' is false for the same sort of reason—especially in the crucial step 8). Thus the uses of T-SUB and Tf-SUB have no parallels with SynSUB, and the conclusion is not valid *de re* in A-logic.

Quine used similar arguments elsewhere against intensionalism[30] but this kind of argument has no force against A-logic and its rule of SynSUB. The policy of extensionality and the principle of substitutivity (T-SUB) are not accepted or used in basic analytic logic from Chapters 1 through 6. The first theorems of A-logic are Syn-theorems. Syn-theorems are presented as true assertions *de dicto*—as truths about language. Specifically, they are intended as assertions about a relationship of meanings of linguistic terms. These theorems are not extensional in Quine's sense. Substitutions are made within these statement by either SynSUB or U-SUB. Neither Tf-SUB (which M-logicians like to call "the Substitutivity of Logical Equivalents") nor T-SUB (Quine's "principle of substitutivity") preserves truth when applied to Syn- or Cont- or Validity-theorems in basic A-logic. Consider:

1) [(P SYN (P & P)] [This is Axiom 1-01]
. 2) Every instantiation of [P SYN (P&P)] is true.
3) 'Tito died' SYN 'Tito died and Tito died' is true [An instantiation of 1) Ax.1-01]
4) 'Tito died' is true in fact. [a contingent truth]
5) 'Hitler died' is also true in fact. [a contingent truth]
6) Substituting 'Hitler died' for an occurrence of 'Tito died' in 3) by T-SUB, yields,
'Hitler died' SYN '(Tito died and Tito died)'—which is false. [3),4),T-SUB]
or 'Tito died' SYN '(Hitler died and Tito died)'—which is false. [3),4),T-SUB]
or 'Tito died' SYN '(Hitler died and Hitler died)'—which is false, etc. [3),4),T-SUB]

T-SUB yields falsehood when applied to Cont-theorems (which reduce to Syn-theorems) and Validity-theorems (which depends on Cont-theorems). Thus in A-logic the policy of extensionality is not accepted as a way to establish logical validity. It is rejected because if followed it would lead to false statements of synonymy.

We remarked earlier that Quine allows a statement B to occur non-truth-functionally within a statement A without conflicting with the policy of extensionality, *provided* B occurs in A in a "referentially opaque" context. A referentially opaque linguistic context is one which robs the singular terms in A of their referents. Thus the true sentence, 'Hitler died' can occur non-truth-functionally in the true statement, ''Hitler died' contains two words' without adverse affect on the policy of extensionality because the occurrence of the word 'Hitler' is not used to refer to the man Hitler. That referent is

---

29. TAUT-theorems in Step 4 and 5) are said to express "logical equivalences" in M-logic.They are TF-biconditional theorems from M-logic expanded to include identity and set theory. They can have no false instances; their denials are inconsistent and never true. The rule of Tf-SUB, which is a rule for substituting "logical equivalents", is used in Steps 6) and 10). It substitutes on of the "logically equivalent" components in 4) or 5) by the other component which occurs in the wffs at step 2) and 9) respectively. But Step 8) follows only because of the assumption that P and Q have the same truth-value; it is the crucial step by T-SUB. The conclusion is provably valid *de dicto* in analytic truth-logic.

30. E.g. , in Willard Van Orman Quine, *Word and Object*, MIT Press, 1960, pp. 197-87 where he attacks modal logic again.

eliminated and what is talked about (its referent in that context) is just the ten-letter sign, 'Hitler died'. Its occurrence is non-truth-functional because its replacement by another true sentence can change the truth-value of the whole.[31] Provided non-truth-functional occurrences of sentences are restricted to occurrences in contexts like ' '…' has two words' which are referentially opaque, no harm is done. Thus Quine acknowledged the existence and legitimacy of referentially opaque statements in which occurrences of sentences within sentences are non-truth-functional, but he made clear that the policy of extensionality, was concerned with the much more important expressions in which the substitution of co-extensive expressions preserved truth-values.

Syn- Cont- and Validity-theorems of A-logic are both true and non-truth-functional, but they are not referentially opaque. The statement " '(Hitler died & Tito died)' Cont 'Tito died' " instantiates theorem T1-37. It is both true and non-truth-functional but it is not referentially opaque. It is non-truth-functional because its truth-value is not preserved if the true statement 'Stalin died' replaces one of the occurrences of 'Tito died' which has the same truth-value. But the referents of 'Hitler died', 'Tito died' and of 'Hitler died and Tito died' are not eliminated or opaque. On the semantic theory we have been using, the referents of these expressions are components of their meanings which are related by referential synonymy and containment.

The actual referent of the term 'Hitler' in my sentence 'Hitler died', is not necessaily some thing-in-itself which exists independently of my mind. It may be viewed as the content of an idea which I carry around with me in my head; namely an entity describable as "an individual human who lived in the actual world and was the dictator of Germany from 1933 to 1946." Any body who understands what I mean when I say 'Hitler died', has already an idea of a person uniquely specifiable in a way which correlates with my description. It is my belief (and I believe this is a common belief) that there was an absolutely real being, utterly independent of me or my ideas (or any body else's thoughts from now on) and that that thing-in-itself in absolute reality corresponded to the definite description I just gave. But that *thing-in-itself* can never have been in my mind. If I am going to converse about the referents of my terms and describe them, e.g., discuss whether they existed in the actual world or are merely fictional, I must refer to things accessible to me or my hearers, namely the contents of ideas in our minds—which the absolute *thing- in-itself*-Hitler, and the objectively real event of Hitler's death, are not. Thus the meaning of 'Hitler died' includes not only the meaning of the predicate 'died', but also the denotational meaning of 'Hitler' as the entity to which I attribute dying.

SynSUB, is like T-SUB, (and unlike U-SUB) in that it permits substitution at *one or more* occurrences of a component in a larger statement. But as a syntactical rule of substitution SynSUB is much stricter than T-SUB or Tf-SUB. The substituted expression must have all and only the same elementary components as the expression it replaces in order to preserve sameness of meaning. T-SUB allows the substituted expression to have any elementary components whatever, provided the substituted expression as whole has the same actual *truth-value* as the expression it replaces. In Tf-SUB the substituted expression can have any elementary components whatever if the *truth-function* of the substituted expression as whole is the same as the *truth-function* of the expression it replaces (i.e., they must have the same final column in their truth-tables).

T-SUB and Tf-SUB are *salve veritate* principles; ones which preserves the truth-value of an expression. They are valid principle for skimming along from the truth-value of one statement to the truth value

---

31. E.g., 'Rome is in Italy' is true; 'Hitler died' is true; and ' 'Hitler died' has two words' is true. But here the context ' '…' *has two words*' is referentially opaque. For, substituting 'Rome is in Italy' for 'Hitler died' to get ' 'Rome is in Italy' has two words' is false. That this use of T-SUB fails, shows that the linguistic context, ' '…' has two words', is non-truth-functional.

of another regardless of how the statements differ in meaning. Such moves does not preserve *sens*, or meaning, except incidently. SynSUB is a *salve sens* principle; one which preserves meaning or sense. The components of Syn-theorems preserve referential meaning exactly. The right-hand component of a Cont-theorem preserves an essential part of the referential meaning of the left-hand component. By the definition of 'logically valid' in A-logic, a conditional or inference is logically valid only if the consequence or conclusion Contains and preserves an essential part of the referential meaning of the antecedent or premiss. It is one thing to say that the t*ruth-value* of the conclusion will be the same as the *truth-value* of the premisses; it is another to say that the conclusion itself (not its truth- value) *follows logically* from the premisses.

Rules for determining truth-values from other truth-values have a legitimate,though minor, role to play in logic; they should not be confused with rules for determining whether one expression follows validly from another. M-logic, which conflates 'P is true' with 'P' in its object language, perpetuates this confusion by using *salve veritate* as the basis of logical validity, and dismissing *salve sens*.

10.2 Problems Due to the Principle of Addition

The Principle of Addition in M-logic is "if P, then either P or Q". Read with a TF-conditional this is symbolized as [P ⊃ (P v Q)]. In M-logic it occurs as an axiom or theorem—M-valid and logically true. In A-logic [P ⊃ (P v Q)] is a tautology; but with a C-conditional in A-logic [P ⇒ (P v Q)] is neither a valid theorem nor a tautology. In line with M-logic's semantic assumptions however, it may also be read as "if P is *true*, then it is *true* that either P or Q". In ordinary language this reading seems incontrovertible, and in analytic truth-logic, with the T-operator, [TP ⇒ T(PvQ)] is valid *de dicto*. But it is not in general valid *de re* and fallacious inferences sometimes follow if the *de dicto* validity of Addition is confused with *de re* validity.

There is a strong support from ordinary usage and from the trivalent truth-tables of A-logic for the claim that [If TP then T(PvQ)] is valid *de dicto*: (1) A conjunction is not true if one of its conjuncts is not true (see T_I7-780). (2) If one of its conjuncts is false, a conjunction is false (see T_I7-781). The principle of Addition is closely connected to these by rules of truth-logic, it says: (3) if P *is true,* this makes (P v Q) *true,* no matter what Q may be (see T_I7-783). As we saw in Sections 7.423 and 8.232, this follows from the meaning of "either…or" and "<1>is true" and from the presuppositions of analytic truth-logic. *Truth* is ascribed to (P v Q) because of the concept that P is true, and the concept of what can make "either P or Q" true.[32] There are many valid *de dicto* principles of *truth*-logic based on these considerations, including,

Ti7-780. Valid_I [~TP .: ~T(P & Q)]	Ti8-780. Valid_I [~TP ⇒ ~T(P & Q)]
Ti7-781. Valid_I [FP .: F(P & Q)]	Ti8-781. Valid_I [FP ⇒ F(P & Q)]
Ti7-783. Valid_I [TQ .: T(P v Q)]	Ti8-783. Valid_I [TQ ⇒ T(P v Q)]

If T-operators are eliminated from any of these principles (e.g., "If P then (P v Q)") the result is not

32. Kleene's 'weak' trivalence truth-tables (see footnote 23, p 343) would yield only ⊨ Valid_I [TQ ⇒ ~F(P v Q)], instead of Ti8-783. Valid_I [TQ ⇒ T(P v Q)] which would be invalid.
Under our "strong" interpretation, (TP v TQ) Syn T(P v Q) .: TP Impl T(P v Q)
Under the "weak" interpretation, (TP v TQ) Cont ~F(P v Q) .: TP Impl ~F(P v Q)
To reformulate our axiom system with this weak interpretation, Ax.7-4 [(T(P v Q) Syn (TP v TQ)] must be replaced by [(0(P v Q) Syn (0P v 0Q)]. The other consequences follow from this change.

valid in A-logic. That these principles are peculiar to the concept of truth is supported by the difference when other non-logical sentential operators are used. For example, intuitively 'It is right that P' does not ordinarily imply 'It is right that either P or Q' where Q is anything whatever.

But the inference from the *truth* of P to the *truth* of [P v Q] ***de re*** can involve a kind of *non sequitur*. P does not by itself logically Contain (P or Q) since it does not Contain Q. Q need not follow from, or be contained in, or be synonymous with, any preceding statement or any part of P. Q is replacable by any statement or well-formed formula whatever. For its *de dicto* validity, there is no reason why any particular statement or wff should be put in the blank of (P v ...) rather than any other. Q can be either true or false or tautologous or inconsistent. Thus if P is a statement which is *true* in some objective field of reference, the statement that (PvQ) is *true* does not add or subtract any possible item of information about the field of reference. The addition of Q in such cases tells us nothing about any field of reference. Thus the Q part of the consequent is a kind of *non sequitur* with respect to *de re* information. The only information that is added is *de dicto* : any *linguistic expression* which is a disjunction with a disjunct which has the property of being *true*, will have the property of being *true* as a whole. This says nothing about the truth or falsity of Q *de re*.

Therefore there are two ways the Principle of Addition can be used in analytic truth-logic:

(i) a way in which the consequent follows validly *de dicto*, but is a *non sequitur* and irrelevant with respect to some conclusion of truth or validity *de re*;
(ii) a way in which all components of the consequent are relevant in some way to the truth or validity *de re* of some conclusion.

More discussion of proper and improper uses of disjunctions in reasoning *de re* (elaborating on Sections 8.23232 and 8.23233) will clarify this distinction. Reasoning *de re* is reasoning about the way things are in some objective field of reference. The role of disjunctions and Addition in *de re* reasoning is the subject of the next two sections.

10.21 On The Proper Uses of Addition in *De Re* Inferences

Bertrand Russell once remarked,

> I do not suppose there is in the world a single disjunctive fact corresponding to 'p or q'. It does not look plausible that in the actual objective world there are facts going about which you could describe as 'p or q'.[33]

The expression "P or Q" can not describe a fact in the way that "P and Q" can. Facts in an objective field of reference are simple or complex. The simple facts consist of individual objects which have specific properties, or n-tuples of individual objects which stand in certain relations to each other.[34] Complex facts are *conjunctions* of facts about individuals or groups of individuals. There is a clear sense

33. Bertrand Russell, "The Philosophy of Logical Atomism", p. 209, in *Logic and Knowledge*; *Essays 1901-1950,* George Allen & Unwin Ltd, London. Reprinted from the *Monist*. Based on lectures in 1918.

34. Individual objects can be complex—a structure of more elementary individuals in fixed relationships. They are the designata of the subject terms, or "arguments", of sentences. Predicates stand for properties or relationships that conceivably may or may not apply to the individual entities designated by subject terms. To assert that a sentence is true or not, is to assert that the predicate of the sentence does or does not apply.

in which an "either…or" statement can not *describe* any single fact, whether complex or simple. Instead, if it has two or more different disjuncts it offers two or more possible facts, either one of which may not correspond to the reality in question. A disjunction of distinct monadic predicates (P<1> v Q<1) can not represent a specific property which might hold of an individual, and a disjunction of polyadic predicates (R<1,2,3> v S<2,13>) can not represent a state of affairs in which some n-tuple of individuals stands in precisely one relationship. Thus as Russell said, no actual facts are disjunctive in any objective field of reference. Disjunctions stand for alternative possibilities and choices, as the existentialists like to point out, except for the idempotence principle Ax.1-04 [P Syn (P v P)].

This does not mean that disjunctive predicates and statements can not refer to *possible* (whether or not actual) facts in an objective field of reference. As such they can often be used in reasoning *de re* aimed at prediction or retrodiction or subjunctive conjectures about present facts. Terms can be defined with disjunctions, and disjunctive classifications can be established based on empirically observed properties and relations in each field of reference. Empirically valid generalizations can be formed with disjunctive components in the antecedent and/or consequent.

E.g., E-Valid (∀x)(x is a cat ⇒ (x is male v x is female))
E-Valid (∀x)((x has feathers or x has gills) ⇒ (x has vertebrae))

The same is true of logically valid generalizations in the field of mathematics;

e.g., Valid [(∀x)((x = (+1. +1) **v** x = (–1. –1)) ⇒ (x = +1)]
Valid [(<1>is a positive integer ⇒ (<1>is odd **v** <1>is even)]

or in pure logic, e.g., Valid[(∀x)((Fx v (Gx & Hx)) ⇒ (Fx v Gx)]. In truth-logic such statement are valid *de re* as well as valid *de dicto*. But still, Russell was correct in saying that the truth of a disjunctive statement never entails that the statement as a whole describes a *specific actual fact*.

The principle of Addition can be invoked in *de re* investigations, when the Q added in the consequent is connected with some *de re* outcome. Often it can serve as a bridge, with T(P v Q) as the middle term, connecting statements about one actual fact with statements about other actual facts by means of the ICI-Syllogism or the CII-Syllogism,

ICI-Syll. [If (Valid_I (TP ⇒ T(PvQ)) and Valid (T(PvQ) ⇒ TR)) then Valid_I(TP ⇒ TR)].
CII-Syll. [If (Valid (T(PvQ) ⇒ TR) and Valid_I (TP ⇒ T(PvQ))) then Valid_I(TP ⇒ TR)]

Correct uses of Addition are always embedded in arguments or inferences in which the disjuncts in the consequent of Addition occur in statements that are valid or true *de re*.

With the aid of Addition rules can be derived for inferring from a description of one fact, through a conditional with a disjunctive antecedent to a description of another fact. For example,

DR-8-9 [(Valid((T(PvQ) ⇒TR) & TP) **.:** TR]
Proof: 1) Valid[T(P v Q) ⇒TR)] [Premiss]
2) TQ [Premiss]
3) Valid_I [TQ ⇒T(P v Q)] **[Addition]**
4) Valid_I[TQ ⇒TR] [1),3),CII-Syll]
5) TR [4),2),MP]
6) Valid[(Valid(T(PvQ) ⇒TR) & TP) **.:** TR]
[1) & 2) to 5),Cond. Proof]

DR-9-9 [(E-Valid((T(PvQ) ⇒TR) & TP) **.:** TR]
Proof: 1) E-Valid[T(P v Q) ⇒TR)] [Premiss]
2) TQ [Premiss]
3) Valid_I [TQ ⇒T(P v Q)] **[Addition]**
4) Valid_I[TQ ⇒TR][1),3),CII-Syll]
5) TR [4),2),MP]
6) Valid[(Valid(T(PvQ) ⇒TR) & TP) **.:** TR]
[1) & 2) to 5),Cond. Proof]

In applications of DR8-9 the major premiss may be a valid theorem: Valid[(Pv(Q&R)) ⇒ (PvR)]

<u>Example 1</u>: [Valid((Pv(Q&R)) ⇒ (PvR)) & TP) ⇒ T(P&R)]

<u>Proof</u>: 1) ((Pv(Q&R)) Cont (PvR) [Premiss from A-logic: Ax.4-04,Df 'Cont']
2) (T(Pv(Q&R)) Cont T(PvR) [1),DR7-1d]
3) TP [Premiss]
4) Valid[(T(Pv(Q&R)) ⇒ T(PvR)] [2),Df 'Valid']
5) Valid_I [TP ⇒ T(Pv(Q&R)) [T8-783 *(Addition)*]
6) Valid_I [TP ⇒ T(PvR)] [5),2),ICI-Syll]
7) T(PvR) [3),6) MP]
8) Valid((Pv(Q&R)) ⇒ (PvR)) & TP) ⇒ T(P&R) [3) to 7).Cond. Proof]

Or the major premiss may be from a definition with a disjunctive definiendum. If any one of the disjuncts in the definiendum is true, then the definiens is true. For example, the XIVth Amendment of the U.S. Constitution says: "All persons born or naturalized in the United States and subject to the jurisdiction thereof are citizens of the United States and of the states wherein they reside." From this it follows that "<1> is a citizen of the U.S." is legally defined by "<1>is born in the United States or <1>is Naturalized". Thus the proof that Jo, who has been naturalized, is a citizen of the U.S. is as follows:

1) [<1> is Cit $\textbf{Syn}_{\textbf{df}}$ (<1>is Born **v** <1> is Nat)] [U.S.Constitution, XIVth Amend]
2) T(Jo is Nat) [Premiss]
3) [T((<1>is Born) v (<1> is Nat)) Syn T(<1> is Cit)] [1),R7-1]
4) Not-Inc[T((<1>is Born) v (<1> is Nat)) & T(<1> is Cit)] [Inspection]
5) Valid[(T(<1>is Born v <1>is Nat) ⇒ T(<1>is Cit)] [3),4),Df 'Valid']
6) Valid[T((Jo is Born v Jo is Nat) ⇒ T(Jo is Cit)] [5),Inst]
7) **$\textbf{Valid}_{\textbf{I}}$[T(Jo is Nat) ⇒ T(Jo is Born v Jo is Nat)]** [T8-783 (**Addition**)]
8) Valid_I[T(Jo is Nat) ⇒ T(Jo is Cit)] [6),7,CII-Syll]
9) T(Jo is Cit) [8),2),MP]

The same use of Addition and CII is needed in the full proof than a wff or statement is inconsistent, based on the definition of inconsistency on page 290 of this book. The definiendum of this definition four disjuncts in its definiendum.

Df 'Inc'. [Inc(P)] Syn_{df} '([(P Syn (Q & ~R)) & (Q Cont R)]
v[(P Syn (Q & R)) & Inc(R)]
v[(P Syn (Q v R)) & Inc(Q) & Inc(R)]
v[(P Syn (Q ⇒ R) & Inc(Q & R))]'

If it is true that a P has the form '((R&~R&S) v (S&~R&~S))' then the full proof that P is inconsistent is based on the following substitution instance of clause (iii):

Substituting '(R&~R&S)' for 'Q' and '(S&~R&~S)' for 'R' throughout the whole definiendum of Df 'Inc' we get a substitution instance of the definiens of 'Inc(P)'. Within it only clause (iii) is true. It says,

((P Syn ((R&~R&S) v (S&~R&~S))) & Inc(R&~R&S) & Inc(S&~R&~S))
(Premiss) (provable by Clause (ii))

Since this disjunct of the whole definiends is true, it follows by Addition that the whole definiens is true. Thus we introduce a substitution instance of **Addition**:

Valid_I [TClause(iii) ⇒ T(Clause (i) v Clause (ii) v Clause (iii) v Clause (iv))

Then by SynSUB from the definition we get:

Valid_I [TClause(iii) ⇒ T(Inc(P))

From which T(Inc(P)) follows by Modus Ponens and the truth of Clause (iii);
I. e., it is true that P is inconsistent, or,

i.e., T(Inc ((R&~R&S) v (S&~R&~S)))

Finally, Addition with CII-Syll may be used to apply E-Valid principles like

Valid_E (∀x)((x has feathers or x has gills) ⇒ (x has vertebrae))

to particular objects. Suppose all that remains of some animal *a* are some leg bones and feathers. By Universal Instantiation we have

1) E-valid[(*a* has feathers or *a* has gills) ⇒ (*a* has vertebrae))	1) E-Valid(T(F*a* v G*a)* ⇒ T(V*a*))
we can then introduce the relevant instance of Addition:	
2) ⊨ (*a* has feathers ⇒ (*a* has feathers or *a* has gills))	2) ⊨ Valid_I (T(Fa) ⇒ T(F*a* v G*a))*
and by CII-Syll, we deduce, validly, that	
3) ⊨ (*a* has feathers ⇒ (*a* has vertebrae))	3) ⊨ (T(F*a*) ⇒ T(V*a*))

which can be used, together with our finding of feathers, to deduce by *Modus Ponens* that T(V*a*)— that the animal had vertebrae.

In all of these appropriate uses of Addition the consequent of Addition was the antecedent of some entailment or Cont-theorem or valid C-conditional.

Besides these uses, Addition is also needed to establish *de re* E-valid general statements e.g., "All cats are either male or female". In this use, the added disjunct in its consequent of addition must represents a property that some cat actually has had at some time. We need instances in which a cat was a male and instances in which a cat was a female. Suppose we have a domain of four entities, a,b,c, and d. Putting 'C' for 'Cat', 'M' for 'Male', 'F' for 'female', suppose also that we find

[Empirical data]	0) TCa & TMa)	&	(TCb & TFb)	&	(~TCc))	&	(TCd & TMa)
<u>It follows that</u>,							
[Simplification]	1) TMa	&	TFb	&	~TCc	&	TMd
[Addition]	2) **(TMa) ⇒ (TMa v TFa)&(TFb) ⇒ (TMb v TFb)**					&	**(TMd) ⇒ (TMd v TFd)**
[1),2) MP]	3) (TMa v TFa)	&	(TMb v TFb)			&	(TMd v TFd)
[0), Simp]	4) TCa	&	TCb			&	TCd
[4),3) ADJ]	5) TCa & (TMa v TFa))	&	TCb & (TMb v TFb))			&	TCd & (TMd v TFd))
[5),Ax t-04]	6) [TCa & T(Ma v Fa))	&	TCb & T(Mb v Fb))			&	TCd & T(Md v Fd))
[6),Ax7-03]	7) [T(Ca & (Ma v Fa))	&	T(Cb & (Mb v Fb))			&	T(Cd & (Md v Fd))
[7),Ax.8-01]	8) [T(Ca ⇒ (Ma v Fa))	&	T(Cb ⇒ (Mb v Fb))	&	~TCc	&	T(Cd ⇒ (Md v Fd))
[8),T18-840]	8) [T(Ca ⇒ (Ma v Fa))	&	T(Cb ⇒ (Mb v Fb))	&	0(Cc ⇒ (Mc v Fc))	&	T(Cd ⇒ (Md v Fd))
[9)T7-13,Df'0']	9) [~F(Ca ⇒ (Ma v Fa))	&	~F(Cb ⇒ (Mb v Fb))	&	~F(Cc ⇒ (Mc v Fc))	&	~FCd ⇒ (Md v Fd)))
	10) The whole row 9, is Syn to ($\forall_4$x)~F(Cx ⇒ (Mx v Fx)) by Df '∀',						

This establishes for that domain of four that $(\forall_4 x)\sim F(Cx \Rightarrow (Mx \text{ v } Fx))$. Of course, E-validity also requires proof that $(\exists_4 x)T(Cx \Rightarrow (Mx \text{ v } Fx))$. This follows immediately from the first and second conjuncts in Step 8). And the E-valid generalization would be false in any domain in which we discovered a cat which is neither male nor female, but this was not the case here. To my knowledge biology has not discovered such cases. But if so, general E-validity must give way to a statement of the high relative frequency of cases in which $(Cat\langle 1\rangle \Rightarrow (Male\langle 1\rangle \text{ v } Female\langle 1\rangle))$ is not-false, unless we adopt a pragmatically useful but inaccurate generalization.

In general, Addition is appropriately used to validate an empirical generalization with a disjunctive consequent, but only if each disjunct in the consequent of Addition is jointly true on some occasion with the antecedent of the generalization.

There are presumably other kinds of cases in which Addition is appropriately used in *de re* reasoning, but the examples above suffice to make clear that there is a difference between proper usages and the imappropriate uses discussed in the next section.

10.22 Irrelevant Disjuncts and Mis-uses of Addition

Since P does not contain Q if P A-implies Q, components of the implicans (the conclusion or consequent) in valid arguments may be utterly irrelevant—i.e., utterly unconnected to the implicandum (premisses, antecedent). There are endless possibilities for chains of valid A-implications which have no significance *de re*, and serve no purpose *de dicto*. We can have A-implicative syllogisms and sorites of any size, these are chains of A-implications which are valid *de dicto*:

1) Valid_I [If TP_1 then $T(P_1vP_2)$]
2) Valid_I [If $T(P_1vP_2)$ then $T(P_1vP_2vP_3)$]
....
n-1) Valid_I[If $T(P_1vP_2vP)$ then $T(P_1vP_2vP_3v\ldots vP_n)$]
.: n) Valid_I [If TP_1 then $T(P_1vP_2vP_3v\ldots vP_n)$]

De dicto validity is preserved although every Pi, $i>1$, in the last formula is instantiated by any wff or statement whatever, provided only that (P1 & (P1vP2vP3v…vPn)) is free of inconsistency. In no case does the consequent (conclusion) follow *de re* from the antecedent (premiss); it only follows ***de dicto***. Such an inference by A-implication conveys information about language, but none about facts. Indeed information about facts is diminished, since (TP v TQ) provides less information than TP or TQ by themselves. From (TP v TQ) one can not know or infer that TP is true or is false, or that TQ is true or is false.

If the principle of the AAA categorical syllogism holds (as it does), so that

$T[(\forall x)(Px \Rightarrow Qx) \;\&\; (\forall x)(Qx \Rightarrow Rx)) \Rightarrow T(\forall x)(Px \Rightarrow Rx)$ is valid *de re*,

then if we treat Addition, $(TP \Rightarrow T(PvQ))$, as we treat *de re* valid entailments, then from any Law of the form $(\forall x)(Px \Rightarrow Qx)$, it would follow that $(\forall x)(Px \Rightarrow (Qx \text{ v } Rx \text{ v } Sx \text{ v } Tx))$ would be valid *de re*. Evidence for $(\forall x)(Px \Rightarrow Qx)$ must be considered to be evidence for all other quantified conditionals which disjoin Qx with any other possible predicate applied to x.

Consider an example: A world atlas usually includes a list of towns of the world, with the countries or states they are in. Sometimes towns in different countries or states have the same name. Suppose I get a letter from Al saying he is going to the town of Mayer. Looking it up in the Atlas I find there are three towns with that name—in Arizona, Minnesota and Chile. I thus form the following factually supported inferential conditional (we assume the Atlas reports facts), with a limited but exhaustive set of alternatives:

1) If Al is going to Mayer, then either Al is going to Arizona or to Minnesota or to Chile.

Symbolizing “<1>is going to<2>” by ‘G<1,2.>’, this has the form,

$$(T(G<a,b>) \Rightarrow (T(G<a,c>) \text{ v } T(G<a,d>) \text{ v } T(G<a,e>))$$

If the Law of Addition is mis-used with no constraints on what can be added, then any disjuncts like ‘T(G<a,…>)’ can be added to the consequent with the names of any states or countries I choose at random in the blank ‘…’—or their denials. By the Law of Addition the data gotten from the Atlas would supports 2) (below) as well as 1):

2) If it is true that Al is going to Mayer,
then it is true that either Al is going to California, or to England, or to France, or to Minnesota,
or to Zambia, or to Peru, or to Chile, or to the Moon, or to Arizona or to Vermont.

Indeed we need not be confined to places. We can throw in any statement from any field of knowledge or fiction, true or false; and the *de dicto* implication will hold as long as the premiss occurs as one disjunct.

Rules of inference in logic are intended to facilitate sound logical reasoning. In cases like this the unconstrained use of the Rule of Addition introduces irrelevancies, tangents and obstacles to efficient deductive reasoning.

If we allow Addition to be used as a logical rule for *de re* inferences are used, every law-like statement that is sound, yields a proliferation of other sound lawlike statements with unlimited alternatives in the consequent.

If “All humans have eyes” is sound,	1) $(x)(T(Hx) \Rightarrow T(\exists x)$	
then, by Addition we would get	2) $(x)(T(Mx) \Rightarrow T(Ex \text{ v } Qx \text{ v } Rx)$	[Addition]
we get an equally sound principle,	3) $(x)(T(Hx) \Rightarrow T(Ex \text{ v } Qx \text{ v } Bx)$	[1),2),Syll]

“All humans either (have a trunk, or have a tail, or have eyes, or have wings)”

Mis-employing Addition to make *de re* Inferences would be a spreading cancer on any sound system of classification. If the initial classification is sound, listing real alternatives at a given level, Addition would allow adding sub-classes indiscriminately for every class. Every sound principle of dichotomy, or trichotomy or n-chotomy, proliferates by Addition into an unlimited number of different and disparate principles.

If the facts show that “All cats are either male or female” is sound *de re*, and Addition is held to be a *de re* principle, then by Syllogism, the following would be an equally sound law: “All cats are either three-headed or two-headed or four legged or one-legged or male or females or neither male nor female”, as would any one of an infinite number of other equally pointless alternatives. At best this is highly misleading and confusing; at worst, simply false (in certain senses of ‘or’). In general when we say, *de re*, that an object is F, or G, or H, we assume that there are actual or possible cases of each F, G and H. If Addition is held to always hold *de re,* this assumption is blown away.

Classificatory hierarchies are useful and are based on observable facts. They differentiate and list the significant sub-classes of a genus. Addition misused would make these same descriptions of Factual alternatives into grounds for eliminating the mutual exclusivity and exhaustive feature of such classes of classes at every level. A-logic recognizes Addition as an implication *de dicto* without constraints on substitution. But in *de re* reasoning A-logic limits the terms that may occur in its consequent if they are not contained in the antecedent.

Goodman's "new riddle of induction" is an example of how unwanted problems are created when the *de dicto* validity of Addition is construed as *de re* validity. Central to the new riddle is the concept of a "grue-type-predicate". Such a predicate is a disjunction with the disjuncts ascribing contrary properties to the same object at different times. Goodman's specific example was expressed as follows: The predicate 'is grue' applies "to all things examined before time t_i just in case they are green, but to other things just in case they are blue".[35]

Let t_1 be the specific year 2100 AD. We define a specific 'grue-type' predicate as follows:

'<1>is grue' Syn_{df} '(∀t)((<1>is green at t & t is before 2100) v (<1>is blue at t & t is after 2100)'

We assume with Goodman that the universal generalization, "All emeralds are green" is a contingent statement and that our belief that it is true is supported by inductive evidence, namely, by finding many emeralds in the past, and finding that all of them were green. Thus we assume there are many true propositions about what has been observed in years before 2100 of the form,

1) It is true that (a_i is an emerald & a_i is green at t_1 & t_1 is before 2100)

These statements have been true *de re*—found true by observation of objects in a field of reference other than the realm of ideas and language. By the Law of Addition, the ascription of truth in 1) analytically implies the ascription of truth to the disjunction in 2) below. Thus the passsage from 1) to 2) would be valid *de dicto* for each observation.

2) True ((a_i is an emerald & a_i is green at t_1 & t_1 is before 2100)
v (a_i is an emerald & a_i is blue at t_1 & t_1 is after 2100))

We all agree that statements of the form 1) help confirm the E-validity of the general law "All emeralds are green." Goodman's "new riddle of induction", assumed that statements of form 2) are valid *de re* consequences of statements of form 1) by the law of Addition. If the truth of 1) is inductive evidence for the truth of 'All emeralds are green', then according to Goodman, by the Law of Addition the truth of 2) must be inductive evidence for the truth of "All emeralds are grue". Whatever confirms "All emeralds are green" also confirms the statement "All emeralds are grue". For if Addition is logically valid *de re*, then truth *de re* of 2) follows from the truth *de re* of 1) and thus 1) supports the generalization that "all emeralds are grue". From the latter it follows in M-logic that all emeralds will be blue after 2100—a contingent generalization which we all expect to be false (and many other strange generalizations including those which replace 'blue' by any other color or property)". But these strange conclusions are clearly *non sequiturs*. No one, including Goodman, really believes they follow logically.

The fallacy in this reasoning occurs at the point where the Law of Addition is used in a *de re* inference from 1) to 2). There is no reason to accept 2) as providing any information *de re*. This move is not based on observed facts, nor is it supported by observed facts. It is a *de dicto* move, inferring from the truth of one sentence to the truth of a another merely because of the meaning of 'either…or' and 'is true'. Analytic logic avoids this in part by replacing '⊃' in universal *de dicto* generalizations by ' ⇒ ', but more importantly by holding that while the Law of Addition is valid *de dicto*, valid inferences about matters of fact must be based on principles of inference which are valid *de re*, which Addition is not.

---

35. Nelson Goodman, *Fact, Fiction and Forecast*, Bobbs-Merrill, 1965, p. 74

There is nothing essentially wrong with grue-type-predicates. Careful observation may show that at certain times an object has a certain property and at other times it has a contrary property, so that the statement that it has either one property at one time and a contrary one at another time can be justified by facts. Science needs such predicates. No one believes 'Grue' applies to emeralds in fact; but *danaus plexippus,* the monarch butterfly, metamorphoses from a 17-legged wingless caterpillar to a 6-legged four-winged flying insect before and after the chrysalis stage (the third and fourth week). An appropriately phrased grue-type predicate can describe this property. Based on many empirical observations of capterpillars that turned into monarch butterflies entymologists have the E-valid Proposition,

If $<1>$ is a *danaus plexippus*
then either ($<1>$ is wingless & 17-legged at t_j & t_j is before $<1>$ is 3-4 weeks old)
or ($<1>$ is winged & 6-legged at t_j & t_j is after $<1>$ is 3-4 weeks old))

The problem in "the new riddle of induction" is not due to grue-type predicates as such, or to any paradoxes inherent in confirmation. It is due to inappropriate employment of the principle of Addition—the *de dicto* principle, that if a sentence is true, then any disjunction containing that sentence as a disjunct is true. This principle can be employed to get true statements about language, but when injected as if it were a *de re* principle, it yields irrelevant *non sequiturs*.

In A-logic empirical generalizations are not analyzed as claims that $(\forall x)$(If Px then Qx) is *true*, for this is equivalent, with the C-conditional, to claiming that for $(\forall x)(Px \,\&\, Qx)$ is *true*. Instead an empirical generalization $(\forall x)(Px \Rightarrow Qx)$ is said to be *E-valid*, if there are no cases in which P is true and Q is false, i.e., $\sim F(\forall x)(Px \Rightarrow Qx)$ and there are some cases in which P and Q are both true, i.e., $T(\exists x)(Px \Rightarrow Qx)$.

To prove: E-valid$[(\forall x)\ (Ex \Rightarrow ((\forall t)(t \geq 2100 \Rightarrow Blue(x,t))))$
you must prove $\sim F(\forall x)\ (Ex \Rightarrow ((\forall t)(t \geq 2100 \Rightarrow Blue(x,t))))$
and $T(\exists x)\ (Ex \Rightarrow ((\forall t)(t \geq 2100 \Rightarrow Blue(x,t))))$

The part of the "grue" predicate which applies at a future times only, can't be false and this allows the non-falsehood of $(\forall x)(Px \Rightarrow Qx)$ far into the future. But it can't be true either, so the second requirement for E-validity will never be met by this particular grue-type predicate; $T(Pa \Rightarrow Qa)$ is never *observed* to be true at a future time. *Empirical* confirmations occur in the present or past only. This, with its constraints upon the uses of A-implications (including Addition) in *de re* reasoning, eliminates the "new Riddle of Induction" for A-logic. This then, has been another example of the undesirable consequences of using the *de dicto* validity of Addition, as *de re* validity-theorems are used.

In M-logic no distinction is made between an indicative sentence and a sentence asserting that the first sentence is true. Addition and Simplication are treated in the same way, so the distinction between the *de dicto* truth of Addition and the amenability to *de re* application of Simplification, can not be made. Thus the distinction between introducing disjunctions with arbitrary, irrelevant and uninformative disjuncts by Addition, and introducing disjunction which represent actual possibilities *de re* can not be made in M-logic.

In A-logic, these distinctions can be made and identified, and the undesirable consequences can be avoided by reference to these distinctions. The root of the problem is not the TF-conditional, for the problem is expressed equally well as a problem of whether 'TP **.:** T(PvQ)' is a valid inference *de re* without transforming that inference into a conditional. It is a problem of the proper employment in truth-inquiries of a rule of inference that is *de dicto* and often not relevant to the establishment of truths about the intended field of reference, and that.can lead to other unwanted rules of inference if these distinctions are not recognized.

10.3 Problems of The TF-conditional and Their Solutions

The problems of the TF-conditional are independent of the problems of M-logic's definition of validity, even though they have parallel defects which seem to reinforce each other. Everything in Section 10.1 and 10.2, could have been said without interpreting any expressions of M-logic as conditional statements. Validity can be viewed as a relation between premisses and a conclusion, and in all cases above any premiss or conclusion which used '⊃' could be expressed equivalently using only conjunction or disjunction and denial.

In ordinary language and in science the words "if…then" or synonyms in the vernacular often occur, and they play a unique role in reasoning. They do jobs that can not be done by conjunction, disjunction, and quantification alone. Competent proponents of M-logic concentrating on the philosophy of science—Carnap, Goodman, Hempel—have rigorously established some of the points at which TF-conditionals cannot do what ordinary conditionals are expected to do in dispositional statements, contrary-to-fact conditionals and confirmation processes. We also show how they fail in the logic of causal statements. Others have shown how the probability of (A ⊃ B) differs radically from the concept of the probability of (B, if A) that is used in science.

There are just three attributes of the TF-conditional that have caused problems apart from the problems of validity—they are:

1) the TF-conditional is true whenever the antecedent is false by itself,
2) the TF-conditional is true whenever the consequent is true by itself, and
3) every conditional statement is either true or false (i.e., none is neither true nor false).

The C-conditional of A-logic avoids all such problems because it does not have any of these three attributes. On the other hand it shares with the TF-conditional the attributes of being false when the antecedent is true and the consequent is false (Ax. 8.02) and being true when the antecedent is true and the the consequent is true (Ax.8.01). This is sufficient to do the jobs logic is expected to do.

Below we discuss some of the many ways the three attributes of the TF-conditional listed above have raised problems and show how the C-conditional avoids or solves them.

10.31 The TF-conditional vs. The VC\VI Principle

The metalogical principle which I have called the "Valid Conditional/Valid Inference" principle, or VC\VI, is expressed formally in A-logic as,

R6-6. [Valid(P **⇒** Q) <=> Valid(P, **.:**Q)]

Independently of its interpretation in A-logic, this principle is accepted in a broad sense by all logicians. It may be expressed without regard to how conditionals or validity are interpreted as,

VC\VI: [If (P_1 &…& P_n) then Q] is a logically valid conditional
if and only if
[P_1,…,P_n, therefore Q] is a logically valid inference.

Although the concept of validity and the concept of a conditional statement can be viewed as independent of each other, commitment to the VC\VI principle together with a given concept of validity will impose requirements on the concept a conditional statement.

In M-logic the VC\VI principle is recognized and accepted in the following sense. Expressions of the form 'If (P_1 &...& P_n) then Q' are symbolized by '((P_1&...& P_n) ⊃ Q'. Whenever such a conditional satisfies the criteria for being logically valid in M-logic, an argument of the form [P_1,...,P_n, **.:** Q] is said to be a valid argument. Conversely, based on the semantics of M-logic, whenever Q is said to be a valid consequence of {P_1,...,P_n}, '[(P_1 &...& P_n) ⊃ Q]' is a theorem of M-logic and is M-valid. This is called "the Deduction Theorem". Thus M-logic's version of VC\VI is: "[P ⊃ Q] is M-valid if and only if [P, **.:** Q] is M-valid".

The problem here is that while all M-valid wffs of the form [P ⊃ Q] are incapable of being false,[36] infinitely many inference schemata of the form, [P, **.:** Q] are *non sequiturs,* as we have repeatedly pointed out. In Section 5.53 we distinguished the classes of M-valid TF-conditionals which yield *non sequiturs* under the VC\VI principle from those which would yield inference schemata which are not *non sequiturs* and are valid in A-logic and ordinary discourse.

Of course the reason some M-valid inferences are *non sequiturs* is that the definition of a valid inference in M-logic's semantics—no case of true premisses and false conclusionis—is too inclusive. As we said above, this criterion is independent of the truth-functional interpretation of the conditional, since nothing is changed if the TF-conditional is replaced at all occurrences by (~ P v Q) or ~ (P& ~ Q).

Nevertheless the credibility of the TF-conditional is involved. For it follows from M-logic's concept of a valid inference *and* the VC\VI principle, that a conditional must be valid in case either its consequent could not be false, or its antecedent could not be true. This means that the *validity* of a conditional can be independent of any relationship between the antecedent and the consequent. This runs counter to the common notion that "If so-and-so then such-and-such" should be *logically* valid only if there is some connection between the antecedent state of affairs and the consequential state of affairs. Given M-logic's semantical definition of validity *and* its commitment to the VC\VI principle, only conditional statements that are made true whenever either the consequent is true or the antecedent is false can fill the bill. But this requisite feature is just what creates the most frequently cited examples of the "paradoxes of material implication" and the attributes 1) and 2) listed above of the TF-conditional.

A-Logic's version of the VC\VI Principle uses a different meaning for "If...then", as well as a different meaning of "logically valid". Since the validity of an inference in A-logic never follows from the premisses being inconsistent, or merely from the conclusion's being always true, the concept of A-validity *with* the VC\VI principle does not entail that the C-conditional must be true if the antecedent is false or the consequent is true. Thus VC\VI can be retained along with A-validity, without the anomalous properties of the TF-conditional. No C-conditional which is logically valid in A-logic entails or implies any member of the infinite class of *non sequiturs* which are called valid inferences in M-logic. Further, because tautologous expressions are never conditionals[37] and are not A-valid, no *non sequiturs* can follow from any tautologies via the VC\VI Principle in A-logic.

---

36. That Valid conditionals are incapable of being false is accepted by both A-logic and M-logic; but A-logic considers this a necessary but not sufficient condition of validity, while for M-logic it is both necessary and sufficient. A-logic requires in addition a formal connection and consistency.

37. Only negations of C-conditionals, e.g., ~(P ⇒ ~P), can be tautologous. A negation is not a conditional.

10.32 Anomalies of Unquantified Truth-functional Conditionals

With respect to common sense and ordinary usage, many of the conditional theorems of M-logic are anomalous. When *Principia Mathematica*, Vol 1, was published in 1913, the truth-functional conditional was called "material implication".[38] As this interpretation of 'if…then' became known outside mathematical circles, it was attacked as not capturing the ordinary meaning of 'implies' or of 'if…then'. Initially the attacked focused on unquantified or particular (rather than generalized) TF-conditionals. In 1919 G.E.Moore, a defender of common sense and an acute analyst of ordinary language, wrote,

> Why logicians should have thus chosen to use the word "implies" as a name for a relation, for which it never is used by any one else, I do not know. … It seems to me quite certain that ["It is not the case that AQ is true and AR is false"] cannot be properly expressed either by "AQ implies AR" or by "if AQ, then AR,"…[39]

He recommended using "P entails Q" to express "the converse of that relation which we assert to hold between a particular proposition *q* and a particular proposition *p*, when we assert that *q follows from p* or *is deducible from p*."[40] Though Moore, following Russell and Whitehead, confusedly called the truth-functional conditional "material implication", he correctly drew attention to the anomalous situation with respect to the identifying ~(P&~Q) with "if…then" or with "following from". Thus A-logic coincides at these points with Moore's basic critique of the 'if…then' of M-logic.

In the United States C.I. Lewis also attacked the truth-functional conditional, listing 28 different theorems of the M-logic's propositional calculus, including $\vdash (P \supset (Q \supset P))$ and $\vdash (\sim P \supset (P \supset Q))$ which he called "paradoxes of material implication".[41] From 1918 to 1932 he developed systems of "strict implication" in which "P strictly implies Q" is equivalent to "it is not possible that both P and not ~Q". This, Lewis thought, accounted for "deducibility", in contrast to "material implication" the TF-condtional.[42] Strict implication is often translated as entailment today. But as we have pointed out (e.g., in Section 10.121), strict implication has its own "paradoxes" that flout the concept of "following logically".

