

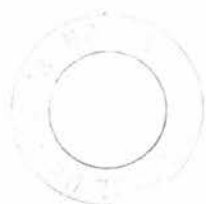
SEMANTICS IN A FREGE STRUCTURE

FAIROUZ DIB KAMAREDDINE

Ph.D.

UNIVERSITY OF EDINBURGH

1988



**IN MEMORY OF MY MOTHER JULIETTE ELAYASS**

### ABSTRACT

This thesis is concerned with the foundations of natural language semantics. Two issues of central importance in semantic theory, namely intensionality and nominalisation, form its central themes. We offer a perspective in which the semantics of natural language constructs are unpacked in terms of Peter Aczel's Frege structures. Along the way we investigate other issues in natural language semantics such as generalised quantifiers, truth, the question of types and the intensional/extensional distinction. This work starts by assessing the foundational problems of the semantics of nominalisation which are classified as mathematical and logical. A new solution to these problems based on Frege structures is offered and is shown to provide promising results for both nominalisation and intensionality. We illustrate how this general framework (using Frege structures) throws light on many puzzling semantic issues.

### DECLARATION

I declare that this thesis has been composed by myself and that the work reported in it is my own.

F.D.KAMAREDDINE

*ACKNOWLEDGEMENTS*

I would like to express my warmest thanks to my supervisor Ewan Klein. He has been throughout my thesis most encouraging, patient and supportive.

Many thanks also go to Patrick Blackburn who did a most efficient job reading and commenting on my thesis. His work on that was very selfless and unforgettable.

I would like also to thank Peter Aczel, Inge Bethke, George Dunbar, Kit Fine, Furio Honsell, Marcus Kracht, David McCarty, Irene Orr, Henry Thompson and Ray Turner for either their helpful comments or having read parts of my thesis.

I cannot find words to say how much I needed Lesley Parker throughout my last two years at Edinburgh. Without her, I would not have been able to survive the toughness of life at times. A special thanks too to Peggy Coonagh who always listened very warmly, to Erica Traill and Elaine Morais for their support and to Peter and Moyra Higgins who will always be very special to me.

Many thanks also to Terry Myers who tried to find every solution to enable me to overcome the problems. Also thanks to the members of the Cowan House Committee at Pollock Halls for their understanding and valuable support.

Finally, my thanks to the following bodies who provided me with finance:

- The Lebanese University,
- The Arabic-British Chamber Foundation,
- The Edinburgh University Women Group,
- The Cowan House Committee,
- The Overseas Research Scheme,
- The University of Edinburgh.

*NOTATIONS*

I use the following notations:

$\Rightarrow$  stands for meta-implication.

$a=b \text{---} \Rightarrow d, e$  stands for If b is true then a=d else a=e.

Bold face words are used to denote objects inside the model rather than inside the formal theory; for example the term John of the formal theory has a denotation **John** in the model.

*SUMMARY OF THE WORK*

This work is divided into seven chapters and an appendix as follows:

*Chapter 1.* This chapter consists of two parts:

*Part A:* Two problems of the semantics of nominalisation are considered. The semantic interpretation of nominalisation requires both a theory and a model: from the theoretical point of view, we are challenged by Russell's paradox, whereas from the other (model existence), we are threatened by Cantor's diagonal argument. (I am not claiming here that they are two separate problems; indeed our syntax and semantics should be tightly related. No inconsistent theory can have a model and sometimes, although a theory is consistent, we cannot readily see what the models look like.)

*Part B:* The two problems explained in part A are problems of theories of predication (set theories) and of models of the lambda calculus respectively. We survey some of the solutions offered so far to overcome these problems in the general sense (i.e. as set theories and models of the lambda-calculus), and then discuss briefly their applications to nominalisation. The solutions to the theoretical problem are to restrict one of the following: the logic, the language or the axioms. Frege structures restrict the logic and thereby solve the theoretical problem, but they have not been applied to nominalisation before. For the problem of model existence, the solutions discussed are Scott domains and Aczel's Frege structures; only Scott domains have previously been applied to nominalisation.

This chapter finds Frege structures to be suitable candidates for solving both problems. One of their advantages, as we shall see in Chapter 2, is their full comprehension principle, which allows us to avoid restricting our nominalised formulae while still evading Russell's paradox. Another advantage (in addition to their being a solution to Cantor's argument) is that they are easy to work with because we can visualize them; they are elegant and simple.

*Chapter 2.* This chapter consists of two parts, as follows:

*Part A:* We start with an informal introduction to Frege structures concentrating on mathematical and philosophical motivations. A more formal account of the structures and of how they can be built is then given; finally, a comparison with Scott domains is made.

*Part B:* A new theory,  $T_{\Omega}$  which has in it a predicate for propositionhood is introduced. The semantics and the proof theory are discussed in detail and it is shown that this theory is sound and complete.

*Chapter 3.* In this chapter we discuss the property theory obtained in this thesis and study the domain of decidable properties. We also introduce a concept of predication which is distinctive from functional application. The concept of truth is dealt with and various familiar theories of truths are accommodated within the framework.

*Chapter 4.* In this chapter we discuss determiners and quantifiers concentrating on the internal definability of determiners and showing that the lack of second order quantification is harmless.