Later in 1946, Lewis sought the basis of empirical knowledge and the relation of perceptual judgements to beliefs in objective states of affairs. He held that the connection lies in "terminating judgments" of the conditional form "If A then E", where A describes an actual perceptual experience and E describes expected or predicted empirical sequent. Turning to the nature of the conditional in these statements, he remarked that contemporary logical studies "throw no light whatever" on its meaning.

> The relation of 'A' to 'E' in "If A then E" is *not* justly interpreted as the relation of material implication which many current developments take as fundamental'; it is *not* what is called in *Principia Mathematica,* a formal implication; and it is *not* a strict implication or entailment such that E is assertedly deducible from 'A'. It is generically the same kind of relation which Hume had in mind when he spoke of 'necessary connections between matters of fact'.[43]

---

38. A.N.Whitehead and B. Russell, *Principia Mathematica*, Vol 1, 1913, p 7
39. G.E. Moore, "External and Internal Relations"(1919), [Reprinted in G.E.Moore, *Philosophical Studies*, Littlefield Adams, 1959, see pp 296 snf 297] In this passage (my brackets) Moore gave as examples of entailments, "being a right angle" entails "being an angle" and "being red" entails "being coloured".
40. Ibid. p. 291.
41. C.I. Lewis, "Interesting theorems in symbolic logic", *Journal of Philosophy Psychology and Scientific Methods,* v. X,(l913), pp 239-42.
42. See Lewis, C.I., *A Survey of Symbolic Logic,* Univ. of California Press, 1918, and Lewis, C.I., and Langford, C.H., *Symbolic Logic,* Appleton Century, 1932.
43. Lewis, C.I., *Analysis of Knowledge and Valuation,* Open Court, l946, p 212.

On Lewis's view, to believe on the basis of an immediate visual perception in the objective truth of "There is a piece of white paper in front me", means that I believe the terminating judgment, "If I turn my eyes to the right, this seen appearance will be displaced to my left". The relation of conditionality in the latter statement is that of a subjunctive or contrary-to-fact conditional; I must believe that even if I don't move my eyes to the right, the seen appearance *would* be displaced to left if I *were* to turn my eyes to the right.[44] The C-conditional of A-logic satisfies the requirements of the kind of conditional that Lewis is talking about here, whether or not one accepts the accompanying theory of knowledge.

A few philosophers have argued that '(P ⊃ Q)' is an adequate representation of the ordinary usage of 'If…then'. For example, Paul Grice held that the ordinary conventions of intelligent conversation ("conversational implicature") prescribe that intelligent persons will not use "if …then" in the anomalous ways that follow from its definition in M-logic.[45]

Others, e.g., Peter Strawson, noting that the truth-functional conditional is best paraphrased as "it is not the case that both p and that not-Q" urged that we "foreswear its more appealing identifications with 'if p then q'".[46]

But the most interesting critiques are those discussed in Section 10.33 below. These deal with failures of specific applications of generalized TF-conditionals in empirical science.

Many standard textbooks of M-logic ignore the issues.

Others recognize the divergence from ordinary language and describe the differences, usually claiming that the advantages of the truth-functional conditional far outweigh its deviations from common usage. This is the position of Quine. Quine's general position is that 'P ⊃ Q' differs from the ordinary 'if…then' but is better suited for scientific purposes. (Cf. his argument in Section 8.12 defending the TF-conditional over the ordinary concept and A-logic's Axiom 8.01). The job of the logician, as an individual who is also physicist, mathematician etc, Quine says, involves an interest in ordinary language

> … only as a means of getting on with physics, mathematics and the rest of science; and he is happy to depart from ordinary language whenever he finds a more convenient device of extraordinary language which is equally adequate to his need of the moment in formulating and developing his physics, mathematics, or the like. He drops 'if--then' in favor of '⊃' without entertaining the mistaken idea that they are synonymous; he makes the change only because he finds that the purposes for which he had been *needing* 'if- then', in connection with his particular scientific work, happen to be satisfactorily manageable also by a somewhat different use of '⊃' and other devices. He makes this and other shifts with a view to streamlining his scientific work, maximizing his algorithmic facility, and maximizing his understanding of what he is doing. He does not care how inadequate his logical notation is as a reflexion of the vernacular, as long as it can be made to serve all the particular needs for which he, in his scientific program, would have otherwise to depend on part of the vernacular.[47]

---

44. Ibid., p 214

45. Grice, H.P. "Logic and Conversation" in *The Logic of Grammar*, by Donald Davidson and Gilbert Harman, 1975, Dickenson Publishing Company. This presents Grice's theory of conversational implicature in which he argues that '⊃' among other connectives of M-logic are in accordance with the way 'if…then' is used in natural language. For a fuller discussion of Grice's treatment of '⊃' see Strawson, P.F., '"If" and "⊃"', in *Philosophical Grounds of Rationality: Intentions, Categories, Ends* [P GRICE], edited by R.E.Grandy and R. Warner, Oxford Clarendon Press, pp 229-42'.

46. P.F.Strawson, *Introduction to Logical Theory*, 1967 Methuen and Co., London (First published in 1952), p. 38.

47. Quine, W.V., "Mr. Strawson on Logical Theory", *MIND*, v. 62, Oct., 1953, reprinted in *Ways of Paradox*, Random House, 1966, pp 148.

Like Quine, A-logic holds that though the differences between the truth-functional conditional and ordinary uses of ‘if…then’ in science and in common sense are real, it is not the job of the formal logician to capture the exact meaning of the vernacular. This is an impossible task since there is no such single, exact meaning in natural language. Further, it agrees with Quine that the meaning logicians assign to ‘if…then’ should be guided in large part by the pragmatic goal of maximizing the understanding and algorithmic facility of scientific reasoning.

But A-logic differs on the following:

First, it has become very clear that the truth-functional conditional does not satisfactorily serve the needs of mathematics and of physics in particular. In view of the true paradoxes, it may even be said that it is not completely satisfactory for mathematics. In the physical sciences it has failed to explicate satisfactorily the confirmation of generalized conditionals or dispositional predicates or the “lawlike” generalizations of science; and it cannot explicate conditional probabilities as the probability of a truth-functional conditional. These failures alone are persuasive reason for seeking another conditional which can do those jobs.

Secondly, the concept of the formal logician’s job “…only as a means of getting on with physics, mathematics and the rest of science” is too narrow. No doubt there should be a logic of physics and a logic of mathematics specifically designed to meet the special concepts and procedures of those disciplines; such logics should be extensions of general formal logic. But Philosophy is not the handmaiden of either the physical sciences or mathematics. Its sub-field, Logic, must deal with rational thinking, logical deductions and derivations in all areas of human activity. Even if it were true that the truth-functional conditional, and truth-logic in general were adequate for mathematics and physics, the traditional and ordinary concept of Logic as a branch of Philosophy demands a broader scope than mathematics and physical science.

Finally, the concern about the gap between ordinary language and the operators and connectives of formal logic, though not determining, has a certain force. The inadequacy of the truth-functional if-then does not lie in its failure to capture some supposed true, fixed, ordinary meaning of ‘if…then’, but it does lie in a failure to define ‘if…then’ in a way which will do the jobs that ‘if…then’ does very well in common sense as well as science. It fails with respect to two different jobs: it gives incorrect tests of what makes a conditional true, and it fails to account for the jobs that can be done by contrary-to-fact, or subjunctive conditionals.

10.321 On the Quantity of Anomalously “True” TF-conditionals

The beginning logic student is rarely aware that M-logic’s theory of the TF-conditional entails that enormous numbers of conditional statements are true, though neither mathematicians, nor scientists, nor common sense, would ordinarily consider them true. There is nothing counter-intuitive or contrary to common sense in saying that $[\sim(\sim P \mathbin{\&} P \mathbin{\&} \sim Q)]$ can not be false, or that $[T\sim(\sim TP \mathbin{\&} TP \mathbin{\&} \sim TQ)]$ is second-level true in all cases. Both of these second-level judgments hold in A-logic and in M-logic. In both M-logic and in A-logic statements of these forms are synonymous with and truth-functionally equivalent to the expressions $((\sim P \mathbin{\&} P) \supset Q)$ and $(\sim Q \supset (P \vee \sim P))$, and to $((\sim TP \mathbin{\&} TP) \supset TQ)$ and $(\sim TQ \supset (TP \vee \sim TP))$. It is only when ‘$\supset$’ is read as ‘if…then’ that the deviations from ordinary usage appear. Again it is not the expressions themselves, but claims that they express forms of *conditional* statements which are never false or always true, that are out of line with the ordinary usage of “if…then” and with A-logic.

To illustrate the extent of the offending anomalies, suppose our field of reference is simply the domain of positive integers from 1 to 10: $\{1,2,3,4,5,6,7,8,9,10\}$. There are 1000 Ordered triples from this set: $\{\langle 1,1,1\rangle, \langle 1,1,2\rangle, \ldots\ldots \langle 10,10,10\rangle\}$. Let ‘$P\langle 1,2,3\rangle$’ stand for the predicate, “$\langle 1\rangle$ is the product of <2> times <3>", i.e., the predicate of "a = b X c". This predicate is true of just 27, or 2.7% of the 1000 ordered triples, namely of

<1,1,1>,	<2,1,2>,	<3,1,3>,	<4,1,4>,	<5,1,5>,	<6,1,6>,	<7,1,7>,	<8,1,8>,	<9,1,9>,	<10,1,10>,
	<2,2,1>,	<3,3,1>,	<4,4,1>,	<5,5,1>,	<6,6,1>,	<7,7,1>,	<8,8,1>,	<9,9,1>,	<10,10,1>,
			<4,2,2>,		<6,2,3>,		<8,2,4>,	<9,3,3>,	<10,2,5>,
					<6,3,2>,		<8,4,2>,		<10,5,2>.

[P<1,2,3>] or "<a = b X c>" is thus false of 973, that is, 97.3%, of the 1,000 ordered triples.

Now, if P<a,b,c> is false, then ~P<a,b,c> will be analytically, mathematically true—a truth or theorem of mathematics. Given the Principle of Addition in M-logic, from any one of the false propositions, we can derive numerous TF-conditionals which must be called True, though most of them would be considered not true or false. For example, provided Q is any well-formed statement in the given universe of discourse, we get a "conditional" theorem of M-logic that no one would call true or valid from each one of the 973 false triples.

⊢ [(1 = 3 X 5) ⊃ Q]

<u>Proof</u>: 1) ⊢ ~((1 = 3 X 5) [Theorem of Math]
2) ⊢ (~P ⊃ (P ⊃ Q)) [Theorem of M-logic]
3) ⊢ (~(1 = 3 X 5) ⊃ ((1 = 3 X 5) ⊃ Q] [2),Instantiation]
4) ⊢ ((1 = 3 X 5) ⊃ Q] [1), 3), TF-MP]

Next, in M-logic, interpreting '⊃' as "if…then' we prove as a truth of arithmetic,

⊢ "If it is true that if 1 equals 3 times 5, then Q is true"

where Q can be any well-formed expression in the domain of discourse. Well, there are 1000 simple unegated well-formed expressions of the form, 'P<1,2,3>', and another thousand well-formed simple expressions of the form '~P<1,2,3>', in this universe of discourse. So here are 2,000 mathematical theorems of the form '⊢ (T(1 = 3 X 5) ⊃ TQ)' with Q replaced in turn by each of the 2,000 just mentioned. Since there are 973 false simple expressions of the same form, we have 973 X 2,000 = 1,946,000 conditional theorems of mathematics, almost all of which would ordinarily be rejected as false or untrue conditionals.[48] According to M-logic they are *all* true conditionals because the premiss is false *a priori*. Among them are cases which are doubly true according to M-logic, because the consequent is one of the 27 true statements.

This is just the starter. There are many more mutually distinct wffs compounded with pairs or n-tuples of the 2,000 elementary wffs which can be put for 'Q', boosting the number of non-sequiturs which are theorems of mathematics in this tiny domain to truly astronomical figures.

As an example, according to M-logic, the following is a theorem:

"If (10 = 1 X 9) then ((8 = 2 X 4) and either (7 X 1 = 9) or (6 X 3 = 8))"

---

48. In the 973 cases with the consequent the same as the antecedent, e.g., '((1= (3X5)) ⊃ (1=(5X3)))', or the 973 with the product commuted, e.g., '((1= (3X5)) ⊃ (1=(5X3)))', could be accepted as true or tautologous. There are a few other cases among the 1,946,000.

This is one of over 9 trillion well-formed conditional “truths” of mathematics formed by putting atomic sentences of the form [a = b X c] for the variables in [P ⊃ (Q&(RvS))]—with a mathematically false one for P. With C-conditionals or ordinary conditionals replacing TF-conditionals, none of these “conditionals” are true, and none are true by A-logic.

In A-logic, a few C-conditionals formed in this way are provably *unfalsifiable*, including 973 cases of the form [P ⇒ P] from [P Cont P]. (By “this way” we mean putting one of the 973 analytically false elementary propositions of the form P<1,2,3> as the antecedent of a conditional, and any other well-formed sentence in the universe of discourse as the consequent). These C-conditionals are not E-valid or true in A-logic because the antecedent is false. In addition, some of the ‘⇒’-for-‘⊃’ conditionals from the trillions of M-valid conditionals will be true in fact (e.g., ‘(2= (1X2)) ⇒ (9 = (3X3))’), and others will be true because derivable from valid general principles, (e.g., ‘(4= (2X2)) ⇒ (8 = (2X2X2))’) or will be *de dicto* implications of A-logic (when the antecedent re-occurs as a disjunct in the consequent). But these cases are very small in number compared to the number of *untrue* C-conditionals which are ‘⇒’-for-‘⊃’ analogues of the many anomalous ‘*true*’ TF-conditionals in this domain.

In a mathematical theory based on A-logic, there will be many significant validity-theorems with C-conditionals—e.g., laws of algebra such as,

$$\models \text{Valid}[(\forall x)(\forall y)(\forall z)(T((x \text{ X } y) = z) \Leftrightarrow T((y \text{ X } x) = z))]$$
$$\models \text{Valid}[(\forall x)(\forall y)(\forall z)(T((x \text{ X } y) = z) \Leftrightarrow T(z = (x \text{ X } y))]$$

These are A-valid, unfalsifiable, useful inferential biconditionals based on the meanings of ‘=’ and ‘the product of *a* and *b*’ in mathematics. Changing C-biconditionals to TF-biconditionals in these cases and prefixing ‘Taut’, we get a sub-class of tautologies in A-logic:

$$\text{Taut}[(\forall x)(\forall y)(\forall z)(T((x \text{ X } y) = z) \equiv T((y \text{ X } x) = z)) .$$
$$\text{Taut}[(\forall x)(\forall y)(\forall z)(T((x \text{ X } y) = z) \equiv T(z = (x \text{ X } y)).$$

A-logic will recognize these Tautologies, along with the trillions of other TF-conditionals which M-logic counts among its theorems. But it will not treat them as “valid” statement forms, because they are not C-biconditionals. It will assert their tautologousness as denials of inconsistencies but will not conflate tautologies with valid principles of inference.

The fact that students of M-logic do not dwell upon or spend time deriving the anomalous conclusions does not change the fact that they all follow from the principles of M-logic. They make ordinary sense only when we reduce them to conjunction or disjunction and denial, without ‘if’ or ‘then’ and employ some version of Addition. Surely the fact that A-logic avoids this kind of anomaly with respect to ‘if ...then’ statements should count in its favor.

10.322 The Anomaly of Self-Contradictory TF-conditionals

In addition to the frequently mentioned anomalies discussed in the previous section, there is a curious, seldom noticed anomaly of two conditionals which are ordinarily considered self-contradictory though their disjunction is logically true in M-logic.

In M-logic and in A-logic ∼P is equivalent to (P ⊃ ∼P) and P is equivalent to (∼P ⊃ P). With ‘⊃’ interpreted as ‘if...then’, these read in M-logic as: ‘If P is not true, then P is true’ is the same as ‘P is true’ and ‘If P is true, then it is not true that P’ is the same as ‘It is not true that P’. In symbols with T-operators: $\vdash$ [TP ≡ (∼TP ⊃ TP)] and $\vdash$ [∼TP ≡ (TP ⊃ ∼TO)].

In A-logic, these are synonymy-theorems:

|= [~TP Syn (TP ⊃ ~TP)]
Proof: 1) P Syn (P & P)
2) TP Syn (TP & TP)
3) ~TP Syn ~(TP & TP)
4) ~TP Syn ~(TP & ~~TP)
5) ~TP Syn (TP ⊃ ~TP)

|= [TP Syn [~TP ⊃ TP]
Proof: 1) TP Syn (TP v TP)
2) TP Syn (~~TP v TP)
3) TP Syn (~TP ⊃ TP)

For most users of ordinary language, "If P is true then P is not true," is a logical contradiction, as is "If P is not true, then P is true". On this account, P by itself, which is not logically inconsistent, can not mean the same as the latter and 'not-P' can not mean the same as the former. In addition, if both conditionals are inconsistent, the disjunction of the two should also be inconsistent. Thus "Either if P is true, then P is not true, or if P is not true then P is true", is should be inconsistent and thus never true.

A-logic agrees with ordinary language. By DR5-5a. [If P Cont Q, then Inc(P & ~Q)] both of these conditionals, and their disjunction, are inconsistent:

|= Inc[TP ⇒ ~TP]
Proof: 1) TP Cont TP
2) Inc[TP & ~TP] [1),DR5-5a]
3) Inc[TP ⇒ ~TP] [2),Df 'Inc ⇒ ']
4) Inc[~TP ⇒ TP] [2),Df 'Inc ⇒ ']

|= Inc[~TP ⇒ TP]
Proof: 1) ~TP Cont ~TP
2) Inc[~TP & ~~TP] [1),DR5-5a]
3) Inc[~TP & TP] [2),DN]

|= Inc[(TP ⇒ ~TP) v (~TP ⇒ TP)]
Proof: 1) Inc[TP ⇒ ~TP] [Theorem above]
2) Inc[~TP ⇒ TP] [Theorem above]
3) Inc[(TP ⇒ ~TP)v(~TP ⇒ TP] [1),2),Df 'Inc',Clause (iii)]

This makes the denials of these wffs tautologous in A-logic;

Inc[TP & ~TP]
.: Inc[TP ⇒ ~TP] Hence, Taut[~(TP ⇒ ~TP)]
.: Inc[~TP ⇒ TP] Hence, Taut[~(~TP ⇒ TP)]
.: Inc[(TP ⇒ ~TP) v (~TP ⇒ TP)] Hence, Taut[~((TP ⇒ ~TP) v (~TP ⇒ TP))]

and by analytic truth-tables, assertions of their non-truth are universal second-order logical truths:

```
|= [~T(TP ⇒ ~TP)]     |= [~T(~TP ⇒ TP)]     |= [~T((TP ⇒ ~TP) v (~TP ⇒ TP))]
    TF F0  0  TF0         TF  TF0  F  F0         TF  F0  0  TF0  0  TF0  F  F0
    TF TT  F  FTT         TF  FTT  0  TT         TF  TT  F  FTT  0  FTT  0  TT
    TF FF  0  TFF         TF  TFF  F  FF         TF  FF  0  TFF  0  TFF  F  FF
    ^                     ^                      ^
```

However, if '⊃' is interpreted as 'if…then', the disjunction of the two TF-conditionals not only is not inconsistent; it is a tautology and *logically true*:

TAUT[(~TP ⊃ TP) v (TP ⊃ ~TP))
Proof: 1) INC[TP & ~TP]
2) INC[(TP & TP) & ~TP]
3) INC[(TP & TP) & (~TP & ~TP)]

```
((~TP ⊃ TP) v (TP ⊃ ~TP))
  TF0 F  F0  T  F0  T TF0
  FTT T  TT  T  TT  F FTT
  TFF F  FF  T  FF  T TFF
```

4) INC[~(~(TP & TP) v ~(~TP & ~TP))]	Syn
5) INC[~((~TP v ~TP) v (TP v TP))]	<u>(TP v~TP)</u>
6) INC[~((~TP v ~TP) v (~~TP v TP))]	F0 T TF0
7) TAUT[(~TP v ~TP) v (~~TP v TP)]	TT T FTT
8) TAUT[(TP ⊃ ~TP) v (~TP ⊃ TP)]	FF T TFT

According to M-logic, this tautology is a theorem of logic, a logical truth, and a valid statement about the disjunction of two conditionals.

Thus, according to M-logic, with '⊃' being interpreted as "if…then",

"Either if P is true then P is not true, or, if P is not true then P is true"

is a valid, logically true theorem-schema. No matter what sentence is put for P, the result of substitution is true and logically true!

This is an anomaly—an apparent contradiction of common sense. The anomaly lies solely in interpreting '⊃' as "if…then". If '(P ⊃ Q)' is replaced by its synonym '(~P v Q)', there is no problem. Common sense, A-logic and M-logic agree that the following two expressions are equivalent and tautologous and that no instantiation of either one can ever be false.

[TAUT((~TP v ~TP) v (~~TP v TP)) Syn TAUT(TP ⊃ ~TP) v (~TP ⊃ TP))]

10.323 Contrary-to-fact and Subjunctive Conditionals

Nelson Goodman published an article 1947 entitled "The Problem of Counterfactual Conditionals". In it he wrote,

> What then, is the *problem* about counterfactual conditionals? Let us confine ourselves to those in which antecedent and consequent are inalterably false—as, for example, when I say of a piece of butter that was eaten yesterday, and that had never been heated,
>
> If that piece of butter had been heated to 150°F., it would have melted.
>
> Considered as a truth-functional compounds, all counterfactuals are of course true, since their antecedents are false. Hence
>
> If that piece of butter had been heated to 150° F., it would not have melted
>
> would also hold.Obviously something different is intended, and the problem is to define the circumstances under which a given counterfactual holds while the opposing conditional with the contradictory consequent fails to hold.[49]

Clearly the problem in this case is based on the fallacy of the false antecedent. In A-logic this version of the problem is solved simply by replacing TF-conditionals with C-conditionals in analytic truth-logic. If a given C-conditional (P ⇒ Q) is true, then the opposing conditional, (P ⇒ ~Q) is false. This is established in T8-18. [T(P ⇒ Q) Syn F(P ⇒ ~Q)]. Thus T8-18 defines "the circumstances under which the given counter-factual holds while the opposing conditional with the contradictory consequent fails to hold," namely, if the C-conditional is true. Theorem T8-18 follows directly from

---

49. Nelson Goodman, *Fact, Fiction and Forecast*, 1955, Chapter 1, page 4. Chapter 1 is a reprint of "The Problem of Countefactual Conditionals", *Journal of Philosophy*, February 1947.

Axioms.8-01 and 8-02 with double negation (Ax.4-05). This solution applies to both contingent, non-analytic conditionals and to analytic conditionals.[50] It works for contingent unquantified conditionals because, when the antecedent is not true (whether false or neither-true-nor-false) the conditional as a whole is neither-true-nor-false. This eliminates the fallacy of the false antecedent and is in accord with ordinary usage. It is also in accord with the trivalent truth-table for [P ⇒ Q], and the principles of inference which are its rules:

				P	Q	(P ⇒ Q)
T8-831.	Valid	[(0P & 0Q) ⇒ 0(P ⇒ Q)]	(For ⇒ Row1)	0	0	0
Ti8-8 ⇒ R2.	Valid_I	[(TP & 0Q) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 2)	T	0	0
T8-832.	Valid	[(FP & 0Q) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 3)	F	0	0
Ti8-8 ⇒ R4.	Valid_I	[(0P & TQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 4)	0	T	0
T8-833.	Valid	[(TP & TQ) ⇒ T(P ⇒ Q)]	(For ⇒ Row 5)	T	T	T
Ti8-8 ⇒ R6.	Valid_I	[(FP & TQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 6)	F	T	0
Ti8-8 ⇒ R7.	Valid_I	[(0P & FQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 7)	0	F	0
T8-834.	Valid	[(TP & FQ) ⇒ F(P ⇒ Q)]	(For ⇒ Row 8)	T	F	F
Ti8-8 ⇒ R9.	Valid_I	[(FP & FQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 9)	F	F	0

The principles of the truth-table for ‘ ⇒ ’ also include the rule that if Q has the value 0, then [P ⇒ Q] has the value 0: Ti8-843 Valid_I [0Q ⇒ 0(P ⇒ Q)].

These rules makes it possible to interpreted C-conditionals as subjunctive. All valid inferential conditionals may be intepreted as subjunctive conditionals. For example, T8-814 Valid [F(P ⇒ Q) ⇒ FQ] may be read as saying that “If (If P then Q) *were* false, then Q *would be* false” is valid. Also, particular *de re* subjunctive conditionals can be derived from non-analytic E-valid quantified conditionals. For example, acceptance of “All butter which is heated to 150° F, melts” entails the particular conditional, “If it should be the case that this piece of butter is heated to 150°F, then this piece of butter would melt” is E-valid by a kind of universal instantiation.

10.33 Anomalies of Quantified TF-conditionals

The re-interpretation of Aristotle’s “universal affirmative” proposition “All Ps are Qs” as a quantified conditional, (∀x)(If Px then Qx), was a great advance. With nested quantifiers and polyadic predicates, it made possible the unprecedented, largely successful reduction of most mathematics to M-logic and set theory. However, in trying to extend the applications of M-logic to generalizations *de re* in natural science and common sense, the anomalies of unquantified or particular TF-conditionals re-appear as anomalies of quantified conditionals.

Quantified conditionals either lie in the scope of a disjunctive (“existential”) quantifier or a conjunctive (“universal”) quantifier, or in the scope of some sequence of both kinds. They respresent conjunctions and/or disjunctions of particular, unquantified conditionals. The following six sub-sections are all concerned with consequences of the fallacies of the false antecedent or true consequent, and of bivalence, on the truth-logic of quantified conditionals. In each we compare quantified TF-conditionals in M-logic with quantified C-conditionals in A-logic.

---

50. By applications to Logically Valid conditionals it adds an enormous number of theorems in analytic truth-logic which have no analogues in M-logic. [If ⊨ Valid(P ⇒ Q) then ⊨ ~T(P ⇒ ~Q)] is a valid rule of inference which yields theorems starting with ⊨ ~T[P ⇒ ~P], ⊨ ~T[(P&Q) ⇒ ~P], ⊨ ~T[(P&Q) ⇒ ~Q], etc.

10.331 Quantified Conditionals, Bivalence and Trivalence
10.332 The "Paradox of Confirmation"—Raven Paradox
10.333 The Fallacy of the Quantified False Antecedent
10.334 Dispositional Predicates/Operational Definitions
10.335 Natural Laws and Empirical Generalizations
10.336 Causal Statements
10.337 Statistical frequencies and Conditional Probability

In each of these the problem with the TF-conditional will be seen to arise from one or more of the three attributes listed at the beginning of Section 10.3 and is eliminated or solved by using the C-conditional in analytic truth-logic.

10.331 Quantified Conditionals, Bivalence and Trivalence

Ordinarily "All ravens are black" is understood to talk only about ravens—not about all things in the universe of discourse. When "(R<1> ⇒ B<1>)" is applied to something other than a raven, it is neither true nor false; for it certainly does not make a true statement about a raven, and to say it is false entails that there is a raven which is not black (in accord with Ax.8-02). The if-clause, "if <1> is a raven", limits the intended referents of the consequent to ravens. To preserve the uniqueness of the subject matter and the specificity of our findings, we need a conditional which can be neither true nor false when the antecedent doesn't apply.

The Law of Bivalence in M-logic—that every indicative conditional sentence must be either true or false and none can be neither true nor false—stands in the way of this important aspect of the ordinary use of quantified "if…then" statements. The principle of Trivalence, $\models$ T[0P v TP v FP], which is proven as T7-71, removes this obstacle.

'(∀x) Px' can be read as "no matter what x may be, x is P". Thus '(∀x)(If Px then Qx)' may be construed as saying that the predicate, 'If P<1> then Q<1>', can be applied to any x, no matter what x may be. To say the conditional predicate can be applied to anything does not entail that it must be either true or false of each thing. It may be true of some, false of others, and neither true nor false of things that do not satisfy the antecedent. So far, on this reading there is no reference to a fixed set of entities of which 'If P<1> then Q<1>' is true or false as a whole.

However, if we also have a field of reference in mind, or we think of '(∀x)(If Px then Qx)' as an expression intended to apply to the extension of some as yet undefined set of entities, then the reading "For *all* x, if x is P then x is Q" or "For *all* members of the intended domain, the predicate, 'if P<1> then Q<1>'applies", seems to be about the set as a whole. It will be false if 'if P<1> then Q<1>' is false of one or more members, true if and only if it applies to all members, and neither true nor false of the set if the antecedent is not satisfied by any member. (For example "All ravens are black" is neither true nor false of the class of black widow spiders.) In this sense it is about the truth, falsity, or neither, of a whole conjunction of instantiations. M-logic treats '(∀x)(If Px then Qx)' extensionally in this manner, except that it does not allow any instantiation to be neither true nor false.

We will retain the non-extensional reading, "no matter what x may be" for '(∀x)' and incorporate the clearly extensional reading "for all x" within it. Thus for any domain of reference, D, the extensional reading is "No matter what x may be if x is a member of D, then if x is P then x is Q". For example, if the domain of reference has four individuals $\{a_1, a_2, a_3, a_4\}$,

'(∀x) (If Px then Qx)' means
'(If Pa_1 then Qa_1 & If Pa_2 then Qa_2 & If Pa_3 then Qa_3 & If Pa_4 then Qa_4)'

Relative to any domain, the generalization '(∀x)(If Px then Qx)' means simply a conjunction of all instantiations of the predicate '(If P<1>then Q<1>)' on the individuals in that domain of reference.

In dealing with quantified conditional predicates, we want to distinguish cases which are *not false* both 1) from those that are true and 2) from those that are both not true and not false.

Let '(∀x)(If Rx then Bx)' abbreviate "All Ravens are Black". We are inclined to say that '(∀x)(If Rx then Bx)' is *true*. But '(∀x)(If Rx then Bx)' is true if and only if each of the four instantiated conditionals are *true* and with a C-conditional '(∀x)(Rx ⇒ Bx)' is clearly *not true* in most cases because not all instantiations are true. Suppose we are talking about a domain of four entities {a_1, a_2, a_3, a_4}; a_1 and a_2 are black ravens, a_3 is a black piece of coal and a_4 is a yellow rose. '(R<1> ⇒ B<1>)' is a POS predicate and what makes it true of an entity a_i is that (Ra_i & Ba_i) is true. This is in accord with Ax. 8-01 and with what Quine said is the ordinary view. But then the result is not true if either conjunct is not true:

'T(∀x) (Rx ⇒ Bx)' Syn '(∀x)T(Rx ⇒ Bx)'
Syn 'T(Ra_1 ⇒ Ba_1) & T(Ra_2 ⇒ Ba_2) & T(Ra_3 ⇒ Ba_3) & T(Ra_4 ⇒ Ba_4)'
[by ⇒ truth-tables.Ch 8] T(T T T) T T(T T T) F F(F 0 T) **F** F(F 0 F)
^

So, unless our field of reference has only ravens as subject, 'T(∀x)(Rx ⇒ Bx)' is false even if all ravens in the field are black.

However, in the same case it is true that '(∀x)(Rx ⇒ Bx)' is *not-false*, for none of its instantiations is false, and at least one must be false to make the whole false.

'~F(∀x)(Rx ⇒ Bx)' Syn '(∀x) ~ F(Rx ⇒ Bx)
Syn '~F(Ra_1 ⇒ Ba_1) & ~F(Ra_2 ⇒ Ba_2) & ~F(Ra_3 ⇒ Ba_3) & ~F(Ra_4 ⇒ Ba_4))'
[by ⇒ Truth-tables.Ch 8] TF(T T T) T TF(T T T) T **TF**(F **0** T) **T** **TF**(F **0** F))
^

In general it would be *false* to say that "no matter what x may be, (P<1> ⇒ Q<1>) is true of x" is *neither true nor false* if the conditional predicate, (P<1> ⇒ Q<1>), has true instantiations and no false ones. This is so even though the predicate does not apply to some entities and thus is neither true nor false of them. It is precisely correct to say, in the case above, that no matter what x may be '(Rx ⇒ Bx)' is *not false*; and this is more accurate than saying "no matter what x may be '(Rx ⇒ Bx)' is neither true nor false".

Also, with the same truth-assignments the quantified expression '(∃x)(Rx ⇒ Bx)' is true.

'T(∃x)(Rx ⇒ Bx)' Syn '(∃x)T(Rx ⇒ Bx)'
Syn 'T(Ra_1 ⇒ Ba_1) v T(Ra_2 ⇒ Ba_2) v T(Ra_3 ⇒ Ba_3) v T(Ra_4 ⇒ Ba_4)'
[by ⇒ Truth-tables.Ch 8] T(T T T) T T(T T T) T F(F 0 T) **T** F(F 0 F)
^

Thus if our field of reference is the natural world, past present and future, one can 1) know *a priori* and *de dicto* that "If <1> is R then <1> is B" is *can not be true of everything* (except in domains such that all members are R), and 2) *know* it is *true of a great many things* (all things that are R that I know about) and at the same time 3) *believe* that it is *not false of anything* (the statement is contingent, since it could still be proved false). This assertion with respect to a contingent conditional is what we have called *empirical validity*. It is, I think, close to what most intelligent people would say if asked to analyze what they literally mean and accept in assertions like "It is *true* that all ravens are black". Most generalized "truth- claims" of common sense and scientific knowledge, can be understood to be E-valid propositions about different classes and subclasses of entities which are taken to exist in the actual world.

In M-logic the quantified TF-conditional is treated as if '(∀x)(If Rx then Bx)' is an assertion about all things, rather than being an assertion about things denoted by its antecedent only. And it gets away with it, for if the conditional is a TF-conditional and principle of bivalence is adopted, then '(∀x)(Rx ⊃ Bx)' is either true or false (exclusively) of the set of all things in any domain of reference. As in A-logic, it is false only if there is at least one object a_i such that Ra_1 is true and Ba_i is false. But if it is *true*, the conjunctively quantified TF-conditional is true of *all* things in the domain.

'T(∀x)(Rx ⊃ Bx)' Syn '(∀x)T(Rx ⊃ Bx)'
Syn 'T(Ra_1 ⊃ Ba_1) & T(Ra_2 ⊃ Ba_2) & T(Ra_3 ⊃ Ba_3) & T(Ra_4 ⊃ Ba_4)'
[by ⊃ truth-tables.Ch 7] T(T T T) T T(T T T) T T(F T T) **T** T(F T F)
^

What makes '(∀x)(Rx ⊃ Bx)' true of *all* things in this case is that (R<1> & ~B<1>) is **not** true in any instantiation; it does not apply to either a_1 or a_2 or a_3 or a_4. The predicate '(R<1> ⊃ B<1>)' is a NEG (negative) predicate. It is Syn to ~(R<1> & ~B<1>) and says that (R<1> & ~B<1>) does **not** apply to either a_1 or a_2 or a_3 or a_4. The trouble is that in M-logic too many other things also imply that '(R<1> ⊃ B<1>)' is true (due to the fallacies of the false consequent and the true antecedent). Thus it fails to be *specifically about* the unique kind of entities ordinarily taken to be the subject of the conditioned predicate. The C-conditional does not have this problem.

10.332 The "Paradox of Confirmation"—Raven Paradox

A major problem for the TF-conditional is exemplified in the "Raven Paradox".[51] In both A-logic and M-logic "All Ravens are Black" is first analyzed as "No matter what x may be, if x is a raven, then x is black", and partially symbolized as '(∀x)(if Rx then Bx)'. The question is, what kinds of observable facts *confirm* the general statement that all ravens are black? A-logic and M-logic give different answers and the difference is due to the difference between TF-conditionals and C-conditionals with respect to whether a false antecedent or a true consequent make the conditional true.

If the TF-conditional is used, "All ravens are black" is symbolized as '(∀x)(Rx ⊃ Bx)', and this is TF-equivalent to, and analytically synonymous with, '(∀x)(~Rx v Bx)', which says that every object, a_i, in the universe of discourse, is either not a raven or is black.[52] Thus confirming instances of this generalization consist of any individual, a_i, of which one of the following three predications is true:

(Ra_i & Ba_i)—"a_i is both a raven and is black" M-implies (~Ra_i v Ba_i)
(~Ra_i & Ba_i)—"a_i is not a raven and is black" M-implies (~Ra_i v Ba_i)
(~Ra_i & ~Ba_i)—"a_i is not a raven and is not black" M-implies (~Ra_i v ~Ba_i).

Further, by the Law of Addition, either Ba_i, or ~Ra_i, implies (~Ra_i v Ba_i), which is synonymous with (Ra_i ⊃ Ba_i). In M-logic the Law of Addition is treated as a valid rule of ***de re*** inference. Thus if a statement that a_i is not a raven is true, then by the Law of Addition, then 'If a_i is a raven then a_i is black' is true, and this in turn supports the claim that the general statement, "All ravens are black", is true. Also if the statement that a_i is black is true (even where a_i is a black object that is not a raven), this similarly

51. See Carl G. Hempel, *Aspects of Scientific Explanation*, Free Press, 1965 pp 3-53, or his "Studies in the Logic of Confirmation", reprinted from MIND (1945) in B.A.Brody and R.E.Grandy, *Readings in the Philosophy of Science*, 1989, espectially Sections 3-5.

52. We assume here that the investigator defines 'Raven' in some way that does not include being black in the definition—i.e., "white raven" is not a contradiction in terms. Thus the question is to be decided by fact, not the meanings of terms.

establishes that ‘if a_i is a raven, then a_i is black’ is true, and thus supports the truth of the general statement. As Hempel, a defender of M-logic, says,

> …we have to recognize as confirming for [‘All ravens are black’] any object which is neither black nor a raven. Consequently, any red pencil, any green leaf, any yellow cow, etc., becomes confirming evidence for the hypothesis that all ravens are black.[53]

Ordinarily, such data would be considered irrelevant, and inferences from them to instances of the generalized conditional would be viewed as non-sequiturs. But as Hempel makes clear, this is what follows by M-logic when we use the TF-conditional in generalizations.

Why does he defend this? His major argument is that to deny it violates Principles of Transposition in M-logic: for example, since $(\sim Q \& \sim P)$ makes $(\sim Q \supset \sim P)$ true, and $(P \supset Q)$ is equivalent to $(\sim Q \supset \sim P)$, therefore $(\sim Q \& \sim P)$ must make $(P \supset Q)$ true. In A-logic, Transposition holds only in restricted forms for C-conditionals, so this objection fails to hold if the “if…then” is a C-conditional (see Section 8.31).

A-logic provides a more satisfactory theory of confirmation. First, there is only one kind of confirming instance instead of three. “It is true that a_i is a raven and a_i is black”, $T(Ra_i \& Ba_i)$, entails $T(Ra_i \Rightarrow Ba_i)$, by Ax. 8-01 & U-SUB and $T(Ra_i \& Ba_i)$, is the only kind of confirming evidence for the general statement $T(\forall x)(Rx \Rightarrow Bx)$. $T(\sim Ra_i \& Ba_i)$ and $T(\sim Ra_i \& \sim Ba_i)$, which confirm $T(\forall x)(Rx \supset Bx)$, do not confirm $T(\forall x)(Rx \Rightarrow Bx)$. In A-logic

If either “a_i is <u>not</u> a raven and a_i is black” is true, $(\sim Ra_i \& Ba_i)$
or “a_i is <u>not</u> a raven and a_i is <u>not</u> black” is true, $(\sim Ra_i \& \sim Ba_i)$
then “If a_i is a raven then a_i is black” is neither true nor false, hence not-true.

Thus both, though they confirm ‘$(\forall x)(Rx \supset Bx)$’, are irrelevant to the confirmation of $(\forall x)(Rx \Rightarrow Bx)$ because the antecedent is not satisfied.

Neither ‘T(a_i is not a raven)’ nor ‘T(a_i is black)’ imply, much less entail. ‘T(If a_i is a raven then a_i is black)’ with a C-conditional. In general, $T\sim Pa_i$ does not entail or imply $T(Pa_i \Rightarrow Qa_i)$ and TQa_i does not entail or imply $T(Pa_i \Rightarrow Qa_i)$. Although $T\sim Pa_i$ and TQa_i each A-implies $T(Pa_i \supset Qa_i)$, i.e., $T(\sim Pa_i \vee Qa_i)$, this is not in general suitable for reaching *de re* conclusions, and in any case ‘$\supset$’ is not considered a genuine conditional in A-logic. Thus none of Hempel’s “paradoxical” statements about confirmation follow if the conditional in the quantified proposition is the C-conditional of A-logic. Neither a red pencil, or a green leaf, or a yellow cow, are in any way confirming evidence for, or are relevant in any way to the hypothesis that all ravens are black.

In A-logic, to say that $(\forall x)(TPx \Rightarrow TQx)$ is *true*, is synonymous with saying that P is true of everything and that Q is true of everything. To say the $(\forall x)(TPx \Rightarrow TQx)$ is *not false*, is much more perspicacious. It says that there is no x such that P is true of x but Q is not. If the field of reference contains an object <u>e</u> which is not a raven, ‘$T(\forall x)(Rx \Rightarrow Bx)$’ is false even if all ravens in the field are black. For $\mathbf{T(\forall x)}(Rx \Rightarrow Bx)$ Syn $\mathbf{(\forall x)T}(Rx \Rightarrow Bx)$ Syn $(\forall x)T(Rx \& Bx)$ and $T(R\underline{e} \& B\underline{e})$ is false, since $R\underline{e}$ is false. In a domain of one thousand entities with just one raven which is black, $\sim F(\forall x)(TRx \Rightarrow TBx)$, i.e., $\sim(\exists x)(TRx \& \sim TBx)$ is true and $T(\forall x)(TRx \Rightarrow TBx)$, i.e., $((\forall x)TRx \& (\forall x)TBx)$ (“Everything is a raven and everything is black”) is as false as can be. This is why we move to the *non-falsity* of $(\forall x)(Rx \Rightarrow Bx)$ instead of its *truth*. The difference between $T(\forall x)(Px \Rightarrow Qx)$ and $\sim F(\forall x)(TPx \Rightarrow Qx)$ is based on Ax. 8-01 $T(P \Rightarrow Q)$ Syn $T(P\&Q)$ and Ax. 8-02.

53. Hempel’s article in Brody and Grandy, Op. Cit., p. 264.

Finally, finding a raven which is not black disconfirms (∀x)(Rx ⇒ Bx) just as (∀x)(Rx ⊃ Bx) is disconfirmed in M-logic. From T(Ra_i & ~Ba_i), by Ti8-793, Valid_I [F(Pa_i) ⇒ F(∀x)Px], we prove that (∀x)(Rx ⇒ Bx) is *false*:

|= [T(Ra_i & ~ Ba_i) ⇒ F(∀x)(Rx ⇒ Bx)]

<u>Proof</u>: 1) T(Ra_i & ~ Ba_i)	[Premiss]
2) F(Ra_i ⇒ Ba_i)	[1),Ax.8-02, SynSUB]
3) Valid_I (F(Ra_i ⇒ Ba_i) ⇒ F(∀x)(Rx ⇒ Bx))	[Ti8-793,U-SUB]
4) F(∀x)(Rx ⇒ Bx)	[2),3),MP]
5) [T(Ra_i & ~ Ba_i) ⇒ F(∀x)(Rx ⇒ Bx)]	[1), to 5), Cond.Pr]

and the conditional (∀x)(TRx ⇒ TBx) is proven false and invalid by T(TRa_i & ~ TBa_i);

<u>Proof</u>: 1) T(TRa_i & ~ TBa_i)	[Premiss]
2) F(TRa_i ⇒ TBa_i)	[1),Ax.8-02,SynSUB]
3) Valid_I (F(TRa_i ⇒ TBa_i) ⇒ F(∀x)(TRx ⇒ TBx)	[Ti8-793,U-SUB]
4) F(∀x)(TRx ⇒ TBx)	[2),3),MP]
5) [T(TRa_i & ~ TBa_i) ⇒ F(∀x)(TRx ⇒ TBx)]	[1), to 5), Cond.Pr]

To see how A logic's theory of confirmation or disconfirmation of the E-validity of "All ravens are black" is worked out in symbols, imagine again two domains of four objects: In case (i) <u>a</u> and <u>b</u> are ravens, <u>c</u> is a black piece of coal, and <u>d</u> is a yellow rose. In case (ii), <u>a</u> and <u>b</u> are black ravens and <u>c</u> is a black piece of coal and <u>e</u> is a raven which is white.

In case (i) we find that all ravens in the domain of reference {a,b,c,d} are black (there are just two ravens) and it is not true that anything in the domain is a raven and not black;

<u>Case (i) Observed data</u>: for "All ravens are black" in field of reference {a,b,c,d}.

T[(T(Ra&Ba) v T(Rb&Bb) v T(Rc&Bc) v T(Rd&Bd)) & (~T(Ra& ~Ba) & ~T(Rb&~Bb) & ~T(Rc &~Bc) & ~T(Rd & ~Bd)
T T TTT TT TTT TF FFT **T**F FFF <u>**T**</u> TF T F FT T TF T F FT T TF F F FT **T** TF F F TF

Since T(Ra&Ba) Syn T(Ra ⇒ Ba) etc. by Ax 8-01 and T(Ra& ~ Ra) Syn F(Ra ⇒ Ba) etc. by Ax.8- 02, it follows logically that the observed data, **confirms** the E-validity of "All ravens are black" in this domain. This is double-checked by application of the truth-table rules for '⇒'.

<u>Confirmation of the E-validity of the Quantified Conditonal (∀x)(Rx ⇒Bx):</u>

[(-----------------(∃x)T(Rx ⇒ Bx)------------------) & (-------------------(∀x) ~ F(Rx ⇒ Bx)-------------------)]
T[(T(Ra ⇒Ba)v T(Rb ⇒Bb)v T(Rc ⇒Bc)v T(Rd⇒Bd)) & (~F(Ra⇒Ba)& ~F(Rb⇒Bb)& ~F(Rc ⇒Bc)& ~F(Rd ⇒Bd))]
T T T T T **T**T T T T **T**F F 0 T **T**F F 0 T <u>**T**</u> TF T T T T TF T T T T TF F 0 T **T** TF F 0 T

In case (ii) we imagine a domain of reference, {a,b,c,e} in which **e** is a third raven which is white, disconfirming the E-validity of "All ravens are black". Assigning the same values, except for sentences about <u>e</u>,

<u>Case (ii) Observed data</u>: for "All ravens are black" in field of reference {a,b,c,**e**}.

T[(T(Ra&Ba) v T(Rb&Bb) v T(Rc&Bc) v **T(Re&Be)**) & (~T(Ra& ~Ba) & ~T(Rb& ~ Bb) & ~T(Rc& ~Bc) & **~T(Re & ~Be)**
T T TTT TT TTT TF FFT TF TFF <u>**F**</u> TF T F FT T TF T F FT T TF F F FT **F** FT T T T F

Since T(Ra & Ba) Syn T(Ra ⇒ Ba) etc., by Ax 8-01 and T(Ra & ~Ra) Syn F(Ra ⇒ Ba) etc. by Ax.8-02, the observed data, for this domain of four entities, **disconfirms** the E-validity of "All ravens are black". This is double-checked by application of the truth-table rules for '⇒'.

Confirmation of the E-validity of the Quantified Conditonal (∀x)(Rx ⇒ Bx):

[(-----------------(∃x)T(Rx ⇒ Bx)------------------) & (-------------------(∀x) ~ F(Rx ⇒ Bx)-------------------)]
T[(T(Ra ⇒Ba)v T(Rb ⇒Bb)v T(Rc ⇒Bc)v **T(Re ⇒Be)**) & (~F(Ra ⇒Ba)& ~F(Rb ⇒Bb)& ~F(Rc ⇒Bc)& **~F(Re ⇒Be))]**
T T T T T **T** T T T T **T** F F 0 T **T** F T F F **T** TF T T T T TF T T T T TF F 0 T **F** FT T F F

As long as all of the ravens in the domain are black (Case (i)) both conjuncts hold. When a raven in the domain is not black (Case (ii)) the second conjunct is false; if all ravens in the domain were not black, both conjuncts would be false. If there were no ravens, the first conjunct would be false though the second would remain as an unproven speculative generalization. I.e., since the first conjunct fails, there is no evidence to support it positively, but no evidence to refute it (the second conjunct holds).

In A-logic generalizations of this sort are also construable as inferential conditionals. These are quantifications of 'If it is true that <1> is a raven, then it is true that <1> is black', symbolized by (∀x)(**T**(Rx) ⇒ **T**(Bx), rather than as assertions that a conditional predicate holds of all entities, symbolized by (∀x)**T**(Rx ⇒ Bx). The important claim for inferential generalizations vis-a-vis the actual world, is the claim that they are not false, rather than the claim that they are true. For this claim is shown false only when we have the antecedent true and the consequent false (a raven which is not black). It is not refuted by irrelevant instantiations for which the antecedent is not true, and it is confirmed by those cases in which both antecedent and consequent are found true.

I believe that when we say we accept an empirical generalization like "All ravens are black" as true, we really mean 'If <1> is a raven then <1> is black' is true of some ravens and is not false of any ravens i.e., it is E-valid. The latter clause—that it is not false of any ravens—requires an inductive leap which can not be justified as a logical entailment or implication from past observations by either M-logic or A-logic. This is the old problem of induction. If the predicate is contingent, nothing in either M-logic or A-logic prevents future ravens from being non-black.

10.333 The Fallacy of the Quantified False Antecedent

Since the TF-conditional is true by implication whenever the antecedent is false, it follows M-logically from "Nothing is P" that "All P's are Q's" is true no matter what Q<1> may be. In symbolic language from (∀x)~Px, it follows logically that (∀x)(Px ⊃ Q) where Q is any statement or combination of statements whatever.[54] When '⊃' is read as 'if…then' this inference schema may be called The Fallacy of the Quantified False Antecedent.

Interpreting '⊃' as 'if…then' and thus '(∀x)(Px ⊃ Qx)' as "All P's are Q's", this fallacy leads to a kind of absurdity. As Strawson pointed out,"It would be a kind of logical absurdity to say 'All John's children are asleep; but John has no children'."[55] Even more absurdly, this suggests that for each person A who has no children, all of the mutually inconsistent statements below follow M-logically from the fact that A has no children:

---

54. Strawson held that (∀x)(If Rx then Bx) "presupposes" the existence of the terms referred to in the antecedent. This is in line with our claim that 'T(∀x)(If Rx then Bx)' entails the existence in the field of reference of entities referred to in antecedent and consequent.

55. P.F.Strawson, *Introduction to Logical Theory*, Menthuen, 1952, p.175

1) At some times all of A's children are asleep and at some times none are.
2) All of A's children are asleep at all times.
3) All of A's children never sleep at any times.
4) Two of A's children never sleep and seven of them sleep at all times.
   etc., (or with any other predicate replacing $Q<1>$ in the consequent)

It should be clear that this "fallacy" or anomaly, is sole the result of interpreting '$\supset$' as 'if…then'. It does not appear in analytic truth-logic even though many variations of the same principle appear among its implication-theorems including:

Ti7-787. Valid_I $[(\forall x)TQx, .\!: (\forall x)(TPx \supset TQx)]$	Ti8-787. Valid_I $[(\forall x)TQx \Rightarrow (\forall x)(TPx \supset TQx)]$
Ti7-788. Valid_I $[(\forall x)TQx, .\!: (\forall x)(FPx \supset TQx)]$	Ti8-788. Valid_I $[(\forall x)TQx \Rightarrow (\forall x)(FPx \supset TQx)]$
Ti7-789. Valid_I $[(\forall x){\sim}TPx, .\!: (\forall x)(TPx \supset FQx)]$	Ti8-789. Valid_I $[(\forall x){\sim}TPx \Rightarrow (\forall x)(TPx \supset FQx]$
Ti7-790. Valid_I $[(\forall x)FPx, .\!: (\forall x)(TPx \supset FQx)]$	Ti8-790. Valid_I $[(\forall x)FPx \Rightarrow (\forall x)(TPx \supset FQx)]$
Ti7-791. Valid_I $[(\forall x)TPx, .\!: (\forall x)({\sim}TPx \supset TQx)]$	Ti8-791. Valid_I $[(\forall x)TPx \Rightarrow (\forall x)({\sim}TPx \supset TQx)]$

But in A-logic, '$(P \supset Q)$' is not interpreted as 'if P then Q'; rather, it abbreviates 'not P or Q' or 'not-(both P and not-Q)'. Interpreted this way these Validity-theorems make sense as *de dicto* principles, though they must be used with care to avoid conflating *de dicto* statements with *de re* statements. Without the interpretation '$(P \supset Q)$' as 'if P then Q' there is no anomaly.

In A-logic there are no analogues of these principles for C-conditionals. Rather, for any predicate, P, if $(\forall x){\sim}TPx$ is true, then it follows by A-logic that any statement of the form $(\forall x)(TPx \Rightarrow Q)$ is neither true nor false, i.e., $(\forall x){\sim}TPx$ A-implies $0(\forall x)T(Px \Rightarrow Q)$. Since the premiss says that the antecedent of the conditional in the consequent is never true, the conditional can never be true or false. Of course $[(\forall x)(T(x \text{ is a child of } A) \Rightarrow Q)]$ *may* be logically or empirically valid even if A has no children. For example, "Any child of A's would be either a son or a daughter" may be an E-valid conditional in the domain of human relationships, whether or not A has children. But in A-logic this can not *follow from* the fact that A has no children, as in M-logic; rather, it could follow by universal instantiation from the assumption that "All human children are either male or female," is E-valid.

Just how $[(\forall x)({\sim}TPx \vee Q)]$ follows logically from $[(\forall x){\sim}TPx]$ may be clarified by pointing out how the Law of Addition applies to each conjunct of the Boolean expansion of the Premiss:

1) $(\forall x)\ {\sim}TPx$			[Premiss]
2) $(\forall x)\ {\sim}TPx$	Syn	$({\sim}TPa_1\ \&{\sim}TPa_2\ \&{\sim}TPa_3\ \&{\sim}TPa_4\ \&\ldots)$	[Df'$(\forall x)$',U-SUB]
3)		$({\sim}TPa_1\ \&{\sim}TPa_2\ \&{\sim}TPa_3\ \&{\sim}TPa_4\ \&\ldots)$	[1),2),SynSUB]
4)		$(({\sim}TPa_1 \vee Q)\ \&\ ({\sim}TPa_2 \vee Q)\ \&\ ({\sim}TPa_3 \vee Q)\ \&\ ({\sim}TPa_4 \vee Q)\ \&\ldots)$	[3), Addition,n times]
5) $(\forall x)({\sim}TPx \vee Q)$	Syn	$(({\sim}TPa_1 \vee Q)\ \&\ ({\sim}TPa_2 \vee Q)\ \&\ ({\sim}TPa_3 \vee Q)\ \&\ ({\sim}TPa_4 \vee Q)\ \&\ldots)$	[Df'$(\forall x)$',U-SUB]
6) $(\forall x)({\sim}TPx \vee Q)$			[5), 4),SynSUB]

Similarly, the conclusion follows from the assertion that Q is true, whether Q is a predicate applied to each member of the domain in turn, or is a fixed unchanging statement. This is called the "Fallacy of the Quantified True Consequent".

The Fallacies of the Quantified False Antecedent play an important role in generating different kinds of anomalies traceable to the TF-conditional. It was implicitly present in the pre-suppositions of Goodman's examples of the "Problem of Counterfactual Conditionals". The supposition that the piece of butter that had never been heated to 150°F must be expressed in the form like, "For all times, t, it is not true that a is heated to 150°F at time t", or, in symbols, "$(\forall t)\ {\sim}T(H<a,t>)$". From this, according to M-logic, "If a is heated to 150°F at any time then… anything can happen”. Similarly for Carnap’s supposition below about a match stick that had never been put in water. We will see that this fallacy is also central to the problem of dispositional predicates, and operational definitions.

10.334 Dispositional Predicates / Operational Definitions

There are a great many adjectives, often ending in ‘..able’, or ‘ible’, etc., which describe potential properties of objects or states of affairs—properties which are not immediately displayed but would be displayed if certain conditions were to obtain. Such predicates, like ‘<1>is soluble in water’, ‘<1>is inflammable’, or ‘<1>boils if heated to 212 degrees F’, are called “dispositional predicates”, and are used to indicate the tendency or disposition of an object, or state of affairs to take on certain properties or relationships under specified conditions.

In common sense and science such predicates are very important, and are best explicated as conditionals. To say a certain vase is fragile means it will break *if* it is jarred; to say a certain plant is poisonous means *if* someone eats it, they will get sick. Every element in the periodic table is defined in terms of several dispositional properties—e.g., that under standard conditionals it will change from liquid to gas *if* heated to a certain temperature, and from a liquid to a solid *if* subjected to a different temperature. *If* an element is subjected to certain chemical reactions involving hydrogen, each of its atoms are will combine with such and such a number of hygrogen atoms (valence). And *if* the temperature and pressure are the standard ones, then the density of the element will be such and such.Definitions of physical entities in terms of testible conditionals of this sort are sometimes called operational definitions. M-logic’s problems with dispositional predicates, have been thought by some to discredit operational definitions.

Note that often an entity’s various dispositions are grounded in mutually exclusive conditions, so it would be inconsistent to conceive of both dispositions being realized at the same time. For example silver melts at if it is heated to 960° C and boils if heated to 1,955° C. It would be a contradiction for the same bit of silver to be at both 960° and 1,955° at the same time. Thus if one state is realized, the other can not be.

It is natural to think such properties should be definable using conditional predicates as follows:

‘<1>is soluble in water’ syn_{df} ‘at any time, t, if <1>is placed in water at t, then <1> dissolves at t’

In 1936, however, Carnap wrote that such disposition-terms can not be defined in this way because it would lead to results that contradicted the intended meaning of the dispositional term.[56] Carnap’s argument assumed that the conditional used in the definition must be a TF-conditional, and it showed that indeed with such a conditional a contradiction could result. His example was essentially as follows:

Letting ‘$Q_3<1>$’ abbreviate ‘<1> is soluble in water’, ‘$Q_1<1,t>$’ abbreviate ‘<1>is placed in water at time t’ and $Q_2<1,t>$’ abbreviate ‘<1> dissolves at time t’, the definition above becomes,

(D:) $Q_3<1>$ Syn_{df} $(\forall t)(Q_1<1,t> \supset Q_2<1,t>)$

i.e., $Q_3<1>$ for “<1>is soluble in water”

is synonymous by definition with

$(\forall t)\ (Q_1<1,t> \supset Q_2<1,t>)$ for “At any time t, if <1>is placed in water at t, then <1> dissolves at t”

But, said Carnap, suppose we had a match, c, which was burnt yesterday and had never been in water. Then the statement 1) ‘$(\forall t) \sim Q_1<c,t>$’ i.e., “c was never placed in water’ would be true, and this,

---

56. Rudolf Carnap, “Testability and Meaning”, *Philosophy of Science*, 3, 1936, Section II, 4, Definitions.

according to M-logic, implies 2) $(\forall t)(Q_1<c,t> \supset Q_2<c,t>)$ i.e., "at any time t if c is placed in water at t, then c dissolves at t" which according to the definition (D) means the same as

3) $Q_3 <c>$, i.e., 'The match c is soluble in water'

which conflicts with what we intended with respect to the meaning of "soluble in water".[57] Chains of definitions would multiply such problems.

The source of Carnap's problem is the fallacy of the quantified false antecedent. Simply stated this is that '$(\forall x)\sim Q_3x$' implies '$(\forall x)(Q_3x \supset Q_4)$' in M-logic, where Q_4 can be any well-formed expression whatever. How this works can be seen by applying the Law of Addition to each conjunct in the Boolean expansion of the premiss in any domain, e.g. in a domain of 3,

4) $(\forall x)\sim Q_3x$	SYN ($\sim Q_3a$	& $\sim Q_3b$	& $\sim Q_3c$)		[Premiss]
5) $(\forall x)(\sim Q_3x \vee Q_4)$	SYN (($\sim Q_3a \vee Q_4$)	& ($\sim Q_3b \vee Q_4$)	& ($\sim Q_3c \vee Q_4$))		[4), Addition, *n* times]
6) $(\forall x)(Q_3x \supset Q_4)$	SYN (($Q_3a \supset Q_4$)	& ($Q_3b \supset Q_4$)	& ($Q_3c \supset Q_4$))		[5),T4-31,SynSUB]

By the Rule of Addition, 'From $\sim P$ infer $(\sim P v Q)$' (or, synonymously, 'From $\sim P$ infer $(P \supset Q)$'), each conjunct of 4) implies the conjunct below it in 5), which is synonymous with 6). The general principle which yields this sort of result is one of the unstated consequences of M-logic: if P is never the case, then if anything is P then it has any and every property whatever.

Regardless of any rationale that may be offered in defense of the Quantified False Antecedent it would be useful for science and common sense to have a different conditional which can define dispositional predicates without its unwanted counter-instances.

The C-conditional does this. As we saw in Section 10.332, if the subject term, a_i, replacing the '1' in '$(P<1> \Rightarrow Q<1>)$' is such that $[Pa_i]$ is false, then the C-conditional is neither true nor false; such cases are irrelevant to its truth or falsity since the consequent is only intended to apply in those cases in which the antecedent is taken to be true. Thus in analytic logic '$<1>$ is soluble in water' may be defined as common sense would define it:

(D':) $Q_3<1>$ syn_{df} $(\forall t)(Q_1<1,t> \Rightarrow Q_2<1,t>)$[58]

More precisely, in A-logic this could be defined as a conditional predicate,

$$Q_3<1> \ \text{syn}_{df} \ ((\text{Time}<2> \ \& \ Q_1<1,2>) \Rightarrow Q_2<1,2>),$$

from which by Instantiation and Generalization we could get

$$(Q_3c \Leftrightarrow (\forall x)((\text{Time}<x> \ \& \ Q_1<c,x>) \Rightarrow Q_2<c,x>),$$

---

57. Carnap proposed as an alternative to such "explicit definitions" a system of "reduction sentences" which put the definiens or its denial in the right-most consequent of the TF-conditional as follows: $(Q1 \supset (Q2 \supset Q3))$ and $(Q4 \supset (Q5 \supset \sim Q3))$. These are Syn to $\sim((Q1 \ \& \ Q2) \ \& \sim Q3)$ and $\sim(Q4 \ \& \ Q5) \ \& \ Q3))$. This avoided undesired results, but had to be hedged to avoid having inconsistent antecedents (which would imply every consequent), and was incapable of the usual semantic reductions through chains of definitions.

58. Or, this can be defined as a conditional predicate, $Q_3<1>$ Syn_{df} $((T<2> \& Q_1<1,2>) \Rightarrow Q_1<1,2>$. By Instantiation and Generalization we can get 'Q_3c' Syn_{df} '$((\forall \hat{x})(Tx \ \& \ Q_1<c,x>) \Rightarrow Q_1<c,x>$, which is equivalent in meaning to $(\forall t)(Q_1<c,t> \Rightarrow Q_2<c,t>)$.

the right-hand side of which is equivalent in meaning to,

(2') $(\forall t)(Q_1\langle c,t\rangle \Rightarrow Q_2\langle c,t\rangle)$

which is the '$\Rightarrow$'-for-'$\supset$' analogue of (2) above. And (2') would be neither true nor false, since the antecedent $Q_1\langle c,t\rangle$ was never true (the match was never put in water). Thus it could not be inferred from (2') that c (the match-stick) is soluble in water. That conclusion would have to be drawn, if at all, by deduction from a synthetic generalization that every piece of wood is soluble in water, with the proved premiss that c is a piece of wood. In any case, Carnap's problem is eliminated.

Thus using C-conditionals, dispositional predicates can be defined without fear of Carnap's consequences, and derivations through chains of definitions by transitivity of 'syn' and '$\Rightarrow$' will be unproblematic provided the definitions are kept free of inconsistencies. The elements of physics and chemistry (oxygen, iron, etc..) can be defined as a cluster of alternative dispositional properties, each one depending on relevant conditions. Further, with C-conditionals classification schemes by genus and differentia and hierarchies of possible or dispositional properties defining different kinds of entities can be established in biology, physics or any sciences, and ***de re*** logical deductions are thereby facilitated.

10.335 Natural Laws and Empirical Generalizations

Truth claims and validity claims can be asserted with respect to many different fields of reference. A particular human language, the set of positive integers, mathematics in general, the field of Greek mythology, theological writings, are all fields of which some statements are true and others false, and some modes of inference are valid and others not. But one field of particular interest, because of its amazingly reliable and useful findings, is field of reference of the natural sciences. The natural sciences are concerned with truths and principles of inference which are useable for prediction, retrodiction and control of events and states of affairs in the actual world. Thus much effort has been given to provide a logical analysis of statements which express laws of nature and the role of conditionals in those statements. M-logic has had problems in this effort, which analytic truth-logic does not have.

To assert of some general statement is a *law of nature* or *natural law*, entails the assertions that it

(i) is expressible as a quantified conditional, "For all x, if x is P, then x is Q",
(ii) is a contingent statement, i.e., not logically unfalsifiable, tautologous, or logically true,
(iii) is true in one and all *known* case in the actual word to which its antecedent condition applies, and
(iv) has no false instantiations in the actual world, past, present and future, whether known or not.

The first two entailed assertions are about logical characteristics of the statement. The last two refer to a special field of reference, the actual world. The concept of the actual world is the concept of a field of reference which is a conjunction of facts and/or events which are what they are regardless of whether any humans, are aware of them. If any state of affairs or event belongs to the actual world, it must exist within a particular time period, and within a unique specifiable, region of space. The concept of a position in Space and Time is conceived as being *absolute* in the actual world—that is, it is what, where and when it is, and nothing else, regardless of what we may think. No actual particular physical object can occupy two disjoint places at the same time. However, every description an individual human or a group of humans can give of the position of any event or object in Space and Time is *relative*; relative to the local spatial arrangements, relative motions, measuring devices, calendars and clocks, etc., of those particular humans.

Events or facts of the actual world are assumed to consist of entities which have attributes and/or which stand in relations to one another. States of affairs and events can be simple or complex, depending upon how they are viewed. Having viewed something as a state of affairs, or an event, we analyze it as a complex whole made up of simpler parts, or we may view it simply as a whole belonging to a larger complex. Whether there are some particular natural objects which are absolutely simple (absolute atoms) is an open question.

It is the fourth entailed assertion which distinguishes the concept of a *natural law* from that of a mere *empirical generalization*. To conceive of a general statement as being a *natural law* entails conceiving it as not being false at any time past, present, or future, and being true at some times and places which are unaccessed to date, or even inaccessible, by humans, as well as those in known human experience.

An *empirical generalization*, in contrast, is a generalization which is asserted to hold for a finite number of entities which have been or are experienced by human agents. To assert that a general statement is a valid *empirical generalization*, is to point out that (i), (ii) and (iii) are true so far as we know of some finite domain, without asserting that it holds of all future and un-exmined cases, i.e., applies to an unlimited, unexperienced domain. An empirical generalization may be proved conclusively E-valid by examination of all its subject-matter. Its projection upon unexamined subjects may be believed or assumed to be E-valid, for pragmatic reasons.

For example, "Everybody in my current elementary logic class knows the basic laws of algebra" could be an empirical generalization about a known, finite class of individuals. Suppose this were proven conclusively by investigation of each individual in my class at some point in time. I may "project" this result on all elementary logic classes in my institution over a certain period of years. This is a guess or an assertion that the same statement holds in a larger domain of individuals; whether I make such a guess will probably be influenced by other considerations such as admissions requirements at my institution. Such a projections can and are used in planning the content of future courses, and in analyzing past performances. This is a pragmatic device, often proved to be useful, but it is not the same as proclaiming a *law of nature*.

To assert that something is a *natural law of the actual world*. is to say that the inferential conditional in its matrix, is not false at any time or place in absolute space-time, and that it is true under certain conditions at some times and places within of space time. Also, what humans call natural laws are restricted to things that could theoretically be falsified on the basis of sensory experience.

Goodman pointed out that the statement "All coins in my pocket on VE day were silver" is not a candidate for being a law of nature even though it may be proved by observed facts and can be expressed as a universal quantified conditional "(∀x)(x was a coin in one of my pockets on VE-day, then x was silver)". It was, indeed, a very particular empirical generalization, but in this case it would be foolish to project it to other times. Goodman's reason for saying it is not a natural law is that the instantiated conditional "if ***a*** were a coin in my pocket on VE-day, then ***a*** would be silver" has false instances, e.g., if ***a*** is a penny. Obviously "(∀t)(∀x)(t is a time & x is a coin in my pocket at t, then x is silver)" is quickly proven not to be E-valid. But the more cogent reason is that the frame of reference specified in the antecedent is a very local and limited region of Space-Time: the day is VE-day (August 19, 1946), the spatial location is 'Goodman's pockets" which were located in his clothes at certain very small specific geographical locations on Earth during that day. To be a law of nature, such as "Sugar dissolves in water" the quantifier must range over all *times* or occasions that certain conditions are met.

To assert that something is a natural law does not make it one. Since the concept of a law of nature is of a contingent universal truth, anything believed to be a law of nature could, without inconsistency, turn out to be false and thus not be a law of nature. Until the end of the 18th century, "all mammals are live-birthed", hence "No mammals lay eggs" $(\forall x)(Mx \Rightarrow \sim LEx)$ was taken to be a law of nature. This was a good candidate for a *Law of Nature* since the subject term 'mammal' is not restricted to any time or place of application (unlike 'the coins in my pocket on VE-Day"). Further, the predicate of the quantification had been found true and never found false by millions of people in millions of cases. Then duck-billed platypusses were discovered (in the late 18th century). They laid eggs, had bills like ducks and webbed feet, but also had mammary glands and the other defining characteristics of mammals. Thus what was believed to be a law of nature, was proved not to be one.

There is no harm in viewing the same general statement as a law of nature and at the same time as an empirical generalization. In drawing inferences for prediction, retrodiction or control in unexplored situations, we must take it as a law of nature if we want the conclusion to follow logically. If we are assessing it critically for reliability, we take it to be an empirical generalization and look for counter-examples which may require us to modify or revise it.

To say either that a statement is a true *empirical generalization* or that it expresses a *law of nature* entails that it has not had any false instantiations as well as the assertion that it is true in some actual cases. In short, it entails it is E-valid and expressible in the form, '$(T(\exists x)(Px \Rightarrow Qx)\ \&\ {\sim}F(\forall x)\ (Px \Rightarrow Qx))$'.

In some cases, if the field of reference for an empirical generalization is denoted by the antecedent, '$T(\forall x)(Px \Rightarrow Qx))$' may be justified by the evidence instead of its weaker entailment '$(T(\exists x)(Px \Rightarrow Qx)\ \&\ {\sim}F(\forall x)(Px \Rightarrow Qx))$'. For example, if my field of reference is simply the books in my library, "All books in my library have less than 5,000 pages" is simply true or false. In symbols, what is asserted is '$T(\forall x)(Bx \Rightarrow Lx))$' where '$B<1>$' is '$<1>$is a book in my library' and '$L<1>$' is '$<1>$ has less than 5000 pages'. Obviously, enormous numbers of generalized conditionals can be formed in this way, and observations can prove them true or false. But this type of generalization can be true only because the antecedent is always true in that field of reference. E.g., if $\{a_1,a_2,a_3, \ldots,a_n\}$ are all the *n* books in my library, then

$$(T(Ba_1 \Rightarrow La_1)\ \&\ T(Ba_2 \Rightarrow La_2)\ \&\ T(Ba_3 \Rightarrow La_3)\ \&\ \ldots\ \&T(Ba_n \Rightarrow La_n)\ \text{Syn}\ T(\forall x)(Bx \Rightarrow Lx))$$

T T T T   T T T T T   T T T T T   T   TT T T T   T T

and if some were false (some book has 5,000 pages) the generalization would be false:

$$(T(Ba_1 \Rightarrow La_1)\ \&\ T(Ba_2 \Rightarrow La_2)\ \&\ T(Ba_3 \Rightarrow La_3)\ \&\ \ldots\ \&T(Ba_n \Rightarrow La_n)\ \text{Syn}\ T(\forall x)(Bx \Rightarrow Lx))$$

T T T T   F T T F F   F T T T T   F   FT T T T   F

In most cases the field of reference is broader than the antecedent of the generalized conditional, and in these cases even empirical generalizations must be formulated with the weaker '$(T(\exists x)(Px \Rightarrow Qx)\ \&\ {\sim}F(\forall x)(Px \Rightarrow Qx))$'. This is because there will be one or more cases in which the antecedent is not true, making the instantiated conditional neither true nor false in those cases. Thus the best we can do is say that it is not false, and add that it is true of some cases. But statements claimed to be *natural laws* must always use the weaker form, for they refer to cases in which it is not known whether the antecedent is satisfied or not.

Both natural laws and empirical generalizations are *disconfirmed* if any instantiation is found false. As long as no false instantiations have be found each new case in which it is found true *confirms* it.

The problems with reducing *natural law* statements to quantified TF-conditionals of M-logic are those mentioned in preceding sections, with more to be mentioned in sections to come. Most of these have been documented by other philosophers and logicians. In concieving, rightly or wrongly, that statements like "All ravens are black", or "Sugar dissolves in water", or "No mammals lay eggs"are possible laws of nature, we simply do not take their meaning to have the form 'for all x (either not-Px or Qx)' or 'it is not the case that for some x (Px and not-Qx)', which they must have if they are interpreted as having the form '$(\forall x)(Px \supset Qx)$'. For, if this is what they mean many unwanted consequences follow. The natural laws that we current accept, would be "confirmed" by irrelevant findings of false antecedents or true consequents. Laws that we currently accept as dispositional statements, will not be expressible as dispositional statements. True statements that certain kinds of things never happen, would imply by the quantified false antecedent, that if these things happen then all imaginable kinds of things result. The list of natural laws will be loaded with non-sensical generalizations. Good natural laws would not be sustainable by contrary-to-fact or subjunctive conditionals, because M-logic does not have an adequate account of these kinds of conditionals. Every natural law would be construed as true of all things in the universe, instead of being about limited natural kinds of entities. Etc., etc.

Specific *Laws of Nature* and specific *Empirical Generalizations* are formulated and established in different branches of science. The logical analysis of specific laws usually involves an analysis of the terms and definitions used in that special branch of science. The meanings of terms, and criteria for determining how they apply to empirical data, are often unique to the science involved. Scientific laws of nature are formulated and discovered by using strict definitions, careful observations of data, and reasoning which stays on track. Our working hypothesis is that analytic truth-logic with C-conditionals provides conceptual tools for each case while avoiding the *non sequiturs* and unwanted consequences which follow from hewing rigorously to M-logic and its TF-conditionals, and makes possible solutions that are more in accord with common sense and science. The hypothesis must be tested in many particular special cases. The discussion of causal statements and probability statements which follow may start the investigation towards more special problems.

10.336 Causal Statements

The problem in using TF-conditionals to try to account for inferences from data to causal statements is that when '$\supset$' is interpreted as "if…then" in M-logic, the data M-implies too much; the fallacies of the false antecedent and the true consequent yield too many opposing conditionals from the data.

This can be illustrated by a simple example. The first light switch inside my front door can be pressed up or down. I assert and believe that "turning the switch up *causes* the entry-way light to go on" and nobody acquainted with my house disputes this. If asked to explain how I know this, I answer,

1) Every time I know about, *if* the switch is turned up, the entry light turns on.
2) Everytime I know about, *if* the switch is not turned up, the entry light does not turn on.
3) At no time, *if* the switch is turned up, does the light not turn on.
4) At no time, *if* the switch is not turned up, does the light turn on.

If pressed, I would have to say that these statements hold under "normal conditions"—i.e., not counting occasions when the power fails or a light bulb burns out, etc. Ignoring these kinds of qualifications for the moment (though they must be taken into account for a full logic of causal statements), the four statements above are my basic reasons for claiming a causal connection.

Each of the statements 1) to 4) is a general statement. The first two are positive statements, based on much positive observational data. Each datum is expressible in the forms,

1') At t_1 the switch is up *and* the light is on.
or 2') At t_2 the switch is not up *and* the light is not on.

True statements of these forms support my assertion of a causal connection. The second pair of statements, 3) and 4) are negative statements, describing what didn't happen (in "normal times"). If, under normal circumstances, I should observe

5') At t_3 the switch is up *and* the light is not on.
or 6') At t_4 the switch is not up *and* the light is on.

these would falsify 3) and 4) (according to both A-logic and M-logic) and my causal statement would be found false and in need of revision.

The generalized statements 1) to 4) are all expressible as quantified conditionals. The question is: How do we get, logically, from the truth of a conjunction of particulars like 1') and 2'), to the quantified conditional statements 1) and 2) which are the grounds of my causal statement?

In the case of A-logic, the answer is simple. By Ax.8-01the truth of each datum of the form 1') or 2') *entails* the truth of just one C-conditional of the form,

1") *If* at t_1 the switch is up *then* the light is on.
or 2") *If* at t_2 the switch is not up *then* the light is not on.

The conjunction of many true conditionals like these supports the E-validity of the causal claim *provided* there are no data with the form or 5') or 6') which would contradict 3) and 4).

They support [T((∃t)(Up(sw,t) ⇒ On(li,t)) & (∃t)(~ Up(sw,t) ⇒ ~ On(li,t))]. The absence of any known cases (under normal conditions) of 5') and 6') being true supports (without proving) the clauses (∀t) ~ F((Up(sw,t) ⇒ On(li,t_1)) and (∀t) ~ F(~ Up(sw,t_1) ⇒ ~ On(li,t_1)). These quantifications of C-conditionals make this portion of my grounds for the causal assertion E-valid.

The basic difficulty with using TF-conditional in the logic of causal relations is that in M-logic the data M-implies too many TF-conditionals. With the principle of Addition, and the fallacies of false antecedent and true consequent, the same data which supports the C-conditionals 1), 2) under A-logic, also supports by M-logic the TF-conditionals which "oppose" 2) and 1). In A-logic, the datum 1') entails just one C-conditional: *If* the switch is UP at t_1 *then* the light is ON at t_1. The same datum, M-implies three TF-conditionals in M-logic, by falsity of the antecedent or the truth of the consequent:

T(Up(sw,t_1) ⇒ On(li,t_1))
T(~Up(sw,t_1) ⇒ ~On(li,t_1))

<u>In A-logic</u>: Valid[T(Up(sw,t_1) & On(li,t_1)) .: 1) T(Up(sw,t_1) ⇒ On(li,t_1))] [By Ax.8-01,Syn SUB]
<u>In M-logic</u>: Valid[T(Up(sw,t_1) & On(li,t_1)) .: 1) T(Up(sw,t_1) ⊃ On(li,t_1))]
and Valid[T(") .: **6) T(~ Up(sw,t_1) ⊃ On(li,t_1))]** [Fallacy of T Conseq]
and Valid[T(") .: 2) T(~ Up(sw,t_1) ⊃ ~ On(li,t_1))] [Fallacy of F Antec]

Similarly with datum 2'): In A-logic, the datum 2') entails just one C-conditional— *If* the switch is not Up t_1 *then* the light is not On at t_1. In M-logic it implies three TF-conditionals by falsity of the antecedent or the truth of the consequent:

<u>In A-logic</u>:Valid[T(~Up(sw,t_1)&~On(li,t_1)) .: 2) T(~ Up(sw,t_1) ⇒ ~ On(li,t_1))] [Ax.8-01,Syn SUB]
<u>In M-logic</u>: Valid[T(~Up(sw,t_1)&~On(li,t_1)).: 2) T(~ Up(sw,t_1) ⊃ ~ On(li,t_1))]
and Valid[T(").: **5) T(Up(sw,t_1) ⊃ ~ On(li,t_1))]** [Fallacy of T Conseq]
and Valid[T(").: 1) T(Up(sw,t_1) ⊃ ~~ On(li,t_1))] [Fallacy of F Antec]

The second of each set of three TF-conditionals (in bold) are conditionals which "oppose" 2) and 1) respectively. Since every datum implies all three TF-conditionals, no matter how much data is gathered each pair of data which supports the quantified TF-conditionals 1) and 2),

1) At any time, *if* the switch is turned up, the entry light turns on.
2) At any time, *if* the switch is not turned up, the entry light *does not* turn on.

and will also support the quantified TF-conditionals which are "opposed" to 1) and 2), namely,

5) At any time, *if* the switch is turned up, then the entry light *does not* turn on.
6) At any time, *if* the switch is not turned up, the entry light *does* turn on.

In M-logic 5) and 6) are neither contradictories nor contraries of 1) and 2), for (P ⊃ Q) and (P ⊃ ~ Q) can be true together. But this is another of the anomalies of the TF-conditional. For ordinarily they are said to be opposites, and 5) and 6) certainly seem to be saying the opposite of what 1) and 2) say.[59] The point is that for causal statements, interpreting the data by using TF-conditionals which follow according to M-logic seems to undermine the ordinary understanding of what a causal statement means. For 5) says the effect Q did not occur at t_2, if the cause P occurred at t_2, while 6) says the effect Q did occur at t_1 if the cause P did not occur at t_1. This effectively negates the claim that c̲ causes e̲. For it follows from 5) and 6) that there are times when the the effect does not occur if the cause occurs and there are times when the effect occurs if the cause does not occur. This contradicts the implicit claim in a causal statement, that (under "normal" conditions) the effect never occurs without the cause, and never fails to occur when the cause occurs.

By contrast in A-logic with C-conditionals the data in 1') and 2') entail 1) and 2), but do not entail 3), 4), 5) or 6), and A-logic avoids these "anomalies of truth-functional causation".

These anomalies are due solely to interpreting '(P ⊃ Q)' as "If P then Q". If we replace '(P ⊃ Q)' by its synonym '(~ P v Q)', the fallacies of the true consequent and the false antecedent no longer have meaning. Instead the first and third cases are simply cases of implication by Addition, while the second case, that was called a fallacy of the true consequent when '⊃' was read as 'if…then', is an entailment of A-logic based on T1-38. [(P&Q) CONT (PvQ)]. This entailment has no logical connection with the logic of C-conditionals.

(P⊃ Q)	(P ⊃~Q)
T T T	T F FT
F **T** T	F **T** FT
T F F	T T TF
F **T** F	F **T** TF

In M-logic: Valid[T(Up(sw,t_1) &On(li,t_1)) .: T(~ Up(sw,t_1) v On(li,t_1))]
and Valid[T(") .: **T(Up(sw,t_1) v On(li,t_1))]** [Fallacy of T Conseq?]
and Valid[T(") .: T(Up(sw,t_1) v ~ On(li,t_1))] [Fallacy of F Antec?]

In M-logic: Valid[T(~Up(sw,t_1) &~On(li,t_1)) .: T(Up(sw,t_1) v ~ On(li,t_1))]
and Valid[T(") .: **T(~ Up(sw,t_1) v ~ On(li,t_1))]** [Fallacy of T Conseq?]
and Valid[T(") .: T(~ Up(sw,t_1) v On(li,t_1))] [Fallacy of F Antec?]

The "fallacies" here are so-called only because when the conclusion of the inference is read as "if…then" the inference would ordinarily be said to be a *non sequitur*—one which, belongs among those anomalies originally mis-called "paradoxes of material implication".

Other aspects of the problem of explicating causal statements with TF-conditionals, and the way these problems are solved using C-conditionals in A-logic, were laid out or suggested in Section 9.34.

59. I use the term 'opposing conditionals' as Nelson Goodman did "The Problem of Counterfactual Conditionals", where he wrotethat "… the problem is to define the circumstances under which a given counterfactual holds while the *opposing* conditional with the contradictory consequent fails to hold". *See p578*.

10.337 Statistical Frequencies and Conditional Probability

The "Problem of Conditional Probability" begins in traditional probability theory, where the expression '$Pr(Q|P)$' is read as 'The probability of Q, given P', and this is said to represent a "conditional probability". The conditional probability of Q given P, is said to be equal to the probability of both P and Q divided by the probability of P. I.e., $Pr(Q|P) = Pr(P\&Q)/Pr(P)$.

The question arises, should not a "conditional probability" be simply the probability of a conditional? Why can we not say that the probability of "Q is true, given that P is true", is simply the probability of "Q is true if P is true", i.e., the probability of 'If P is true, then Q is true'? Why is '$Pr(Q|P)$' not synonymous with 'Pr(If P then Q)'?

At first it might be thought that the TF-conditional could fill the bill; that Pr(if P then Q) $= Pr(P \supset Q) = Pr(Q|P)$. But this is quickly eliminated. For $(P \supset Q)$ is synonymous with $(\sim P \vee Q)$; therefore $Pr(P \supset Q)$ must equal $Pr(\sim P \vee Q)$. But, as we saw in Section 9.353, the probability of $(\sim P \vee Q)$ is not usually equal to $Pr(P\&Q)/Pr(P)$. If $Pr(P)=.40$ and $Pr(P\&Q)=.12$, then $Pr(P \supset Q)$, i.e., $Pr(\sim P \vee Q)=.72$ while $Pr(Q|P)=.30$, a discrepancy of .42. Again, if $Pr(Q)=.12$ and $Pr(Q\&S)=.02$, then $Pr(S|Q)=.17$ while $Pr(Q\&\sim S) = .10$ so that $Pr(Q \supset S), = .90$. Thus $Pr(\sim P \vee Q)$ diverges widely from $Pr(Q|P)$ in probability theory.[60] Only if $Pr(P) = 1.00$, or $Pr(P) = Pr(P\&Q)$, is $Pr(\sim P \vee Q)$ the same as $Pr(Q|P)$.

The problem is how to reconcile the the intuitively plausible view that conditional probability is the probability of a conditional statement with the discrepancies between conditional probabilities in probability theory and the probability of truth-functional conditionals in probability theory.

A-logic's answer is to abandon the notion that '$(P \supset Q)$' is a conditional statement form, and use the C-conditional, '$(P \Rightarrow Q)$' in its stead. With the C-conditional, and judicious use of the truth-operator, a solution is provided. The solution depends on distinquishing the probability that a C-conditional is true, i.e., $PrT(P \Rightarrow Q)$, from the probability of an inferential condititional asserting that the consequent is true if the antecedent is true, i.e., $Pr(TP \Rightarrow TQ)$.

Suppose we want to know the probability that $(P \Rightarrow Q)$ is true, in the field of reference. Obviously since $\models [T(P \Rightarrow Q) \text{ Syn } T(P\&Q)]$ by Ax. 8-01, $Pr(T(P \Rightarrow Q)) = Pr(T(P\&Q))$. In other words, the probability that (If P then Q) is *true*, is the same as the probability that (P and Q) is *true*. If x is the number of times (P&Q) is true in a reference class with *n* members, then x/n is the ratio of the times $(P \Rightarrow Q)$ is true of members of that reference class to the total number *n* of members of the reference class. Hence, $PrT(P \Rightarrow Q)$ is the probability that $(P \Rightarrow Q)$ is true = the ratio x/n (where $0 \leq x/n \leq 1$).

But the probability that Q is true, *if* P is true, i.e., $Pr(TP \Rightarrow TQ)$ is a different matter. Here we have the probability of an inferential conditional, not of the truth of a conditional: the probability of $(TP \Rightarrow TQ)$, not the probability of $T(P \Rightarrow Q)$. The expression '$(TP \Rightarrow TQ)$' says "If P is true, then Q is true" which means, "In those *cases in which* P is true, then Q is also true" or "In those *cases in which* P is true, (P&Q) is true". We want to know what is the proportion of those cases in which P is true, to those in which Q is *also* true (i.e., both P and Q are true). The answer is, obviously, the number of cases in which both (P & Q) is true divided by the number of cases in which P itself is true. The denominator of the probability ratio will be the number of instances in which P is true (not the number of members of the reference class) and the numerator will be the number of members in the reference class

60. See Section 9. 353 for more details, and TABLE 9-2 in Section 9.352

of which (P&Q), hence (P ⇒ Q), is true. This is precisely the conditional probability, Pr (Q|P) of standard probability theory.

$$\mathrm{Pr(TP \Rightarrow TQ)} = \mathrm{Pr(Q|P)} = \frac{\mathrm{PrT(P\&Q)}}{\mathrm{PrT(P)}}$$

For a more thorough discussion of the solution to this problem the reader is referred back to Section 9.35. The explanation above of the solution to the problem of conditional probability differs from the main argument-line of Section 9.35, where we talked of 1st- and 2nd-level frequency ratios, and used the expression "$\mathrm{rf_2T(P \Rightarrow Q)}$" instead of "Pr(TP ⇒ TQ)" which we used here. Though the terminology and notation differs, the basic solution to the problem is the same in both cases. The terminology and notation are adjustable to reconcile the differences.

The problem in equating conditional probability with the probability of a TF-conditional is not as easily related to the fallacies of the false antecedent and the true consequent as previous problems have been. The problem is first seen rather as that of mistakenly identifying an expression which is synonymous with "not-P or Q" and "not both P and not-Q)" with "if P then Q".

What is needed is an account of "If P then Q" which is true or false only in cases where the antecedent P satisfied. Thus it must not allow that the falsity of the antecedent P can make 'If P and Q' true. Only by separating out the cases in which P is true, and distinguishing within them whether (P&Q) is true or false in those cases, can the concept of conditional probability be explicated. The requirement that "for all x, if Px then Qx" can be true only if it is true in every instantiation stands in the way of separating the relevant cases (when P holds) from the irrelevant ones (when P is not the case). The law of trivalence in A-logic removes this obstacle, facilitating the analysis of both lawlike causal laws and probabilistic laws based on empirical data.

Appendices

Theorems from Chapters 1 to 8

Appendix I—Theorems and Rules, Chapters 1 & 2

Axioms:

T1-01 Ax.1-01. [P SYN (P&P)] [&-IDEM]
T1-02 Ax.1-02. [P SYN (PvP)] [v-IDEM]
T1-03 Ax.1-03. [(P&Q) SYN (Q&P)] [&-COMM]
T1-04 Ax.1-04. [(PvQ) SYN (QvP)] [v-COMM]
T1-05 Ax.1-05. [(P&(Q&R)) SYN ((P&Q)&R)] [&-ASSOC]
T1-06 Ax.1-06. [(Pv(QvR)) SYN ((PvQ)vR)] [v-ASSOC]
T1-07 Ax.1-07. [(Pv(Q&R)) SYN ((PvQ)&(PvR))] [v&-DIST-1]
T1-08 Ax.1-08. [(P&(QvR)) SYN- ((P&Q)v(P&R))] [&v-DIST-1]

The Rule of SynSUB: If $\models$ P and $\models$ (R Syn S) then $\models$ P (S//R)

Applied to SYN statements:

R1-1 [If ((P SYN Q) & (R SYN S)) then P SYN Q(**S**//**R**)]

Derived Rules

DR1-01. [If (P SYN Q) then (Q SYN P)]
hence DR1-01b [If ((P SYN Q) & (R SYN S)) then P SYN Q(**R**//**S**)]
DR1-01c [If ((P SYN Q) & (R SYN S)) then P (**S**//**R**) SYN Q]
DR1-01d [If ((P SYN Q) & (R SYN S)) then P(**R**//**S**) SYN Q].
DR1-02. [If (P SYN Q) and (Q SYN R), then (P SYN R)]

DR1-03—"&-ORD". Any complex wff containing a string of components connected only by conjunction signs is SYN to any other grouping or ordering of just those conjunctive components with or without conjuncts occurring more than once.