*Chapter 5.* In this chapter we consider the problems of intensionality; Montague's incomplete success in dealing with propositional attitudes was due to his use of a weakly intensional approach based on possible world semantics. Our approach is highly intensional and this allows many problems of belief sentences to be solved. We discuss the problem of trying to construct extensionality out of intensionality, and whether there is a congruence relation which defines extensionality while enabling us to remain consistent. Possible worlds and modalities are also discussed within this highly intensional framework.

*Chapter 6.* This chapter deals with type theory inside our type free framework. We build domains inductively inside a Frege structure and build a typed theory  $T_{\text{pol}}$

where the various types take denotations in the constructed domains. We also present a small Montague fragment of English which deals with nominalisation and intensionality.

*Chapter 7.* In this chapter we summarise the work and compare the concept of quantification in both Frege structures and Scott domains.

*Appendix I:* This appendix is a self-contained introduction to Dana Scott's domain  $E_{\infty}$  showing it can be built with the property that  $E_{\infty} \cong [E_{\infty} \rightarrow E_{\infty}]$ .



TABLE OF CONTENTS

ABSTRACT	3
DECLARATION	3
ACKNOWLEDGEMENTS	4
NOTATIONS	5
SUMMARY OF THE WORK	6
TABLE OF CONTENTS	9
CHAPTER 1. SET THEORY AND NOMINALISATION	14
PART A. THE PROBLEMS	14
A.1 The problem of the formal theory	14
A.1.1 Ontology, Concepts, Predicates, Properties, Sets	14
A.1.2 A language of nominalisation and the problem	20
A.1.3 The ontological status of sets	24
A.2 The problem of the existence of the models	27
A.2.1 What a model should look like	27
A.2.2 Difficulties with such models	31
A.2.3 Existence of models	31
PART B. THE DIFFERENT ATTEMPTS AT A SOLUTION	33
B.1 Solution to the theoretical problem	33
B.1.1 Notes on set theory	33
B.1.1.1 Altering the language	33
B.1.1.2 Altering the axioms	35
B.1.1.3 Altering the logic	39
B.1.1.4 Feferman and the foundational issues	43
B.1.2 Effects of set theory on nominalisation	44
B.1.2.1 Language and nominalisation	44
B.1.2.2 Axioms and nominalisation	45
B.1.2.3 Logic and nominalisation	47
B.1.2.4 The place of logic in the above applications	48
B.2 Solution to model existence	50
B.2.1 Lambda calculus and its models	50
B.2.1.1 Frege structures	52
B.2.2 Using those models for nominalisation	53
B.2.2.1 Scott domains and nominalisation	53
Conclusion and Comparison with Cocchiarella	54
CHAPTER 2. FREGE STRUCTURES AND NOMINALISATION	57
PART A. SUMMARY OF FREGE STRUCTURES	57
A.1 Informal introduction	58
A.2 The models	62
A.3 Frege structures as models and comparison with Scott domains	74
PART B. A THEORY OF PROPERTIES, ITS SEMANTICS AND PROOF THEORY	79
B.1 The theory $T_{\Omega}$	80
B.2 The metatheory of $T_{\Omega}$	86
B.3 The semantics of $T_{\Omega}$	92
B.4 Soundness and completeness	94

CHAPTER 3. A THEORY OF PROPERTIES AND THEORIES OF TRUTH	102
PART A. A THEORY OF PROPERTIES	102
A.1 Closure conditions on properties	102
A.2 Decidable properties	111
PART B. THEORIES OF TRUTH	117
B.1 Various truth theories	120
CHAPTER 4. DETERMINERS AND QUANTIFIERS IN A FREGE STRUCTURE	126
PART A. TWO EXAMPLES OF DETERMINERS	127
PART B. NON INTERNAL DEFINABILITY	129
PART C. THE DETERMINER "the"	132
PART D. HOW TO SHOW SOMETHING IS A DETERMINER	133
PART E. CHARACTERISTICS OF DETERMINERS AND QUANTIFIERS	135
PART F. NO LOSS OF QUANTIFICATION WITH FIRST ORDER THEORIES	138
CHAPTER 5. INTENSIONALITY AND EXTENSIONALITY USING A FREGE STRUCTURE	141
PART A. AN INTENSIONAL SOLUTION TO PROPOSITIONAL ATTITUDES	141
PART B. AN EXTENSIONAL STRUCTURE	148
PART C. THOMASON'S APPROACH	158
PART D. POSSIBLE WORLDS	160
CHAPTER 6. TYPE THEORY AND THE MONTAGUE FRAGMENT IN A FREGE STRUCTURE	162
PART A. MONTAGUE'S IL	162
PART B. A TYPE THEORY $T_{pol}$	165
PART C. INTERPRETING $PTQ$ in $T_{pol}$	172
CHAPTER 7. SUMMARY AND COMPARISON	185
PART A. QUANTIFICATION	186
PART B. FURTHER ADVANTAGES	190
APPENDIX I. FOUNDATIONS OF DOMAIN THEORY	194
BIBLIOGRAPHY	228

INTRODUCTION

*Nominalisation:* Semanticists tend to define nominalisation as being the process which transforms what acts semantically as a predicate into something which acts semantically as an object and is subject to predication. The following pairs of sentences illustrate the phenomenon of nominalisation:

- (1a). The dress is red.
- (1b). Red is a colour.
- (2a). The necklace is diamond.
- (2b). Diamond is a stone.
- (3a). John went home.
- (3b). I love going home.
- (4a). Mary swims regularly.
- (4b). Swimming in the warm Mediterranean is something I miss.
- (5a). John runs regularly
- (5b). To run is fun.
- (6a). I always disliked John.
- (6b). That I always disliked John used to upset my mother.
- (7a). Bill is honest.
- (7b). Honesty is a virtue.