DR1-04—"v-ORD". Any complex wff containing a string of components connected only by disjunction signs is SYN to any other grouping or ordering of just those disjunctive components with or without disjuncts occurring more than once.

DR1-05. Generalized &v-DIST. If P is a conjunction with n conjuncts, and one or more of the conjuncts are disjunctions, P SYN Q if Q is a disjunction formed by disjoining all of the distinct conjunctions which are formed by taking just one disjunct from each conjunct of P.

DR1-06. Generalized v&-DIST. If P is a disjunction with n disjuncts and one or more of the disjuncts are conjunctions and Q is a conjunction formed by conjoining all of the distinct disjunctions which are formed by taking just one conjunct from each disjunct of P, then [P SYN Q].

SYN-theorems

Note: All even-numbered theorems from T1-12 to T1-35 have conjunctions on the right side; odd-numbered theorems are their duals with disjunctions on the right. CONT-theorems are derivable by Df 'CONT' from even-numbered theorems only.

T1-11. [P SYN P]
T1-12. [((P&Q) & (R&S)) SYN ((P&R) & (Q&S))] [From &-ASSOC]
T1-13. [((PvQ) v (RvS)) SYN ((PvR) v (QvS))] [From v-ASSOC]
T1-14. [(P & (Q&R)) SYN ((P&Q) & (P&R))] [From &-IDEM and &-ASSOC]
T1-15. [(P v (QvR)) SYN ((PvQ) v (PvR))] [From v-IDEM and v-ASSOC]
T1-16. [(Pv(P&Q)) SYN (P&(PvQ))]
T1-17. [(P&(PvQ)) SYN (Pv(P&Q))]
T1-18. [(P&(Q&(PvQ))) SYN (P&Q)] [&-Max]
T1-19. [(Pv(Qv(P&Q))) SYN (PvQ)] [v-Max]
T1-20. [(P&(Q&R)) SYN (P&(Q&(R&(Pv(QvR)))))]

T1-21. [(Pv(QvR)) SYN (Pv(Qv(Rv(P&(Q&R)))))]

T1-22. [(Pv(P&(Q&R))) SYN (P&((PvQ)&((PvR)&(Pv(QvR)))))]

T1-23. [(P&(Pv(QvR))) SYN (Pv((P&Q)v((P&R)v(P&(Q&R)))))]

T1-24. [(Pv(P&(Q&R))) SYN (P&(Pv(QvR)))]

T1-25. [(P&(Pv(QvR))) SYN (Pv(P&(Q&R)))]

T1-26. [(P&(PvQ)&(PvR)&(Pv(QvR))) SYN (P&(Pv(QvR)))] [&-Expansion]

T1-27. [(Pv(P&Q)v(P&R)v(P&(Q&R))) SYN (Pv(P&(Q&R)))] [v-Expansion]

T1-28. [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & (PvR))]

T1-29. [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v (P&R))]

T1-30. [((P&Q)&(RvS)) SYN ((P&Q) & ((P&R)v(Q&S)))]

T1-31. [((PvQ)v(R&S)) SYN ((PvQ) v ((PvR)&(QvS)))]

T1-32. [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) & (PvRv(Q&S))] [Praeclarum]

T1-33. [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) v (P&R&(QvS))]]

T1-34. [((P&Q)v(R&S)) SYN (((P&Q)v(R&S)) & (PvR) & (QvS))]

T1-35. [((PvQ)&(RvS)) SYN (((PvQ)&(RvS)) v (P&R) v (Q&S))]

CONT-theorems

Df 'Cont': [(P Cont Q) Syn_{df} (P Syn (P&Q))]

Derived Rule: "Df 'Cont'": If (P Syn (Q_1 & Q_2 &…Q_n) then (P Cont Q_i) ($1 \leq i \leq n$)

T1-36. [(P&Q) CONT P]

T1-37. [(P&Q) CONT Q]

T1-38. [(P&Q) CONT (PvQ)]

T1-39. [(P&(QvR)) CONT ((P&Q)vR)]

From T1-22. [(Pv(P&(Q&R))) SYN (P&((PvQ)&((PvR)&(Pv(QvR)))))] by Df 'Cont':

1 (2) (3) (4)

T1-22c(1). [(Pv(P&(Q&R))) CONT P] [T1-22.Df 'CONT']

T1-22c(2). [(Pv(P&(Q&R))) CONT (PvQ)] [T1-22.Df 'CONT']

T1-22c(3). [(Pv(P&(Q&R))) CONT (PvR)] [T1-22.Df 'CONT']

T1-22c(4). [(Pv(P&(Q&R))) CONT (Pv(QvR))] [T1-22.Df 'CONT']

T1-22c(1,2). [(Pv(P&(Q&R))) CONT (P&(PvQ))] [T1-22.Df 'CONT']

T1-22c(1,3). [(Pv(P&(Q&R))) CONT (P&(PvR))] [T1-22.Df 'CONT']

T1-22c(2,3). [(Pv(P&(Q&R))) CONT ((PvQ)&(PvR))] [T1-22.Df 'CONT']

T1-22c(2,4). [(Pv(P&(Q&R))) CONT ((PvQ)&(Pv(QvR)))] [T1-22.Df 'CONT']

T1-22c(3,4). [(Pv(P&(Q&R))) CONT ((PvR)&(Pv(QvR)))] [T1-22.Df 'CONT']

Derived Rules for getting to CONT-theorems

DR1-01. If [P SYN Q] then [Q SYN P]

DR1-02. If [P SYN Q] and [Q SYN R], then [P SYN R]

DR1-11. If [P SYN Q] then [P CONT Q]

DR1-12. If [P SYN Q] then [Q CONT P]

DR1-13. If [A CONT (B&C)] then [A CONT B & A CONT C)]

DR1-14. If [A SYN B] then [(A CONT B) & (B CONT A)]

DR1-15. If [A SYN (B&C)], then [A CONT B]

DR1-16. If [A SYN B] and [C CONT D], then [C CONT D(A//B)]

DR1-17. If [A CONT B], then [(CvA) CONT (CvB)]

DR1-18. If [A CONT B] and [B CONT A] then [A SYN B]

DR1-19. If [A CONT B] and [B CONT C], then [A CONT C]

DR1-20. If [A_1 CONT A_2] and [A_2 CONT A_3] and…and [A_{n-1} CONT A_n], then [A_1 CONTA_n].

DR1-21. If [A CONT B] then [(C&A) CONT (C&B)]

DR1-22. If [A CONT B] then [(C&A) CONT (CvB)]
DR1-23. If [A CONT C] and [B CONT C], then [(AvB) CONT C]
DR1-24. If [A CONT C] and [B CONT C], then [(A&B) CONT C]
DR1-25. If [A CONT B] and [A CONT C], then [A CONT (B&C)]
DR1-26. If [A CONT B] and [A CONT C], then [A CONT (BvC)]
DR1-27. If [A CONT C] and [B CONT D], then [(AvB) CONT (CvD)]
DR1-28. If [A CONT C] and [B CONT D], then [(A&B) CONT (C&D)]

In Chapter 2, members of {P_1,P_2,...,P_n} are treated as predicate letters rather that propositional or sentence letters. New wffs are added. The Rule of U-SUB (universal substitution) is augmented from R1-2, to R2-2 which permits introducing predicate schemata at all occurrences of a simpler predicate schema. The Rule of Instantiation, permitting the application of predicate schemata in a logical theorem to individual constants is added. The two rules, for unquantified wffs of Chapter 1, are as follows:

<u>R2-2</u>. If ⊨ R and (i) P_i<t_1,...,t_n> occurs in R,
and (ii) Q is an h-adic wff, where $h \geq n$,
and (iii) Q has occurrences of all numerals 1 to n,
then ⊨ [R(P_i<t_1,...,t_n>/Q) may be inferred U-SUB **Revised**
<u>R2-3</u>. If ⊨ [P<1>] then ⊨ [Pa] may be inferred INST **Added**

From every theorem of Chapter 1, an unlimited number of new theorems may be derived using these rules. For example, as a starter, from T1-16, the following seven new theorems are derived.

T1-16. [(P v (P & Q)) SYN (P & (P v Q))]
1) [(P<1> v (P<1> & Q)) SYN (P<1> & (P<1> v Q))] [T1-16,R2-2]
2) [(P<1> v (P<1> & R<1>)) SYN (P<1> & (P<1> v R<1>))] [2),R2-2]
3) [(P<1> v (P<1> & Ra)) SYN (P<1> & (P<1> v Ra))] [3),INST]
4) [(Pa v (Pa & Q)) SYN (Pa & (Pa v Q))] [1).INST]
5) [(Pa v (Pa & R<2>)) SYN (Pa & (Pa v R<2>))] [4),R2-2]
6) [(Q<2,b,1> v (Q<2,b,1> & Ra)) SYN (Q<2,b,1> & (Q<2,b,1> v Ra))] [3),R2-2]
7) [(P<1> v (P<1> & Q<2,b,a>)) SYN (P<1> & (P<1> v Q<2,b,a>))] [3),R2-2]

R2-2, The Rule of Instantiation (INST) will become the basis in Chapter 3 for theorems with quantificational wffs.

7) ⊨ [(P<1> v (P<1>& Q<2,b,a>)) SYN (P<1> & (P<1> v Q<2,b,a>))] [3),R2-2]
⊨ [(Pa v (Pa & Q<2,b,a>)) SYN (Pa & (Pa v Q<2,b,a>))] [7),INST]
⊨ [(Pb v (Pb & Q<2,b,a>)) SYN (Pb & (Pb v Q<2,b,a>))] [7),INST]
⊨ [(Pc v (Pc & Q<2,b,a>)) SYN (Pc & (Pc v Q<2,b,a>))] [7),INST]
⊨ [(Pd v (Pd & Q<2,b,a>)) SYN (Pd & (Pd v Q<2,b,a>))] [7),INST]

Hence, in Chapter 3 for the domain of four, {a,b,c,d}, we can derive:
.: ⊨ [(∀x)[(Px v (Px & Q<2,b,a>)) SYN (∀x)(Px & (Px v Q<2,b,a>))]

Appendix II—Chapter 3 Rules and Theorems

Definitions:

T3-11 $((\forall_n x)\ Px)$ $\mathbf{SYN_{df}}$ $(Pa_1\ \&\ P_2\ \&\ldots\&\ P_n)$ — <u>Added</u>

T3-12 $((\exists_n x)\ Px)$ $\mathbf{SYN_{df}}$ $(Pa_1\ v\ P_2\ v\ldots v\ P_n)$ — <u>Added</u>

Rules of Inference:

Alph Var. If P is an alphabetic variant of Q, then [P SYN Q]. — **<u>Added</u>**

R3-2. If ⊨ R <u>and</u> (i) $P_i<t_1,\ldots,t_n>$ occurs in R, — [U-SUB]
<u>and</u> (ii) Q is an h-adic wff, where $h \geq n$,
<u>and</u> (iii) Q has occurrences of all numerals 1 to n,
<u>and</u> (iv) no variable in Q occurs in R or S,
then ⊨ $R(P_i<t_1,\ldots,t_n>/Q)$ — **<u>Revised</u>**

DR3-3a. If ⊨ $P<1>$ then ⊨ (∀x)Px — "UG" [CG—Conjunctive Generalization]

DR3-3b. If ⊨ $P<1>$ then ⊨ (∃x)Px — "EG" [DG—Disjunctive Generalization]

DR3-3c. If [$P<1>$ SYN $Q<1>$], then [(∀x)Px SYN (∀x)Qx] — <u>Added</u>

DR3-3d. If [$P<1>$ SYN $Q<1>$], then [(∃x)Px SYN (∃x)Qx] — <u>Added</u>

DR3-3e. If [$P<1>$ CONT $Q<1>$], then [(∀x)Px CONT (∀x)Qx] — <u>Added</u>

DR3-3f. If [$P<1>$ CONT $Q<1>$], then [(∃x)Px CONT (∃x)Qx] — <u>Added</u>

Major SYN Theorems		**(Compare to Quine's Metatheorems in ML)**
T3-13. [(∀x)(Px & Qx) SYN ((∀x)Px & (∀x)Qx)]		*140
T3-14. [(∃x)(Px v Qx) SYN ((∃x)Px v (∃x)Qx)]		*141
T3-15. [(∀x)(∀y)Rxy SYN (∀y)(∀x)Rxy]		*119
T3-16. [(∃x)(∃y)Rxy SYN (∃y)(∃x)Rxy]		*138
T3-17. [(∀x)(P&Qx) SYN (P & (∀x)Qx)]	Rules of Passage	*157
T3-18. [(∃x)(PvQx) SYN (P v (∃x)Qx)]	Rules of Passage	*160
T3-19. [(∃x)(P&Qx) SYN (P & (∃x)Qx)]	Rules of Passage	*158
T3-20. [(∀x)(PvQx) SYN (P v (∀x)Qx)]	Rules of Passage	*159
T3-21. [(∀x)Px SYN ((∀x)Px & (∃x)Px)]		
T3-22. [(∃x)Px SYN ((∃x)Px v (∀x)Px)]		
T3-23. [(∃x)(Px & Qx) SYN ((∃x)(Px & Qx) & (∃x)Px)]		
T3-24. [(∀x)(Px v Qx) SYN ((∀x)(Px v Qx) v (∀x)Px)]		
T3-25. [((∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx))]		
T3-26. [((∃x)Px v (∀x)Qx) SYN ((∃x)Px v (∀x)(Px v Qx))]		
T3-27. [(∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]		
T3-28. [(∀y)(∃x)Rxy SYN ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]		
T3-29. [(∀x)(Px v Qx) SYN ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx))]		
T3-30. [(∃x)(Px & Qx) SYN ((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx))]		
T3-31. [(∀x)(∀y)Rxy SYN ((∀x)(∀y)Rxy & (∀x)Rxx)]		
T3-32. [(∃x)(∀y)Rxy SYN ((∃x)(∀y)Rxy & (∃x)Rxx)]		

3.27. Theorems of Quantificational Containment

T3-33. [(∀x)Px CONT Pa_i]	$(1 \geq i \geq n)$	
T3-34. [(∀x)(∀y)Rxy CONT (∀x)Rxx]		
T3-35. [(∃x)(∀y)Rxy CONT (∃x)Rxx]		
T3-36. [(∀x)Px CONT (∃x)Px]		*139
T3-37. [(∃y)(∀x)Rxy CONT (∀x)(∃y)Rxy]		*139

T3-38. [((∀x)Px v (∀x)Qx) CONT (∀x)(Px v Qx)] *143
T3-39. [(∀x)(Px v Qx) CONT ((∃x)Px v (∀x)Qx))] *144
T3-40. [(∀x)(Px v Qx) CONT ((∀x)Px v (∃x)Qx))] *145
T3-41. [((∀x)Px v (∃x)Qx) CONT (∃x)(Px v Qx)] *146
T3-42. [((∃x)Px v (∀x)Qx) CONT (∃x)(Px v Qx)] *147
T3-43. [(∀x)(Px & Qx) CONT ((∃x)Px & (∀x)Qx)] *152
T3-44. [(∀x)(Px & Qx) CONT ((∀x)Px & (∃x)Qx)] *153
T3-45. [((∀x)Px & (∃x)Qx) CONT (∃x)(Px & Qx)] *154
T3-46. [((∃x)Px & (∀x)Qx) CONT (∃x)(Px & Qx)] *155
T3-47. [(∃x)(Px & Qx) CONT ((∃x)Px & (∃x)Qx)] *156
T3-48. [(∀y) ((∀x)Pxy CONT Pyy)] Related to Quine's *103
T3-139 [(∀x)(Px & (Qx v Rx)) CONT (∀x)((Px & Qx) v Rx))]

[Note: asterisked numbers are numbers of metatheorems in Quine's *Mathematical Logic* (1982)] from which '⊃'-for-'CONT' analogues of theorems on the left can be drawn.]

Appendix III—Inductive Proofs of Selected Theorems, Ch. 3

1) T3-13. $[(\forall x)(Px \,\&\, Qx)$ SYN $((\forall x)Px \,\&\, (\forall x)Qx)]$ (Yields dual, T3-14)
2) T3-19. $[(\exists x)(P \,\&\, Qx)$ SYN $(P \,\&\, (\exists x)Qx)]$ (Yields dual, T3-20)
3) T3-21. $[(\forall x)Px$ SYN $((\forall x)Px \,\&\, (\exists x)Px)]$
   (Yields dual,T3-22, and CONT-theorem T3-36, helps with CONT-theorems T3-43,T3-44)
4) T3-23. $[(\exists x)(Px \,\&\, Qx)$ SYN $((\exists x)(Px \,\&\, Qx) \,\&\, (\exists x)Px)]$
   (Yields dual,T3-24, and CONT-Theorem T3-47)
5) T3-25. $[((\forall x)Px \,\&\, (\exists x)Qx)$ SYN $((\forall x)Px \,\&\, (\exists x)(Px \,\&\, Qx))]$
   (Yields Dual, T3-26, and CONT-theorems T3-45, T3-46)
6) T3-27. $[(\exists y)(\forall x)Rxy$ SYN $((\exists y)(\forall x)Rxy \,\&\, (\forall x)(\exists y))Rxy)]$
   (Yields dual,T3-28, and CONT-Theorem T3-37]

---

1) **T3-13. $[(\forall x)(Px \,\&\, Qx)$ SYN $((\forall x)Px \,\&\, (\forall x)Qx)]$**

Proof: Let $x_n = \cup\, a_n = \{a_1,a_2,\ldots,a_k,a_{k+1},\ldots\}$, $n \in N$.

A. Basis step: For $n=1$, we have $x_1 = \{a_1\}$ and:

1) $(\forall x_1)(Px \,\&\, Qx)$ SYN $(Pa_1 \,\&\, Qa_1)$ [$\forall_1$-Exp]
2) $(\forall x_1)Px$ SYN Pa_1 [$\forall_1$-Exp]
3) $(\forall x_1)Qx$ SYN Qa_1 [$\forall_1$-Exp]
4) $(\forall x_1)(Px \,\&\, Qx)$ SYN $((\forall x_1)Px \,\&\, Qa_1)$ [1),2),R1]
5) $(\forall x_1)(Px \,\&\, Qx)$ SYN $(((\forall x_1)Px \,\&\, (\forall x_1)Qx)$ [4),3),R1]

B. Inductive step: For $n=k$ we have $x_n = x_k = \{a_1,a_2,\ldots,a_k\}$,
for $n=k+1$ we have $x_n = x_{k+1} = \{a_1,a_2,\ldots,a_k,a_{k+1}\}$, and:

1) $(\forall x_k)(Px \,\&\, Qx)$ SYN $((\forall x_k)Px \,\&\, (\forall x_k)Qx)$ [Assumption]
2) $(\forall x_k)(Px \,\&\, Qx)$ SYN $((Pa_1 \,\&\, Qa_1)\&(Pa_2 \,\&\, Qa_2)\&\ldots\&(Pa_k \,\&\, Qa_k))$ [$\forall_k$-Exp]
3) $(\forall x_k)Px$ SYN $(Pa_1 \,\&\, Pa_2 \,\&\ldots\&\, Pa_k)$ [$\forall_k$-Exp]
4) $(\forall x_k)Qx$ SYN $(Qa_1 \,\&\, Qa_2 \,\&\ldots\&\, Qa_k)$ [$\forall_k$-Exp]
5) $(\forall x_k)(Px \,\&\, Qx)$ SYN $((Pa_1 \,\&\, Pa_2 \,\&\ldots\&\, Pa_k) \,\&\, (\forall x_k)Qx)$ [1),3)DR1-1,R1]
6) $(\forall x_k)(Px \,\&\, Qx)$ SYN $((Pa_1 \,\&\, Pa_2 \,\&\ldots\&\, Pa_k) \,\&\, (Qa_1 \,\&\, Qa_2 \,\&\ldots\&\, Qa_k))$ [5),4)DR1-1,R1]
7) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $((Pa_1 \,\&\, Qa_1)\&((Pa_2 \,\&\, Qa_2)\& (\ldots\&((Pa_k \,\&\, Qa_k)\&(Pa_{k+1} \,\&\, Qa_{k+1}))\ldots)))$ [$\forall_{k+}$1-Exp]
8) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $(((Pa_1 \,\&\, Qa_1)\&(Pa_2 \,\&\, Qa_2)\&\ldots\&(Pa_k \,\&\, Qa_k)) \,\&(Pa_{k+1} \,\&\, Qa_{k+1}))$ [7),&-ORD]
9) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $((\forall x_k)(Px \,\&\, Qx) \,\&\, (Pa_{k+1} \,\&\, Qa_{k+1}))$ [8),2),R1]
10) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $(((Pa_1 \,\&\, Pa_2 \,\&\ldots\&\, Pa_k)\&(Qa_1\&Qa_2\&\ldots\&Qa_k)) \,\&\, (Pa_{k+1}\&Qa_{k+1}))$ [9),6)DR1-1,R1]
11) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $(((Pa_1 \,\&\, Pa_2 \,\&\ldots\&\, Pa_k \,\&\, Pa_{k+1})\&((Qa_1 \,\&\, Qa_2 \,\&\ldots\&\, Qa_k \,\&\, Qa_{k+1}))$ [10),&-ORD]
12) $(\forall x_{k+1})Px$ SYN $(Pa_1 \,\&\, Pa_2 \,\&\ldots\&\, Pa_k \,\&\, Pa_{k+1})$ [$\forall_{k+}$1-Exp]
13) $(\forall x_{k+1})Qx$ SYN $(Qa_1 \,\&\, Qa_2 \,\&\ldots\&\, Qa_k \,\&\, Qa_{k+1}))$ [$\forall_{k+}$1-Exp]
14) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $((\forall x_{k+1})Px \,\&\, (Qa_1 \,\&\, Qa_2 \,\&\ldots\&\, Qa_k \,\&\, Qa_{k+1}))$ [11),12),R1]
15) $(\forall x_{k+1})(Px \,\&\, Qx)$ SYN $((\forall x_{k+1})Px \,\&\, (\forall x_{k+1})Qx)$ [14),13),R1]

Hence:

16) If $((\forall x_k)(Px \,\&\, Qx)$ SYN $((\forall x_k)Px \,\&\, (\ Vx_k)Qx)$
    then $((\forall x_{k+1})(Px \,\&\, Qx)$ SYN $((\forall x_{k+1})Px \,\&\, (\forall x_{k+1})Qx)$ [By 1)-15), Conditional Proof]

Hence, T3-13. $[(\forall x)(Px \,\&\, Qx)$ SYN $((\forall x)Px \,\&\, (\forall x)Qx)]$ [By Steps A and B, Math Ind.]

2) T3-19. [$(\exists x)(P \& Qx)$ SYN $(P \& (\exists x)Qx)$]

Proof: Let $x_n = \cup a_n = \{a_1, a_2, \ldots, a_k, a_{k+1}, \ldots\}$, $n \in N$.

A. Basis step:

For $n=1$, we have $x_1 = \{a_1\}$ and:

1) $(\exists_1 x)Qx$ SYN Qa_1 [$\exists_1$-Exp]
2) $(\exists_1 x)(P \& Qx)$ SYN $(P \& Qa_1)$ [$\exists_1$-Exp]
3) $(\exists_1 x)(P \& Qx)$ SYN $(P \& (\exists_1 x)Qx)$ [2),1)DR1-1,R1]

B. Inductive step: For $n=k$ we have $x_n = x_k = \{a_1, a_2, \ldots, a_k\}$,
for $n=k+1$ we have $x_n = x_{k+1} = \{a_1, a_2, \ldots, a_k, a_{k+1}\}$, and:

1) $(\exists_k x)(P \& Qx)$ SYN $(P \& (\exists_k x)Qx)$ [Assumption]
2) $(\exists_k x)Qx$ SYN $(Qa_1 \text{ v } Qa_2 \text{ v} \ldots \text{v } Qa_k)$ [$\exists_k$-Exp]
3) $(\exists_k x)(P\&Qx)$ SYN $((P\&Qa_1)\text{v}(P\&Qa_2)\text{v}\ldots\text{v}(P\&Qa_k))$ [$\exists_k$-Exp]
4) $(\exists_{k+1} x)Qx$ SYN $(Qa_1 \text{ v } Qa_2 \text{ v} \ldots \text{ v } Qa_k \text{ v } Qa_{k+1})$ [$\exists_{k+}1$-Exp]
5) $(\exists_{k+1} x)Qx$ SYN $((Qa_1 \text{ v } Qa_2 \text{ v} \ldots \text{ v } Qa_k) \text{ v } Qa_{k+1})$ [4),v-ORD]
6) $(\exists_{k+1} x)Qx$ SYN $((\exists_k x)Qx \text{ v } Qa_{k+1})$ [5),2),R1]
7) $(\exists_{k+1} x)(P\&Qx)$ SYN $((P\&Qa_1)\text{v}(P\&Qa_2)\text{v}\ldots\text{v}((P\&Qa_k)\text{v}(P\&Qa_{k+1}))$ [$\exists_{k+}1$-Exp]
8) $(\exists_{k+1} x)(P\&Qx)$ SYN $(((P\&Qa_1)\text{v}(P\&Qa_2)\text{v}\ldots\text{v}(P\&Qa_k))\text{v}(P\&Qa_{k+1}))$ [7),v-ORD]
9) $(\exists_{k+1} x)(P\&Qx)$ SYN $((\exists_k x)(P\&Qx)\text{v}(P\&Qa_{k+1}))$ [8),3),R1]
10) $(\exists_{k+1} x)(P\&Qx)$ SYN $((P\&(\exists_k x)Qx)\text{v}(P\&Qa_{k+1}))$ [9),1)(DR1-1),R1]
11) $(\exists_{k+1} x)(P\&Qx)$ SYN $(P \& ((\exists_k x)Qx \text{ v } Qa_{k+1}))$ [10),A4b,R1]
12) $(\exists_{k+1} x)(P\&Qx)$ SYN $(P \& (\exists_{k+1} x)Qx)$ [11),6),R1]
13) If $(\exists_k x)(P \& Qx)$ SYN $(P \& (\exists_k x)Qx)$ then $(\exists_{k+1} x)(P \& Qx$ SYN $(P \& (\exists_{k+1} x)Qx)$ [1)-13),C. Pr.]

Hence, [$(\exists x)(P \& Qx)$ SYN $(P \& (\exists x)Qx)$] [Steps 1 v 2, Math Induct]

3) T3-21. [$(\forall x)Px$ SYN $((\forall x)Px \& (\exists x)Px)$]

Proof: Let $x_n = \cup a_n = \{a_1, a_2, \ldots, a_k, a_{k+1}, \ldots\}$, $n \in N$.

A. Basis step: For $n=1$, we have $x_1 = \{a_1\}$ and:

1) $(\exists_1 x)Px$ SYN Pa_1 [$\exists_1$-Exp]
2) $(\forall_1 x)Px$ SYN Pa_1 [$\forall_1$ -Exp]
3) $((\forall_1 x)Px\&(\exists_1 x)Px)$ SYN $((\forall_1 x)Px\&(\exists_1 x)Px)$ [T1-11]
4) $((\forall_1 x)Px\&(\exists_1 x)Px)$ SYN $((\forall_1 x)Px\&Pa_1)$ [3),1),R1]
5) $((\forall_1 x)Px\&(\exists_1 x)Px)$ SYN $(Pa_1\&Pa_1)$ [4),2),R1]
6) $((\forall_1 x)Px\&(\exists_1 x)Px)$ SYN Pa_1 [5),A1a,R1]
7) $(\forall_1 x)Px$ SYN $((\forall_1 x)Px\&(\exists_1 x)Px)$ [2),6),DR1-1,R1]

B. Inductive step: For $n=k$ we have $x_n = x_k = \{a_1, a_2, \ldots, a_k\}$,
for $n=k+1$ we have $x_n = x_{k+1} = \{a_1, a_2, \ldots, a_k, a_{k+1}\}$, and:

1) $(\forall_k x)Px$ SYN $((\forall_k x)Px\&(E_k x)Px)$ [Assumption]
2) $(\forall_k x)Px$ SYN $(Pa_1\&Pa_2\&\ldots\&Pa_k)$ [$\forall_k$-Exp]
3) $(\exists_k x)Px$ SYN $(Pa_1\text{v}Pa_2\text{v}\ldots\text{v}Pa_k)$ [$\exists_k$-Exp]
4) $(\forall_{k+1} x)Px$ SYN $(Pa_1\&Pa_2\&\ldots\&Pa_k)\&P_{k+1})$ [$\forall_{k+}1$-Exp]
5) $(\forall_{k+1} x)Px$ SYN $((\forall_k x)Px\&Pa_{k+1})$ [7),3),R1]
6) $(\exists_{k+1} x)Px$ SYN $(Pa_1\text{v}Pa_2\text{v}\ldots\text{v}Pa_k\text{v}Pa_{k+1})$ [$\exists_{k+}1$-Exp]
7) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ [T1-11]
8) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((\forall xk+1)Px\&(Pa_1\text{v}Pa_2\text{v}\ldots\text{v}Pa_k\text{v}Pa_{k+1}))$ [7),6),R1]
9) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $((Pa_1\&Pa_2\&\ldots\&Pa_k\&P_{k+1})\ \&(Pa_1\text{v}Pa_2\text{v}\ldots\text{v}Pa_k\text{v}Pa_{k+1}))$ [8),4),R1]
10) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(((\forall_k x)Px\&Pa_{k+1})\&(Pa_1\text{v}Pa_2\text{v}\ldots\text{v}Pa_k\text{v}Pa_{k+1}))$ [9),2),R1]
11) $((\forall_{k+1} x)Px\&(\exists_{k+1} x)Px)$ SYN $(((\forall_k x)Px\&P_a k_{+1})\&((\exists_k x)Px \text{ v } Pa_{k+1}))$ [10),6),R1]

12) $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ SYN $((((\forall_k x)Px\&Pa_{k+1}\&(\exists_k x)Px)$ v $(((\forall_k x)Px\&Pa_{k+1})\&Pa_{k+1}))$ [11),A1-4b(DR1-1),R1]

13) $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ SYN $((((\forall_k x)Px\&(\exists_k x)Px\&Pa_{k+1})$ v $((\forall_k x)Px\&(Pa_{k+1}\&Pa_{k+1})))$ [12),&-ORD,twice]

14) $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ SYN $(((\forall_k x)Px\&Pa_{k+1})v(\forall_k x)Px \;\&\; (Pa_{k+1} \;\&\; Pa_{k+1})))$ [13),1),R1]

15) $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ SYN $(((\forall_k x)Px\&Pa_{k+1})v((\forall_k x)Px \;\&\; Pa_{k+1}))$ [14),A1a,R1]

16) $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ SYN $((\forall_k x)Px\&Pa_{k+1}$ [15),A3a,R1]

17) $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ SYN $(\forall_{k+1}x)Px$ [16),5),R1]

18) $(\forall_{k+1}x)Px$ SYN $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ [17),DR1-1]

19) If $(\forall_k x)Px$ SYN $((\forall_k x)Px\&(\exists_k x)Px)$ then $(\forall_{k+1}x)Px$ SYN $((\forall_{k+1}x)Px\&(\exists_{k+1}x)Px)$ [1) -19),C.P.]

Hence, $[(\forall x)Px$ SYN $((\forall x)Px \;\&\; (\exists x)Px)]$ [Steps 1 & 2, Math Induct]

4) T3-23. $[((\exists x)(Px \;\&\; Qx)$ SYN $((\exists x)(Px \;\&\; Qx) \;\&\; (\exists x)Px)]$

Proof: Let $x_n = \bigcup a_n = \{a_1,a_2,\ldots,a_k,a_{k+1},\ldots\}$, $n \in N$.

A. Basis step: For $n=1$, we have $x_1=\{a_1\}$ and:

1) $(\exists_1 x)(Px \;\&\; Qx)$ SYN $(Pa_1 \;\&\; Qa_1)$ [$\exists$1-Exp]

2) $(\exists_1 x)Px$ SYN Pa_1 [$\exists$1-Exp]

3) $(Pa_1 \;\&\; Qa_1)$ SYN $(Pa_1 \;\&\; Qa_1)$ [T1-11]

4) $(Pa_1 \;\&\; Qa_1)$ SYN $((Pa_1 \;\&\; Pa_1) \;\&\; Qa_1)$ [3),Ax.1-1a(DR1-1),R1]

5) $(Pa_1 \;\&\; Qa_1)$ SYN $((Pa_1 \;\&\; Qa_1) \;\&\; Pa_1)$ [4),&-ORD]

6) $(\exists_1 x)(Px \;\&\; Qx)$ SYN $((Pa_1 \;\&\; Qa_1) \;\&\; Pa_1)$ [5),1),R1c]

7) $(\exists_1 x)(Px \;\&\; Qx)$ SYN $((\exists_1 x)(Px \;\&\; Qx) \;\&\; Pa_1))$ [6),1),R1]

8) $(\exists_1 x)(Px \;\&\; Qx)$ SYN $(((\exists_1 x)(Px \;\&\; Qx) \;\&\; (\exists_1 x)Px)$ [7),2),R1]

B. Inductive step:

For $n=k$ we have $x_n=x_k=\{a_1,a_2,\ldots,a_k\}$, for $n=k+1$ we have $x_n=x_{k+1}=\{a_1,a_2,\ldots,a_k,a_{k+1}\}$, and:

1) $(\exists_k x)(Px \;\&\; Qx)$ SYN $((\exists_k x)(Px \;\&\; Qx) \;\&\; (\exists_k x)Px)$ [Assumption]

2) $(\exists_k x)(Px \;\&\; Qx)$ SYN $((Pa_1 \;\&\; Qa_1)$v$(Pa_2 \;\&\; Qa_2)$v…v$(Pa_k \;\&\; Qa_k)$ [$\exists_k$-Exp]

3) $(\exists_k x)Px$ SYN $(Pa_1$ v Pa_2v…v $Pa_k)$ [$\exists_k$-Exp]

4) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((Pa_1 \;\&\; Qa_1)$v$(Pa_2 \;\&\; Qa_2)$v…v$(Pa_k \;\&\; Qa_k)$ v$(Pa_{k+1} \;\&\; Qa_{k+1}))$[$\exists_{k+}$1-Exp]

5) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_k x)(Px \;\&\; Qx)$ v $(Pa_{k+1} \;\&\; Qa_{k+1}))$ [4),2),R1]

6) $(\exists_{k+1}x)Px$ SYN $(Pa_1$vPa_2v…vPa_kv$Pa_{k+1})$ [$\exists_{k+}$1-Exp]

7) $(\exists_{k+1}x)Px$ SYN $((\exists_k x)Px$ v $Pa_{k+1})$ [6),3),R1]

8) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $(((\exists_k x)(Px \;\&\; Qx) \;\&\; (\exists_k x)Px)$ v $(Pa_{k+1} \;\&\; Qa_{k+1}))$ [5),1)DR1-1,R1]

9) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_k x)(Px \;\&\; Qx)$ v $(Pa_{k+1} \;\&\; Qa_{k+1}))$
$\&(((\exists_k x)Px$ v $(Pa_{k+1} \;\&\; Qa_{k+1}))$ [8),A4a(DR1-1),R1]

10) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_{k+1}x)(Px \;\&\; Qx) \;\&(((\exists_k x)Px$ v $(Pa_{k+1} \;\&\; Qa_{k+1}))$ [9),5),R1]

11) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_{k+1}x)(Px \;\&\; Qx) \;\&((\exists_k x)Px$ v $Pa_{k+1}) \;\&((\exists_k x)Px$ v $Qa_{k+1}))$ [10),A4a(DR1-11),R1]

12) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_{k+1}x)(Px \;\&\; Qx) \;\&(((\exists_k x)Px$ v $Pa_{k+1})$
$\&((\exists_k x)Px$ v $Pa_{k+1})) \;\&((\exists_k x)Px$ v $Qa_{k+1}))$ [11),Ax.01(DR1-1),R1]

13) $(\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $(((\exists_{k+1}x)(Px \;\&\; Qx) \;\&\; (((\exists_k x)Px$ v $Pa_{k+1}) \;\&\; ((\exists_k x)Px$ v $Qa_{k+1})))$
$\&(\exists_k x)Px$ v $Pa_{k+1})$ [12),&-ORD]

14) $((\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_{k+1}x)(Px \;\&\; Qx) \;\&\; (\exists_k x)Px$ v $Pa_{k+1}))$ [13),11),R1]

15) $((\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_{k+1}x)(Px \;\&\; Qx) \;\&\; (\exists_{k+1}x)Px)$ [14),7),R1]

Hence: If $((\exists_k x)(Px \;\&\; Qx)$ SYN $((\exists_k x)(Px \;\&\; Qx) \;\&\; (\exists_k x)Px)$
then $((\exists_{k+1}x)(Px \;\&\; Qx)$ SYN $((\exists_{k+1}x)(Px \;\&\; Qx) \;\&\; (\exists_{k+1}x)Px)$ [1) to 15),C.Pr.]

Hence, T3-23.$[(\exists x)(Px \;\&\; Qx)$ SYN $((\exists x)(Px \;\&\; Qx) \;\&\; (\exists x)Px)]$ [By Steps 1&2, Math Ind.]

5) T3-25. [((∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx))]

Proof: Let $x_n = \bigcup a_n = \{a_1, a_2, \ldots, a_k, a_{k+1}, \ldots\}$, $n \in N$.

A. Basis step: For $n=1$, we have $x_1 = \{a_1\}$ and:

1) $(\forall_1 x)Px$ SYN Pa_1 [_1-Exp]
2) $(\exists_1 x)Qx$ SYN Qa_1 [_$_1$ -Exp]
3) $(\exists_1 x)(Px \ \& \ Qx)$ SYN $(Pa_1 \ \& \ Qa_1)$ [_1-Exp]
4) $((\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $((\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ [T1-11,U-SUB]
5) $((\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $(Pa_1 \ \& \ (\exists_1 x)Qx)$ [4),2),R1]
6) $((\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $(Pa_1 \ \& \ Qa_1)$ [5),1),R1]
7) $((\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $((Pa_1 \ \& \ Pa_1) \ \& \ Qa_1))$ [6),A1a,R1]
8) $(\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $(Pa_1 \ \& \ (Pa_1 \ \& \ Qa_1))$ [7),A3a,R1]
9) $(\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $(((\forall_1 x)Px \ \& \ (Pa_1 \ \& \ Qa_1))$ [8),2),R1]
10) $(\forall_1 x)Px \ \& \ (\exists_1 x)Qx)$ SYN $(((\forall_1 x)Px \ \& \ (\exists_1 x)(Px \ \& \ Qx))$ [9),3),R1]

B. Inductive step: For $n=k$ we have $x_n = x_k = \{a_1, a_2, \ldots, a_k\}$, for $n=k+1$ we have $x_n = x_{k+1} = \{a_1, a_2, \ldots, a_k, a_{k+1}\}$, and:

1) $((\forall_k x)Px \ \& \ (\exists_k x)Qx)$ SYN $((\forall_k x)Px \ \& \ ((\exists_k x)(Px \ \& \ Qx))$ [Assumption]
2) $(\forall_k x)Px$ SYN $(Pa_1 \& Pa_2 \& \ldots \& Pa_k)$ [$\exists_k$-Exp]
3) $(\exists_k x)Qx$ SYN $(Qa_1 v Qa_2 v \ldots v Qa_k)$ [$\forall_k$-Exp]
4) $(\exists_k x)(Px \ \& \ Qx)$ SYN $((Pa_1 \ \& \ Qa_1) v (Pa_2 \ \& \ Qa_2) v \ldots v (Pa_k \ \& \ Qa_k)$ [$\exists_k$-Exp]
5) $(\forall_{k+1} x)Px$ SYN $(Pa_1 \& Pa_2 \& \ldots \& Pa_k \& Pa_{k+1})$ [$\forall_{k+}$1-Exp]
6) $(\forall_{k+1} x)Px$ SYN $((\forall_k x)Px \ \& \ Pa_{k+1})$ [5),2),R1]
7) $(\exists_{k+1} x)Qx$ SYN $(Qa_1 v Qa_2 v \ldots v Qa_k v Qa_{k+1})$ [$\exists_{k+}$1-Exp]
8) $(\exists_{k+1} x)Qx$ SYN $((\exists_k x)Qx \ v \ Qa_{k+1})$ [7),3),R1]
9) $(\exists_{k+1} x)(Px \ \& \ Qx)$ SYN $((Pa_1 \ \& \ Qa_1) v (Pa_2 \ \& \ Qa_2) v \ldots v (Pa_k \ \& \ Qa_k) \ v (Pa_{k+1} \ \& \ Qa_{k+1}))$[$\exists_{k+}$1-Exp]
10) $(\exists_{k+1} x)(Px \ \& \ Qx)$ SYN $((\exists_k x)(Px \ \& \ Qx) \ v \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [9),4),R1]
11) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ [T1-11]
12) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ (\exists_{k+1} x)Qx)$ [11),6)DR1-1,R1]
13) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ ((\exists_k x)Qx \ v \ Qa_{k+1})$ [12),8)DR1-1,R1]
14) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $(((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ (\exists_k x)Qx)$
 $v((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ Qa_{k+1}))$ [13),A4b(DR1-1),R1]
15) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $(((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ ((\forall_k x)Px \ \& \ (\exists_k x)Qx))$
 $v((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [&-ORD]
16) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $(((\forall_k x)Px \& Pa_{k+1}) \ \&$ $\mathbf{((\forall_k x)Px \ \& \ ((\exists_k x)(Px \ \& \ Qx)))}$
 $v((\forall_k x)Px \& Pa_{k+1}) \ \& \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [15),1),R1]
17) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $($$\mathbf{((\forall_k x)Px \ \& \ (\forall_k x)Px)}$ $\& \ Pa_{k+1}) \ \& \ ((\exists_k x)(Px \ \& \ Qx)))$
 $v((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [16),&-ORD]
18) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $((($$\mathbf{(\forall_k x)Px}$ $\& \ Pa_{k+1}) \ \& \ ((\exists_k x)(Px \ \& \ Qx)))$
 $v((\forall_k x)Px \ \& \ Pa_{k+1}) \ \& \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [17),A1a,R1]
19) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $(((\forall_k x)Px \& Pa_{k+1})$
 $\&((\exists_k x)(Px \ \& \ Qx)) \ v \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [18),A4b,R1]
20) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $($$\mathbf{(\forall_k x)Pa_{k+1}}$
 $\&((\exists_k x)(Px \ \& \ Qx)) \ v \ (Pa_{k+1} \ \& \ Qa_{k+1}))$ [19),6),R1]
21) $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $((\forall_{k+1} x)Px \ \&$ $\mathbf{(\exists_{k+1} x)(Px \ \& \ Qx)}$$)$ [20),10),R1]
22) If $((\forall_k x)Px \ \& \ (\exists_k x)Qx)$ SYN $((\forall_k x)Px \ \& \ ((\exists_k x)(Px \ \& \ Qx))$
 then $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)Qx)$ SYN $((\forall_{k+1} x)Px \ \& \ (\exists_{k+1} x)(Px \ \& \ Qx))$ [By 1)-21), Cond. Pr.]

Hence: T3-25. [((∀x)Px & (∃x)Qx) SYN ((∀x)Px & (∃x)(Px & Qx))] [Math. Ind. Step 1 & Step 2]

6) T3-27: [($\exists$y)($\forall$x)Rxy SYN (($\exists$y)($\forall$x)Rxy & ($\forall$x)($\exists$y)Rxy)]

<u>Proof</u>: Let $x_n = \cup a_n = \{a_1, a_2, \ldots, a_k, a_{k+1}, \ldots\}$, $n \in N$.

A. <u>Basis step</u>: (For n=1, we have $x_1 = \{a_1\}$)

1) $(\forall_1 x)(\exists_1 y)Rxy$ SYN $(\exists_1 y)Ra_1y$ [$\forall_1$ -Exp]
2) $(\exists_1 y)Ra_1y$ SYN Ra_1a_1 [$\exists_1$-Exp]
3) $(\forall_1 x)(\exists_1 y)Rxy$ SYN Ra_1a_1 [1),2),R1b]
4) $(\exists_1 y)(\forall_1 x)Rxy$ SYN $(\forall_1 x)Rxa_1$ [$\exists_1$-Exp]
5) $(\forall_1 x)Rxa_1$ SYN Ra_1a_1 [$\forall_1$-Exp]
6) $(\exists_1 y)(\forall_1 x)Rxy$ SYN Ra_1a_1 [4),5), R1b]
7) $(\exists_1 y)(\forall_1 x)Rxy$ SYN $(Ra_1a_1$ & $Ra_1a_1)$ [6), Ax.1-1a(DR1-1),R1]
8) $(\exists_1 y)(\forall_1 x)Rxy$ SYN $(Ra_1a_1$ & $(\forall_1 x)(\exists_1 y)Rxy$ [7),3),R1]
9) $(\exists_1 y)(\forall x)Rxy$ SYN $((\exists_1 y)(\forall x)Rxy$ & $(\forall x)(\exists_1 y)Rxy)$ [8),6),R1]

B. <u>Inductive step</u>: (For n=k, let $x_n = x_k = \{a_1, a_2, \ldots, a_k\}$, for n=k+1, let $x_n = x_{k+1} = \{a_1, a_2, \ldots, a_k, a_{k+1}\}$, and:)

1) $(\exists_k y)(\forall_k x)Rxy$ SYN $((\exists_k y)(\forall_k x)Rxy$ & $(\forall_k x)(\exists_k y)Rxy)$ [Assumption]
2) $(\exists_k y)(\forall_k x)Rxy$ SYN $((\forall_k x)Rxa_1$ v $(\forall_k x)Rxa_2$ v…v $(\forall_k x)Rxa_k)$ [$\exists_k$-Exp]
3) $((\forall_k x)Rxa_i$ SYN $((Ra_1a_i$ & Ra_2a_i &…& $Ra_ka_i)$ [$\forall$k-Exp]
4) $((\forall_k x)Rxa_1$ v $(\forall_k x)Rxa_2$ v…v $(\forall_k x)Rxa_k)$ …
SYN $((Ra_1a_1$ & Ra_2a_1 &…& $Ra_ka_1)$ v $(Ra_1a_2$ & Ra_2a_2 &…& $Ra_ka_2)$ v…**v** $(Raa_k$ & Ra_2a_k &…& $Ra_ka_k))$
[2),3)($\forall_k$-Exp), k times]
5) $(\exists_k y)(\forall_k x)Rxy$ SYN $((Ra_1a_1$ & Ra_2a_1 &…& $Ra_ka_1)$ v$(Ra_1a_2$ & Ra_2a_2 &…& $Ra_ka_2)$ v….
… v$(Ra_1a_k$ & Ra_2a_k &…& $Ra_ka_k))$ 2),4), R1b]
6) $(\forall_{k+1} x)Rxa_i$ SYN $(Ra_1a_i$ & Ra_2a_i &…& Ra_ka_i & $Ra_{k+1}a_i)$ [$\forall_{k+}$1Exp]
7) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $((\forall_{k+1} x)Rxa_1$ v $(\forall_{k+1} x)Rxa_2$ v…v $(\forall_{k+1} x)Rxa_k$ v $(\forall_{k+1} x)Rxa_{k+1})$[$\exists_{k+1}$y–Exp]
8) $((\forall_{k+1} x)Rxa_1$ v $(\forall_{k+1} x)Rxa_2$ v…v $(\forall_{k+1} x)Rxa_k$ v $(\forall_{k+1} x)Rxa_{k+1})$
SYN $((Ra_1a_1$ & Ra_2a_1 &…& Ra_ka_1 & $Ra_{k+1}a_1)$ v $(Ra_1a_2$ & Ra_2a_2 &…& Ra_ka_2 & $Ra_{k+1}a_2$ **)** **v**….
… v$(Ra_1a_k$ & Ra_2a_k &…& Ra_ka_k & $Ra_{k+1}a_k)$ v $(Ra_1a_{k+1}$ & Ra_2a_{k+1} &…& $Ra_{k+1}a_k$ & $Ra_{k+1}a_{k+1}))$
[(7),6)($\forall_{k+}$1-Exp), k+1 times]
9) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN
$((Ra_1a_1$ & Ra_2a_1 &…& Ra_ka_1 & $Ra_{k+1}a_1)$ v $(Ra_1a_2$ & Ra_2a_2 &….& $(Ra_ka_2$ & $Ra_{k+1}a_2)$ **v**….
…v$(Ra_1a_k$ & Ra_2a_k &…& Ra_ka_k & $Ra_{k+1}a_k)$ v $(Ra_1a_{k+1}$ & Ra_2a_{k+1} &…& $Ra_{k+1}a_k$ & $Ra_{k+1}a_{k+1}))$
[7),8),R1b]
10) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $(((Ra_1a_1$ & Ra_2a_1 &…& Ra_ka_1 & $Ra_{k+1}a_1)$
v$(Ra_1a_2$ & Ra_2a_2 &…& Ra_ka_2 & $Ra_{k+1}a_2)$
v….
v$(Ra_1a_k$ & Ra_2a_k &…& Ra_ka_k & $Ra_{k+1}a_k))$
v$(Ra_1a_{k+1}$ & Ra_2a_{k+1} &…& $Ra_{k+1}a_k$ & $Ra_{k+1}a_{k+1}))$
&$(Ra_1a_1$ v Ra_1a_2 v…v Ra_1a_k v $Ra_1a_{k+1})$
&$(Ra_2a_1$ v Ra_2a_2 v…v Ra_2a_k v $Ra_2a_{k+1})$
&….
&$(Ra_ka_1$ v Ra_ka_2 v…v Ra_ka_k v $Ra_ka_{k+1}))$
&$(Ra_{k+1}a_1$ v $Ra_{k+1}a_2$ v…v $Ra_{k+1}a_k$ v $Ra_{k+1}a_{k+1})$ [9),GEN v&-DIST]
11) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $((\exists_{k+1} y)(\forall_{k+1} x)Rxy$ &
$((Ra_1a_1$ v Ra_1a_2 v…v Ra_1a_k v $Ra_1a_{k+1})$
&$(Ra_2a_1$ v Ra_2a_2 v…v Ra_2a_k v $Ra_2a_{k+1})$
&….
&$(Ra_ka_1$ v Ra_ka_2 v…v Ra_ka_k v $Ra_ka_{k+1}))$
&$(Ra_{k+1}a_1$ v $Ra_{k+1}a_2$ v…v $Ra_{k+1}a_k$ v $Ra_{k+1}a_{k+1}))$ [10),9),R1]

12) $(\exists_k y)Ra_i y$ SYN $((Ra_i a_1 \text{ v } Ra_i a_2 \text{ v} \ldots \text{v } Ra_i a_k)$ [$\exists_k$-Exp]

13) $(\exists_{k+1} y)Ra_i y$ SYN $(Ra_i a_1 \text{ v } Ra_i a_2 \text{ v} \ldots \text{v } Ra_i a_k \text{ v } Ra_i a_{k+1})$ [$\exists_{k+}$1-Exp]

14) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $((\exists_k y)(\forall_{k+1} x)Rxy$
$\&(((\exists_k y)Ra_1 y \text{ v } Ra_1 a_{k+1}) \& ((\exists_k y)Ra_2 y \text{ v } Ra_2 a_{k+1}) \& \ldots.$
$\ldots.\&((\exists_k y)Ra_k y \text{ v } Ra_k a_{k+1})) \&((\exists_k y)Ra_{k+1} y \text{ v } Ra_{k+1} a_{k+1})))$ [11),13),R1 k+1 times]

15) $(\exists_{k+1} y)Ra_i y$ SYN $((\exists_k y)Ra_i y \text{ v } Ra_i a_{k+1})$ [13),12),R1]

16) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $((\exists_{k+1} y)(\exists_{k+1} x)Rxy \& (\exists_{k+1} y)Ra_1 y$
$\&(\exists_{k+1} y)Ra_2 y \& \ldots.\&(\exists_{k+1} y)Ra_k y \&(\exists_{k+1} y)Ra_{k+1} y))$
[14),15),R1-k+1 times]

17) $(\forall_{k+1} x)(\exists_{k+1} y)Rxy$ SYN $((\exists_{k+1} y)Ra_1 y \& (\exists_{k+1} y)Ra_2 y \& \ldots.\&(\exists_{k+1} y)Ra_k y \&(\exists_{k+1} y)Rya_{k+1})$
[$\forall_{k+}$1-Exp]

18) $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $((\exists_{k+1} y)(\forall_{k+1} x)Rxy \& (\forall_{k+1} x)(\exists_{k+1} y)Rxy)$ [17),16),R1]

<u>Hence:</u> 19) If $(\exists_k y)(\forall_k x)Rxy$ SYN $((\exists_k y)(\forall_k x)Rxy \& (\forall_k x)(\exists_k y)Rxy)$
then $(\exists_{k+1} y)(\forall_{k+1} x)Rxy$ SYN $((\exists_{k+1} y)(\forall_{k+1} x)Rxy \& (\forall_{k+1} x)(\exists_{k+1} y)Rxy$ [By 1)-18), C.P.]

<u>Hence</u>, T3-27: [(∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy)] [By Math. Ind, Step 1 & Step 2]

To see how step 10) works, consider a proof of T3-27 in Ra domain of 3:

T3-27. [(∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]

<u>Proof in a domain of n=(k+1)=3</u> 1) (∃y)(∀x)Rxy SYN (∃y)(∀x)Rxy] [T1-11]

2) (∃y)(∀x)Rxy SYN (∃y)(Ray & Rby & Rcy)] [$\forall_3$-Exp]

<u>Step 9):</u> 3) (∃y)(∀x)Rxy SYN ((Raa & Rba & Rca)v(Rab & Rbb & Rcb)v(Rac & Rbc & Rcc)) [$\exists_3$-Exp]

<u>Step 10):</u> 4) (**Raa** & Rba & Rca)v(**Rab** & Rbb & Rcb)v(**Rac** & Rbc & Rcc))
SYN (**(Raa v Rab v Rac)** & (Rba v Rbb v Rbc) & (Rca v Rcb v Rcc)) [9),GEN v&-DIST]

<u>Step 11):</u> 5) (∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & ((Raa v Rab v Rac)&(Rba v Rbb v Rbc)
&(Rca v Rcb v 0cc))) [5),4),R1]

<u>Step 16):</u> 6) (∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & ((∃y)Ray & (∃y)Rby & (∃y)Rcy) [5),$\exists_3$-Contr]

<u>Step 17):</u> 7) (∀x)(∃y)Rxy SYN ((∃y)Ray & (∃y)Rby & (∃y)Rcy) [$\forall_3$-Exp]

<u>Step 18):</u> 8) [(∃y)(∀x)Rxy SYN ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))] [16),17),SynSUB]

Appendix IV—Rules and Theorems, Chapter 4

Primitive symbol: Negation sign: ~ (Introduced for first time) **Added**

Rule of formation: FR4-1. If P is wff, then [~P] is wff. **Added**

Rule of Inference: R4-3. If [P SYN Q] then [~P SYN ~Q] **Added**

Definitions:

D5. [(P v Q) SYN$_{df}$ ~(~P & ~Q)] [Df 'v'] **Added**

D6. [(P ⊃ Q) SYN$_{df}$ ~(P & ~Q)] [Df '⊃'] **Added**

D7. [(P ≡ Q) SYN$_{df}$ ((P ⊃ Q)&(Q ⊃ P))] [Df '≡'] **Added**

D8. [(∃x)P SYN$_{df}$ ~(∀x)~P] [Df '(∃x)'] **Added**

Axioms:

Ax.4-01. [P SYN (P&P)] [&-IDEM] [= Ax.1-01, Drop Ax.1-02]

Ax.4-02. [(P&Q) SYN (Q&P)] [&-COMM] [= Ax.1-03, Drop Ax.1-04]

Ax.4-03. [(P&(Q&R)) SYN ((P&Q)&R)] [&-ASSOC] [= Ax.1-05, Drop Ax.1-06]

Ax.4-04. [(Pv(Q&R)) SYN ((PvQ)&(PvR))] [&v-DIST] [= Ax.1-07, Drop Ax.1-08]

Ax.4-05. [P SYN ~~P] [DN] **Added**

DR4-1. If ⊨ [P SYN Q] then ⊨ [~P SYN ~Q]

R4-2. If [R SYN S] and (i) $P_i<t_1,\ldots,t_n>$ occurs in R,
and (ii) Q is an h-adic wff, where h $\geq$ n,
and (iii) Q has occurrences of all numerals 1 to n,
and (iv) no individual variable in Q occurs in R or S,
then it may be inferred that [R($P_i<t_1,\ldots,t_n>$/Q) SYN S($P_i<t_1,\ldots,t_n>$/Q)]. [U-SUB]

R4-2a ("U-SUBa")
If [R SYN S] and (i) $P_i<t_1,\ldots,t_n>$ occurs only POS in R and S,
and (ii) Q is an h-adic negation-free wff, where h $\geq$ n,
and (iii) Q has occurrences of all numerals 1 to n,
and (iv) no predicate letter in Q occurs NEG in R or S,
and (v) no variable in Q occurs in R or S,
then it may be inferred that [R($P_i<t_1,\ldots,t_n>$/Q) SYN S($P_i<t_1,\ldots,t_n>$/Q)]. [U-SUBa]

R4-2b ("U-SUBb")
If [R SYN S] and (i) $P_i<t_1,\ldots,t_n>$ occurs in R and S, (POS or NEG)
and (ii) Q is '~ $P_i<t_1,\ldots,t_n>$'
then it may be inferred that [R($P_i<t_1,\ldots,t_n>$/Q) SYN S($P_i<t_1,\ldots,t_n>$/Q)]. [U-SUBb]

R4-3. If ⊨ [P$<1>$] then ⊨ [Pa_i] [INST]

DR4-3a. If ⊨ [~P$<1>$] then ⊨ [(∀x)~Px] 'CG' (Conjunctive Generalization) [R3-3a,R4-2b]

DR4-3b. If ⊨ [~P$<1>$] then ⊨ [(∃x)~Px] 'DG' (Disjunctive Generalization) [R3-3b,R4-2b]

DR4-3c If ⊨ [P$<1>$ Syn Q$<1>$] then ⊨ [(∀x)~Px Syn (∀x)~Qx]

DR4-3d If ⊨ [P$<1>$ Syn Q$<1>$] then ⊨ [(∃x)~Px Syn (∃x)~Qx]

DR4-3e If ⊨ [P$<1>$ Cont Q$<1>$] then ⊨ [(∀x)~Px Cont (∀x)~Qx]

DR4-3f If ⊨ [P$<1>$ Cont Q$<1>$] then ⊨ [(∃x)~Px Cont (∃x)~Qx]

Theorems:

T4-11. [(P&Q) SYN ~(~Pv~Q)] [DeM1]

T4-12. [(PvQ) SYN ~(~P&~Q)] [Df 'v'] [DeM2]

T4-13. [(P&~Q) SYN ~(~PvQ)] [DeM3]

T4-14. [(P v~Q) SYN ~(~P&Q)] [DeM4]

T4-15. [(~P&Q) SYN ~(Pv~Q)] [DeM5]

T4-16. [(~PvQ) SYN ~(P&~Q)] [DeM6]

T4-17. [(~P&~Q) SYN ~(PvQ)] [DeM7]

T4-18. [(~Pv~Q) SYN ~(P&Q)] [DeM8]

T4-19. [P SYN (PvP)] [v-IDEM] [Same as Ax.1-02]
T4-20. [(PvQ) SYN (QvP)] [v-COMM] [Same as Ax.1-04]
T4-21. [(Pv(QvR)) SYN ((PvQ)vR)] [v-ASSOC] [Same as Ax.1-06]
T4-22. [(P&(QvR)) SYN ((P&Q)v(P&R))] [v&-DIST] [Same as Ax.1-08]
T4–23. [(P& ~ Q)) CONT (~ (Q& ~ P)]
T4-24. [(∃x) ~ Px SYN ~ (∀x)Px] [Q-Exch2]
T4-25. [(∀x) ~ Px SYN ~ (∃x)Px] [Q-Exch3]
T4-26. $[(\exists x_1)\ldots(\exists a_n) \sim P<x_1,\ldots,x_n>$ SYN $\sim(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n>]$ [Q-Exch4]
T4-27. $[(\forall x_1)\ldots(\forall x_n) \sim P<x_1,\ldots,x_n>$ SYN $\sim(\exists x_1)\ldots(\exists x_n)P<x_1,\ldots,x_n>]$ [Q-Exch5]

<u>Theorems with TF-conditionals by df. and DN only,</u>

T4-28 [((P ⊃ R) & (Q ⊃ S)) CONT ((P & Q) ⊃ (R & S))] [Praeclarum]
T4-30. [(P ⊃ Q) SYN ~ (P& ~ Q)]
T4-31. [(~ PvQ) SYN (P ⊃ Q)]
T4-32. [(P ⊃ Q) SYN (~ Q ⊃ ~ P)]
T4-33. [(∃x)(Px ⊃ Qx) SYN ((∀x)Px ⊃ (∃x)Qx)] ML*142
T4-34. [(∃x)(Px ⊃ Q) SYN ((∀x)Px ⊃ Q)] ML*162
T4-35. [(∀x)(Px ⊃ Q) SYN ((∃x)Px ⊃ Q)] ML*161
T4-36. [((∃x)Px ⊃ (∀x)Qx) CONT (∀x)(Px ⊃ Qx)] ML*148
T4-37. [(∀x)(Px ⊃ Qx) CONT ((∀x) Px ⊃ (∀x)Qx)] ML*101
T4-38. [(∀x)(P ⊃ Qx) SYN (P ⊃ (∀x)Qx)))] [Rule of Passage]
T4-39. [(∀x)(Px ⊃ Qx) CONT ((∃x)Px ⊃ (∃x)Qx)] ML*149
T4-40. [((∃x)Px ⊃ (∃x)Qx) CONT (∃x)(Px ⊃ Qx)] ML*150
T4-41. [((∀x)Px ⊃ (∀x)Qx) CONT (∃x)(Px ⊃ Qx)] ML*151

Appendix V—Rules and Theorems of Chapter 5

III. Definitions: All definitions of Chapter 4, plus

Df'Inc': 'Inc[P]' Syn_{df} '(((P Syn (Q& ~ R)) & (Q Cont R) [Df 'Inc']
v ((P Syn (Q&R)) & Inc(R))
v ((P Syn (QvR)) & Inc(Q) & Inc(R))'

Df'Taut': 'Taut[P]' Syn_{df} 'Inc[~ P]' [Df 'Taut']

V. Derived Transformation Principles: All transformation rules of Chapter 4, Plus

DR5-5f.[If (Inc(P) & Inc(~ P&Q)) then Inc(Q)] [Inc-Detachment]

DR5-5f′. [If (Taut(P) & Taut(P⊃ Q)) then Taut(Q)] [Taut-Detachment]

Derived Rules. 5.211. Principles Asserting that Syn and Cont preserve Inc and Taut:

DR5-5 [If (P Syn Q & Inc(R)) then Inc(R(Q//P)]

DR5-5′. [If (P Syn Q & Taut(R) then Taut(R(Q//P)]

Principles for Deriving Inc and TAUT-theorems, from Syn and Cont-theorems

DR5-5a. [If P Cont Q, then Inc(P & ~ Q)] DR5-5a′. [If P Cont Q, then Taut(P ⊃ Q)]

DR5-5b. [If P Syn Q, then Inc(P & ~ Q)] DR5-5b′. [If P Syn Q, then Taut(P ⊃ Q)]

DR5-5d. [If P Syn Q, then Inc((P& ~ Q)&(Q& ~ P))] DR5-d′. [If P Syn Q, then Taut(P ≡ Q)]

5.212. Principles from Df 'Inc', for Deriving Inc and TAUT-theorems with 'and' and 'or'.

MT5-21 If Inc[P] then Inc[P&Q] MT5-21b If Taut[P] then Taut[PvQ]

MT5-22. If Inc[Q v R] then Inc[Q] & Inc(R) MT5-22b If Taut[Q&R] then Taut[Q] and Taut[R]

MT5-23 If (Inc[P] & Inc[Q]) then Inc[PvQ] MT5-23b If (Taut[P] & Taut[Q]) then Taut[P&Q]

MT5-24 Inc[P] & Inc[Q] iff Inc[PvQ] MT5-24b Taut[P] & Taut[Q] iff Taut[P&Q]

Detachment Principles of Chapter 5 (This is TF-Modus Ponens, used in Quine, Rosser, Thomason, etc.)

DR5-25 If INC(P) and INC[~ P & Q)], then INC(Q). [INC-Det]

DR5-25b If TAUT(P) and TAUT[P ⊃ Q], then TAUT(Q) [TAUT-Det, Quine's *104]

Selected TAUT-Theorems:

T5-501 TAUT[P ⊃ (P & P)] (Rosser's Axiom 1}

T5-02. TAUT[P ⊃ (Q v P)] ("Addition")

T5-03. TAUT[P ⊃ (Q ⊃ P)] (Thomason's Axiom 1)

T5-04. TAUT[(~ P ⊃ ~ Q) ⊃ (Q ⊃ P)] (Thomason's Axiom 3)

T5-05. TAUT[(P ⊃ Q) ⊃ (~ (Q&R) ⊃ ~ (R&P))] (Rosser Axiom 3)

T5-06. TAUT[(P ⊃ Q) ⊃ ((Q ⊃ R) ⊃ (P ⊃ R))] ("Hyp Syllogism"}

T5-07. TAUT[((P ⊃ (Q ⊃ R) ⊃ ((P ⊃ Q) ⊃ (P ⊃ R)) (Thomason's Axiom 2)

T5-08. TAUT[(∀y)((∀x)Px ⊃ Py)]

T5-09. TAUT[(∀y)((∀x)Pxy ⊃ Pyy)] (Rosser's Axiom 6)

T5-10. TAUT[(∀x)(Px ≡ Qx) ⊃ ((∀x)Px ≡ (∀x)Qx))] (Quine's *116 in ML)

T5-136c. TAUT[((P&Q) ⊃ P) (Rosser's Axiom 2)

T5-333c. TAUT[(∀x)(Px ⊃ Pa)] (Thomason's Axiom 6)

T5-438c. TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)] (Thomason's Axiom 4)

T5-437c. TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))]
(Quine's *101,Rosser's Axiom 4, Thomason's Axiom 5)

5.213 Principles of INC and TAUT with Instantiation and Generalization

DR5-30. If INC[P < 1 >] then INC[Pa] DR5-30b. If TAUT[P < 1 >] then TAUT[Pa]

DR5-31. If INC[P<1>] then (∀x)INC[Px]
DR5-32. If INC[P<1>] then INC[(∃x)Px]]
DR5-33. If INC[P<1>] then (∃x)INC[Px]
DR5-34. If INC[P<1>] then INC[(∀x)Px]
DR5-35. If INC[Pa] then INC[P<1>]
DR5-36. If INC[Pa] then (∀x)INC[Px]
DR5-37. If INC[Pa] then INC (∃x)[Px]
DR5-38. If INC[Pa] then (∃x)INC[Px]
DR5-39. If INC[Pa] then INC[(∀x)Px]
DR5-40. If (∀x)INC[Px] then INC[P<1>]
DR5-41. If (∀x)INC[Px] then INC[Pa]
DR5-42. If (∀x)INC[Px] then INC[(∃x)Px]
DR5-43. If (∀x)INC[Px] then (∃x)INC[Px]
DR5-44. If (∀x)INC[Px] then INC[(∀x)Px]
DR5-45. If INC(∃x)[Px] then INC[P<1>]
DR5-46. If INC(∃x)[Px] then INC[Pa]
DR5-47. If INC[(∃x)Px] then (∀x)INC[Px]
DR5-48. If INC[(∃x)Px] then (∃x)INC[Px]
DR5-49. If INC[(∃x)Px] then INC[(∀x)Px]
DR5-50. If (∃x)INC[Px] then INC[(∀x)Px]
DR5-51. If INC[(∀x)Px] then (∃x)INC[Px]
DR5-52. INC[Pa] iff INC[P<1>]
DR5-53. (∀x)INC[Px]iff INC[P<1>]
DR5-54. INC[(∃x)Px] iff INC[P<1>]
DR5-55. INC[Pa] iff (∀x)INC[Px]
DR5-56. INC[Pa] iff INC(∃x)[Px]
DR5-57. INC[(∃x)Px] iff (∀x)INC[Px]
DR5-58. INC[(∀x)Px] iff (∃x)INC[Px]

DR5-31b. If TAUT[P<1>] then (∀x)TAUT[Px]
DR5-32b. If TAUT[P<1>] then TAUT[(∀x)Px]
DR5-33b. If TAUT[P<1>] then (∃x)TAUT[Px]
DR5-34b. If TAUT[P<1>] then TAUT[(∃x)Px]
DR5-35b. If TAUT[Pa] then TAUT[P<1>]
DR5-36b. If TAUT[Pa] then (∀x)TAUT[Px]
DR5-37b. If TAUT[Pa] then TAUT (∃x)[Px]
DR5-38b. If TAUT[Pa] then (∃x)TAUT[Px]
DR5-39b. If TAUT[Pa] then TAUT[(∃x)Px]
DR5-40b. If (∀x)TAUT[Px] then TAUT[P<1>]
DR5-41b. If (∀x)TAUT[Px] then TAUT[Pa]
DR5-42b. If (∀x)TAUT[Px] then TAUT[(∃x)Px]
DR5-43b. If (∀x)TAUT[Px] then (∃x)TAUT[Px]
DR5-44b. If (∀x)TAUT[Px] then TAUT[(∃x)Px]
DR5-45b. If TAUT(∀x)[Px] then TAUT[P<1>]
DR5-46b. If TAUT(∀x)[Px] then TAUT[Pa]
DR5-47b. If TAUT[(∀x)Px] then (∀x)TAUT[Px]
DR5-48b. If TAUT[(∀x)Px] then (∃x)TAUT[Px]
DR5-49b. If TAUT[(∀x)Px] then TAUT[(∃x)Px]
DR5-50b. If (∃x)TAUT[Px] then TAUT[(∃x)Px]
DR5-51b. If TAUT[(∀x)Px] then (∃x)TAUT[Px]
DR5-52b.TAUT[Pa] iff TAUT[P<1>]
DR5-53b. (∀x)TAUT[Px]iff TAUT[P<1>]
DR5-54b. TAUT[(∀x)Px] iff TAUT[P<1>]
DR5-55b. TAUT[Pa] iff (∀x)TAUT[Px]
DR5-56b. TAUT[Pa] iff TAUT[(∀x)Px]
DR5-57b. TAUT[(∀x)Px] iff (∀x)TAUT[Px]
DR5-58b. TAUT[(∃x)Px] iff (∃x)TAUT [Px]

Taut- and Inc-theorems at the base of M-logic: Thomason's Axioms

T5-03. (Thomason's AS1) TAUT[P ⊃ (Q ⊃ P)]
T5-07. (Thomason's AS2) TAUT[(P ⊃ (Q ⊃ R)) ⊃ ((P ⊃ Q) ⊃ (P>R)]
T5-04. (Thomason's AS3) TAUT[(~P ⊃~Q) ⊃ (Q ⊃ P)]
T5-438c. (Thomason's AS4) TAUT[(∀x)(P ⊃ Qx) ⊃ (P ⊃ (∀x)Qx)]
T5-437c. (Thomason's AS5) TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx))](Quine's *101)
T5-333c. (Thomason's AS6) TAUT[(∀x)Px ⊃ Pa] (Included in Quine's *102)

Rosser's Axioms

T5-333c. (Rosser's Axiom 1) TAUT[(P ⊃ (P& P)]
T5-136b. (Rosser's Axiom 2) TAUT[(P & Q) ⊃ P]
T5-05 (Rosser's Axiom 3) TAUT[(P ⊃ Q) ⊃ (~(Q&R) ⊃~(R&P))]
T5-437c. (Rosser's Axiom 4) TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)] (Quine's *101)
Rosser's Axiom 5, like Quine's*102, introduces vacuous quantifiers. A-logic uses DR5-32b instead.
T5-09. (Rosser's Axiom 6) TAUT[(∀y)((∀x)Rxy ⊃ Ryy] (In lieu of Quine's *103)

Quine's "Metatheorems"

T5-437c. (Quine's *101) TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)] [T4-37,DR5-13b]

T5-10. (Quine's *116) TAUT[(∀x)(Px ≡ Qx) ⊃ ((∀x)Px ≡ (∀x)Qx))] [T5-10,DR5-13b]
T5-315c (Quine's *119) TAUT[(∀x)(∀y)Pxy ⊃ (∀y)(∀x)Pxy] [T3-15,DR5-13b]
T5-424c. (Quine's *130) TAUT[~ (∀x)Px ≡ (∃x) ~ Px] [T4-24,DR1,DR5-15b]
T5-425c. (Quine's *131) TAUT[~ (∃x)Px ≡ (∀x) ~ Px] [T4-25,DR1,DR5-15b]
T5-426c. (Quine's *132) TAUT[$\sim (\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n> \equiv (Ex_1)\ldots(Ea_n) \sim P<x_1,\ldots,x_n>$] [T4-26,DR5-15b]
T5-427c. (Quine's *133) TAUT[$\sim (Ex_1)\ldots(Ex_n)P<x_1,\ldots,x_n> \equiv (\forall x_1)\ldots(\forall x_n) \sim P<x_1,\ldots,x_n>$] [T4-27,DR5-15b]
T5-336c. (Quine's *136) TAUT[(∀x)Px ⊃ (∃x)Px] [T3-36,DR5-13b]
T5-316c. (Quine's *138) TAUT[(∃x)(∃y)Pxy ≡ (∃y)(∃x)Pxy] [T3-16,DR5-15b]
T5-337c. (Quine's *139) TAUT[(∃y)(∀x)Pxy ⊃ (∀x)(∃y)Pxy] [T3-37,DR5-13b]
T5-313c. (Quine's *140) TAUT[(∀x)(Px & Qx) ≡ ((∀x)Px & (∀x)Qx] [T3-13,DR5-15b]
T5-314c. (Quine's *141) TAUT[(∃x)(Px v Qx) ≡ ((∃x)Px v (∃x)Qx)] [T3-14,DR5-15b]
T5-433c. (Quine's *142) TAUT[(∃x)(Px ⊃ Qx) ⊃ (∀x)Px ⊃ (∃x)Qx)] [T4-33,DR5-15b]
T5-338c. (Quine's *143) TAUT[((∀x)Px v (∀x)Qx) ⊃ (∀x)(Px v Qx)] [T3-38,DR5-13b]
T5-339c. (Quine's *144) TAUT[(∀x)(Px v Qx) ⊃ ((∃x)Px v (∀x)Qx))] [T3-39,DR5-13b]
T5-340c. (Quine's *145) TAUT[(∀x)(Px v Qx) ⊃ ((∀x)Px v (∃x)Qx))] [T3-40,DR5-13b]
T5-341c. (Quine's *146) TAUT[((∀x)Px v (∃x)Qx) ⊃ (∃x)(Px v Qx)] [T3-41,DR5-13b]
T5-342c. (Quine's *147) TAUT[((∃x)Px v (∀x)Qx) ⊃ (∃x)(Px v Qx)] [T3-42,DR5-13b]
T5-436c. (Quine's *148) TAUT[((∃x)Px ⊃ (∀x)Qx) ⊃ (∀x)(Px ⊃ Qx)] [T4-36,DR5-13b]
T5-439c. (Quine's *149) TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∃x)Px ⊃ (∃x)Qx)] [T4-39,DR5-13b]
T5-440c. (Quine's *150) TAUT[((∃x)Px ⊃ (∃x)Qx) ⊃ (∃x)(Px ⊃ Qx)] [T4-40,DR5-13b]
T5-441c. (Quine's *151) TAUT[((∀x)Px ⊃ (∀x)Qx) ⊃ (∃x)(Px ⊃ Qx)] [T4-41,DR5-13b]
T5-343c. (Quine's *152) TAUT[(∀x)(Px & Qx) ⊃ ((∃x)Px & (∀x)Qx)] [T3-43,DR5-13b]
T5-344c. (Quine's *153) TAUT[(∀x)(Px & Qx) ⊃ ((∀x)Px & (∃x)Qx)] [T3-44,DR5-13b]
T5-345c. (Quine's *154) TAUT[((∀x)Px & (∃x)Qx) ⊃ (∃x)(Px & Qx)] [T3-45,DR5-13b]
T5-346c. (Quine's *155) TAUT[((∃x)Px & (∀x)Qx) ⊃ (∃x)(Px & Qx)] [T3-46,DR5-13b]
T5-347c. (Quine's *156) TAUT[(∃x)(Px & Qx) ⊃ ((∃x)Px & (∃x)Qx)] [T3-47,DR5-13b]
T5-317c. (Quine's *157) TAUT[(∀x)(P&Qx) ≡ (P&(∀x)Qx)] [T3-17,DR5-15b]
T5-319c. (Quine's *158) TAUT[(∃x)(P&Qx) ≡ (P&(∃x)Qx)] [T3-19,DR5-15b]
T5-320c. (Quine's *159) TAUT[(∀x)(PvQx) ≡ (Pv(∀x)Qx)] [T3-20,DR5-15b]
T5-318c. (Quine's *160) TAUT[(∃x)(PvQx) ≡ (Pv(∃x)Qx)] [T3-18,DR5-15b]
T5-435c. (Quine's *161) TAUT[(∀x)(Px ⊃ Q) ≡ ((∃x)Px ⊃ Q)] [T4-35,DR5-15b]
T5-434c. (Quine's *162) TAUT[(∃x)(Px ⊃ Q) ≡ ((∀x)Px ⊃ Q)] [T4-34,DR5-15b]
<u>Alpha.Variance</u>, "If P and Q are alphabetic Variants, ⊨ Taut(P) iff Taut(Q)" [cf. Quine's ML*170 and *171]

Miscellaneous TAUT and INC-theorems

T5-136a TAUT[(P&Q) ⊃ P] [T1-36,DR5-13b]
T5-137a TAUT[(P&Q) ⊃ Q] [T1-37,DR5-13b]
T5-138a TAUT[(P&Q) ⊃ (PvQ)] [T1-38,DR5-13b]
T5-139a TAUT[(P&(QvR)) ⊃ ((P&Q)vR)] [T1-39,DR5-13b]
T5-333a. INC[(∀x)Px & ~ Pa] [T3-33,DR5-13a]
T5-333c TAUT[(∀x)Px ⊃ Pa] [T3-33,DR5-13b]
T5-335c TAUT[((∃y)(∀x)Rxy ⊃ (∃x)Rxx] [T3-35,DR5-13b]
T5-335c TAUT[((∃y)(∀x)Rxy ⊃ (∀x)(∃y)Rxy] [T3-37,DR5-13b]
T5-320f. TAUT[(∀x)(Px v Qx) ≡ (P v (∀x)Qx)] [T3-20,DR5-13b]
T5-437c TAUT[(∀x)(Px ⊃ Qx) ⊃ ((∀x)Px ⊃ (∀x)Qx)] [T4-37,DR5-13b]

Logically Valid Inference Schemata in Chapter 5

Rules:

DR5-6a. [If (P Cont Q) and not-Inc(P & Q)] then Valid:(P, **.:**Q)]

<u>Proof</u>: 1) (P Cont Q) and not-Inc(P & Q) [Assumption]
2) Valid (P, **.:**Q) [1), Df’‘Valid’]
3) [If (P Cont Q) and not-Inc(P & Q)] then Valid:(P, **.:**Q)] [1)-2), Cond.Pr]

DR5-6b.[If (P Syn Q) and not-Inc (P & Q) then Valid:(P, **.:**Q)]

<u>Proof</u>: 1) (P Syn Q) and not-Inc(P & Q) [Assumption]
2) Valid (P, **.:**Q) [1), Df’‘Valid’]
3) [If (P Syn Q) and not-Inc(P & Q)] then Valid:(P, **.:**Q)] [1)-2), Cond.Pr]

<u>From SYN-theorems in Chapter 1</u>:

T5-101b. VALID[P, **.:** (P&P)] [&-IDEM] [Ax.1-01, DR5-6b]
T5-102b. VALID[P, **.:** (PvP)] [v-IDEM] [Ax.1-02,DR5-6b]
T5-103b. VALID[(P&Q), **.:** (Q&P)] [&-COMM] [Ax.1-03,DR5-6b]
T5-104b. VALID[(PvQ), **.:** (QvP)] [v-COMM] [Ax.1-04,DR5-6b]
T5-105b. VALID[(P&(Q&R)), **.:** ((P&Q)&R)] [&-ASSOC] [Ax.1-05,DR5-6b]
T5-106b. VALID[(Pv(QvR)), **.:** ((PvQ)vR)] [v-ASSOC] [Ax.1-06,DR5-6b]
T5-107b. VALID[(Pv(Q&R)), **.:** ((PvQ)&(PvR))] [v&-DIST-1] [Ax.1-07,DR5-6b]
T5-108b. VALID[(P&(QvR)), **.:** ((P&Q)v(P&R))] [&v-DIST-1] [Ax.1-08,DR5-6b]
T5-111b. VALID[P, **.:** P] [T1-11,DR5-6b]
T5-112b. VALID[((P&Q) & (R&S)), **.:** ((P&R) & (Q&S))] [T1-12,DR5-6b]
T5-113b. VALID[((PvQ) v (RvS)), **.:** ((PvR) v (QvS))] [T1-13,DR5-6b]
T5-114b. VALID[(P & (Q&R)), **.:** ((P&Q) & (P&R))] [T1-14,DR5-6b]
T5-115b. VALID[(P v (QvR)), **.:** ((PvQ) v (PvR))] [T1-15,DR5-6b]
T5-116b. VALID[(Pv(P&Q)), **.:** (P&(PvQ))] [T1-16,DR5-6b]
T5-117b. VALID[(P&(PvQ)), **.:** (Pv(P&Q))] [T1-17,DR5-6b]
T5-118b. VALID[(P&(Q&(PvQ))), **.:** (P&Q)] [T1-18,DR5-6b]
T5-119b. VALID[(Pv(Qv(P&Q))), **.:** (PvQ)] [T1-19,DR5-6b]
T5-120b. VALID[(P&(Q&R)), **.:** (P&(Q&(R&(Pv(QvR)))))] [T1-20,DR5-6b]
T5-121b. VALID[(Pv(QvR)), **.:** (Pv(Qv(Rv(P&(Q&R)))))] [T1-21,DR5-6b]
T5-122b. VALID[(Pv(P&(Q&R))), **.:** (P&((PvQ)&((PvR)&(Pv(QvR)))))] [T1-22,DR5-6b]
T5-123b. VALID[(P&(Pv(QvR))), **.:** (Pv((P&Q)v((P&R)v(P&(Q&R)))))] [T1-23,DR5-6b]
T5-124b. VALID[(Pv(P&(Q&R))), **.:** (P&(Pv(QvR)))] [T1-24,DR5-6b]
T5-125b. VALID[(P&(Pv(QvR))), **.:** (Pv(P&(Q&R)))] [T1-25,DR5-6b]
T5-126b. VALID[(P&(PvQ)&(PvR)&(Pv(QvR))), **.:** (P&(Pv(QvR)))] [T1-26,DR5-6b]
T5-127b. VALID[(Pv(P&Q)v(P&R)v(P&(Q&R))), **.:** (Pv(P&(Q&R)))] [T1-27,DR5-6b]
T5-128b. VALID[((P&Q)v(R&S)), **.:** (((P&Q)v(R&S)) & (PvR))] [T1-28,DR5-6b]
T5-129b. VALID[((PvQ)&(RvS)), **.:** (((PvQ)&(RvS)) v (P&R))] [T1-29,DR5-6b]
T5-130b. VALID[((P&Q)&(RvS)), **.:** ((P&Q) & ((P&R)v(Q&S)))] [T1-30,DR5-6b]
T5-131b. VALID[((PvQ)v(R&S)), **.:** ((PvQ) v ((PvR)&(QvS)))] [T1-31,DR5-6b]
T5-132b. VALID[((PvQ)&(RvS)), **.:** (((PvQ)&(RvS)) & (PvRv(Q&S))] “Praeclarum” [T1-32,DR5-6b]
T5-133b. VALID[((P&Q)v(R&S)), **.:** (((P&Q)v(R&S)) v (P&R&(QvS))] [T1-33,DR5-6b]
T5-134b. VALID[((P&Q)v(R&S)), **.:** (((P&Q)v(R&S)) & (PvR) & (QvS))] [T1-34,DR5-6b]
T5-135b. VALID[((PvQ)&(RvS)), **.:** (((PvQ)&(RvS)) v (P&R) v (Q&S))] [T1-35,DR5-6b]

<u>From CONT-theorems in Chapter 1:</u>

T5-136a. VALID [(P&Q), **.:** P] [T1-36,DR5-6a]

T5-137a. VALID [(P&Q), **.:** Q] [T1-37,DR5-6a]

T5-138a. VALID [(P&Q), **.:** (PvQ)] [T1-38),DR5-6a]

T5-139a. VALID [(P&(QvR), **.:** ((P&Q)vR)] [T1-39),DR5-6a]

T5-122a(1) VALID[(Pv(P&(Q&R))), **.:** P] [T1-22c(1),DR5-6a]

T5-122a(1) VALID[(Pv(P&(Q&R))), **.:** (PvQ)] [T1-22c(2),DR5-6a]

T5-122a(2) VALID[(Pv(P&(Q&R))), **.:** (PvR)] [T1-22c(3),DR5-6a]

T5-122a(4) VALID[(Pv(P&(Q&R))), **.:** (Pv(QvR))] [T1-22c(4),DR5-6a]

T5-122a(1,2) VALID[(Pv(P&(Q&R))), **.:** (P&(PvQ))] [T1-22c(1,2),DR5-6a]

T5-122a(1,3) VALID[(Pv(P&(Q&R))), **.:** (P&(PvR))] [T1-22c(1,3),DR5-6a]

T5-122a(2,3) VALID[(Pv(P&(Q&R))), **.:** ((PvQ)&(PvR))] [T1-22c(2,3),DR5-6a]

T5-122a(2,4) VALID[(Pv(P&(Q&R))), **.:** ((PvQ)&(Pv(QvR)))] [T1-22c(2,4),DR5-6a]

T5-122a(3,4) VALID[(Pv(P&(Q&R))), **.:** ((PvR)&(Pv(QvR)))] [T1-22c(3,4),DR5-6a]

<u>From CHAPTER 3 SYN and CONT theorems with Quantifiers</u>

<u>From SYN-theorems in Chapter 3</u>:

Theorem		Cf. Quine's Metatheorems	
T5-311b. VALID[((∀x)$_n$ Px), **.:** (Pa$_1$ & P$_2$ &…& P$_n$)]			[T3-11,DR5-6b]
T5-312b. VALID[((∃x)$_n$ Px), **.:** (Pa$_1$ v P$_2$ v…v P$_n$)]			[T3-12,DR5-6b]
T5-313b. VALID[(∀x)(Px & Qx), **.:** ((∀x)Px & (∀x)Qx)]		ML*140	[T3-13,DR5-6b]
T5-314b. VALID[(∃x)(Px v Qx), **.:** ((∃x)Px v (∃x)Qx)]		ML*141	[T3-14,DR5-6b]
T5-315b. VALID[(∀x)(∀y)Rxy, **.:** (∀x)(∀y) Rxy]		ML*119	[T3-15,DR5-6b]
T5-316b. VALID[(∃x)(∃y)Rxy, **.:** (∃y)(∃x)Rxy]		ML*138	[T3-16,DR5-6b]
T5-317b. VALID[(∀x)(P & Qx), **.:** (P & (∀x)Qx)]	"Rules of Passage"	ML*157	[T3-17,DR5-6b]
T5-318b. VALID[(∃x)(P v Qx), **.:** (P v (∃x)Qx)]	"Rules of Passage"	ML*160	[T3-18,DR5-6b]
T5-319b. VALID[(∃x)(P & Qx), **.:** (P & (∃x)Qx)]	"Rules of Passage"	ML*158	[T3-19,DR5-6b]
T5-320b. VALID[(∀x)(P v Qx), **.:** (P v (∀x)Qx)]	"Rules of Passage"	ML*159	[T3-20,DR5-6b]
T5-321b. VALID[(∀x)Px, **.:** ((∀x)Px & (∃x)Px)]			[T3-21,DR5-6b]
T5-322b. VALID[(∃x)Px, **.:** ((∃x)Px v (∀x)Px)]			[T3-22,DR5-6b]
T5-323b. VALID[(∃x)(Px & Qx), **.:** ((∃x)(Px & Qx) & (∃x)Px)]			[T3-23,DR5-6b]
T5-324b. VALID[(∀x)(Px v Qx), **.:** ((∀x)(Px v Qx) v (∀x)Px)]			[T3-24,DR5-6b]
T5-325b. VALID[((∀x)Px & (∃x)Qx), **.:** ((∀x)Px & (∃x)(Px & Qx))]			[T3-25,DR5-6b]
T5-326b. VALID[((∃x)Px v (∀x)Qx), **.:** ((∃x)Px v (∀x)(Px v Qx))]			[T3-26,DR5-6b]
T5-327b. VALID[(∃y)(∀x)Rxy, **.:** ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]			[T3-27,DR5-6b]
T5-328b. VALID[(∀y)(∃x)Rxy, **.:** ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]			[T3-28,DR5-6b]
T5-329b. VALID[(∀x)(Px v Qx), **.:** ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx))]			[T3-29,DR5-6b]
T5-330b. VALID[(∃x)(Px & Qx), **.:** ((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx))]			[T3-30,DR5-6b]
T5-331b. VALID[(∀x)(∀y)Rxy, **.:** ((∀x)(∀y)Rxy & (∀x)Rxx)]			[T3-31,DR5-6b]
T5-332b. VALID[(∃x)(∀y)Rxy, **.:** ((∃x)(∀y)Rxy & (∃x)Rxx)]			[T3-32,DR5-6b]

<u>From CONT-theorems in Chapter 3:</u>

T5-333a. VALID [(∀x)Px, **.:** Pa]		[T3-33,DR5-6a]
T5-334a. VALID [(∀x)(∀y)Rxy, **.:** (∀x)Rxx]		[T3-34,DR5-6a]
T5-335a. VALID [(∃x)(∀y)Rxy, **.:** (∃x)Rxx]		[T3-35,DR5-6a]
T5-336a. VALID [(∀x)Px, **.:** (∃x)Px]	ML*136	[T3-36,DR5-6a]
T5-337a. VALID [(∃y)(∀x)Rxy, **.:** (∀x)(∃y)Rxy]	ML*139	[T3-37,DR5-6a]
T5-338a. VALID [((∀x)Px v (∀x)Qx), **.:** (∀x)(Px v Qx)]	ML*143	[T3-38,DR5-6a]
T5-339a. VALID [(∀x)(Px v Qx), **.:** ((∃x)Px v (∀x)Qx))]	ML*144	[T3-39,DR5-6a]
T5-340a. VALID [(∀x)(Px v Qx), **.:** ((∀x)Px v (∃x)Qx))]	ML*145	[T3-40,DR5-6a]

T5-341a. VALID [((∀x)Px v (∃x)Qx), .: (∃x)(Px v Qx)]		ML*146	[T3-41,DR5-6a]
T5-342a. VALID [((∃x)Px v (∀x)Qx), .: (∃x)(Px v Qx)]		ML*147	[T3-42,DR5-6a]
T5-343a. VALID [(∀x)(Px & Qx), .: ((∃x)Px & (∀x)Qx)]		ML*152	[T3-43,DR5-6a]
T5-344a. VALID [(∀x)(Px & Qx), .: ((∀x)Px & (∃x)Qx)]		ML*153	[T3-44,DR5-6a]
T5-345a. VALID [((∀x)Px & (∃x)Qx), .: (∃x)(Px & Qx)]		ML*154	[T3-45,DR5-6a]
T5-346a. VALID [((∃x)Px & (∀x)Qx), .: (∃x)(Px & Qx)]		ML*155	[T3-46,DR5-6a]
T5-347a. VALID [(∃x)(Px & Qx), .: ((∃x)Px & (∃x)Qx)]		ML*156	[T3-47,DR5-6a]

From SYN-theorems in Chapter 4:

T5-411b. VALID [(P&Q), .: ~(~Pv~Q)]		[DeM2]	[T4-11,DR5-6b]
T5-412b. VALID [(PvQ), .: ~(~P&~Q)]	[Df ‘v’]	[DeM1]	[T4-12,DR5-6b]
T5-413b. VALID [(P&~Q), .: ~(~PvQ)]		[DeM3]	[T4-13,DR5-6b]
T5-414b. VALID [(Pv~Q), .: ~(~P&Q)]		[DeM4]	[T4-14,DR5-6b]
T5-415b. VALID [(~P&Q), .: ~(Pv~Q)]		[DeM5]	[T4-15,DR5-6b]
T5-416b. VALID [(~PvQ), .: ~(P&~Q)]		[DeM6]	[T4-16,DR5-6b]
T5-417b. VALID [(~P&~Q), .: ~(PvQ)]		[DeM7]	[T4-17,DR5-6b]
T5-418b. VALID [(~Pv~Q), .: ~(P&Q)]		[DeM8]	[T4-18,DR5-6b]
T5-419b. VALID [P, .: (PvP)]		[v-IDEM]	[T4-19,DR5-6b]
T5-420b. VALID [(PvQ), .: (QvP)]		[v-COMM]	[T4-20,DR5-6b]
T5-421b. VALID [(Pv(QvR)), .: ((PvQ)vR)		[v-ASSOC]	[T4-21,DR5-6b]
T5-422b. VALID [(P&(QvR)), .: ((P&Q)v(P&R))]		[v&-DIST]	[T4-22,DR5-6b]
T5-424b. VALID [(∃x)~Px, .: ~(∀x)Px]	[Q-Exch2]	ML*130	[T4-24,DR5-6b]
T5-425b. VALID [(∀x)~Px, .: ~(∃x)Px]	[Q-Exch3]	ML*131	[T4-25,DR5-6b]
T5-426b. VALID $[(Ex_1)\ldots(Ea_n){\sim}P<x_1,\ldots,x_n>, .: {\sim}(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n>]$	[Q-Exch4]	ML*132	[T4-26,DR5-6a]
T5-427b. VALID $[(\forall x_1)\ldots(\forall x_n){\sim}P<x_1,\ldots,x_n>, .: {\sim}(Ex_1)\ldots(Ex_n)P<x_1,\ldots,x_n>]$	[Q-Exch5]	ML*133	[T4-27,DR5-6b]

From CHAPTER 4, SYN- and CONT-theorems (with TF-conditionals).