There are many other criteria that should be taken into account by a theory of nominalisation. For example, linguists have drawn attention to a semantical distinction between infinitives as in (5b) and gerunds as in (3b) and (4b).<sup>1</sup> It has also been argued that bare plurals and mass nouns are cases of nominalisation, but it is not clear whether they act in such a way that a single theory could fit both. However, these distinctions are not going to be taken into account in this thesis. I intend instead

---

<sup>1</sup> Cf [CH3].

to assume the above definition of nominalisation and work out some applications of my proposal to its semantics.

*Intensionality:* Both extensional and weakly intensional theories (e.g. Montague's PTQ) face a problem concerned with propositional attitudes. The problem can be illustrated as follows:

Consider the two concepts groundhog and woodchuck.

According to extensional interpretations, we have:

for any  $x$ ,  $\text{groundhog}(x)$  is true iff  $\text{woodchuck}(x)$  is true.

Also, according to possible worlds semantics, we have in any possible world:

for any  $x$ ,  $\text{groundhog}(x)$  is true iff  $\text{woodchuck}(x)$  is true.

Hence, according to extensional theories, the two concepts groundhog and woodchuck are the same and therefore:

(1) John believes that  $\text{groundhog}(a) \rightarrow$  John believes that  $\text{woodchuck}(a)$ .

Also, according to possible world semantics, two concepts are the same if they hold of the same objects in the same possible worlds. Therefore according to possible world semantics (1) above is true.

From a certain perspective on the nature of belief statements, namely one which insists that John might believe that something is a groundhog without believing it to be a woodchuck, this state of affairs is unacceptable.

The approach that I put forward in this thesis is highly intensional, and hence throws some light on the above problem. The solution can be summarised as follows: groundhog and woodchuck are two propositional functions which give equivalent values for all objects, but this equivalence does not entail equality. According to our approach, two objects cannot be equal unless they are the same object. They can both have the same truth value but this will not make them equal as objects.<sup>2</sup> Working

---

<sup>2</sup> Note that our approach is not committed to non-extensionality of functions and we assume our construction of a Frege structure to be based on extensional domains such as  $E_\infty$ .

with intensional problems creates a host of interesting puzzles, and I try to accommodate solutions to many of these in the thesis. For instance, the concept of truth is given a lot of attention, as is the definition of extensionality in terms of intensionality and the relationship between Montague's intensional semantics and the one proposed here.

## CHAPTER 1. SET THEORY AND NOMINALISATION

The main thesis of this chapter is that the basic problems of nominalisation are those of set theory. We shall therefore explain the problems of set theory and their various solutions; we shall then assess the influence of these matters on nominalisation.

### PART A. THE PROBLEMS

Let us start by examining the problem of the semantics of nominalisation. I shall look at it from two angles, the first related to the formal theory, the second concerned with the existence of models.

#### *A.1. The problem of the formal theory*

Any theory of nominalisation<sup>3</sup> which is to be interpreted should be accompanied by some ontological views on concepts - for predicates and open well-formed formulae act semantically as concepts. This is vague, however, if only because where I use the word concept, someone else might use class, predicate, set, property or even system (Dedekind). This terminological profusion is hardly surprising, for we are touching on the problem of universals, a problem philosophers have been debating for hundreds of years. (This new term - universal - may be more confusing than any of the others, but we may use Aristotle as a preliminary guide and define a universal to be that which can be predicated of things.) The aim of this section is not to take a standpoint on any of the philosophical theories of universals; rather it is to show that, no matter what approach we adopt, nominalisation is going to generate a problem.

*A.1.1 Ontology, concepts, predicates, properties and sets:* It does seem that the main

---

<sup>3</sup> I consider nominalisation here to be defined as in the introduction to this thesis.

problem of nominalisation is an old problem of set theory. If one takes an open sentence, according to Quine in [QU3], page 1,

"the notion of a class is such that there is supposed to be, to the various things of which that sentence is true, a further thing which is the class having each of those things and no others as member."

As an example we take the sentence being an x such that the colour of x is red. We have in our universe various things of which this sentence is true; but perhaps we can also say that the class of all those things which are red also exists in our universe. I say perhaps because it will be shown shortly that if we let any open sentence determine an object which is the class of all those things of which the sentence is true, we run into difficulties.

To see this clearly it is important that the reader bear in mind the following four notions: the Comprehension Principle, Quantification, Interpretation and Russell's Paradox. I shall comment here on how each such notion is to be understood in the present context.

*The Comprehension Principle:* This is the principle which decides which open sentence in our theory determines a class (or set) of precisely those entities that satisfy it.

*Quantification:* Take a class which stands for an open sentence (i.e. the class of all those objects which when substituted for the free variables in the open sentence returns true). Does this class act exactly like any other object in our universe? If so, should we be able to quantify over it?

*Interpretation:* Should we keep to a full classical interpretation or use a non-classical one? If we keep to a full classical interpretation, and assume that the comprehension principle applies to each open sentence and that we have full quantification, we will fall foul of Russell's paradox. (It is worth mentioning, however, that the paradox does not occur only under the classical interpretation but under many other interpretations, as we shall see later.)

*Russell's paradox:* The paradox derives from assumptions similar to the following: Let S be the set of all sets that do not contain themselves. Such an assumption is contradictory for we can deduce from it that S is in S iff S is not in S.

The important point to concentrate on is how these four notions interact, and in particular to note that an assumption of full comprehension (i.e. every open sentence determines a class) and of full quantification (i.e. every class acts exactly like any object and can be quantified over) will, under some interpretations, lead to Russell's paradox. This point will be presented in more detail in the following section.

We will now describe the four main conceptions of universals, all of which will have to face up to this sort of problem.

1. *Realistic conception (Platonism):* Platonists take concepts to be real properties. That is, concepts are language/observer independent entities. Platonists also subscribe to an unrestricted (or full) comprehension principle, i.e. to each well defined condition, there exists a set (or class) of all entities satisfying the condition. Moreover, this set is an entity in its own right and can be quantified over. According to this conception, interpretation is much more important than language and therefore it seems obligatory to use the referential interpretation.

2. *Formalist conception (Nominalism):* Formalists, of whom Hilbert was the father, insist on the paramount importance of language.<sup>4</sup> According to the formalists, concepts are predicate expressions which do not exist beyond our linguistic expressions. Open sentences are excluded from standing for concepts, and furthermore the comprehension principle is restricted. As language is the most important thing for them, interpretation is secondary. Thus it seems that the

---

<sup>4</sup> Hilbert's program, as it is well known, consisted of separating signs and meaning and only allowing finitary arguments in the proof theory. Had the program worked, it would have made it easy to prove things about the theory inside the theory itself. Gödel's result made apparent the impossibility of carrying out this aim - and as has been said by Quine:

"Gödel's proof is beyond doubt, we can philosophise about it but we can not philosophise it away."



obvious semantics should be based on a substitutional interpretation. (A substitutional interpretation of the quantifiers involves truth clauses of the following kind:

$[[ \exists x \Phi ] ]_{S,g}$  is true  $\leftrightarrow$  for some name  $\underline{a}$  in the language,  $[[ \Phi[a/x] ] ]_{S,g}$  is true.

$[[ \forall x \Phi ] ]_{S,g}$  is true  $\leftrightarrow$  for every name  $\underline{a}$  in the language,  $[[ \Phi[a/x] ] ]_{S,g}$  is true.

By contrast, Referential interpretation treats quantifiers as follows:

$[[ \exists x \Phi ] ]_{R,g}$  is true  $\leftrightarrow$  for some object  $\underline{a}$  in the model,  $[[ \Phi[a/x] ] ]_{R,g}$  is true.

$[[ \forall x \Phi ] ]_{R,g}$  is true  $\leftrightarrow$  for every object  $\underline{a}$  in the model,  $[[ \Phi[a/x] ] ]_{R,g}$  is true.)

3. *Conceptualism*: Borrowing a sentence from Fraenkel (at the end of [FR2], page 336):

Conceptualists are

"attracted neither by the luscious jungle flora of platonism nor by the ascetic desert landscape of neo-nominalism."

Concepts here are neither predicate expressions nor real properties. They are not objects but unsaturated entities, the saturation of which results in a mental act and not necessarily a truth value. Some conceptualists are constructive and construct only those sets that correspond to predicative conditions; some others accept an unrestricted comprehension principle. However all of them care for interpretation, and in a semantics for a conceptualistic theory one should consider a referential interpretation where the meaning of a concept applied to an object does not necessarily have to be a truth value.

4. *Fregean conception*: It might be said that Frege is both a realist and a conceptualist but; he is anti-formalist and tends to lean towards conceptualism. The ontology assumed by Frege of concepts was that they are functions of one argument whose values are always truth-values. Concepts, according to him, are unsaturated, and the behaviour of a concept is predicative even if something is being asserted about it. The unsaturation of a concept comes from the fact that concepts can never themselves be objects and only by applying the concept to an object can we obtain a saturated

element (an object which is a truth-value). Assertions that are made about concepts do not apply to objects: for example, existence is a property of concepts and not of objects. However, the way we attach properties to concepts consists in predicating the property not of the concept but of the concept-correlate. This concept-correlate is the extension of the concept, according to Frege, and is an object. We said that concepts here are functions: thus the graphs of functions are objects even though functions themselves are not. This is exactly the case with concepts and their extensions. The extensions are objects but the concepts themselves are not. The extension of a concept does not fully determine the concept, for we can have two extensions which are the same while the concepts themselves are not. Frege always warned against confusing a concept with its extension and defined sets and classes to be the extensions of the concepts, not the concepts themselves:

"sets and classes are objects whereas concepts are anything but objects."