T5-430b. VALID [(P ⊃ Q), .: ~(P&~Q)]	[Df ‘⊃’]		[T4-30,DR5-6b]
T5-431b. VALID [(~PvQ), .: (P ⊃ Q)]			[T4-31,DR5-6b]
T5-432b. VALID [(P ⊃ Q), .: (~Q ⊃ ~ P)]			[T4-32,DR5-6b]
T5-433b. VALID [(∃x)(Px ⊃ Qx), .: ((∀x)Px ⊃ (∃x)Qx)]		ML*142	[T4-33,DR5-6b]
T5-434b. VALID [(∃x)(Px ⊃ Q), .: ((∀x)Px ⊃ Q)]	} “Rules of	ML*162	[T4-34,DR5-6b]
T5-435b. VALID [(∀x)(Px ⊃ Q), .: ((∃x)Px ⊃ Q)]	Passage”	ML*161	[T4-35,DR5-6b]
T5-436a. VALID [((∃x)Px ⊃ (∀x)Qx), .: (∀x)(Px ⊃ Qx)]		ML*148	[T4-36,DR5-6a]
T5-437a. VALID [(∀x)(Px ⊃ Qx), .: ((∀x)Px ⊃ (∀x)Qx)]		ML*101	[T4-37,DR5-6a]
T5-438a. VALID [(∀x)(P ⊃ Qx), .: (P ⊃ (∀x)Qx)]		ML*101	[T4-37,DR5-6a]
T5-439a. VALID [(∀x)(Px ⊃ Qx), .: ((∃x)Px ⊃ (∃x)Qx)]		ML*149	[T4-39,DR5-6a]
T5-440a. VALID [((∃x)Px ⊃ (∃x)Qx), .: (∃x)(Px ⊃ Qx)]		ML*150	[T4-40,DR5-6a]
T5-441a. VALID [((∀x)Px ⊃ (∀x)Qx), .: (∃x)(Px ⊃ Qx)]		ML*151	[T4-41,DR5-6a]

From CHAPTER 5, CONT-theorem

T5-510. VALID [(∀x)(Px ≡ Qx), .: ((∀x)Px ≡ (∀x)Qx)))]		ML*116	[T5-10,DR5-6a]

Appendix VI—Principles and Theorems of Chapter 6

The full system of **Purely Formal Analytic Logic**

1. Primitives:

Grouping devices	), (,], [, >, <		
Predicate letters:	$P_1, P_2, \ldots, P_n$.	[Abbr.'P','Q','R']	PL
Argument Place holders:	1, 2, 3,…		APH
Individual Constants:	$a_1, a_2, \ldots a_n$,	[Abbr.'a', 'b', 'c']	IC
Individual variables:	$x_1, x_2, \ldots x_n$,	[Abbr.'x', 'y', 'z']	IV
Predicate operators:	&, ~, ⇒		
Quantifier:	$(\forall x_i)$		
Primitive (2nd level) Predicate of Logic:	Syn		

2. Formation Rules

FR1. $[P_i]$ is a wff
FR2. If P and Q are wffs, [(P&Q)] is a wff.
FR3. If P is a wff, [~P] is a wff.
FR4. If P and Q are wffs, [(P ⇒ Q)] is a wff.
FR5. If P_i is a wff and each t_j $(1 \leq j \leq k)$ is an APH or a IC, then $P_i < t_1, \ldots, t_k >$ is a wff
FR6. If $P_i < 1 >$ is a wff, then $[(\forall x_j) P_i x_j]$ is a wff.

3. Abbreviations, definitions

Predicate Operators:

Df6-1. $[(P \,\&\, Q \,\&\, R)\ SYN_{df}\ (P \,\&\, (Q \,\&\, R))]$
Df6-2. $[(\forall_k x)Px\ SYN_{df}\ (Pa_1 \,\&\, Pa_2 \,\&\, \ldots \,\&\, Pa_k)]$ [Df 'V']
Df6-3. $[(P \vee Q)\ SYN_{df} \sim(\sim P \,\&\, \sim Q)]$ [Df 'v',DeM1]
Df6-4. $[(P \supset Q)\ SYN_{df} \sim(P \,\&\, \sim Q)]$ [Df '⊃']
Df6-5. $[(P \equiv Q)\ SYN_{df}\ ((P \supset Q)\&(Q \supset P))]$ [Df '≡']
Df6-6. $[(\exists x)Px\ SYN_{df} \sim(\forall x) \sim Px]$ [Df '(∃x)']
Df6-7. $[(P \Leftrightarrow Q)\ SYN_{df}\ ((P \Rightarrow Q)\&(Q \Rightarrow P))]$ [Df '⇔']

Logical Predicates

Df 'Cont'. $[(P\ Cont\ Q)\ Syn_{df}\ (P\ Syn\ (P\&Q))]$
Df 'Inc'. $[\text{'Inc(P)'}\ Syn_{df}$ '((P Syn (Q&~R) & (Q Cont R))
v (P Syn (Q&R) & Inc(R))
v (P Syn (QvR) & Inc(Q) & Inc(R))
v (P Syn (Q ⇒ R) & Inc(Q&R))]'
Df 'Taut'. $[Taut(P)\ Syn_{df}\ Inc(\sim P)]$
Df 'Valid'. $[Valid(P, .:\ Q)\ Syn_{df}\ ((P\ Cont\ Q) \,\&\, \text{not-}Inc(P\&Q))]$

4. Axioms

Ax.6-1. $\models$ [P Syn (P&P)] [&-IDEM1]
Ax.6-2. $\models$ [(P&Q) Syn (Q&P)] [&-COMM]
Ax.6-3. $\models$ [(P&(Q&R)) Syn ((P&Q)&R)] [&-ASSOC1]
Ax.6-4. $\models$ [(P&(QvR)) Syn ((P&Q)v(P&R))] [&v-DIST1]
Ax.6-5. $\models$ [P Syn ~~P] [DN]
Ax 6-6. $\models$ [(P & (P ⇒ Q)) Syn (P & (P ⇒ Q) & Q)] [MP]

5. Principles of Inference

R6-1. If $\models$ P, and Q is a component of P, and $\models$ [Q Syn R] then $\models$ P(Q//R) SynSUB]

R6-2. If $\models$ R and $P_i<t_1,\ldots,t_n>$ occurs in R,
and Q is a h-adic wff, where h > n,
and Q has an occurrence of each numeral 1 to n,
and no individual variable in Q occurs in R,
then $\models$ $[R(P_i<t_1,\ldots,t_n>/Q)$ [U-SUB]

R6-3. If $\models$ $P<t_1,\ldots,t_n>$ then $\models$ $P<t_1,\ldots,t_n>(t_i/a_j)$ [INST]

R6-6. Valid[P $\Rightarrow$ Q] if and only if Valid[P, **.:**Q] [VC\VI]

Selected SYN and CONT-theorems in Chapter 6:

T6-11. [(P & (P $\Rightarrow$ Q)) SYN ((P&Q) & (P $\Rightarrow$ Q))] [Ax.6, &-ORD]
T6-12. [(P & (P $\Rightarrow$ Q)) CONT (P&Q)] [T6-06,Df 'CONT']
T6-13. [((P $\Rightarrow$ Q) & P) CONT Q] [T6-06,Df 'CONT']
T6-14. [(P $\Leftrightarrow$ Q) CONT (Q $\Rightarrow$ P)] [Df '$\Leftrightarrow$',Df 'CONT']
T6-15. [(P $\Rightarrow$ Q) CONT (P $\Rightarrow$ Q)] [T1-11,U-SUB]
T6-16. [(P $\Leftrightarrow$ Q) CONT (P $\Rightarrow$ Q)] [T1-11,Df '$\Leftrightarrow$',Df 'CONT',R1]
T6-20. [(($\forall$x)((Px $\Rightarrow$ Qx) & Px) CONT ($\forall$x)Qx)]
T6-21. [(($\forall$x)((Px $\Rightarrow$ Qx) & ($\forall$x)Px) CONT ($\forall$x)Qx)]

Selected INC and TAUT-Theorems in Chapter 6

T6-30. INC[(P $\Rightarrow$ ~P)]
T6-31. TAUT[~(P $\Rightarrow$ ~P)]
T6-32. INC((P&Q) $\Rightarrow$ ~P)]
T6-33. TAUT[~((P&Q) $\Rightarrow$ ~P)]
T6-34. TAUT[~((P&~P) $\Rightarrow$ Q]

Derived Principles of Inference with C-conditionals, Expressed in A-logic:
To derive new kinds of Inc- and Taut-theorems

DR6-5a. If [P CONT Q] then INC(P $\Rightarrow$ ~Q)
DR6-5b. If [P CONT Q] then TAUT[~(P $\Rightarrow$ ~Q)]
DR6-5c. If TAUT[P $\supset$ Q] then INC[P $\Rightarrow$ ~Q]
DR6-5d. If TAUT[P $\supset$ Q] then TAUT[~(P $\Rightarrow$ ~Q)]

To yield valid conditionals, from the definition of validity with VC\VI:

DR6-6a. [If [P CONT Q and not-INC (P&Q)] then VALID[P $\Rightarrow$ Q]
DR6-6b. If [P SYN Q and not-INC (P&Q)] then VALID[P $\Rightarrow$ Q]
DR6-6c. If [P SYN Q and not-INC (P&Q)] then VALID[Q $\Rightarrow$ P]
DR6-6d. If [(P Syn Q) & not-Inc(P & Q)], then Valid[P $\Leftrightarrow$ Q]
DR6-6e. If Valid[P $\Rightarrow$ Q] then Valid[P, **.:** Q]
DR6-6f. If Valid[P, **.:** Q] then Valid[P $\Rightarrow$ Q]
DR6-10. If A is a consistent ordered set of wffs, $<A_1, A_2,\ldots,A_n>$ $(1 \leq i \leq n)$
and for each A_i, $(i>1)$ either $\models$ A_i
or $(g,h<i)$ and A_h is $[A_g \Rightarrow A_i]$ and $\models$ Valid$[A_g \Rightarrow A_i]$
then Valid$(A_1 \Rightarrow A_n)$ "Conditional Proof"

VALIDITY THEOREMS:

From SYN-theorems in Chapter 1:

T6-101. VALID[P <=> (P&P)]	[&-IDEM]	[Ax.1-01, DR6-6d]
T6-102. VALID[P <=> (PvP)]	[v-IDEM]	[Ax.1-02, DR6-6d]
T6-103. VALID[(P&Q) <=> (Q&P)]	[&-COMM]	[Ax.1-03, DR6-6d]
T6-104. VALID[(PvQ) <=> (QvP)]	[v-COMM]	[Ax.1-04, DR6-6d]
T6-105. VALID[(P&(Q&R)) <=> ((P&Q)&R)]	[&-ASSOC]	[Ax.1-05, DR6-6d]
T6-106. VALID[(Pv(QvR)) <=> ((PvQ)vR)]	[v-ASSOC]	[Ax.1-06, DR6-6d]
T6-107. VALID[(Pv(Q&R)) <=> ((PvQ)&(PvR))]	[v&-DIST-1]	[Ax.1-07, DR6-6d]
T6-108. VALID[(P&(QvR)) <=> ((P&Q)v(P&R))]	[&v-DIST-1]	[Ax.1-08, DR6-6d]
T6-111. VALID[P <=> P]		[T1-11,DR6-6d]
T6-112. VALID[((P&Q) & (R&S)) <=> ((P&R) & (Q&S))]		[T1-12,DR6-6d]
T6-113. VALID[((PvQ) v (RvS)) <=> ((PvR) v (QvS))]		[T1-13,DR6-6d]
T6-114. VALID[(P & (Q&R)) <=> ((P&Q) & (P&R))]		[T1-14,DR6-6d]
T6-115. VALID[(P v (QvR)) <=> ((PvQ) v (PvR))]		[T1-15,DR6-6d]
T6-116. VALID[(Pv(P&Q)) <=> (P&(PvQ))]		[T1-16,DR6-6d]
T6-117. VALID[(P&(PvQ)) <=> (Pv(P&Q))]		[T1-17,DR6-6d]
T6-118. VALID[(P&(Q&(PvQ))) <=> (P&Q)]		[T1-18,DR6-6d]
T6-119. VALID[(Pv(Qv(P&Q))) <=> (PvQ)]		[T1-19,DR6-6d]
T6-120. VALID[(P&(Q&R)) <=> (P&(Q&(R&(Pv(QvR)))))]		[T1-20,DR6-6d]
T6-121. VALID[(Pv(QvR)) <=> (Pv(Qv(Rv(P&(Q&R)))))]		[T1-21,DR6-6d]
T6-122. VALID[(Pv(P&(Q&R))) <=> (P&((PvQ)&((PvR)&(Pv(QvR)))))]		[T1-22,DR6-6d]
T6-123. VALID[(P&(Pv(QvR))) <=> (Pv((P&Q)v((P&R)v(P&(Q&R)))))]		[T1-23,DR6-6d]
T6-124. VALID[(Pv(P&(Q&R))) <=> (P&(Pv(QvR)))]		[T1-24,DR6-6d]
T6-125. VALID[(P&(Pv(QvR))) <=> (Pv(P&(Q&R)))]		[T1-25,DR6-6d]
T6-126. VALID[(P&(PvQ)&(PvR)&(Pv(QvR))) <=> (P&(Pv(QvR)))]		[T1-26,DR6-6d]
T6-127. VALID[(Pv(P&Q)v(P&R)v(P&(Q&R))) <=> (Pv(P&(Q&R)))]		[T1-27,DR6-6d]
T6-128. VALID[((P&Q)v(R&S)) <=> (((P&Q)v(R&S)) & (PvR))]		[T1-28,DR6-6d]
T6-129. VALID[((PvQ)&(RvS)) <=> (((PvQ)&(RvS)) v (P&R))]		[T1-29,DR6-6d]
T6-130. VALID[((P&Q)&(RvS)) <=> ((P&Q) & ((P&R)v(Q&S)))]		[T1-30,DR6-6d]
T6-131. VALID[((PvQ)v(R&S)) <=> ((PvQ) v ((PvR)&(QvS)))]		[T1-31,DR6-6d]
T6-132. VALID[((PvQ)&(RvS)) <=> (((PvQ)&(RvS)) & (PvRv(Q&S))]	"Praeclarum"	[T1-32,DR6-6d]
T6-133. VALID[((P&Q)v(R&S)) <=> (((P&Q)v(R&S)) v (P&R&(QvS))]		[T1-33,DR6-6d]
T6-134. VALID[((P&Q)v(R&S)) <=> (((P&Q)v(R&S)) & (PvR) & (QvS))]		[T1-34,DR6-6d]
T6-135. VALID[((PvQ)&(RvS)) <=> (((PvQ)&(RvS)) v (P&R) v (Q&S))]		[T1-35,DR6-6d]

From CONT-theorems in Chapter 1:

T6-136. VALID [(P&Q) => P]	[T1-36,DR6-6a]
T6-137. VALID [(P&Q) => Q]	[T1-37,DR6-6a]
T6-138. VALID [(P&Q) => (PvQ)]	[T1-38),DR6-6a]
T6-139. VALID [(P&(QvR)) => ((P&Q)vR)]	[T1-39),DR6-6a]
T6-122c(1) VALID[(Pv(P&(Q&R))) => P]	[T1-22c(1),DR6-6a]
T6-122c(1) VALID[(Pv(P&(Q&R))) => (PvQ)]	[T1-22c(2),DR6-6a]
T6-122c(2) VALID[(Pv(P&(Q&R))) => (PvR)]	[T1-22c(3),DR6-6a]
T6-122c(4) VALID[(Pv(P&(Q&R))) => (Pv(QvR))]	[T1-22c(4),DR6-6a]
T6-122c(1,2) VALID[(Pv(P&(Q&R))) => (P&(PvQ))]	[T1-22c(1,2),DR6-6a]
T6-122c(1,3) VALID[(Pv(P&(Q&R))) => (P&(PvR))]	[T1-22c(1,3),DR6-6a]
T6-122c(2,3) VALID[(Pv(P&(Q&R))) => ((PvQ)&(PvR))]	[T1-22c(2,3),DR6-6a]

T6-122c(2,4) VALID[(Pv(P&(Q&R))) => ((PvQ)&(Pv(QvR)))] [T1-22c(2,4),DR6-6a]
T6-122c(3,4) VALID[(Pv(P&(Q&R))) => ((PvR)&(Pv(QvR)))] [T1-22c(3,4),DR6-6a]

<u>From CHAPTER 3 SYN and CONT theorems with Quantifiers</u>

<u>From SYN-theorems in Chapter 3</u>: Cf. Quine's Metatheorems

T6-311. VALID[$((\forall_n x)\ Px) \Leftrightarrow (Pa_1\ \&\ P_2\ \&\ldots\&\ P_n)$] [T3-11,DR6-6d]
T6-312. VALID[$((\exists_n x\ )\ Px) \Leftrightarrow (Pa_1\ v\ P_2\ v\ldots v\ P_n)$] | [T3-12,DR6-6d]
T6-313. VALID[(∀x)(Px & Qx) <=> ((∀x)Px & (∀x)Qx)] ML*140 [T3-13,DR6-6d]
T6-314. VALID[(∃x)(Px v Qx) <=> ((∃x)Px v (∃x)Qx)] ML*141 [T3-14,DR6-6d]
T6-315. VALID[(∀x)(∀y)Rxy <=> (∀y)(∀x)Rxy] ML*119 [T3-15,DR6-6d]
T6-316. VALID[(∃x)(∃y)Rxy <=> (∃y)(∃x)Rxy] ML*138 [T3-16,DR6-6d]
T6-317. VALID[(∀x)(P & Qx) <=> (P & (∀x)Qx) ML*157 [T3-17,DR6-6d]
T6-318. VALID[(∃x)(P v Qx) <=> (P v (∃x)Qx)] "Rules of Passage" ML*160 [T3-18,DR6-6d]
T6-319. VALID[(∃x)(P & Qx) <=> (P & (∃x)Qx)] ML*158 [T3-19,DR6-6d]
T6-320. VALID[(∀x)(P v Qx) <=> (P v (∀x)Qx)] ML*159 [T3-20,DR6-6d]
T6-321. VALID[(∀x)Px <=> ((∀x)Px & (∃x)Px)] [T3-21,DR6-6d]
T6-322. VALID[(∃x)Px <=> ((∃x)Px v (∀x)Px)] [T3-22,DR6-6d]
T6-323. VALID[(∃x)(Px & Qx) <=> ((∃x)(Px & Qx) & (∃x)Px)] [T3-23,DR6-6d]
T6-324. VALID[(∀x)(Px v Qx) <=> ((∀x)(Px v Qx) v (∀x)Px)] [T3-24,DR6-6d]
T6-325. VALID[((∀x)Px & (∃x)Qx) <=> ((∀x)Px & (∃x)(Px & Qx))] [T3-25,DR6-6d]
T6-326. VALID[((∃x)Px v (∀x)Qx) <=> ((∃x)Px v (∀x)(Px v Qx))] [T3-26,DR6-6d]
T6-327. VALID[(∃y)(∀x)Rxy <=> ((∃y)(∀x)Rxy & (∀x)(∃y)Rxy)] [T3-27,DR6-6d]
T6-328. VALID[(∀y)(∃x)Rxy <=> ((∀x)(∃y)Rxy v (∃y)(∀x)Rxy)] [T3-28,DR6-6d]
T6-329. VALID[(∀x)(Px v Qx) <=> ((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx))] [T3-29,DR6-6d]
T6-330. VALID[(∃x)(Px & Qx) <=> ((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx))] [T3-30,DR6-6d]
T6-331. VALID[(∀x)(∀y)Rxy <=> ((∀x)(∀y)Rxy & (∀x)Rxx)] [T3-31,DR6-6d]
T6-332. VALID[(∃x)(∀y)Rxy <=> ((∃x)(∀y)Rxy & (∃x)Rxx)] [T3-32,DR6-6d]

<u>From CONT-theorems in Chapter 3</u>:

T6-333. VALID [(∀x)Px => Pa] [T3-33,DR6-6a]
T6-334. VALID [(∀x)(∀y)Rxy => (∀x)Rxx] [T3-34,DR6-6a]
T6-335. VALID [(∃x)(∀y)Rxy => (∃x)Rxx] [T3-35,DR6-6a]
T6-336. VALID [(∀x)Px => (∃x)Px] ML*136 [T3-36,DR6-6a]
T6-337. VALID [(∃y)(∀x)Rxy => (∀x)(∃y)Rxy] ML*139 [T3-37,DR6-6a]
T6-338. VALID [((∀x)Px v (∀x)Qx) => (∀x)(Px v Qx)] ML*143 [T3-38,DR6-6a]
T6-339. VALID [(∀x)(Px v Qx) => ((∃x)Px v (∀x)Qx)] ML*144 [T3-39,DR6-6a]
T6-340. VALID [(∀x)(Px v Qx) => ((∀x)Px v (∃x)Qx)] ML*145 [T3-40,DR6-6a]
T6-341. VALID [((∀x)Px v (∃x)Qx) => (∃x)(Px v Qx)] ML*146 [T3-41,DR6-6a]
T6-342. VALID [((∃x)Px v (∀x)Qx) => (∃x)(Px v Qx)] ML*147 [T3-42,DR6-6a]
T6-343. VALID [(∀x)(Px & Qx) => ((∃x)Px & (∀x)Qx)] ML*152 [T3-43,DR6-6a]
T6-344. VALID [(∀x)(Px & Qx) => ((∀x)Px & (∃x)Qx)] ML*153 [T3-44,DR6-6a]
T6-345. VALID [((∀x)Px & (∃x)Qx) => (∃x)(Px & Qx)] ML*154 [T3-45,DR6-6a]
T6-346. VALID [((∃x)Px & (∀x)Qx) => (∃x)(Px & Qx)] ML*155 [T3-46,DR6-6a]
T6-347. VALID [(∃x)(Px & Qx) => ((∃x)Px & (∃x)Qx)] ML*156 [T3-47,DR6-6a]

<u>From SYN-theorems in Chapter 4</u>:

T6-411. VALID [(P&Q) <=> ~(~Pv~Q)] [DeM2] [T4-11,DR6-6d]
T6-412. VALID [(PvQ) <=> ~(~P&~Q)] [From Df 'v'] [DeM1] [T4-12,DR6-6d]
T6-413. VALID [(P&~Q) <=> ~(~PvQ)] [DeM3] [T4-13,DR6-6d]
T6-414. VALID [(Pv~Q) <=> ~(~P&Q)] [DeM4] [T4-14,DR6-6d]

T6-415. VALID $[(\sim P\&Q) \Leftrightarrow \sim(Pv\sim Q)]$	[DeM5]		[T4-15,DR6-6d]
T6-416. VALID $[(\sim PvQ) \Leftrightarrow \sim(P\&\sim Q)]$	[DeM6]		[T4-16,DR6-6d]
T6-417. VALID $[(\sim P\&\sim Q) \Leftrightarrow \sim(PvQ)]$	[DeM7]		[T4-17,DR6-6d]
T6-418. VALID $[(\sim Pv\sim Q) \Leftrightarrow \sim(P\&Q)]$	[DeM8]		[T4-18,DR6-6d]
T6-419. VALID $[P \Leftrightarrow (PvP)]$	[v-IDEM]		[T4-19,DR6-6d]
T6-420. VALID $[(PvQ) \Leftrightarrow (QvP)]$	[v-COMM]		[T4-20,DR6-6d]
T6-421. VALID $[(Pv(QvR)) \Leftrightarrow ((PvQ)vR)$	[v-ASSOC]		[T4-21,DR6-6d]
T6-422. VALID $[(P\&(QvR)) \Leftrightarrow ((P\&Q)v(P\&R))]$	[v&-DIST]		[T4-22,DR6-6d]
T6-424. VALID $[(\exists x)\sim Px \Leftrightarrow \sim(\forall x)Px]$	[Q-Exch2]	ML*130	[T4-24,DR6-6d]
T6-425. VALID $[(\forall x)\sim Px \Leftrightarrow \sim(\exists x)Px]$	[Q-Exch3]	ML*131	[T4-25,DR6-6d]
T6-426. VALID $[(\exists x_1)\ldots(Ea_n)\sim P<x_1,\ldots,x_n> \Leftrightarrow \sim(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n>]$	[Q-Exch4]	ML*132	[T4-26,DR6-6d]
T6-427. VALID $[(\forall x_1)\ldots(\forall x_n)\sim P<x_1,\ldots,x_n> \Leftrightarrow \sim(\exists x_1)\ldots(\exists x_n)P<x_1,\ldots,x_n>]$	[Q-Exch5]	ML*133	[T4-27,DR6-6d]

<u>From CHAPTER 4, SYN- and CONT-theorems (with TF-conditionals).</u>

T6-430. VALID $[(P \supset Q) \Leftrightarrow -(P\&\sim Q)]$	[Df '⊃']		[T4-30,DR6-6d]
T6-431. VALID $[(\sim PvQ) \Leftrightarrow (P \supset Q)]$			[T4-31,DR6-6d]
T6-432. VALID $[(P \supset Q) \Leftrightarrow (\sim Q \supset \sim P)]$			[T4-32,DR6-6d]
T6-433. VALID $[(\exists x)(Px \supset Qx) \Leftrightarrow ((\forall x)Px \supset (\exists x)Qx)]$		ML*142	[T4-33,DR6-6d]
T6-434. VALID $[(\exists x)(Px \supset Q) \Leftrightarrow ((\forall x)Px \supset Q)]$	“Rules of	ML*162	[T4-34,DR6-6d]
T6-435. VALID $[(\forall x)(Px \supset Q) \Leftrightarrow ((\exists x)Px \supset Q)]$	Passage”	ML*161	[T4-35,DR6-6d]
T6-436. VALID $[((\exists x)Px \supset (\forall x)Qx) \Rightarrow (\forall x)(Px \supset Qx)]$		ML*148	[T4-36,DR6-6a]
T6-437. VALID $[(\forall x)(Px \supset Qx) \Rightarrow ((\forall x)Px \supset (\forall x)Qx)]$		ML*101	[T4-37,DR6-6a]
T6-438. VALID $[(\forall x)(Px \equiv Qx) \Rightarrow ((\forall x)Px \equiv (\forall x)Qx)]$		ML*116	[T4-38,DR6-6d]
T6-439. VALID $[(\forall x)(Px \supset Qx) \Rightarrow ((\exists x)Px \supset (\exists x)Qx)]$		ML*149	[T4-39,DR6-6a]
T6-440. VALID $[((\exists x)Px \supset (\exists x)Qx) \Rightarrow (\exists x)(Px \supset Qx)]$		ML*150	[T4-40,DR6-6a]
T6-441. VALID $[((\forall x)Px \supset (\forall x)Qx) \Rightarrow (\exists x)(Px \supset Qx)]$		ML*151	[T4-41,DR6-6a]

<u>From SYN- and CONT-theorems in Chapter 6.</u>

T6-611. VALID$[(P\&(P \Rightarrow Q)) \Leftrightarrow ((P\&Q)\&(P \Rightarrow Q))]$			[T6-11,DR6-6]
T6-612. VALID$[(P \& (P \Rightarrow Q)) \Rightarrow (P\&Q)]$			[T6-12,DR6-6a]
T6-613. VALID$[((P \Rightarrow Q) \& P) \Rightarrow Q]$'MP', “Modus Ponens”		ML*104	[T6-13,DR6-6a]
T6-614. VALID$[(P \Leftrightarrow Q) \Rightarrow (Q \Rightarrow P)]$			[T6-14,DR6-6a]
T6-615. VALID$[(P \Rightarrow Q) \Rightarrow (P \Rightarrow Q)]$			[T6-15,DR6-6a]
T6-616. VALID$[(P \Leftrightarrow Q) \Rightarrow (P \Rightarrow Q)]$			[T6-16,DR6-6a]
T6-620. VALID$[(\forall x)((Px \Rightarrow Qx) \& Px) \Rightarrow (\forall x)Qx)]$			[T6-20,DR6-6a]
T6-621. VALID$[((\forall x)(Px \Rightarrow Qx) \& (\forall x)Px) \Rightarrow (\forall x)Qx]$			[T6-21,DR6-6a]

Appendix VII—Principles and Theorems of Chapter 7

7.41 The Logistic Base = The Logistic Base of Chapter 5, plus:

Rules of Formation: RF 7-1 .If A is wff, T(A) is wff
Abbreviation: Df 'F'. [F(P) Syn_{df} T ~ P]
Df '0'. [0(P) Syn_{df} (~ TP & ~ FP)]
Axioms: Ax.7-1. [TP Syn (TP & ~ FP)]
Ax.7-2. [FTP Syn ~ TP]
Ax.7-3. [T(P & Q) Syn (TP & TQ)]
Ax.7-4. [T(P v Q) Syn (TP v TQ)]
Ax.7-5. [((TP & ~ TP)v TQ)) Cont TQ]
Rules of Inference: R7-1. [If (P Syn Q) then (TP Syn TQ)]
R7-2. [If Inc(P) then $\models$ ~ T(P)]

7.42 Theorems and Rules of Inference.

7.421. Rules for deriving Syn and Cont-theorems T-wffs from Chs 1 to 4:

R7-1. [If (P SYN Q) then (TP Syn TQ)]
DR7-1a. [If (P SYN Q) then (FP Syn FQ)]
DR7-1b. [If (P SYN Q) then (~ TP Syn ~ TQ)]
DR7-1c. [If (P SYN Q) then (~ FP Syn ~ FQ)]
DR7-1d. [If (P CONT Q) then (TP Cont TQ)]
DR7-1e. [If (P CONT Q) then (~ FP Cont ~ FQ)]
DR7-NF. Normal Form Theorem for T-wffs

Rules Inc-wffs to Inc and Taut T-wffs

DR7-5. [If Inc(P) then Inc(T(P))]
DR7-5′. [If Taut(P) then Taut(~ F(P))]
DR7-5a. [If P Cont Q, then Inc(T(P & ~ Q))]
DR7-5a′. [If P Cont Q, then Taut(~ F(~ P v Q))]
DR7-5b. [If P Syn Q, then Inc(T(P & ~ Q))]
DR7-5b′. [If P Syn Q, then Taut(~ F(~ P v Q))]

Rules for Unsatisfiability and Unfalsifiability

R7-2. [If Inc(P) then ~ T(P)]
DR7-2′. [If Taut(P) then ~ F(P)]
DR7-2a. [If (P Cont Q), then ~ T(P& ~ Q)]
DR7-2a′. [If (P Cont Q), then ~ F(~ P v Q)]
DR7-2b. [If (P Syn Q), then ~ T(P& ~ Q)]
DR7-2b′. [If (P Syn Q), then ~ F(~ P v Q)]

Rules for Valid Inference Schemata

DR7-6a. [If (P Cont Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]
DR7-6b. [If (P Syn Q) and not-Inc (P&Q) then Valid (TP, .:TQ)]
DR7-6c. [If (P Syn Q) and not-Inc (P&Q) then Valid (TQ, .:TP)]
DR7-6d. [If (P Syn Q) and not-Inc (P&Q) then Valid ((TP, .:TQ) & (TQ, .:TP))]
DR_I7-6g. [If (P Impl Q) and not-Inc (P&Q) then Valid_I (TP, .:TQ)]

Rules for Implication

DR_I7-4a. [If P CONT Q and TQ Impl TR and Not-(TP Cont TR) then TP Impl TR] "CIISyll"
DR_I7-4b. [If TP Impl TQ and Q CONT R and Not-(TP Cont TR) then TP Impl TR] "ICISyll"

Theorems

<u>Syn and Cont-theorems of Chapter 7.</u>

T7-01. [TP Syn (TP & ~ FP)]
T7-02. [FTP Syn ~ TP]
T7-03. [T(P & Q) Syn (TP & TQ)]
T7-04. [T(P v Q) Syn (TP v TQ)]
T7-05. [T((TP & ~ TP) v Q) Cont TQ]
T7-06. [F(P) Syn T ~ P]
T7-07. [0(P) Syn (~ TP & ~ FP)]

7.42121. Syn and Cont-theorems from Axioms 7-1 and Df 'F'

T7-11. [TP Syn TP]
T7-12. [FP Syn FP]
T7-13. [TP Cont ~ FP] (not[~ FP Cont TP])
T7-14. [FP Syn (FP & ~ TP)]
T7-15. [FP Cont ~ TP] (not[~ TP Cont FP])
T7-16. [TP Syn F ~ P]

7.42122. Reduction to Normal Form T-wffs, from Axioms 7-1 to 7-4

T7-17. [FFP Syn ~ FP]
T7-18. [FP Cont FTP]
T7-19. [TP Cont FFP] } 1st level to 2nd-level or vice versa
T7-20. [TTP Syn TP]
T7-21. [TFP Syn FP]
T7-22. [F(P & Q) Syn (FP v FQ)]
T7-23. [F(P v Q) Syn (FP & FQ)]
T7-24. [T(∀x)Px Syn (∀x)TPx]
T7-25. [T(∃x)Px Syn (∃x)TPx]
T7-26. [~ F(∀x)Px Syn (∀x) ~ FPx]
T7-27. [~ F(∃x)Px Syn (∃x) ~ FPx]
T7-28. [F(∃x)Px Syn (∀x)FPx]
T7-29. [F(∀x)Px Syn (∃x)FPx]
T7-30. [~ T(∃x)Px Syn (∀x) ~ TPx]
T7-31. [~ T(∀x)Px Syn (∃x) ~ TPx]

7.42123. Other Syn- and Cont-theorems from Axioms 7-1 to 7-4

T7-32. [~ FTP Syn TTP] } 2nd-level
T7-33. [~ FFP Syn TFP]
T7-34. [T(P ⊃ Q) Cont (TP ⊃ TQ)]
T7-35. [T(P ⊃ Q) Syn T(~ Q ⊃ ~ P)]
T7-36. [(TP ⊃ TQ) Syn (~ TQ ⊃ ~ TP)]
T7-37. [(FP & FQ) Cont F(P & Q)] (For &-table, Row 9)
T7-38. [(TP & TQ) Cont T(P v Q)] (For v-table, Row 5)
T7-39. [(FP & TQ) Cont T(P ⊃ Q)] (For ⊃-table, Row 6)
T7-40. [(TP & FQ) Syn F(P ⊃ Q)] (For ⊃-table, Row 8)
T7-41. [F(TP & ~ TQ) Syn ~ T(TP & ~ TQ)] } 2nd-level
T7-42. [T(~ TP v TQ) Syn ~ F(~ TP v TQ)]
T7-43. [T(∀x)(Px ⊃ Qx) Cont (T(∀x)Px ⊃ T(∀x)Qx]

7.42124 Detachment theorems from Ax.7-5)

T7-44. [(TP & (~TP v TQ)) Cont TQ] “Alternative Syllogism #1”
T7-44⊃. [(TP & (TP ⊃ TQ)) Cont TQ] “TF-Modus Ponens #1”
T7-45. [((~TP v TQ) & ~TQ) Cont ~TP] “Alternative Syllogism #2”
T7-45⊃. [((TP ⊃ TQ) & ~TQ) Cont ~TP] “TF-Modus Tollens #1”
T7-46. [(TP & T(~P v Q)) Cont TQ] “Alternative Syllogism #3”
T7-46⊃.[(TP & T(P ⊃ Q)) Cont TQ] “TF-Modus Ponens #2”
T7-47. [(T(~P v Q) & FQ) Cont FP] “Alternative Syllogism #4”
T7-47⊃. [(T(P ⊃ Q) & FQ) Cont FP] “TF-Modus Tollens #2”

7.42125. Theorems about non-True and non-false expressions, from Df ‘0’

T7-48. [0P Syn ~(TP v FP)]
T7-49. [~0P Syn (TP v FP)]
T7-50. [0P Cont FTP] } 2nd-level not[FTP Cont OP]
T7-51. [0P Cont FFP] } not[FFP Cont OP]
T7-52. [0~P Syn 0P]
T7-53. [0P Syn T(0P)]
T7-54. [~F(0P) Syn 0P]
T7-55. [(0P v TP) Cont ~FP]
T7-56. [(0P v FP) Cont ~TP]
T7-57. [(0P & 0Q) Syn (0(P&Q) & 0(PvQ))]
T7-58. [(0P & 0Q) Cont 0(P&Q)]
T7-59. [(0P & 0Q) Cont 0(PvQ)]
T7-60. [(0P & 0Q) Cont 0(P⊃Q)]
T7-61. [0TP Syn (~TP & TP)] }
T7-62. [0FP Syn (~FP & FP)] } 2nd-level
T7-63. [00P Syn (F0P & ~F0P)] }
T7-64. [(T(P&Q)vF(P&Q)v0(P&Q)) & FP) Cont F(P&Q)]
T7-65. [((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) Cont ~T(P&Q)]
T7-66. [((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0P)) Cont (0(P&Q)]
T7-67. [(0P v TP v FP) Syn ((~TP& ~FP) v (TP& ~FP) v (FP& ~TP)]
T7-68. [(0P v TP v FP) Syn ((~TP v TP) & (~TP v ~FP) & (~FP v FP))]
T7-69. [T(0P v TP v FP) Syn (T(~TP v TP) & T(~TP v ~FP) & T(~FP v FP))]
T7-70.Taut[0P v TP v FP]

7.422. Logical Presuppositions of Analytic Truth Logic

7.4221 Basic Taut and Inc Theorems for T-wffs

From these we get the 1st-Level Inc- and TAUT-theorems,

⊨ Inc[TP & FP]	Three laws	[T7-13,DR5-13a]
⊨ Inc[TP & ~TP]	of	[T7-11,DR5-13a]
⊨ Inc[FP & ~FP]	non-contradiction	[T7-12,DR5-13a]
⊨ Inc[(~TP & ~FP) & ~0P]	Law of	[T7-07,DR5-13a]
⊨ Taut[(0P v TP v FP]	Trivalence	[T7-70]
⊨ Taut[(~TP v ~FP)]	Three Laws	[T7-13,DR5-4b]
⊨ Taut[(~TP v TP)]	of	[T7-11,DR5-4e]
⊨ Taut[(~FP v FP)]	“excluded middle”	[T7-12,DR5-4e]

|= Inc[~ FTP & ~ TTP]	(Syn ‘ |= Inc[0(TP)]’)	[T7-32,DR5-13a]	2nd-level
|= Inc[~ FFP & ~ TFP]	(Syn ‘ |= Inc[0(FP)]’)	[T7-33,DR5-13a]	
|= Inc[~ T0P & ~ F0P]	(Syn ‘ |= Inc[0(0P)]’)	[T7-55,DR5-13a]	

7.4222 2nd Level UnSatisfiability and UnFalsifiability-Theorems

From R7-2 to DR7-2f we get 2nd-Level theorems of Unsatifiability and Unfalsifiability

|= ~ T[TP & FP]	Three laws	[T7-13,DR7-2]
|= ~ T[TP & ~ TP]	of	[T7-11,DR7-2]
|= ~ T[FP & ~ FP]	non-contradiction	[T7-12,DR7-2]
|= ~ T[(~ TP & ~ FP) & ~ 0P]	Law of	[T7-07,DR7-2b]
|= ~ F[(0Pv(TP v FP))]	Trivalence	[T7-49,DR7-2b′]
|= ~ F[(~ TP v ~ FP)]	Three Laws	[T7-13,DR7-2a]
|= ~ F[(~ TP v TP)]	of	[T7-11,DR7-2a]
|= ~ F[(~ FP v FP)]	“excluded middle”	[T7-12,DR7-2a]
|= ~ T[~ FTP & ~ TTP]	(Syn ‘ |= ~ T[0(TP)]’)	[T7-32,DR7-2a′]
|= ~ T[~ FFP & ~ TFP]	(Syn ‘ |= ~ T[0(FP)]’)	[T7-33,DR7-2a′]
|= ~ T[~ T0P & ~ F0P]	(Syn ‘ |= ~ T[0(0P)]’)	[T7-53,T7-54,DR7-2a′]

7.4223 2nd Level Logical Truth and Logical Falsehood; LT- and LF-theorems

From these we get 2nd-level theorems of Logical Falsehood and Logical Truth:

|= F[TP & FP]	Three laws
|= F[TP & ~ TP]	of
|= F[FP & ~ FP]	non-contradiction
|= F[(~ TP & ~ FP) & ~ 0P]	Law of
|= T[((TP v FP) v 0P]	Trivalence
|= T[(~ TP v ~ FP)]	Three Laws
|= T[(~ TP v TP)]	of
|= T[(~ FP v FP)]	“excluded middle”

7.4224 2nd Level Logical Presuppositions of Analytic Truth-logic: Trivalence.

T7-71. LT[0P v TP v FP]	The Law of Trivalence	
T7-72. LT[~ TP v TP]	Three Laws	
T7-73. LT[~ TP v ~ FP]	of	
T7-74. LT[~ FP v FP]	Excluded Middle	
T7-75. LT[~ (TP & ~ TP)]	Three Laws	Syn |= LF[TP & ~ TP]
T7-76. LT[~ (TP & FP)]	of	Syn |= LF[TP & FP]
T7-77. LT[~ (FP & ~ FP)]	Non-Contradiction	Syn |= LF[FP & ~ FP]

7.423 Implication Theorems

7.4231—Basic Implications

Ti7-80. [~ TP Impl ~ T(P & Q)]
Ti7-81. [FP Impl F(P & Q)]
Ti7-82. [~ FP Impl ~ F(P v Q)]
Ti7-83. [TQ Impl T(P v Q)]
Ti7-84. [TP Impl F(0P)]
Ti7-85. [FP Impl F(0P)]

7.4233—Principles of the truth-tables.