Something falls under a concept and the grammatical predicate stands for this concept. A name of an object is incapable of being used as a grammatical predicate. For Frege, the saturation of a concept results in a truth-value and according to him each open sentence denotes a class. Those classes are objects and can be quantified over. Being an anti-formalist he insisted on interpretation, but as is well known he paid a high price for these relaxed conditions: his theory, known as the naive theory, was found to be subject to Russell's paradox, since the concept set of all those things that do not belong to themselves has an extension K which is a proper object. Thus his theory is contradictory.

These then are the four main conceptions of universals. In constructing a theory of nominalisation corresponding to any of those conceptions, we have to embody its distinctive features either in the language or in the interpretation or both. However problems can occur in any of these theories if we are not careful about the way we bind together the comprehension principle, the quantification techniques and the

interpretation (no matter what philosophical background we assume). To avoid inconsistency, some people restricted their comprehension principle but still allowed unlimited quantification; others restricted both quantification and comprehension. Yet if one is not careful in setting out the theory, a paradox can still be derivable.

I shall in most of this thesis use Frege's views on concepts and objects, with a relaxation of both comprehension and quantification, without falling into Russell's paradox: Aczel's Frege structures enable us to maintain this stance while staying paradox free. I adopt Frege's conception for three reasons. The first is that all the scholars whose results I intend to compare with my own seem to have used it. The second is that formalism has lost its attraction after Gödel's famous results. The third reason is that the Fregean conception seems to be a solution between conceptualism and realism, and I do not have anything against either of the two latter conceptions. Adopting the Fregean stance means that I am committed to defining nominalisation as the phenomenon of turning what was at one stage a predicate into something which will act as an object - something to which properties can be attributed. This new object is different from our initial objects and will act as a concept-correlate in our semantics. What in the language acts as a name should be mapped in the semantics as an object. What in the language acts as a predicate should be mapped as a concept. And what in the language acts as a nominalised predicate should be mapped as a concept-correlate.

Once again, it is vital to keep in mind that a concept-correlate is an object and not a concept. As an example, consider the syntactic discourse which has the following: tall (a predicate), being tall (the nominal of tall), John (a name). The semantic universe has: a property **tall** which holds of all the tall things in the universe, a set A of all tall things and a man **John** called John. Then we have the following semantic interpretation: tall is mapped into **tall**, being tall is mapped into A and John is mapped into **John**.

We shall now examine in detail how the Russell paradox can threaten theories of nominalisation; and in section B we shall meet the solutions to the problem.

*A.1.2. A language of nominalisation:* If we are going to assume a first order language of nominalisation and we are going to let any open well-formed formula stand for a concept, then we might fall into the paradox. This is shown as follows: take a first order calculus and add to it a new primitive relation  $\in$  and the two axioms:

*Comprehension:* For each open well-formed formula  $\Phi$ ,

$\exists y \forall x [(x \in y) \leftrightarrow \Phi(x)]$  where  $y$  is not free in  $\Phi(x)$ .

*Extensionality:*  $\forall x \forall y [\forall z [(z \in x) \leftrightarrow (z \in y)] \rightarrow x = y]$ .

This theory is obviously inconsistent, for take  $\Phi(x)$  to be  $\neg(x \in x)$ . Then we get:

$\exists y \forall x [(x \in y) \leftrightarrow \neg(x \in x)]$

$\implies$

$\forall x [(x \in y) \leftrightarrow \neg(x \in x)]$

$\implies$

$[(y \in y) \leftrightarrow \neg(y \in y)]$

Is<sup>5</sup> it the assumption that the class  $x$  exists? In this theory of nominalisation, we assume that each open well-formed expression determines a concept whose extension exists and is the set of all those elements which satisfy the concept. We could restrict our comprehension principle so that  $\Phi(x)$  stands for everything except  $\neg(x \in x)$ ; but this will not save us from paradox. To see this let  $\Phi(x)$  stand for  $\neg(x \in_2 x)$  where

$(x \in_2 y)$  abbreviates  $(\exists z)((x \in z) \& (z \in y))$ .

Again, ruling out this instance is not enough for we will still get the paradox if we take  $\Phi(x)$  to be  $\neg(x \in_3 x)$ . This process continues ad infinitum. We could rule out all such instances - but the problem will persist, for take a sentence  $\Phi(x)$  like:

$\neg(\exists z_1, z_2, \dots) [\dots (z_3 \in z_2) \& (z_2 \in z_1) \& (z_1 \in x)]$

and let  $y$  be the class obtained from the comprehension axiom for  $\Phi(x)$ .

<sup>5</sup> Note here that the axiom of extensionality did not have any role in the proof of the inconsistency.

If  $(y \in y)$  then  $\neg (\exists z_1, z_2, \dots) [\dots \in z_2) \& (z_2 \in z_1) \& (z_1 \in y)]$ .

But we can take  $z_1 = z_2 = \dots = y$ , and get a contradiction.

If  $\neg (y \in y)$  then  $(\exists z_1, z_2, \dots) [\dots \in z_2) \& (z_2 \in z_1) \& (z_1 \in y)]$ .

But as  $(z_1 \in y)$  then  $\Phi(y)$ ; but we have that  $\neg \Phi(y)$ . Contradiction.  $\square$

We have assumed above a first order language of nominalisation. Although I shall leave the discussion of whether we need higher order languages for later chapters, allow me to remark en passant that it seems we do not need to go higher than second order languages for the semantics of nominalisation - for according to Frege's conception, we stop at second level concepts, but these can be mapped into first order concepts which in turn can be mapped into objects. So when we come to quantify over properties, we really quantify over their extensions which are objects. We shall discuss quantification in Frege structures in more detail in Chapter 4, but we shall here try to answer the question of whether we still face the problem with higher order languages. I cannot find a better way to show that we do than by looking at a second order theory due to Cocchiarella. This language essentially embodies Frege's conceptions of concepts and objects summarised above, according to which we need to quantify over our predicates, and predicate quantifiers have a referential significance, even though predicates themselves are not singular terms. I shall start by writing down the axioms and rules of a second order language which will accommodate nominalised predicates. If this language is to allow us to talk about nominalisation, it should have a device which can turn any sentence, open wff (well formed formula) or predicate into a singular term. For example, we should turn run into to run, the sun is grey into that the sun is grey and so on. I shall add such a device to the language and refer to it by  $\lambda$ . As I said earlier the language used here is based on Cocchiarella's formulation of second order logic with nominalised predicates and will be used to illustrate the problem.

*The typing of the language is as follows:*

0 represents the type of all singular terms,

1 represents the type of propositions,

$n+1$  represents the type of  $n$ -place predicates.

For each  $n > 0$  assume the existence of denumerably many variables. I shall use the following metavariables:

$M, N, \dots$  refer to both individual and predicate variables

$F_n, G_n, H_n, \dots$  refer to  $n$ -place predicate variables, We can get rid of the subscript when no confusion occurs.

$x, y, z, w, \dots$  refer to individual variables.

$a, b, \dots$  refer to singular terms.

*The primitive symbols of the language are:*  $\rightarrow, \neg, =, \forall, \lambda$ . The others are defined in the metalanguage.

The meaningful expressions of any type  $n$ ,  $ME_n$  are defined recursively as:

(1) Every individual variable is in  $ME_0$ .

Every  $n$ -place predicate is in both  $ME_0$  and  $ME_{n+1}$ .

(2) For  $a, b$  in  $ME_0$ ,  $(a = b)$  is in  $ME_1$ .

(3)  $F$  in  $ME_{n+1}$ ,  $a_1, \dots, a_n$  in  $ME_0 \implies F(a_1, \dots, a_n)$  in  $ME_1$ .

(4)  $\Phi$  in  $ME_1$  and  $x_1, \dots, x_n$  are pairwise distinct variables, where  $n \geq 1$ ,  
 $\implies [\lambda x_1 \dots x_n \Phi]$  is in  $ME_{n+1}$ .

(5)  $\Phi$  in  $ME_1 \implies \neg \Phi$  in  $ME_1$ .

(6)  $\Phi, \Psi$  in  $ME_1 \implies (\Phi \rightarrow \Psi)$  is in  $ME_1$ .

(7)  $\Phi$  in  $ME_1$  and  $a$  is an individual or predicate variable  
 $\implies \forall a \Phi$  is in  $ME_1$ .

(8)  $\Phi$  in  $ME_1 \implies [\lambda \Phi]$  in  $ME_0$ .

(9) For all  $n > 1$ ,  $ME_n$  is included in  $ME_0$ .<sup>6</sup>

---

<sup>6</sup> Note that 9 does not follow from 1

AXIOMS:

(A0\*) All tautologous well formed formulae.<sup>7</sup>

(A1\*)  $\forall u(\Phi \rightarrow \Psi) \rightarrow (\forall u\Phi \rightarrow \forall u\Psi)$

where  $u$  is an individual or a  
predicate variable.

(A2\*)  $\Phi \rightarrow \forall u\Phi$  where  $u$  is an individual or a  
predicate variable not free in  $\Phi$ .

(A3\*)  $\exists x (a = x)$  where  $a$  is singular term in which  
 $x$  is not free.

( $\lambda L^*$ )  $(a = b) \rightarrow (\Phi \leftrightarrow \Psi)$  where  $a, b$  are singular terms and  
 $\Psi$  comes from  $\Phi$  by replacing one or  
more free occurrences of  $b$  by free  
occurrences of  $a$ .

(CP\*)  $\exists F_n \forall x_1 \dots x_n [F_n(x_1, \dots, x_n) \leftrightarrow \Phi]$   
where  $F_n$  does not occur free in  $\Phi$   
and  $x_1, \dots, x_n$  are distinct vars.

( $\lambda$ -CONV\*)  $[\lambda x_1, \dots, x_n \Phi] (a_1, \dots, a_n) \leftrightarrow \Phi(a_1/x_1, \dots, a_n/x_n)$   
where  $a_1, \dots, a_n$  are singular terms  
and each  $a_i$  is free for  $x_i$  in  $\Phi$ .

(ID $\lambda^*$ )  $[\lambda x_1, \dots, x_n R(x_1, \dots, x_n)] = R$   
where  $R$  is an  $n$ -place predicate  
variable or constant.

*Inference Rules:* The two inference rules are MP and UG.<sup>8</sup>

Note that (CP\*) is an instance of (CP $\lambda^*$ ) where :

(CP $\lambda^*$ )  $\exists F_n ([\lambda x_1, \dots, x_n \Phi] = F_n)$  where  $F_n$  is not free in  $\Phi$ .

<sup>7</sup> Classical tautologies.