For Truth Table of '&':	From	
T7-&R1. [(0P & 0Q) Cont 0(P & Q)]	(**T7-58**)	(For &-table, Row 1)
Ti7-&R2. [(TP & 0Q) Impl 0(P & Q)]		(For &-table, Row 2)
Ti7-&R3. [(FP & 0Q) Impl F(P & Q)]		(For &-table, Row 3)
Ti7-&R4. [(0P & TQ) Impl 0(P & Q)]		(For &-table, Row 4)
T7-&R5. [(TP & TQ) Syn T(P & Q)]	(**Ax**.**7-3**)	(For &-table, Row 5)
Ti7-&R6. [(FP & TQ) Impl F(P & Q)]		(For &-table, Row 6)
Ti7-&R7. [(0P & FQ) Impl F(P & Q)]		(For &-table, Row 7)
Ti7-&R8. [(TP & FQ) Impl F(P & Q)]		(For &-table, Row 8)
T7-&R9. [(FP & FQ) Cont F(P & Q)]	(**T7-37**)	(For &-table, Row 9)
For Truth Table of 'v':		
T7-vR1. [(0P & 0Q) Cont 0(P v Q)]	(**T7-59**)	(For v-table, Row 1)
Ti7-vR2. [(TP & 0Q) Impl T(P v Q)]		(For v-table, Row 2)
Ti7-vR3. [(FP & 0Q) Impl 0(P v Q)]		(For v-table, Row 3)
Ti7-vR4. [(0P & TQ) Impl T(P v Q)]		(For v-table, Row 4)
T7-vR5. [(TP & TQ) Cont T(P v Q)]	(**T7-38**)	(For v-table, Row 5)
Ti7-vR6. [(FP & TQ) Impl T(P v Q)]		(For v-table, Row 6)
Ti7-vR7. [(0P & FQ) Impl 0(P v Q)]		(For v-table, Row 7)
Ti7-vR8. [(TP & FQ) Impl T(P v Q)]		(For v-table, Row 8)
T7-vR9. [(FP & FQ) Syn F(P v Q)]	(**T7-23**)	(For v-table, Row 9)
For Truth Table of '⊃':		
T7⊃-R1. [(0P & 0Q) Cont 0(P ⊃ Q)]	(**T7-60**)	(For ⊃-table, Row 1)
Ti7⊃-R2. [(TP & 0Q) Impl 0(P ⊃ Q)]		(For ⊃-table, Row 2)
Ti7⊃-R3. [(FP & 0Q) Impl T(P ⊃ Q)]		(For ⊃-table, Row 3)
Ti7⊃-R4. [(0P & TQ) Impl T(P ⊃ Q)]		(For ⊃-table, Row 4)
Ti7⊃-R5. [(TP & TQ) Impl T(P ⊃ Q)]		(For ⊃-table, Row 5)
T7⊃-R6. [(FP & TQ) Cont T(P ⊃ Q)]	(**T7-39**)	(For ⊃-table, Row 6)
Ti7⊃-R7. [(0P & FQ) Impl 0(P ⊃ Q)]		(For ⊃-table, Row 7)
T7⊃-R8. [(TP & FQ) Impl F(P ⊃ Q)]	(**T7-40**)	(For ⊃-table, Row 8)
Ti7⊃-R9. [(FP & FQ) Impl T(P ⊃ Q)]		(For ⊃-table, Row 9)

7.4234—A-implication in Q-Theory

Ti7-86. [(∀x)TQx Impl (∀x)(~0Px v TQx)]
Ti7-87. [(∀x)TQx Impl (∀x)(~TPx v TQx)]
Ti7-88. [(∀x)TQx Impl (∀x)(~FPx v TQx)]
Ti7-89. [(∀x)~TPx Impl (∀x)(~TPx v FQx)]
Ti7-90. [(∀x)FPx Impl (∀x)(~TPx v FQx)]
Ti7-91. [(∀x)TPx Impl (∀x)(TPx v TQx)]
Ti7-92. [T(Pa_i) Impl T(∃x)Px]
Ti7-93. [F(Pa_i) Impl F(∀x)Px]
Ti7-94. [~T(Pa_i) Impl ~T(∀x)Px]
Ti7-95. [~F(Pa_i) Impl ~F(∃x)Px]

Valid Inference Schemata

Replacing 'P' by 'TP' under U-SUBa in the theorems T5-101 to T5-105 of Chapter 5),

T7-5101b Valid [TP, **.:** (TP & TP)]	[&-IDEM]	[U-SUBa,(P/TP)]
T7-5102b Valid [TP, **.:** (TP v TP)]	[v-IDEM]	[U-SUBa,(P/TP)]
T7-5103b Valid [(TP & TQ), **.:** (TQ & TP)]	[&-COMM]	[U-SUBa,(P/TP)]
T7-5104b Valid [(TP v TQ), **.:** (TQ v TP)]	[v-COMM]	[U-SUBa,(P/TP)]
T7-5105b Valid [(TP & (TQ & TR)), **.:** ((TP & TQ) & TR)]	[&-ASSOC]	[U-SUBa,(P/TP)]

Replacing 'P' by '~P' under U-SUBb in the preceding theorems,

T7-5101b Valid [T~P, **.:** (T~P & T~P)]	[&-IDEM]	[U-SUBb,(P/~P)]
T7-5102b Valid [T~P, **.:** (T~P v T~P)]	[v-IDEM]	[U-SUBb,(P/~P)]
T7-5103b Valid [(T~P & TQ), **.:** (TQ & T~P)]	[&-COMM]	[U-SUBb,(P/~P)]
T7-5104b Valid [(T~P v TQ), **.:** (TQ v T~P)]	[v-COMM]	[U-SUBb,(P/~P)]
T7-5105b Valid [(T~P & (TQ & TR)), **.:** ((T~P & TQ) & TR)]	[&-ASSOC]	[U-SUBb,(P/~P)]

Replacing P by ~P under U-SUBb in initial theorems,T5-101 to T5-105 of Chapter 5,

T7-5101b Valid [~P, **.:** (~P & ~P)]	[&-IDEM]	[U-SUBb (P/~P)]]
T7-5102b Valid [~P, **.:** (~P v ~P)]	[v-IDEM]	[U-SUBb (P/~P)]
T7-5103b Valid [(~P & Q), **.:** (Q & ~P)]	[&-COMM]	[U-SUBb (P/~P)]
T7-5104b Valid [(~P v Q), **.:** (Q v ~P)]	[v-COMM]	[U-SUBb (P/~P)]
T7-5105b Valid [(~P & (Q & R)), **.:** ((~P & Q) & R)]	[&-ASSOC]	[U-SUBb (P/~P)]

Then replacing 'P' by '~TP' in the initial theorems,T5-101 to T5-105 of Chapter 5,

T7-5101b Valid [~TP, **.:** (~TP & ~TP)]	[&-IDEM]	[U-SUBb,(P/~TP)]
T7-5102b Valid [~TP, **.:** (~TP v ~TP)]	[v-IDEM]	[U-SUBb,(P/~TP)]
T7-5103b Valid [(~TP & TQ), **.:** (TQ & ~TP)]	[&-COMM]	[U-SUBb,(P/~TP)]
T7-5104b Valid [(~TP v TQ), **.:** (TQ v ~TP)]	[v-COMM]	[U-SUBb,(P/~TP)]
T7-5105b Valid [(~TP & (TQ&TR)), **.:** ((~TP&TQ) & TR)]	[&-ASSOC]	[U-SUBb,(P/~TP)]

Replacing P by '~T~P' in the initial theorems T5-101 to T5-105 of Chapter 5 by U-SUBa,

T7-5101b Valid [~T~P, **.:** (~T~P & ~T~P)]	[&-IDEM]	[U-SUBa,(P/~T~P)]
T7-5102b Valid [~T~P, **.:** (~T~P v ~T~P)]	[v-IDEM]	[U-SUBa,(P/~T~P)]
T7-5103b Valid [(~T~P & TQ), **.:** (TQ & ~T~P)]	[&-COMM]	[U-SUBa,(P/~T~P)]
T7-5104b Valid [(~T~P v TQ), **.:** (TQ v ~T~P)]	[v-COMM]	[U-SUBa,(P/~T~P)]
T7-5105b Valid [(~T~P& (TQ&TR)), **.:** ((~T~P &TQ) &TR)]	[&-ASSOC]	[U-SUBa,(P/~T~P)]

From CHAPTER 7, Valid Inference-Schemata directly from Definitions and Axioms in Chapter 7

T7-701 Valid [TP, **.:** (TP & ~FP)]	[T7-01,DR5-6b,U-SUB]
T7-702. Valid [FTP, **.:** ~TP]	[T7-02,DR5-6b,U-SUB]
T7-703. Valid [T(P & Q), **.:** (TP & TQ)]	[T7-03,DR5-6b,U-SUB]
T7-704. Valid [T(P v Q), **.:** (TP v TQ)]	[T7-04,DR5-6b,U-SUB]
T7-705. Valid [T((TP & ~TP) v Q)), **.:** TQ]	[T7-05,DR5-6a,U-SUB]
T7-706. Valid [F(P), **.:** T~P]	[T7-06,DR5-6b,U-SUB]
T7-707. Valid [0(P), **.:** (~TP & ~FP)]	[T7-07,DR5-6b,U-SUB]
T7-711. Valid [TP, **.:** TP]	[T7-11,DR5-6b,U-SUB]
T7-712. Valid [FP, **.:** FP]	[T7-12,DR5-6b,U-SUB]
T7-713. Valid [TP, **.:** ~FP]	[T7-13,DR5-6a,U-SUB]
T7-714. Valid [FP, **.:** (FP & ~TP)]	[T7-14,DR5-6b,U-SUB]
T7-715. Valid [FP, **.:** ~TP]	[T7-15,DR5-6a,U-SUB]
T7-716. Valid [F~P, **.:** TP]	[T7-16,DR5-6b,U-SUB]

From 7.42122. Valid Inference-Schemata for Reduction to Normal Form TWffs from Ax.7-1 to 7-4

T7-717. Valid [FFP, .: ~ FP] [T7-17,DR5-6b,U-SUB]

T7-718. Valid [FP, .: FTP] [T7-18,DR5-6a,U-SUB]

T7-719. Valid [TP, .: FFP] [T7-19,DR5-6a,U-SUB]

T7-720. Valid [TTP, .: TP] [T7-20,DR5-6b,U-SUB]

T7-721. Valid [TFP, .: FP] [T7-21,DR5-6b,U-SUB]

T7-722. Valid [F(P & Q), .: (FP v FQ)] [T7-22,DR5-6b,U-SUB]

T7-723. Valid [F(P v Q), .: (FP & FQ)] [T7-23,DR5-6b,U-SUB]

T7-724. Valid [T(∀x)Px, .: (∀x)TPx] [T7-24,DR5-6b,U-SUB]

T7-725. Valid [T(∃x)Px, .: (∃x)TPx] [T7-25,DR5-6b,U-SUB]

T7-726. Valid [~ F(∀x)Px, .: (∀x) ~ FPx] [T7-26,DR5-6b,U-SUB]

T7-727. Valid [~ F(∃x)Px, .: (∃x) ~ FPx] [T7-27,DR5-6b,U-SUB]

T7-728. Valid [F(∃x)Px, .: (∀x)FPx] [T7-28,DR5-6b,U-SUB]

T7-729. Valid [F(∀x)Px, .: (∃x)FPx] [T7-29,DR5-6b,U-SUB]

T7-730. Valid [~ T(∃x)Px, .: (∀x) ~ TPx] [T7-30,DR5-6b,U-SUB]

T7-731. Valid [~ T(∀x)Px, .: (∃x) ~ TPx] [T7-31,DR5-6b,U-SUB]

From 7.42123 Valid Inference-Schemata from Axioms 7-1 to 7-4

T7-732. Valid [~ FTP, .: TTP] [T7-32,DR5-6b,U-SUB]

T7-733 .Valid [~ FFP, .: TFP] [T7-33,DR5-6b,U-SUB]

T7-734. Valid [T(P ⊃ Q), .: (TP ⊃ TQ)] [T7-34,DR5-6a,U-SUB]

T7-735. Valid [T(P ⊃ Q), .: T(~ Q ⊃ ~ P)] [T7-35,DR5-6b,U-SUB]

T7-736. Valid [(TP ⊃ TQ), .: (~ TQ ⊃ ~ TP)] [T7-36,DR5-6b,U-SUB]

T7-737. Valid [(FP & FQ), .: F(P & Q)] (For &-table, Row 9) [T7-37,DR5-6a,U-SUB]

T7-738. Valid [(TP & TQ), .: T(P v Q)] (For v-table, Row 5) [T7-38,DR5-6a,U-SUB]

T7-739. Valid [(FP & TQ), .: T(P ⊃ Q)] (For⊃-table, Row 6) [T7-39,DR5-6a,U-SUB]

T7-740. Valid [(TP & FQ), .: F(P ⊃ Q)] (For⊃-table, Row 8) [T7-40,DR5-6b,U-SUB]

T7-741. Valid [F(TP & ~ TQ), .: ~ T(TP & ~ TQ)] [T7-41,DR5-6b,U-SUB]

T7-742. Valid [T(~ TP v TQ), .: ~ F(~ TP v TQ)] [T7-42,DR5-6b,U-SUB]

T7-743. Valid [T(∀x)(Px ⊃ Qx), .: (T(∀x)Px ⊃ T(∀x)Qx] [T7-43,DR5-6a],U-SUB]

From 7.42124. Valid Inference-Schemata of Detachment, from Ax.7-5)

T7-744. Valid [(TP & (~ TPvTQ)), .: TQ] "Alternative Syllogism #1" [T7-44,DR5-6a,U-SUB]

T7-745.Valid [((~ TPvTQ) & ~ TQ), .: ~ TP] "Alternative Syllogism #2" [T7-45,DR5-6a,U-SUB]

T7-746.Valid [(TP & T(~ PvQ)), .: TQ] "Alternative Syllogism #3" [T7-46,DR5-6a,U-SUB]

T7-747.Valid [(T(~ PvQ) & FQ), .: FP] "Alternative Syllogism #4" [T7-47,DR5-6a,U-SUB]

From 7.42125. Valid Inference-Schemata about non-true and non-false expressions, from Df '0'

T7-748. Valid [0P, .: ~ (TP v FP)] [T7-48,DR5-6b,U-SUB]

T7-749. Valid [~ 0P, .: (TP v FP)] [T7-49,DR5-6b,U-SUB]

T7-750. Valid [0P, .: ~ TP] [T7-50,DR5-6a,U-SUB]

T7-751. Valid [0P, .: ~ FP] [T7-51,DR5-6a,U-SUB]

T7-752. Valid [0 ~ P, .: 0P] [T7-52,DR5-6b,U-SUB]

T7-753. Valid [0P, .: T(0P) [T7-53,DR5-6b,U-SUB]

T7-754. Valid [~ F0P, .: 0P] [T7-54,DR5-6b,U-SUB]

T7-755. Valid [(TP v 0P), .: ~ FP] [T7-55,DR5-6a,U-SUB]

T7-756. Valid [(FP v 0P), .: ~ TP] [T7-56,DR5-6a,U-SUB]

T7-757. Valid [(0P & 0Q), .: (0(P&Q) & 0(PvQ))] [T7-57,DR5-6b,U-SUB]

(T7-758,. T7-759 and T7-760 are the first principles of inference for the &-, v- and ⊃-truth-tables)

T7-761. Valid [0TP, .: (~TP & TP)] [T7-61,DR5-6b,U-SUB]
T7-762. Valid [0FP, .: (~FP & FP)] [T7-62,DR5-6b,U-SUB]
T7-763. Valid [00P, .: (F0P & ~F0P)] [T7-63,DR5-6b,U-SUB]
T7-764. Valid [((T(P&Q) v F(P&Q) v 0(P&Q)) & FP), .: F(P&Q)] [T7-64,DR5-6a,U-SUB]
T7-765. Valid [(T(P&Q) v F(P&Q) v 0(P&Q)) & ~TP), .: ~T(P&Q)] [T7-65,DR5-6a,U-SUB]
T7-766. Valid [((T(P&Q) v F(P&Q) v 0(P&Q)) & (TP & 0P)), .: 0(P&Q)] [T7-66,DR5-6a,U-SUB]
T7-767. Valid [(0PvTPvFP), .: ((~TP& ~FP) v (TP& ~FP) v (FP& ~TP)] [T7-67,DR5-6a,U-SUB]
T7-768. Valid [(0PvTPvFP), .: ((~TP vTP) & (~TPv ~FP) & (~FpvFP))] [T7-68,DR5-6a,U-SUB]
T7-769. Valid [T(0PvTPvFP), .: (T(~TPvTP) & T(~TPv ~FP) & T(~FPvFP))] [T7-69,DR5-6a,U-SUB]

From 7.4231 Valid Inference-Schemata based on Basic Implication Theorems

Ti7-780. Valid$_I$ [~TP, .: ~T(P & Q)] [Ti7-80,DR7-6g]
Ti7-781. Valid$_I$ [FP, .: F(P & Q)] [Ti7-81,DR7-6g]
Ti7-782. Valid$_I$ [~FP, .: ~F(P v Q)] [Ti7-82,DR7-6g]
Ti7-783. Valid$_I$ [TQ, .: T(P v Q)] [Ti7-83,DR7-6g]
Ti7-784. Valid$_I$ [TP, .: (TP v FP)] [Ti7-84,DR7-6g]
Ti7-785. Valid$_I$ [FP, .: (TP v FP)] [Ti7-85,DR7-6g]

From 7.4233 Valid Principles of Inference for truth-tables (some are not A-implications).

For the truth ~ table of (P& Q)

T7-758a.	Valid [(0P & 0Q), .: 0(P&Q)]	(For &-Row 1)	[T7-58,DR5-6a,U-SUB]
Ti7-7&R2.	Valid$_I$ [(TP & 0Q), .: 0(P & Q)]	(For &-Row 2)	[Ti7-&R2,DR7-6g]
Ti7-7&R3.	Valid$_I$ [(FP & 0Q), .: F(P & Q)]	(For &-Row 3)	[Ti7-&R3,DR7-6g]
Ti7-7&R4.	Valid$_I$ [(0P & TQ), .: 0(P & Q)]	(For &-Row 4)	[Ti7-&R4,DR7-6g]
T7-703c.	Valid [TP & TQ), .: (T(P &Q)]	(For &-Row 5)	[Ax..7-3,DR5-6c,U-SUB]
Ti7-7&R6.	Valid$_I$ [(FP & TQ), .: F(P & Q)]	(For &-Row 6)	[Ti7-&R6,DR7-6g]
Ti7-7&R7.	Valid$_I$ [(0P & FQ), .: F(P & Q)]	(For &-Row 7)	[Ti7-&R7,DR7-6g]
Ti7-7&R8.	Valid$_I$ [(TP & FQ), .: F(P & Q)]	(For &-Row 8)	[Ti7-&R8,DR7-6g]
T7-737a.	Valid [(FP & FQ), .: F(P & Q)]	(For &-Row 9)	[T7-37,DR5-6a,U-SUB]

For the truth-table of (P v Q)

T7-759a.	Valid [(0P & 0Q), .: 0(P v Q)]	(For v-Row 1)	[T7-59,DR5-6a,U-SUB]
Ti7-7vR2.	Valid$_I$ [(TP & 0Q), .: T(P v Q)]	(For v-Row 2)	[Ti7-vR2,DR7-6g]
Ti7-7vR3.	Valid$_I$ [(FP & 0Q), .: 0(P v Q)]	(For v-Row 3)	[Ti7-vR3,DR7-6g]
Ti7-7vR4.	Valid$_I$ [(0P & TQ), .: T(P v Q)]	(For v-Row 4)	[Ti7-vR4,DR7-6g]
T7-738a.	Valid [(TP & TQ), .: T(P v Q)]	(For v-Row 5)	[T7-38,DR5-6a,U-SUB]
Ti7-7vR6.	Valid$_I$ [(FP & TQ), .: T(P v Q)]	(For v-Row 6)	[Ti7-vR6,DR7-6g]
Ti7-7vR7.	Valid$_I$ [(0P & FQ), .: 0(P v Q)]	(For v-Row 7)	[Ti7-vR7,DR7-6g]
Ti7-7vR8.	Valid$_I$ [(TP & FQ), .: T(P v Q)]	(For v-Row 8)	[Ti7-vR8,DR7-6g]
T7-723c.	Valid [(FP & FQ), .: F(P v Q)]	(For v-Row 9)	[T7-23,DR5-6c,U-SUB]

For the truth-table of (P ⊃ Q)

T7-760a.	Valid [(0P & 0Q), .: 0(P ⊃ Q)]	(For ⊃-Row 1)	[T7-60,DR5-6a,U-SUB]
Ti7-7⊃-R2.	Valid$_I$ [(TP & 0Q), .: 0(P ⊃ Q)]	(For ⊃-Row 2)	[Ti7⊃-R2,DR7-6g]
Ti7-7⊃-R3.	Valid$_I$ [(FP & 0Q), .: T(P ⊃ Q)]	(For ⊃-Row 3)	[Ti7⊃-R3,DR7-6g]
Ti7-7⊃-R4.	Valid$_I$ [(0P & TQ), .: T(P ⊃ Q)]	(For ⊃-Row 4)	[Ti7⊃-R4,DR7-6g]
Ti7-7⊃-R5.	Valid$_I$ [(TP & TQ), .: T(P ⊃ Q)]	(For ⊃-Row 5)	[Ti7⊃-R5,DR7-6g]
T7-739a.	Valid [(FP & TQ), .: T(P ⊃ Q)]	(For ⊃-Row 6)	[T7-39,DR5-6a,U-SUB]
Ti7-7⊃-R7.	Valid$_I$ [(0P & FQ), .: 0(P ⊃ Q)]	(For ⊃-Row 7)	[Ti7⊃-R7,DR7-6g]
T7-740a.	Valid [(TP & FQ), .: F(P ⊃ Q)]	(For ⊃-Row 8)	[T7-40,DR5-6b,U-SUB]
Ti7-7⊃-R9.	Valid$_I$ [(FP & FQ), .: T(P ⊃ Q)]	(For ⊃-Row 9)	[Ti7⊃-R8,DR7-6g]

From 7.4234—A-implication in Q-Theory

Ti7-786. Valid_I [(∀x)TQx, .: (∀x)(0Px ⊃ TQx)] [Ti7-86,DR7-6g]
Ti7-787. Valid_I [(∀x)TQx, .: (∀x)(TPx ⊃ TQx)] [Ti7-87,DR7-6g]
Ti7-788. Valid_I [(∀x)TQx, .: (∀x)(FPx ⊃ TQx)] [Ti7-88,DR7-6g]
Ti7-789. Valid_I [(∀x) ~ TPx, .: (∀x)(TPx ⊃ FPx)] [Ti7-89,DR7-6g]
Ti7-790. Valid_I [(∀x)FPx, .: (∀x)(TPx ⊃ FQx)] [Ti7-90,DR7-6g]
Ti7-791. Valid_I [(∀x)TPx, .: (∀x)(~ TPx ⊃ TQx)] [Ti7-91,DR7-6g]
Ti7-792. Valid_I [T(Pai), .: T(∃x)Px] [Ti7-92,DR7-6g]
Ti7-793. Valid_I [F(Pai), .: F(∀x)Px] [Ti7-93,DR7-6g]
Ti7-794. Valid_I [~ T(Pai), .: ~ T(∀x)Px] [Ti7-94,DR7-6g]
Ti7-795. Valid_I [~ F(Pai), .: ~ F(∃x)Px] [Ti7-95,DR7-6g]

Appendix VIII—Rules and Theorems, Chapter 8

8.21 The Logistic Base: Definitions, Axioms, Rules of Inference

I. Primitives:

Grouping devices: (,), [,], $<$, $>$.

Predicate letters: $P_1, P_2, \ldots, P_n$. [Abbr. 'P', 'Q', 'R'] PL

Argument Place holders: 1, 2, 3,… APH

Individual Constants: $a_1, a_2, \ldots a_n$, [Abbr. 'a', 'b', 'c'] IC

Individual variables: $x_1, x_2, \ldots x_n$, [Abbr. 'x', 'y', 'z'] IV

Predicate operators: & ~ ⇒ T

Conjunctive Quantifier: $(\forall v_i)$

Primitive (2nd Order) Predicate of Logic: Syn

II. Formation Rules:

FR1. $[P_i]$ is a wff

FR2. If P and Q are wffs, [(P&Q)] is a wff.

FR3. If P_i is a wff and $t_j (1 \leq j \leq k)$ is an APH or a IC, then $P_i < t_1, \ldots, t_k >$ is a wff.

FR4. If $P_i < 1 >$ is a wff, then $[(\forall v_j) P_i v_j]$ is a wff.

FR5. If P is a wff, [~P] is a wff.

FR6. If P and Q are wffs, [(P ⇒ Q)] is a wff.

FR7. If P is wff, T(P) is wff

III. Abbreviations, Definitions

Predicate Operators:

Df 1-1.	[(P & Q & R)	SYN_{df} (P & (Q & R))]	
Df 3-1.	$[(\forall_k x)Px$	SYN_{df} $(Pa_1 \& Pa_2 \& \ldots \& Pa_k)]$	[Df '(∀x)']
Df 4-1.	[(P v Q)	SYN_{df} ~(~P & ~Q)]	[Df 'v',DeM1]
Df 4-2.	[(P ⊃ Q)	SYN_{df} ~(P & ~Q)]	[Df '⊃']
Df 4-3.	[(P ≡ Q)	SYN_{df} ((P ⊃ Q) & (Q ⊃ P))]	[Df '≡']
Df 4-4.	[(∃x)Px	SYN_{df} ~(∀x)~Px]	[Df '(∃x)']
Df 6-1.	[(P ⇔ Q)	SYN_{df} ((P ⇒ Q) & (Q ⇒ P))]	[Df '⇔']
Df 7-1.	[F(P)	Syn_{df} T(~P)]	[Df 'F']
Df 7-2.	[0(P)	Syn_{df} (~TP & ~FP)]	[Df '0']

Logical Predicates

Df 'Cont'. [(P Cont Q) Syn_{df} '(P Syn (P&Q))]

Df 'Inc'. [Inc(P) Syn_{df} ((P Syn (Q&-R)) & (Q cont R))
v ((P Syn (Q&R)) & Inc(R))
v ((P Syn (QvR)) & Inc(Q) & Inc(R))
v ((P Syn (Q ⇒ R) &Inc(Q&R)]

Df 'Taut' [Taut(P) Syn_{df} Inc(~P)]

Df 'P Ent Q' [(P Ent Q) Syn_{df} ((P Cont Q) & ~Inc(P&Q))]

Df 'Impl' [(TP Impl TQ) Syn_{df} (i) ((0Q v TQ v FQ) & TP) Cont TQ)
(ii) & Not: (TP Cont TQ)
(iii) & Not: (0Q v TQ v FQ) Cont TQ]

Df 'Valid'[Valid(P ⇒ Q) Syn_{df} ((P Ent Q) v (P Impl Q)) & not-Inc(P&Q))]

IV. Axioms and Transformation Rules

1. Axioms Ax.6-01. [P SYN (P&P)] "&-IDEM1"
   Ax.6-02. [(P&Q) SYN(Q&P)] "&-COMM"
   Ax.6-03. [(P&(Q&R)) SYN ((P&Q)&R)] "&-ASSOC1"
   Ax.6-04. [(P&(QvR)) SYN ((P&Q)v(P&R))] "&v-DIST1"
   Ax.6-05. [P SYN ~~P] "DN"
   Ax 6-06. $\models$[((P$\Rightarrow$Q) & P) SYN (((P$\Rightarrow$Q) & P) & Q)]
   Ax.7-01. [TP Syn (TP & ~FP)
   Ax.7-02. [FTP Syn ~TP
   Ax.7-03. [T(P & Q) Syn (TP & TQ)
   Ax.7-04. [T(P v Q) Syn (TP v TQ)
   Ax.7-05. [(TP & ~TP) v TQ)) Cont TQ]
   Ax.8-01. T(P$\Rightarrow$Q) Syn T(P&Q)
   Ax.8-02. F(P$\Rightarrow$Q) Syn T(P&~Q)

2. Principles of Inference

R1. If $\models$P and $\models$[Q Syn R] then $\models$P(Q//R) [SynSUB]

R3-2. If = R and P_i<t_1,...,t_n> occurs in R, [U-SUB]
and Q is an h-adic wff, where h $\geq$ n,
and Q has occurrences of all numerals 1 to n,
and no variable in Q occurs in R or S
then $\models$[R(P_i<t_1,...,t_n>/Q)

R2-3. If $\models$[P<1>] then $\models$[Pa] "INST"

R7-2. If Inc(P) then ~T(P).

From Section 8.22. Syn- and Cont-Theorems with 'T' and '$\Rightarrow$'

T8-01. [T(P&Q) Syn T(P8Q)] [Ax.8-01,DR1-01]
T8-02. [T(P&~Q) Syn F(P$\Rightarrow$Q)] [Ax.8-02,DR1-01]
T8-11. [T(P $\Rightarrow$Q) Cont TP]
T8-12. [F(P $\Rightarrow$Q) Cont TP]
T8-13. [T(P $\Rightarrow$Q) Cont TQ]
T8-14. [F(P $\Rightarrow$Q) Cont FQ]
T8-15. [(T(P$\Rightarrow$Q) & T(Q$\Rightarrow$R)) Cont T(P $\Rightarrow$R)]

DR6-119. [VALID (P $\Rightarrow$Q) & VALID(Q $\Rightarrow$R)) $\Rightarrow$VALID(P $\Rightarrow$R)] and by U-SUBa,
DR8-119. [Valid(TP $\Rightarrow$TQ) & Valid(TQ $\Rightarrow$TR)) $\Rightarrow$Valid(TP $\Rightarrow$ TR)], and
DR6-120.[VALID(P_1 $\Rightarrow$$P_2$) & VALID($P_2$ $\Rightarrow$$P_3$}&...& VALID($P_{n-1}$ $\Rightarrow$$P_n$)) $\Rightarrow$VALID($P_1$$\Rightarrow$$P_n$]

T8-16. [T(P$\Rightarrow$Q) Syn T(Q$\Rightarrow$P)]
T8-17. [T(P$\Rightarrow$Q) Syn T(TP$\Rightarrow$TQ)]
T8-18. [T(P $\Rightarrow$Q) Syn F(P $\Rightarrow$ ~Q)]
T8-19. [F(P $\Rightarrow$Q) Syn T(P $\Rightarrow$ ~Q)]
T8-20. [T(P $\Rightarrow$Q) Cont ~T(P $\Rightarrow$ ~Q)]
T8-21. [F(P $\Rightarrow$Q) Cont ~F(P $\Rightarrow$ ~Q)]
T8-22. [F(P$\Rightarrow$Q) Syn F(~Q$\Rightarrow$~P)]
T8-23. [~F(P$\Rightarrow$Q) Syn ~F(~Q$\Rightarrow$~P)]
T8-24. [T(P$\Rightarrow$Q) Cont ~F(~Q$\Rightarrow$~P)]
T8-25 [F(P$\Rightarrow$Q) Syn F(P $\supset$ Q)]
T8-26. [(T(P$\Rightarrow$Q) v F(P$\Rightarrow$Q)) Syn (TP & (TQ v FQ))]
T8-27. [~0(P$\Rightarrow$Q) Cont TP]
T8-28. [~0(P$\Rightarrow$Q) Cont ~0Q]

T8-29. [0(P ⇒ Q) Syn (~ TP v 0Q)]
T8-30. [0(P ⇒ Q) Syn 0(P ⇒ ~ Q)]
T8-31. [(0P & 0Q) Cont 0(P ⇒ Q)] (For Row 1, of (P ⇒ Q) Table)
T8-32. [(FP & 0Q) Cont 0(P ⇒ Q)] (For Row 3, of (P ⇒ Q) Table)
T8-33. [(TP & TQ) Syn T(P ⇒ Q)] (For Row 5, of (P ⇒ Q) Table)
T8-34. [(TP & FQ) Syn F(P ⇒ Q)] (For Row 8, of (P ⇒ Q) Table)
T8-35. [(T(P ⇒ Q)vF(P ⇒ Q)v0(P ⇒ Q)) Syn ((TPv0(P ⇒ Q)) & (~ 0Qv0(P ⇒ Q)))]
T8-36. [((T(P ⇒ Q)vF(P ⇒ Q)v0(P ⇒ Q)) & ~ TP) Cont ~ T(P ⇒ Q)]
T8-37. [((T(P ⇒ Q)vF(P ⇒ Q)v0(P ⇒ Q)) & 0Q) Cont (0(P ⇒ Q)]
T8-38. [(T(TP ⇒ TQ)v(F(TP ⇒ TQ)v(0(TP ⇒ TQ) & (TP ⇒ TQ) & ~ TQ) Cont ~ TP]
T8-39. [((∀x)(TPx ⇒ TQx) & (∀x)TPx) Cont (∀x)TQx]

8.222 Impl-Theorems with T-operators and '⇒'.

Ti8-40. [~ TP Impl 0(P ⇒ Q)] .: |= [~ TP Impl ~ T(P ⇒ Q)]
Ti8-41. [FP Impl 0(P ⇒ Q)] .: |= [FP Impl ~ T(P ⇒ Q)]
Ti8-42. [0P Impl 0(P ⇒ Q)]
Ti8-43. [0Q Impl 0(P ⇒ Q)]
Ti8-44. [(TP ⇒ TQ) & ~ TQ) Impl ~ TP]
|= [TQ Impl (TP ⊃ TQ)] .: |= [FQ Impl (TP ⊃ FQ)], but not: |= [TQ Impl (TP ⇒ TQ)]
|= [~ TP Impl (TP ⊃ TQ)] .: |= [FP Impl (TP ⊃ TQ)], but not: |= [FP Impl (TP ⇒ TQ)]
|= [TQ Impl (TP ⊃ TQ)] .: |= [FQ Impl (TP ⊃ FQ)], but not: |= [TQ Impl (TP ⇒ TQ)]

For Truth-table of '⇒':

Ti8 ⇒ r2. [(TP & 0Q) Impl 0(P ⇒ Q)] [For Row 2, truth-table of '⇒']
Ti8 ⇒ r4. [(0P & TQ) Impl 0(P ⇒ Q)] [For Row 4, truth-table of '⇒']
Ti8 ⇒ r6. [(FP & TQ) Impl 0(P ⇒ Q)] [For Row 6, truth-table of '⇒']
Ti8 ⇒ r7. [(0P & FQ) Impl 0(P ⇒ Q)] [For Row 7, truth-table of '⇒']
Ti8 ⇒ r9. [(FP & FQ) Impl 0(P ⇒ Q)] [For Row 9, truth-table of '⇒']

Section 8.24 Inc- and Taut-theorems With C-conditionals and T-wffs

Derived rules whereby New Inc- or Taut-theorems re: truth assertions of C-conditionals can be derived from every Cont-, Syn- or Impl-theorem in A-logic:

DR8-5a. [If (P Cont Q) then Inc(T(P ⇒ ~ Q))]
DR8-5a'.[If (P Cont Q) then Taut(~ T(P ⇒ ~ Q))]
DR8-5d. [If (P Impl Q) then IncT(P ⇒ ~ Q)]
DR8-5d'.[If (P Impl Q) then Taut ~ T(P ⇒ ~ Q)]

For example:

|=Inc[TP ⇒ ~ TP], |=Taut ~ [TP ⇒ ~ TP] [T7-13,DR6-5a, DR6-5a']
|=Inc[FP ⇒ ~ FP], |=Taut ~ [FP ⇒ ~ FP] [T7-12,DR6-5a, DR6-5a']
|= Inc[(T(P & Q) ⇒ ~ TP], |=Taut ~ [T(P & Q) ⇒ ~ TP] [T7-136,DR6-5a, DR6-5a']
|= Inc[T(P) ⇒ ~ T(PvQ)], |=Taut ~ [T(P) ⇒ ~ T(PvQ)] [T8-83,DR6-5d, DR6-5d']
|= Inc[F(Pa_i) ⇒ T(∀x)Px], |=Taut ~ [F(Pa_i) ⇒ T(∀x)Px] [Ti8-93,DR6-5d, DR6-5d']

Section 8.23. Validity Theorems with C-conditionals and T-wffs

Derived Rules:

DR8-6a. [If (P Cont Q) and not-Inc (P&Q) then Valid (TP ⇒ TQ)]
DR8-6b. [If (P Syn Q) and not-Inc (P&Q) then Valid (TP ⇒ TQ)]
DR8-6c. [If (P Syn Q) and not-Inc (P&Q) then Valid (TQ ⇒ TP)]
DR8-6d. [If (P Syn Q) and not-Inc (P&Q) then Valid (TP **⇐⇒** TQ)]
DR8-6g. [If (P Impl Q) and not-Inc(P&Q) then $Valid_I$ (P ⇒ Q)]

From SYN- and CONT-theorems in Chapter 1:

T8-101d. Valid[TP <=> T(P&P)] [&-IDEM] [Ax.1-01, DR8-6d]
T8-102d. Valid[TP <=> T(PvP)] [v-IDEM] [Ax.1-02, DR8-6d]
T8-103d. Valid[T(P&Q) <=> T(Q&P)] [&-COMM] [Ax.1-03, DR8-6d]
T8-104d. Valid[T(PvQ) <=> T(QvP)] [v-COMM] [Ax.1-04, DR8-6d]
T8-105d. Valid[T(P&(Q&R)) <=> T((P&Q)&R)] [&-ASSOC] [Ax.1-05, DR8-6d]
T8-106d. Valid[T(Pv(QvR)) <=> T((PvQ)vR)] [v-ASSOC] [Ax.1-06, DR8-6d]
T8-107d. Valid[T(Pv(Q&R)) <=> T((PvQ)&(PvR))] [v&-DIST-1] [Ax.1-07, DR8-6d]
T8-108d. Valid[T(P&(QvR)) <=> T((P&Q)v(P&R))] [&v-DIST-1] [Ax.1-08, DR8-6d]
T8-111d. Valid[TP <=> TP] [T1-11,DR8-6d]
T8-112d. Valid[T((P&Q) & (R&S)) <=> T((P&R) & (Q&S))] [T1-12,DR8-6d]
T8-113d. Valid[T((PvQ) v (RvS)) <=> T((PvR) v (QvS))] [T1-13,DR8-6d]
T8-114d. Valid[T(P & (Q&R)) <=> T((P&Q) & (P&R))] [T1-14,DR8-6d]
T8-115d. Valid[T(P v (QvR)) <=> T((PvQ) v (PvR))] [T1-15,DR8-6d]
T8-116d. Valid[T(Pv(P&Q)) <=> T(P&(PvQ))] [T1-16,DR8-6d]
T8-117d. Valid[T(P&(PvQ)) <=> T(Pv(P&Q))] [T1-17,DR8-6d]
T8-118d. Valid[T(P&(Q&(PvQ))) <=> T(P&Q)] [T1-18,DR8-6d]
T8-119d. Valid[T(Pv(Qv(P&Q))) <=> T(PvQ)] [T1-19,DR8-6d]
T8-120d. Valid[T(P&(Q&R)) <=> T(P&(Q&(R&(Pv(QvR)))))] [T1-20,DR8-6d]
T8-121d. Valid[T(Pv(QvR)) <=> T(Pv(Qv(Rv(P&(Q&R)))))] [T1-21,DR8-6d]
T8-122d. Valid[T(Pv(P&(Q&R))) <=> T(P&((PvQ)&((PvR)&(Pv(QvR)))))] [T1-22,DR8-6d]
T8-123d. Valid[T(P&(Pv(QvR))) <=> T(Pv((P&Q)v((P&R)v(P&(Q&R)))))] [T1-23,DR8-6d]
T8-124d. Valid[T(Pv(P&(Q&R))) <=> T(P&(Pv(QvR)))] [T1-24,DR8-6d]
T8-125d. Valid[T(P&(Pv(QvR))) <=> T(Pv(P&(Q&R)))] [T1-25,DR8-6d]
T8-126d. Valid[T(P&(PvQ)&(PvR)&(Pv(QvR))) <=> T(P&(Pv(QvR)))] [T1-26,DR8-6d]
T8-127d. Valid[T(Pv(P&Q)v(P&R)v(P&(Q&R))) <=> T(Pv(P&(Q&R)))] [T1-27,DR8-6d]
T8-128d. Valid[T((P&Q)v(R&S)) <=> T(((P&Q)v(R&S)) & (PvR))] [T1-28,DR8-6d]
T8-129d. Valid[T((PvQ)&(RvS)) <=> T(((PvQ)&(RvS)) v (P&R))] [T1-29,DR8-6d]
T8-130d. Valid[T((P&Q)&(RvS)) <=> T((P&Q) & ((P&R)v(Q&S)))] [T1-30,DR8-6d]
T8-131d. Valid[T((PvQ)v(R&S)) <=> T((PvQ) v ((PvR)&(QvS)))] [T1-31,DR8-6d]
T8-132d. Valid[T((PvQ)&(RvS)) <=> T(((PvQ)&(RvS)) & (PvRv(Q&S))] [T1-32,DR8-6d]
T8-133d. Valid[T((P&Q)v(R&S)) <=> T(((P&Q)v(R&S)) v (P&R&(QvS))] [T1-33,DR8-6d]
T8-134d. Valid[T((P&Q)v(R&S)) <=> T(((P&Q)v(R&S)) & (PvR) & (QvS))] [T1-34,DR8-6d]
T8-135d. Valid[T((PvQ)&(RvS)) <=> T(((PvQ)&(RvS)) v (P&R) v (Q&S))] [T1-35,DR8-6d]

From CONT-theorems:

T8-136a. Valid [T(P&Q) => TP] [T1-36,DR8-6a]
T8-137a. Valid [T(P&Q) => TQ] [T1-37,DR8-6a]
T8-138a. Valid [T(P&Q) => T(PvQ)] [T1-38),DR8-6a]
T8-122c(1) Valid[T(Pv(P&(Q&R))) => TP] [T1-22c(1),DR8-6a]
T8-122c(1) Valid[T(Pv(P&(Q&R))) => T(PvQ)] [T1-22c(2),DR8-6a]
T8-122c(2) Valid[T(Pv(P&(Q&R))) => T(PvR)] [T1-22c(3),DR8-6a]
T8-122c(4) Valid[T(Pv(P&(Q&R))) => T(Pv(QvR))] [T1-22c(4),DR8-6a]
T8-122c(1,2) Valid[T(Pv(P&(Q&R))) => T(P&(PvQ))] [T1-22c(1,2),DR8-6a]
T8-122c(1,3) Valid[T(Pv(P&(Q&R))) => T(P&(PvR))] [T1-22c(1,3),DR8-6a]
T8-122c(2,3) Valid[T(Pv(P&(Q&R))) => T((PvQ)&(PvR))] [T1-22c(2,3),DR8-6a]
T8-122c(2,4) Valid[T(Pv(P&(Q&R))) =>T((PvQ)&(Pv(QvR)))] [T1-22c(2,4),DR8-6a]
T8-122c(3,4) Valid[T(Pv(P&(Q&R))) =>T((PvR)&(Pv(QvR)))] [T1-22c(3,4),DR8-6a]

From CHAPTER 3 SYN and CONT theorems with Quantifiers

From SYN-theorems in Chapter 3:

		Analogues of Quine's Metatheorems:	
T8-311d. Valid[T((∀x) Px) ⇔ T(Pa$_1$ & P$_2$ &…& P$_n$)]			[T3-11,DR8-6d]
T8-312d. Valid[T((∃x) Px) ⇔ T(Pa$_1$ v P$_2$ v…v P$_n$)]		|	[T3-12,DR8-6d]
T8-313d. Valid[T(∀x)(Px & Qx) ⇔ T((∀x)Px & (∀x)Qx)]		ML*140	[T3-13,DR8-6d]
T8-314d. Valid[T(∃x)(Px v Qx) ⇔ T((∃x)Px v (∃x)Qx)]		ML*141	[T3-14,DR8-6d]
T8-315d. Valid[T(∀x)(∀y)Rxy ⇔ T(∀y)(∀x)Rxy]		ML*119	[T3-15,DR8-6d]
T8-316d. Valid[T(∃x)(∃y)Rxy ⇔ T(∃y)(∃x)Rxy]		ML*138	[T3-16,DR8-6d]
T8-317d. Valid[T(∀x)(P & Qx) ⇔ T(P & (∀x)Qx)		ML*157	[T3-17,DR8-6d]
T8-318d. Valid[T(∃x)(P v Qx) ⇔ T(P v (∃x)Qx)]	} "Rules of	ML*160	[T3-18,DR8-6d]
T8-319d. Valid[T(∃x)(P & Qx) ⇔ T(P & (∃x)Qx)]	Passage"	ML*158	[T3-19,DR8-6d]
T8-320d. Valid[T(∀x)(P v Qx) ⇔ T(P v (∀x)Qx)]		ML*159	[T3-20,DR8-6d]
T8-321d. Valid[T(∀x)Px ⇔ T((∀x)Px & (∃x)Px)]			[T3-21,DR8-6d]
T8-322d. Valid[T(∃x)Px ⇔ T((∃x)Px v (∀x)Px)]			[T3-22,DR8-6d]
T8-323d. Valid[T(∃x)(Px & Qx) ⇔ T((∃x)(Px & Qx) & (∃x)Px)]			[T3-23,DR8-6d]
T8-324d. Valid[T(∀x)(Px v Qx) ⇔ T((∀x)(Px v Qx) v (∀x)Px)]			[T3-24,DR8-6d]
T8-325d. Valid[T((∀x)Px & (∃x)Qx) ⇔ T((∀x)Px & (∃x)(Px & Qx))]			[T3-25,DR8-6d]
T8-326d. Valid[T((∃x)Px v (∀x)Qx) ⇔ T((∃x)Px v (∀x)(Px v Qx))]			[T3-26,DR8-6d]
T8-327d. Valid[T(∃y)(∀x)Rxy ⇔ T((∃y)(∀x)Rxy & (∀x)(∃y)Rxy))]			[T3-27,DR8-6d]
T8-328d. Valid[T(∀y)(∃x)Rxy ⇔ T((∀x)(∃y)Rxy v (∃y)(∀x)Rxy))]			[T3-28,DR8-6d]
T8-329d. Valid[T(∀x)(Px v Qx) ⇔ T((∀x)(Px v Qx) & ((∃x)Px v (∀x)Qx)]			[T3-29,DR8-6d]
T8-330d. Valid[T(∃x)(Px & Qx) ⇔ T((∃x)(Px & Qx) v ((∀x)Px & (∃x)Qx)]			[T3-30,DR8-6d]
T8-331d. Valid[T(∀x)(∀y)Rxy ⇔ T((∀x)(∀y)Rxy & (∀x)Rxx)]			[T3-31,DR8-6d]
T8-332d. Valid[T(∃x)(∀y)Rxy ⇔ T((∃x)(∀y)Rxy & (∃x)Rxx)]			[T3-32,DR8-6d]

From CONT-theorems in Chapter 3:

T8-333a. Valid [T(∀x)Px ⇒ TPa		[T3-33,DR8-6a]
T8-334a. Valid [T(∀x)(∀y)Rxy ⇒ T(∀x)Rxx		[T3-34,DR8-6a]
T8-335a. Valid [T(∃x)(∀y)Rxy ⇒ T(∃x)Rxx		[T3-35,DR8-6a]
T8-336a. Valid [T(∀x)Px ⇒ T(∃x)Px]	ML*136	[T3-36,DR8-6a]
T8-337a. Valid [T(∃y)(∀x)Rxy ⇒ T(∀x)(∃y)Rxy]	ML*139	[T3-37,DR8-6a]
T8-338a. Valid [T((∀x)Px v (∀x)Qx) ⇒ T(∀x)(Px v Qx)]	ML*143	[T3-38,DR8-6a]
T8-339a. Valid [T(∀x)(Px v Qx) ⇒ T((∃x)Px v (∀x)Qx))]	ML*144	[T3-39,DR8-6a]
T8-340a. Valid [T(∀x)(Px v Qx) ⇒ T((∀x)Px v (∃x)Qx))]	ML*145	[T3-40,DR8-6a]
T8-341a. Valid [T((∀x)Px v (∃x)Qx) ⇒ T(∃x)(Px v Qx)]	ML*146	[T3-41,DR8-6a]
T8-342a. Valid [T((∃x)Px v (∀x)Qx) ⇒ T(∃x)(Px v Qx)]	ML*147	[T3-42,DR8-6a]
T8-343a. Valid [T(∀x)(Px & Qx) ⇒ T((∃x)Px & (∀x)Qx)]	ML*152	[T3-43,DR8-6a]
T8-344a. Valid [T(∀x)(Px & Qx) ⇒ T((∀x)Px & (∃x)Qx)]	ML*153	[T3-44,DR8-6a]
T8-345a. Valid [T((∀x)Px & (∃x)Qx) ⇒ T(∃x)(Px & Qx)]	ML*154	[T3-45,DR8-6a]
T8-346a. Valid [T((∃x)Px & (∀x)Qx) ⇒ T(∃x)(Px & Qx)]	ML*155	[T3-46,DR8-6a]
T8-347a. Valid [T(∃x)(Px & Qx) ⇒ T((∃x)Px & (∃x)Qx)]	ML*156	[T3-47,DR8-6a]

From SYN-theorems in Chapter 4:

T8-411d. Valid [T(P&Q) ⇔ T~(~Pv~Q)]		[DeM2]	[T4-11,DR8-6d]
T8-412d. Valid [T(PvQ) ⇔ T~(~P&~Q)]	[Df 'v']	[DeM1]	[T4-12,DR8-6d]
T8-413d. Valid [T(P&~Q) ⇔ T~(~PvQ)]		[DeM3]	[T4-13,DR8-6d]
T8-414d. Valid [T(Pv~Q) ⇔ T~(~P&Q)]		[DeM4]	[T4-14,DR8-6d]
T8-415d. Valid [T(~P&Q) ⇔ T~(Pv~Q)]		[DeM5]	[T4-15,DR8-6d]
T8-416d. Valid [T(~PvQ) ⇔ T~(P&~Q)]		[DeM6]	[T4-16,DR8-6d]

T8-417d. Valid [T(~P&~Q) <=> T~(PvQ)] [DeM7] [T4-17,DR8-6d]
T8-418d. Valid [T(~Pv~Q) <=> T~(P&Q)] [DeM8] [T4-18,DR8-6d]
T8-419d. Valid [TP <=> T(PvP)] [v-IDEM] [T4-19,DR8-6d]
T8-420d. Valid [T(PvQ) <=> T(QvP)] [v-COMM] [T4-20,DR8-6d]
T8-421d. Valid [T(Pv(QvR)) <=> T((PvQ)vR) [v-ASSOC] [T4-21,DR8-6d]
T8-422d. Valid [T(P&(QvR)) <=> T((P&Q)v(P&R))] [v&-DIST] [T4-22,DR8-6d]
T8-424d. Valid [T(∃x)~Px <=> T~(∀x)Px] [Q-Exch2] ML*130 [T4-24,DR8-6d]
T8-425d. Valid [T(∀x)~Px <=> T~(∃x)Px] [Q-Exch3] ML*131 [T4-25,DR8-6d]
T8-426d. Valid $[T(Ex_1)\ldots(Ea_n)\sim P<x_1,\ldots,x_n> \;<=> T\sim(\forall x_1)\ldots(\forall x_n)P<x_1,\ldots,x_n>]$
[Q-Exch4] ML*132 [T4-26,DR8-6d]
T8-427d. Valid $[T(\forall x_1)\ldots(\forall x_n)\sim P<x_1,\ldots,x_n> \;<=> T\sim(Ex_1)\ldots(Ex_n)P<x_1,\ldots,x_n>]$
[Q-Exch5] ML*133 [T4-27,DR8-6d]

From CHAPTER 4, SYN- and CONT-theorems (with TF-conditionals).