<sup>8</sup> Modus Ponens and Universal Generalisation, where MP is: infer from  $\Phi \rightarrow \Psi$  and  $\Phi$  that  $\Psi$ . UG is infer from  $\Phi$  that  $\forall x\Phi$ .

The system (just described) is subject to Russell's paradox, for take the special instance of:

$$(CP^*): \exists F \forall x [F(x) \leftrightarrow \neg x(x)],$$

one can then derive:  $F(F) \leftrightarrow \neg F(F)$ .

So the system is inconsistent and we need ways of making it consistent. In part B, we shall come to solutions for such a problem. For the moment, note that the presence of (CP\*) is necessary for second-order logics with nominalised predicates and that the problem comes from (CP\*) together with (A3\*) under various logical laws. We shall see in part B the different solutions that have been offered and the effects on nominalisation. However, before closing this section, I would like to comment on the ontological status of sets and on the nature of Russell's paradox, as the solutions depend on both issues.

*A.1.3. The ontological status of sets:* There are two main views of sets: the mathematical conception of set and the logical conception. According to the mathematical conception, a set is determined by the elements that belong to it. E.g.  $\{1,2,3\}$  is the set of the numbers 1 2 and 3. The logical conception, on the other hand, regards sets as existing according to their defining concepts, and not their constituent objects; so here  $\{1,2,3\}$  might be the set of positive integers less than 4. Frege's conception of set was a logical one, and is known in the literature as the naive conception of set. According to this view, any predicate has an extension and sets are extensions of predicates. However, under the classical laws of logic and especially the law of excluded middle (LEM) and non-free logic (where not necessarily each element denote), this notion of set is subject to Russell's paradox.<sup>9</sup> I shall illustrate the occurrence of the paradox by assuming both LEM and that every predicate has an extension. Now, if one chooses  $P(x)$  to be  $\neg(x \in x)$ , then  $\{x: \neg(x \in x)\}$  is an  $r$  to which LEM applies. So we have either  $(r \in r)$  or  $\neg(r \in r)$ . In both cases we get a contradiction.

---

<sup>9</sup> However, the paradox holds even in minimal logic and other non-classical logics, e.g. we can derive the paradox without the use of LEM which means that the paradox is intuitionistically derivable.



So the theory contains a paradox (a contradictory statement is provable in it) even though the axioms seem true and the rules of inference valid. We get a theory that is inconsistent even though we were very careful in building it.

After Frege's naive set theory was shown to be inconsistent, set theorists were anxious to solve the problem, and many directions were followed to overcome the paradox. Frege himself had something to say about the paradox. He stated that if one abandoned the naive conception and the use of full comprehension, it would not be obvious how to define numbers (see [FR3], Frege on Russell's paradox). This follows because the essential definition of numbers in Frege's theory was based on the existence of extensions of concepts - thus the paradox shook Frege's whole theory. Frege suggested that the solution lay in either banishing LEM for classes, or forbidding some concepts from having extensions. He was not satisfied with the first solution because he wanted classes to be full objects - and full objects obey LEM. If classes are to be considered as improper objects then this will create an infinite number of types in the theory, for we are going to have functions that apply to proper, improper or mixed arguments. Frege was not in favour of that solution, and preferred to acknowledge the existence of concepts that have no extensions. This would affect axiom (V)<sup>10</sup> (in [FR3], which Frege was not satisfied with from the beginning) and in particular (Vb) which is:<sup>11</sup>

$$(Vb) \quad z'f(z) = z'g(z) \implies \forall x (x \text{ falls under } f \leftrightarrow x \text{ falls under } g)$$

This axiom states that if two concepts are equal in extension then whatever falls under one falls under the other. Frege made only general remarks about the inconsistency and did not pin down what caused the problem. He sometimes felt the problem lay in (Vb) and at other times thought that the assumption of the existence of an extension to each concept was to blame. (Va)<sup>12</sup> is acceptable as it takes us from

<sup>10</sup>  $(V) \quad z'f(z) = z'g(z) \iff \forall x (x \text{ falls under } f \leftrightarrow x \text{ falls under } g)$ , where  $z'f(z)$  stands for the extension of  $f$ .

<sup>11</sup> See [FR3], pages 214-224 for a good account of the following discussion

<sup>12</sup> I.e. the opposite direction of (Vb):  $\forall x (x \text{ falls under } f \leftrightarrow x \text{ falls under } g) \implies z'f(z) = z'g(z)$ .

equality that holds in general to an equality that holds of graphs (or extensions). But according to Frege (in [FR3] page 219),

"We cannot in general take the words

the function  $\Phi(\xi)$  has the same graph as the function  $\Psi(\xi)$

to mean the same thing as the words

the functions  $\Phi(\xi)$  and  $\Psi(\xi)$  always have the same value for the same argument;

and we must take into account the possibility that there are concepts with no extension (...)."

However, Frege did not realise that his domain of concepts was far too big. Concepts are propositional functions but according to Frege's conception, there are far more propositions than there should be. For each object  $a$ ,  $\neg a$  (the content of  $a$ ) is a proposition even though  $a$  was not. Thus Frege has far too many concepts and some paradoxical sentences stand for concepts when they should not do. Accordingly, a way of ruling out the paradox might be to restrict the number of concepts. Let us look again at the paradoxical sentence: the set of all things that do not belong to themselves. Under the restriction strategy, we cannot tell whether this sentence stands for a concept or not, as we do not know if this is a propositional function or not so we cannot think of its extension.