T8-430d. Valid [T(P ⊃ Q) <=> T~(P&~Q)] [Df '⊃'] [T4-30,DR8-6d]
T8-431d. Valid [T(~PvQ) <=> T(P ⊃ Q)] [T4-31,DR8-6d]
T8-432d. Valid [T(P ⊃ Q) <=> T(~Q ⊃ ~ P)] [T4-32,DR8-6d]
T8-433d. Valid [T(∃x)(Px ⊃ Qx) <=> T((∀x)Px ⊃ (∃x)Qx)] ML*142 [T4-33,DR8-6d]
T8-434d. Valid [T(∃x)(Px ⊃ Q) <=> T((∀x)Px ⊃ Q)] "Rules of ML*162 [T4-34,DR8-6d]
T8-435d. Valid [T(∀x)(Px ⊃ Q) <=> T((∃x)Px ⊃ Q)] Passage" ML*161 [T4-35,DR8-6d]
T8-436a. Valid [T((∃x)Px ⊃ (∀x)Qx) => T(∀x)(Px ⊃ Qx)] ML*148 [T4-36,DR8-6a]
T8-437a. Valid [T(∀x)(Px ⊃ Qx) => T((∀x)Px ⊃ (∀x)Qx)] ML*101 [T4-37,DR8-6a]
T8-438a. Valid [T(∀x)(Px ≡ Qx) => T((∀x)Px ≡ (∀x)Qx)))] ML*116 [T4-38,DR8-6d]
T8-439a. Valid [T(∀x)(Px ⊃ Qx) => T((∃x)Px ⊃ (∃x)Qx)] ML*149 [T4-39,DR8-6a]
T8-440a. Valid [T((∃x)Px ⊃ (∃x)Qx) => T(∃x)(Px ⊃ Qx)] ML*150 [T4-40,DR8-6a]
T8-441a. Valid [T((∀x)Px ⊃ (∀x)Qx) => T(∃x)(Px ⊃ Qx)] ML*151 [T4-41,DR8-6a]

8.2312 Valid C-conditionals from Syn- and Cont-theorems in Chapter 6

T8-611d. Valid [T(P & (P => Q)) <=> T((P & Q &(P => Q))] [T6-11,DR8-6d]
T8-612a. Valid [TP & (P => Q)) => T(P&Q)] [T6-12,DR8-6a]
T8-613a. Valid [T(P => Q) & P) => TQ] 'MP', "Modus Ponens" [T6-13,DR8-6a]
T8-614a. Valid [T(P <=> Q) => T(Q => P)] [T6-14,DR8-6a]
T8-615a. Valid [T(P <=> Q) => T(P => Q)] [T6-15,DR8-6a]
T8-620a. Valid [T((∀x)((Px => Qx) & Px) => T(∀x)Qx)] [T6-20,DR8-6a]
T8-621a. Valid [T((∀x)((Px => Qx) & (∀x)Px) => T (∀x)Qx] [T6-21,DR8-6a]

Also, by the T-Normal Form Theorem on theorems above, or simple U-SUBa,

|= Valid [(TP & (TP => TQ)) <=> ((TP & TQ)&(TP => TQ))] [T6-11,U-SUBa]
|= Valid [(TP & (TP => TQ)) => (TP&TQ)] [T6-12,U-SUBa]
|= Valid [((TP => TQ) & TP) => TQ] 'MP', "Modus Ponens" [T6-13,U-SUBa]
|= Valid [(TP <=> TQ) => (TQ => TP)] [T6-14,U-SUBa]
|= Valid [(TP <=> TQ) => (TP => TQ)] [T6-15,U-SUBa]
|= Valid [((∀x)((TPx => TQx) & TPx) => (∀x)TQx)] [T6-20,U-SUBa]
|= Valid [((∀x)((TPx => TQx) & (∀x)TPx) => (∀x)TQx] [T6-21,U-SUBa]
|= Valid [((~P => Q) & ~P) => Q] [T6-13,U-SUBb('~P'for'P')]
|= Valid [((~TP => TQ) & ~TP) => TQ] [U-SUB('TP'for'P','TQ'for'Q')]
|= Valid [((T~P => TQ) & T~P) => TQ] [T8-613,U-SUBb('~P'for'P','TQ'for'Q')]
|= Valid [((FP => TQ) & FP) => TQ] [Df'F'(twice)]

8.2311 Validity-Theorems from Syn- and Cont-theorems in Chapter 7 and 8

From Axioms and Definitions in the Base of Truth-logic.

T8-701d. Valid [TP <=> (TP & ~FP)] [T7-01,DR6-6d,U-SUB]
T8-702d. Valid [FTP <=> ~TP] [T7-02,DR6-6d,U-SUB]
T8-703d. Valid [T(P & Q) <=> (TP & TQ)] [T7-03,DR6-6d,U-SUB]
T8-704d. Valid [T(P v Q) <=> (TP v TQ)] [T7-04,DR6-6d,U-SUB]
T8-705a. Valid [T((TP & ~TP)vQ)) => TQ] [T7-05,DR6-6a,U-SUB]
T8-706d. Valid [F[P] <=> T~P] [T7-06,DR6-6d,U-SUB]
T8-707d. Valid [0[P] <=> (~TP & ~FP)] [T7-07,DR6-6d,U-SUB]

From Section 7.42121. Theorems derived from Axioms 7-1 and Df'F'

T8-711d. Valid [TP <=> TP] [T7-11,DR6-6d,U-SUB]
T8-712d. Valid [FP <=> FP] [T7-12,DR6-6d,U-SUB]
T8-713a. Valid [TP => ~FP] [T7-13,DR6-6a,U-SUB]
T8-714d. Valid [FP <=> (FP & ~TP)] [T7-14,DR6-6d,U-SUB]
T8-715a. Valid [FP => ~TP] [T7-15,DR6-6a,U-SUB]
T8-716d. Valid [F~P <=> TP] [T7-16,DR6-6d,U-SUB]

From Sect. 7.42122. Theorems for Reduction to Normal Form T-wffs (from Ax, 7-1 to Ax. 7-4)

T8-717d. Valid [FFP <=> ~FP] [T7-17,DR6-6d,U-SUB]
T8-718a. Valid [FP => FTP] [T7-18,DR6-6a,U-SUB]
T8-719a. Valid [TP => FFP] [T7-19,DR6-6a,U-SUB]
T8-720d. Valid [TTP <=> TP] [T7-20,DR6-6d,U-SUB]
T8-721d. Valid [TFP <=> FP] [T7-21,DR6-6d,U-SUB]
T8-722d. Valid [F(P & Q) <=> (FP v FQ)] [T7-22,DR6-6d,U-SUB]
T8-723d. Valid [F(P v Q) <=> (FP & FQ)] (For v-table, Row 9) [T7-23,DR6-6d,U-SUB]
T8-724d. Valid [T(∀x)Px <=> (∀x)TPx] [T7-24,DR6-6d,U-SUB]
T8-725d. Valid [T(∃x)Px <=> (∃x)TPx] [T7-25,DR6-6d,U-SUB]
T8-726d. Valid [~F(∀x)Px <=> (∀x)~FPx] [T7-26,DR6-6d,U-SUB]
T8-727d. Valid [~F(∃x)Px <=> (∃x)~FPx] [T7-27,DR6-6d,U-SUB]
T8-728d. Valid [F(∃x)Px <=> (∀x)FPx] [T7-28,DR6-6d,U-SUB]
T8-729d. Valid [F(∀x)Px <=> (∃x)FPx] [T7-29,DR6-6d,U-SUB]
T8-730d. Valid [~T(∃x)Px <=> (∀x)~TPx] [T7-30,DR6-6d,U-SUB]
T8-731d. Valid [~T(∀x)Px <=> (∃x)~TPx] [T7-31,DR6-6d,U-SUB]

From Section 7.42123. Other Syn- and Cont-theorems from Axioms 7-1 to 7-4

T8-732d. Valid [~FTP <=> TTP] [T7-32,DR6-6d,U-SUB]
T8-733d. Valid [~FFP <=> TFP] [T7-33,DR6-6d,U-SUB]
T8-734a. Valid [T(P ⊃ Q) => (TP ⊃ TQ)] [T7-34,DR6-6a,U-SUB]
T8-735d. Valid [T(P ⊃ Q) <=> T(~Q ⊃ ~P)] [T7-35,DR6-6d,U-SUB]
T8-736d. Valid [(TP ⊃ TQ) <=> (~TQ ⊃ ~TP)] [T7-36,DR6-6d,U-SUB]
T8-737a. Valid [(FP & FQ) => F(P & Q)] (For &-table, Row 9) [T7-37,DR6-6a,U-SUB]
T8-738a. Valid [(TP & TQ) => T(P v Q)] (For v-table, Row 5) [T7-38,DR6-6a,U-SUB]
T8-739a. Valid [(FP & TQ) => T(P ⊃ Q)] (For ⊃ table, Row 6) [T7-39,DR6-6a,U-SUB]
T8-740d. Valid [(TP & FQ) <=> F(P ⊃ Q)] (For ⊃ table, Row 8) [T7-40,DR6-6d,U-SUB]
T8-741d. Valid [F(TP & ~TQ) <=> ~T(TP & ~TQ)] [T7-41,DR6-6d,U-SUB]
T8-742d. Valid [T(~TP v TQ) <=> ~F(~TP v TQ)] [T7-42,DR6-6d,U-SUB]
T8-743a. Valid [T(∀x)(Px ⊃ Qx) => (T(∀x)Px ⊃ T(∀x)Qx] [T7-43,DR6-6a,U-SUB]

From Section 7.42124. Detachment theorems from Ax.7-5)

(Note that TF-Modus Ponens is derivable with a C-conditional as the main connective.)

T8-744a. Valid [(TP & (~TPvTQ)) ⇒ TQ]   “Alternative Syllogism #1”   [T7-44,DR6-6a,U-SUB]
  ⊨ [(TP & (TP ⊃ TQ)) ⇒ TQ]   “TF-Modus Ponens #1”   [T8-744a,T4-31,SynSUB]
T8-745a. Valid [((~TPvTQ) & ~TQ) ⇒ ~TP]   ”Alternative Syllogism #2”   [T7-45,DR6-6a,U-SUB]
  ⊨ [((TP ⊃ TQ) & ~TQ) ⇒ ~TP]   “TF-Modus Tollens #1”   [T8-745a,T4-31,SynSUB]
T8-746a. Valid [(TP & T(~P v Q)) ⇒ TQ]   “Alternative Syllogism #3”   [T7-46,DR6-6a,U-SUB]
  ⊨ [(TP & T(P ⊃ Q)) ⇒ TQ]   “TF-Modus Ponens #2”   [T8-746a,T4-31,SynSUB]
T8-747a. Valid [(T(~P v Q) & FQ) ⇒ FP]   “Alternative Syllogism #4”   [T7-47,DR6-6a,U-SUB]
  ⊨ [(T(P ⊃ Q) & FP) ⇒ TQ]   “TF-Modus Ponens #2”   [T8-747a,T4-31,SynSUB]

From Section 7.42125. Theorems about non-true and non-false expressions, from Df ‘0’

T8-748d. Valid [0P ⇔ ~(TP v FP)]   [T7-48,DR6-6d,U-SUB]
T8-749d. Valid [~0P ⇔ (TP v FP)]   [T7-49,DR6-6d,U-SUB]
T8-750a. Valid [0P ⇒ ~TP]   not-Valid: [~TP ⇒ OP]   [T7-50,DR6-6a,U-SUB]
T8-751a. Valid [0P ⇒ ~FP]   not-Valid: [~FP ⇒ OP]   [T7-51,DR6-6a,U-SUB]
T8-752d. Valid [0~P ⇔ 0P]   0P ⇔ 0~P ⇔ T0P ⇔ F0P   [T7-52,DR6-6d,U-SUB]
T8-753d. Valid [0P ⇔ T(0P)]   ~0P ⇔ ~T0P ⇔ F0P   [T7-53,DR6-6d,U-SUB]
T8-754d. Valid [~F(0P) ⇔ 0P]   [T7-54,DR6-6d,U-SUB]
T8-755a. Valid [(TP v 0P) ⇒ ~FP]   [T7-55,DR6-6a,U-SUB]
T8-756a. Valid [(FP v 0P) ⇒ ~TP]   [T7-56,DR6-6a,U-SUB]
T8-757d. Valid [(0P & 0Q) ⇔ (0(P&Q) & 0(PvQ))]   [T7-57,DR6-6d,U-SUB]
T8-758a. Valid [(0P & 0Q) ⇒ 0(P&Q)]   (For &-table, Row 1)   [T7-58,DR6-6a,U-SUB]
T8-759a. Valid [(0P & 0Q) ⇒ 0(PvQ)]   (For v-table, Row 1)   [T7-59,DR6-6a,U-SUB]
T8-760a. Valid [(0P & 0Q) ⇒ 0(P ⊃ Q)]   (For ⊃-table, Row 1)   [T7-60,DR6-6a,U-SUB]
T8-761d. Valid [0TP ⇔ (~TP & TP)]   [T7-61,DR6-6d,U-SUB]
T8-762d. Valid [0FP ⇔ (~FP & FP)]   [T7-62,DR6-6d,U-SUB]
T8-763d. Valid [00P ⇔ (F0P & ~F0P)]   [T7-63,DR6-6d,U-SUB]
T8-764a. Valid [((T(P&Q)vF(P&Q)v0(P&Q)) & FP) ⇒ F(P&Q)]   [T7-64,DR6-6a,U-SUB]
T8-765a. Valid [((T(P&Q)vF(P&Q)v0(P&Q)) & ~TP) ⇒ ~T(P&Q)]   [T7-65,DR6-6a,U-SUB]
T8-766a. Valid [((T(P&Q)vF(P&Q)v0(P&Q)) & (TP & 0P)) ⇒ (0(P&Q)]   [T7-66,DR6-6a,U-SUB]

From Section 8.221 Syn- and Cont theorems with T-operators and ‘⇒’.

T8-801d. Valid [T (P & Q) ⇔ T(P ⇒ Q)]   [T8-01,DR6-6d,U-SUB]
T8-802d. Valid [T(P&~Q) ⇔ F(P ⇒ Q)]   [T8-02, DR6-6d,U-SUB]
T8-811a. Valid [T(P ⇒ Q) ⇒ TP]   [T8-11, DR6-6a,U-SUB]
T8-812a. Valid [F(P ⇒ Q) ⇒ TP]   [T8-12,DR6-6a,U-SUB]
T8-813a. Valid [T(P ⇒ Q) ⇒ TQ]   [T8-13,DR6-6a,U-SUB]
T8-814a. Valid [F(P ⇒ Q) ⇒ FQ]   [T8-13,DR6-6a,U-SUB]
T8-815a. Valid [(T(P ⇒ Q) & T(Q ⇒R)) ⇒ T(Q ⇒P)]   [T8-15,DR6-6a,U-SUB]
T8-816d. Valid [T(P ⇒ Q) ⇔ T(Q ⇒P)]   [T8-16,DR6-6d,U-SUB]
T8-817d. Valid [T(P ⇒ Q) ⇔ T(TP ⇒TQ)]   [T8-17,DR6-6d,U-SUB]
T8-818d. Valid [T(P ⇒ Q) ⇔ F(P ⇒ ~Q)]   [T8-18,DR6-6d,U-SUB]
T8-819d. Valid [F(P ⇒ Q) ⇔ T(P ⇒ ~Q)]   [T8-19,DR6-6d,U-SUB]
T8-820d. Valid [T(P ⇒ Q) ⇒ ~T(P ⇒ ~Q)]   [T8-20,DR6-6a,U-SUB]
T8-821d. Valid [F(P ⇒ Q) ⇒ ~F(P ⇒ ~Q)]   [T8-21,DR6-6a,U-SUB]
T8-822d. Valid [F(P ⇒ Q) ⇔ F(~Q ⇒ ~P)]   [T8-22,DR6-6d,U-SUB]
T8-823d. Valid [~F(P ⇒Q) ⇔ ~F(~Q ⇒ ~P)]   [T8-23,DR6-6d,U-SUB]
T8-824a. Valid [T(P ⇒ Q) ⇒ ~F(~Q ⇒ ~P)]   [T8-24,DR6-6a,U-SUB]

T8-825d. Valid [F(P ⇒ Q) ⇔ F(P ⊃ Q)] [T8-25,DR6-6d,U-SUB]
T8-826d. Valid [(T(P ⇒ Q) v F(P ⇒ Q)) ⇔ (TP & (TQ v FQ))] [T8-26,DR6-6d,U-SUB]
T8-827a. Valid [~0(P ⇒ Q) ⇒ TP] [T8-27,DR6-6a,U-SUB]
T8-828a. Valid [~0(P ⇒ Q) ⇒ ~0Q] [T8-28,DR6-6a,U-SUB]
T8-829d. Valid [0(P ⇒ Q) ⇔ (~TP v 0Q)] [T8-29,DR6-6d,U-SUB]
T8-830d. Valid [0(P ⇒ Q) ⇔ 0(P ⇒ ~Q)] [T8-30,DR6-6d,U-SUB]
T8-826d. Valid [(T(P ⇒ Q) v F(P ⇒ Q)) ⇔ (TP & (TQ v FQ))] [T8-26,DR6-6d,U-SUB]
T8-827a. Valid [~0(P ⇒ Q) ⇒ TP] [T8-27,DR6-6a,U-SUB]
T8-828a. Valid [~0(P ⇒ Q) ⇒ ~0Q] [T8-28,DR6-6a,U-SUB]
T8-829d. Valid [0(P ⇒ Q) ⇔ (~TP v 0Q)] [T8-29,DR6-6d,U-SUB]
T8-830d. Valid [0(P ⇒ Q) ⇔ 0(P ⇒ ~Q)] [T8-30,DR6-6d,U-SUB]
T8-831a. Valid [(0P & 0Q) ⇒ 0(P ⇒ Q)] [T8-31,DR6-6a]
T8-832a. Valid [(FP & 0Q) ⇒ 0(P ⇒ Q)] [T8-32,DR6-6a]
T8-833d. Valid [(TP & TQ) ⇔ T(P ⇒ Q) [T8-33,DR6-6d]
T8-834d. Valid [(TP & FQ) ⇔ F(P ⇒ Q)] [T8-34,DR6-6d]

8.2322 Valid Conditionals from A-implications in Chapters 7 and 8

From Section 8.2112 - A-implications

Ti8-840. Valid$_I$ [~TP ⇒ 0(P ⇒ Q)] [Ti8-40,DR8-6g]
Ti8-841. Valid$_I$ [FP ⇒ 0(P ⇒ Q)] [Ti8-41,DR8-6g]
Ti8-842. Valid$_I$ [0P ⇒ 0(P ⇒ Q)] [Ti8-42,DR8-6g]
Ti8-843. Valid$_I$ [0Q ⇒ 0(P ⇒ Q)] [Ti8-43,DR8-6g]
Ti8-844. Valid$_I$ [(TP ⇒ TQ) & ~TQ) ⇒ ~TP] [Ti8-44,DR8-6g]

From Section 7.423 A-implication Theorems of Chapter 7

Ti8-780. Valid$_I$ [~TP ⇒ ~T(P & Q)] [Ti7-80,DR8-6g]
Ti8-780a. Valid$_I$ [~TP ⇒ ~T(P ⇒ Q)] [Ti8-780,Ax.8-01,SynSUB]
Ti8-781. Valid$_I$ [FP ⇒ F(P & Q)] [Ti7-81,DR8-6g]
Ti8-782. Valid$_I$ [~FP ⇒ ~F(P v Q)] [Ti7-82,DR8-6g]
Ti8-783. Valid$_I$ [TQ ⇒ T(P v Q)] [Ti7-83,DR8-6g]
Ti8-784. Valid$_I$ [TP ⇒ (TP v FP)] [Ti7-84,DR8-6g]
Ti8-785. Valid$_I$ [FP ⇒ (TP v FP)] [Ti7-85,DR8-6g]

From 7.4234 - A-implication in Q-Theory

Ti8-786. Valid$_I$ [(∀x)TQx ⇒ (∀x)(0Px ⊃ TQx)] [Ti7-86,DR8-6g]
Ti8-787. Valid$_I$ [(∀x)TQx ⇒ (∀x)(TPx ⊃ TQx)] [Ti7-87,DR8-6g]
Ti8-788. Valid$_I$ [(∀x)TQx ⇒ (∀x)(FPx ⊃ TQx)] [Ti7-88,DR8-6g]
Ti8-789. Valid$_I$ [(∀x)~TPx ⇒ (∀x)(TPx ⊃ FQx)] [Ti7-89,DR8-6g]
Ti8-790. Valid$_I$ [(∀x)FPx ⇒ (∀x)(TPx ⊃ FQx)] [Ti7-90,DR8-6g]
Ti8-791. Valid$_I$ [(∀x)TPx ⇒ (∀x)(~TPx ⊃ TQx)] [Ti7-91,R8-6g]
Ti8-792. Valid$_I$ [T(Pa_i) ⇒ T(∃x)Px] [Ti7-92,DR8-6g]
Ti8-793. Valid$_I$ [F(Pa_i) ⇒ F(∀x)Px] [Ti7-93,DR8-6g]
Ti8-794. Valid$_I$ [~T(Pa_i) ⇒ ~T(∀x)Px] [Ti7-94,DR8-6g]
Ti8-795. Valid$_I$ [~F(Pa_i) ⇒ ~F(∃x)Px] [Ti7-95,DR8-6g]

8.3223. Principles of Truth-tables.

For the Truth-table of (P & Q)

T8-758.	Valid [(0P & 0Q) ⇒ 0(P & Q)]	(For &-Row 1)	[T7-58,DR8-6a]
Ti8-7&R2.	Valid$_I$ [(TP & 0Q) ⇒ 0(P & Q)]	(For &-Row 2)	[Ti7-&R2,DR8-6g]
Ti8-7&R3.	Valid$_I$ [(FP & 0Q) ⇒ F(P & Q)]	(For &-Row 3)	[Ti7-&R3,DR8-6g]
Ti8-7&R4.	Valid$_I$ [(0P & TQ) ⇒ 0(P & Q)]	(For &-Row 4)	[Ti7-&R4,DR8-6g]
T8-702.	Valid [(TP & TQ) ⇒ T(P & Q)]	(For &-Row 5)	[Ax.7-02,R8-6d]
Ti8-7&R6.	Valid$_I$ [(FP & TQ) ⇒ F(P & Q)]	(For &-Row 6)	[Ti7-&R6,DR8-6g]
Ti8-7&R7.	Valid$_I$ [(0P & FQ) ⇒ F(P & Q)]	(For &-Row 7)	[Ti7-&R7,DR8-6g]
Ti8-7&R8.	Valid$_I$ [(TP & FQ) ⇒ F(P & Q)]	(For &-Row 8)	[Ti7-&R8,DR8-6g]
T8-737.	Valid [(FP & FQ) ⇒ F(P & Q)]	(For &-Row 9)	[T7-37,R8-6a]

For the Truth-table of (P v Q)

T8-763.	Valid [(0P & 0Q) ⇒ 0(P v Q)]	(For v-Row 1)	[T7-59,DR8-6a]
Ti8-7vR2.	Valid$_I$ [(TP & 0Q) ⇒ T(P v Q)]	(For v-Row 2)	[Ti7-vR2,DR8-6g]
Ti8-7vR3.	Valid$_I$ [(FP & 0Q) ⇒ 0(P v Q)]	(For v-Row 3)	[Ti7-vR3,DR8-6g]
Ti8-7vR4.	Valid$_I$ [(0P & TQ) ⇒ T(P v Q)]	(For v-Row 4)	[Ti7-vR4,DR8-6g]
T8-740.	Valid [(TP & TQ) ⇒ T(P v Q)]	(For v-Row 5)	[T7-38,DR8-6a]
Ti8-7vR6.	Valid$_I$ [(FP & TQ) ⇒ T(P v Q)]	(For v-Row 6)	[Ti7-vR6,DR8-6g]
Ti8-7vR7.	Valid$_I$ [(0P & FQ) ⇒ 0(P v Q)]	(For v-Row 7)	[Ti7-vR7,DR8-6g]
Ti8-7vR8.	Valid$_I$ [(TP & FQ) ⇒ T(P v Q)]	(For v-Row 8)	[Ti7-vR8,DR8-6g]
T8-723.	Valid [(FP & FQ) ⇒ F(P v Q)]	(For v-Row 9)	[T7-23,R8-6d]

For the truth-table of (P ⊃ Q)

T8-760.	Valid [(0P & 0Q) ⇒ 0(P ⊃ Q)]	(For ⊃-Row 1)	[T7-60,DR8-6a]
Ti8-7⊃-R2.	Valid$_I$ [(TP & 0Q) ⇒ 0(P ⊃ Q)]	(For ⊃-Row 2)	[Ti7⊃-R2,DR8-6g]
Ti8-7⊃-R3.	Valid$_I$ [(FP & 0Q) ⇒ T(P ⊃ Q)]	(For ⊃-Row 3)	[Ti7⊃-R3,DR8-6g]
Ti8-7⊃-R4.	Valid$_I$ [(0P & TQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 4)	[Ti7⊃-R4,DR8-6g]
Ti8-7⊃-R5.	Valid$_I$ [(TP & TQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 5)	[Ti7⊃-R5,DR8-6g]
T8-739.	Valid [(FP & TQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 6)	[T7-39,R8-6a]
Ti8-7⊃-R7.	Valid$_I$ [(0P & FQ) ⇒ 0(P ⊃ Q)]	(For ⊃-Row 7)	Ti7⊃-R7,DR8-6g]
T8-740.	Valid [(TP & FQ) ⇒ F(P ⊃ Q)]	(For ⊃-Row 8)	[T7-40,R8-6d]
Ti8-7⊃-R9.	Valid$_I$ [(FP & FQ) ⇒ T(P ⊃ Q)]	(For ⊃-Row 9)	Ti7⊃-R8,DR8-6g]

For the truth-table of (P = > Q):

T8-831.	Valid [[0P & 0Q) ⇒ 0(P ⇒ Q)]	(For ⇒ Row1)	[T8-31,DR8-6a]
Ti8-8 ⇒ R2.	Valid$_I$ [(TP & 0Q) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 2)	[Ti8 ⇒ R2,DR8-6g]
T8-832.	Valid [(FP & 0Q) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 3)	[T8-32,DR8-6a]
Ti8-8 ⇒ R4.	Valid$_I$ [(0P & TQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 4)	[Ti8 ⇒ R4,DR8-6g]
T8-833.	Valid [(TP & TQ) ⇒ T(P ⇒ Q)]	(For ⇒ Row 5)	[T8-33,DR8-6d]
Ti8-8 ⇒ R6.	Valid$_I$ [(FP & TQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 6)	[Ti8 ⇒ R6,DR8-6g]
Ti8-8 ⇒ R7.	Valid$_I$ [(0P & FQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 7)	[Ti8 ⇒ R7,DR8-6g]
T8-834.	Valid [(TP & FQ) ⇒ F(P ⇒ Q)]	(For ⇒ Row 8)	[T8-34,DR8-6d]
Ti8-8 ⇒ R9.	Valid$_I$ [(FP & FQ) ⇒ 0(P ⇒ Q)]	(For ⇒ Row 9)	[Ti8 ⇒ R9,DR8-6g]

Bibliography

Ackerman, W., (See Hilbert, D. and)

Ackermann, W., [1956] "Begrundungeiner Strengen Implikation", *Journal of Symbolic Logic*, v. 21.

Adams, Ernest, [1965] "The Logic of Conditionals", *Inquiry*, Vol 8, pp 166-97.

———, [1975] *The Logic of Conditionals*, Reidel.

———, [1998] *A Primer of Probability Logic*, CLSI Publications.

Ammerman, Robert E., [1965] *Classics of Analytic Philosophy* McGraw-Hill.

Anderson, A.R., and Belnap, N.D., [1962] "Tautological Entailments", *Philosophical Studies*, v. 13.

———, [1975] *Entailment,* Princeton Univrsity Press.

Angell, R.B., [1962] "A Propositional Logic with Subjunctive Conditionals", *J. of Symbolic Logic*, v.27.

———, [1967] "Three Logics of Subjunctive Conditionals", (Abstract) *J. of Sym.Logic*, v.32, pp 556-7.

———, [1973] "A Unique Form for Synonyms in the Propositional Calculus" (abstract), *Journal of Symbolic Logic*, v.38, p. 350.

———, [1978] "Tre Logiche dei condizionali congiuntivi", in *Leggi di natura modalit*à *ipotesi*,(Claudio Pizzi, ed.) Milan,Feltinelli, 1978, pp 156-81.

———, [1981] "Analytic Truth-Tables", (Abstract) *J. of Symbolic Logic*, Vol. 46, No 3, p 677.

———, [1986] "Truth-functional Conditionals and Modern vs. Traditional Logic", *Mind,* pp 210-23.

———, [1989] "Deducibility, Entailment and Analytic Containment", In *Directions in Relevant Logic,* Jean Norman and Richard Sylvan, Editors, Ch. 3, pp 119-143.

Aristotle, *Prior Analytics*, *Posterior Analytics*.

Brody, B.A. and Grandy, R.E., [1989] *Readings in the Philosophy of Science*, Prentice Hall, NJ,2nd Ed.

Carnap, Rudolf, [1936-7] "Testability and Meaning", *Philosophy of Science*, v.3 & v.4 (Reprinted in Feigl, and Brodbeck, *Readings in the Philosophy of Science*, 1953 and in Ammerman, Robert E., [1965] *Classics of Analytic Philosophy*)

———, [1947, 1956] *Meaning and Necessity*, Univ. of Chicago, 2nd Edition, 1956.

———, [1952] "Meaning Postulates", *Philosophical Studies*, 3 (1952), pp 65-73. (Reprinted in Supplement, 2nd Ed., 1956, of Carnap, Rudolf, Meaning and Necessity)

———, [1955] "Meaning and Synonymy in Natural Languages", *Philosophical Studies*,, pp 33-47, (Reprinted in the Supplement of the 2nd Ed., of Carnap, *Meaning and Necessity*, 1956)

Chisholm, R.M., [1946] "The Contrary-to-Fact Conditional", *Mind*, v. 55.

Church, Alonzo, [1956] *Introduction to mathematical Logic*, Vol I, Princeton University Press

Davidson, Donald,- [1960] "Truth and Meaning", Synthese, XVII, No. 3, pp 304-323. (Reprinted in Jay Rosenberg, *Readings in the Philosophy of Language*, 1971)

——, [1967] “Causal Relations”, *Journal of Philosophy*, 64, pp. 691-703.
(Reprinted in Ernest Sosa, (Ed.), *Causation and Conditionals*, Oxford, 1975, pp 82-94).

De Finetti, Bruno, [1936] “La Logique de Probabilité”, Actualités Scientifiques et Industrialles, 391, *Actes du Congrès Internationale de Philosophie Scientifique*, Sorbonne Paris, 1935, IV. Induction and Probabilit, Hermann et CLe, Editeurs. Paris. 1936, pp 31-39.
(English translation by R.B.Angell, as “The Logic of Probability”, *Phil. Studies*, v.77, 1995).

Eaton, Ralph M., [1931] *General Logic*, Charles Scribners.

Feigl, H. and Brodbeck, May, [1953] *Readings in the Philosophy of Science*, Appleton-Century-Crofts.

Feigl, H. and Sellars, Wilfrid, [1949] *Readings in the Philosophical Analysis*, Appleton-Century-Crofts.

Follestal, Dagfin, [1966] “Quantification into Causal Contexts”, in *Boston Studies in the Philosophy of Science*, ii, (Editors, R.S. Cohen and W.W.Wartofsky, New York 1966, pp 263-74.

Frege, Gottlob, [1879] *Begriffsschrift*, Halle (Chapter 1 translated in Geach and Black, (below), pp 1-20).

——, [1891] “Funktion und Begriff”, Address given to the *Jenaische Gesellshaft fur Medecin und Naturwissenschaft* Jan 9, 1891. (Translated in Geach and Black (below) pp 21-41).

——, [1892] “Ueber Sinn und Bedeutung”, *Zeitschrift fur Philosophie und Philosophische Kritik*, vol. 100 , pp 215-252. (Translated in Geach and Black, (below) pp 56-78).

Geach, Peter and Black, Max, [1970] *Translations from the Philosophical Writings of Gottlob Frege*, Basil Blackwell.

Goodman, Nelson, [1947] “The Problem of Counterfactual Conditionals”, *Journal of Philosophy*, v.44 (Reprinted in Goodman, *Fact, Fiction and Forecast*, 1955,1965, and in Linsky, below).

——, [1949] “On Likeness of meaning”, *Analysis*, v.10.
(Reprinted in Linsky, Leonard, *Semantics and the Philosophy of Language).*

——, [1955] *Fact, Fiction and Forecast*, Univerity of London, 1955 , 2nd Ed. 1965.

——, [1965] “New Riddle of Induction”, in *Fact, Fiction and Forecast*, 2nd Ed.

Grandy, Richard E. (See Brody, B.A. and Grandy, R.E.).

Grice, H.P., [1975] “Logic and Conversation” in *The Logic of Grammar*, by Davidson and Harman.

Grice, H.P. and Strawson,P.F., [1956] “In Defense of A Dogma”, *The Philosophical Review*, v. LXV.
Hempel, Carl .G., [1945] “Studies in the Logic of Confirmation”, *Mind*, v.54, pp 1-26
(Reprinted in Brody, B.A. and Grandy, R.E.)

——, [1965] *Aspects of Scientific Explanation*, Free Press.

Herbrand, Jacques, *Logical Writings; Jacques Herbrand*, Warren Goldfarb, (Ed.), Harvard,1971.

Hilbert, D. and Ackerman,W. [1950]*Principles of Mathematical Logic*, Chelsea Publishing Company. NYC (Translation of *Grundzüge der Theoretischen Logik*, 2nd Ed.,1938).

Kim, Jaegwon, [1971] “Causes and Events: Mackie on Causation”, *Journal of Philosophy,* v 68.
(Reprinted in *Causation and Conditionals*, Ernest Sosa (Ed).

Kleene, Stephen Cole, [1952] *Introduction to Metamathematics*,Van Nostrand, Princeton NJ.

Kneale, William and Martha, [1964] *The Development of Logic*, Oxford at the Clarendon Press.

Kolmogorov, A. N., [1950] *Foundations of the Theory of Probability*, New York, 1950.

Kripke, Saul, [1959] “A Completeness Theorem in Modal Logic”, *J. of Symbolic Logic*, v. 24., pp 1-14.

——, [1963] “Semantical considerations on modal logics”, *Acta Philosophica Fennica*.

——, [1963] “Semantical Analysis of modal logic I, Normal propositional calculi”, *Zeitschrift fur mathematische Logik und Grundlagen der Mathematik,* v. 9.

Lewis, C.I., [1912] “Implication and the algebra of logic”, *Mind*, n.s.21 pp. 522-31.

——, [1913] “Interesting theorems in symbolic logic”, *Journal of Philosophy Psychology and Scientific Methods*, v. X, pp 239-42.

——, [1917] “The issues concerning material implication”, *Journal of Philosophy Psychology and Scientific Methods*, v. XIV, pp350-56.

——, [1918] *A Survey of Symbolic Logic*, Univ. of California Press. (see esp. pp 324-39).
——, [1932] and Langford, C.H., *Symbolic Logic,* Appleton Century.
——, [1943] "The Modes of Meaning", *Philosophy and Phenomenological Research*, pp 236-249. (Reprinted in Linsky, Leonard, *Semantics and the Philosophy of Language).*
——, [1946] *Analysis of Knowledge and Valuation*, Open Court..
Lewis, David, [1973] *Counterfactuals,* Harvard University Press.
——, [1976] "Probabilities of Conditionals and Conditional Probabilities", *Philosophical Review*, v 85 .
Linsky, Leonard, (Ed.) [1952] *Semantics and the Philosophy of Language*, University Of Illinois.
Mackie, J. L. [1965] "Causes and Conditions" *American Philosophical Quarterly,* (Reprinted in Sosa (Ed.), *Causations and Conditionals*.)
Mates, Benson, [1950] "Synonymity", *University of California Publications in Philosophy*, 25. (Reprinted in Linsky, Leonard, *Semantics and the Philosophy of Language).*
McCall, Storrs, [1966] "Connexive Implication", *Journal of Symbolic Logic*, v. 31, pp83-90.
Moore, G.E., [1922] *Philosophical Studies*, Routledge, and Kegan Paul.
Moore, G.E., [1920] "External and Internal Relations", *Proceedings of the Aristotelian Society*, 1919-20, (Reprinted in G.E.Moore, *Philosophical Studies*, Routledge and Kegan Paul, 1922).
Nelson, E.J., [1930] "Intensional Relations", *Mind*, n.s., v.39, pp 440-453.
——, [1933] "On Three Logical Principles in Intension", *The Monist*, v.43, pp 268-284.
Nicod, Jean, [1924] *Le probleme logique de l'induction*, Paris. (Translated by P.P.Weiner as *Foundations of Geometry and Induction*, London, 1930.)
Norman, Jean and Sylvan,Richard (Editors) [1989] *Directions in Relevant Logic*, Kluwer Academic Dordrecht, Netherlands.
Parry, W.T., [1932] "Implication", *Summaries of Ph.D. Dissertations*,Harvard Grad.School, pp 332-5.
——, [1933] "Ein Axiomen system fur eine neue Art von Implikation", *Ergebnisse eines math Kolloquiums*, Heft 4 (1933), pp 5-6.
Pollock, J.L., [1976] *Subjunctive Reasoning*, Reidel.
Quine, Willard Van Orman, [1940] *Mathematical Logic*, Harvard University Press, Revised Edition 1951, tenth printing of the revised edition, 1983.
——, [1941] *Elementary Logic*, Revised Edition, 1964, Harper.
——, [1953] *Methods of Logic*, Harvard University Press, 1953.
——, [1959] *Methods of Logic,* 2nd Ed, Henry Holt, 1950.
——, [1972] *Methods of Logic,* 3nd Ed, Holt, Rinehart & Winston, 1972.
——, [1982] *Methods of Logic,* 4nd Ed, Harvard University Press, 1982.
——, [1951] "Two Dogmas of Empiricism", *Philosophical Review* v. 60, (1951), pp 20-43. (Reprinted in W.V.O.Quine, *From a Logical Point of View*, Harvard, 1953).
——, [1953] "Three grades of Modal Involvement", *Proceedings of the XIth International Congress of Philosophy*, Brussels, 1953, Volume 14 (Amsterdam: North-Holand Publishing Co.) (Reprinted in Quine, *Ways of Paradox*, Random House, 1966).
——, [1953] "Mr. Strawson on Logical Theory", *MIND*, v. 62. (Reprinted in Quine, *Ways of Paradox*, Random House, 1966)
——, [1953] *From a Logical Point of View*, Harvard University Press.
——, [1960] *Word and Object*, M.I.T.Press.
——, [1970] *Philosophy of Logic*, Prentice-Hall
Reichenbach, Hans, [1954] "Nomological Statements and Admissible Operations".
Rosenberg, Jay, [1971] *Readings in the Philosophy of Language*, Prentice-Hall
Rosser, J. Barkley, [1953] *Logic for Mathematicians*, McGraw-Hill.

Routley, R. and Montgomery, H., [1968] "On Systems Containing Aristotle's Thesis", *Journal of Symbolic Logic*, v.33, pp 82-96.

Russell, Bertrand [1918] "The Philosophy of Logical Atomism", *The Monist*. (Reprinted in Russell, B., *Logic and Knowledge,* George Allen & Unwin, 1956, pp 177-282).

———, 1913, 1927 With Whitehead, A.N, *Principia Mathematica*,Vol 1, Cambridge.

Ryle, Gilbert [1949] *The Concept of Mind*, Hutchinson's Universal Library, London.

Sosa, Ernest, (Ed.) [1975]*Causation and Conditionals*, Oxford University Press.

Stalnaker, R.C.,and Thomason, R.C., [1970] "A Semantic Analysis of Conditional Logic", *Theoria,* v.36.

Stalnaker, R.C., [1968] "A Theory of Conditionals", in *Studies in Logical Theory* (ed. N.Rescher).

Stevenson, C.L., [1970] "If-iculties", *Philosophy of Science*, v.37 (1970), pp 27-49.

Strawson, P.F.,[1952] *Introduction to Logical Theory*, Methuen, London, 1967 (first published 1952).

———, [1950] "On Referring", *Mind*, v. LIX (1950).

Suranyi, Janos, [1950] *Acta Mathematica Academiae Scientiarum Hungaricae*, Vol 1 (1950), pp 261-271.

Tarski, Alfred, [1941] *Introduction to Logic and to the Methodology of Deductive Sciences,* Oxford University Press.

———, "The Semantic Conception of Truth", *Philosophy and Phenomenological Research*,v.4. (Reprinted in Feigl and Sellars, above, and in Linsky, Leonard, *Semantics and …)*

van Fraassen, Bas C., [1966] "Singular Terms, Truth-value Gaps, and Free Logic", *J. of Philosophy*, v.63.

Thomason, Richmond H., [1970] *Symbolic Logic; An Introduction*, MacMillan, London.

Wittgenstein, Ludwig, [1921] *Tractatus Logico-Philosophicus*, Translation by Pears and McGuiness, Routledge,& Kegan Paul, 1961.

Whitehead, A.N. and B. Russell, [1913] *Principia Mathematica*,Vol.1,Cambridge,England; 2nd Ed, 1927.

Subject Index

Name Index