We could say that there were two ways of reformulating set theory. One is to abandon Frege's definition of set and use the mathematical notion of set. The second is to keep to the logical definition of set and try to make it consistent. We shall, in part B, see reformulations of set theory in both directions.

To conclude this section, it is worth drawing attention to the role self reference plays in these set theoretic paradoxes. Paradoxes involving self reference are well known in the literature, and are of two kinds: logical and semantical paradoxes. Russell's paradox has been classified under the logical category, as have the barber's paradox

and Cantor's paradox. So far we have not said anything about the semantical paradoxes. As they are important to intensionality I shall illustrate them with some examples:

*Grelling's paradox:* Some adjectives possess the property that they denote (e.g. English, Polysyllabic) and some do not (eg French). Call the second type heterological; then

heterological is heterological iff heterological is not heterological.

Another example of this paradox is: A concept is predicable if it can be predicated of itself, otherwise it is impredicable. Hence,

impredicable is impredicable iff impredicable is not impredicable.

Another very important semantical paradox is

*The liar's paradox:* Assume that John Doe utters on December 1st, 1970 the following English sentence and nothing else all day:

"The only sentence uttered by John Doe on December 1st, 1970 is false."

This sentence is true iff it is false.

Paradoxes of this sort should not lead us to reject self-reference, which is needed for many disciplines. We have to find a solution which will allow self-reference without any contradiction.

### *A.2. The problem of the existence of models*

The theory discussed in A.1. is inconsistent, so it does not have models. But even in the case of a theory whose consistency we are sure of, we still sometimes cannot imagine what the models look like. This section describes what a model of nominalisation should be, and what the difficulties of constructing such models are.

*A.2.1. What a model should look like:* A model of nominalisation will be roughly as follows:  $M = \langle U, P, f \rangle$  where  $U$  is the domain of objects,  $P$  is the domain of functions from  $U$  into  $\{0,1\}$  and  $f$  is the nominalisation function.  $f$  is a function from  $P$  into  $U$

which should be injective. This implies that  $P$  is a subset of  $U$  up to an isomorphism.

Let me describe in more detail what this means. In trying to build our semantic function which maps each syntactic entity into a semantic one, we should do the following:

(1) Map individual variables and singular terms into objects in  $U$ .

(2) Map the predicates into  $P$ , the domain of the first order properties. The nominalised items are singular terms and they are mapped into  $U$ . The function  $f$  acts as a nominalisation function, assigning to each element  $p$  of  $P$ , an element in  $U$  called the correlate of  $p$ . This correlate is the denotation of the nominalised item that corresponds to the predicate.

*$f:P \rightarrow f(P)$  is an isomorphism because:*

(i)  $f$  is a well defined function: We assume that each property has a single correlate.

(ii)  $f$  is injective: We assume that each two distinct properties in  $P$  have distinct correlates in  $f(P)$ .

(iii)  $f$  is surjective: Because every element in  $f(P)$  corresponds to an element in  $P$ .

So in constructing a model of nominalisation, we should construct three domains such as  $U$ ,  $P$  and  $f(P)$  satisfying the condition that  $P$  (or  $f(P)$ ) is a subset of  $U$ . According to Cantor's diagonal theorem, we cannot take  $P$  to be the set of all functions from  $U$  to  $\{0,1\}$ . We have to restrict  $P$ , but we should not restrict it too much, for we would like to obtain the nominalisation of all the desired items.

In the above construction of  $f$ , I assumed that two distinct predicates have distinct nominals. It should not be assumed that this requires the principle of extensionality to hold in the domain  $P$ . If we have two predicates which are both true of the same objects but are distinct then their extensions, and hence nominalisations, must also be distinct. In fact one of the main issues in a Frege structure is, as Aczel

puts it in [AC3]:

"The point is that extensional equality between sets must not be confused with the equality relation between sets as objects."

The above construction of  $f$  in no way assumes extensionality. To see this consider the following example: take the two concepts positive integer less than or equal to 2 and positive integer which divides 2. We know that the extension of the first concept contains 1, 2 and only those numbers. The extension of the second concept contains also 1, 2 and only those. However, this in no way implies that the two extensions as objects are the same. So, although the theory of properties itself is intensional, the principle of extensionality still holds of predicates. In [AC6] for example, this line is assumed. There, properties are propositional functions, yet the predication of a property to an individual is not necessarily function application except in the case where the property  $P$  is itself basic. In this theory, although properties should be treated intensionally, we can still assume extensionality on predicates. This is because the predication of a property  $P$  to an object  $a$  is not always the application of the propositional function  $P$ . It is only application when  $P$  is basic, which is fine because we obviously know everything about basic properties. (Basic properties are things like red, tall, etc whereas non-basic properties are those obtained from open formulas.)<sup>13</sup> I hope by now it is obvious that assuming that the function  $f$  above is injective does not entail assuming extensionality of properties: our example [AC6] demonstrates this; if this is still not clear, the reader can refer to part C of Chapter 2.

I am trying to say here that two distinct predicates have two distinct nominals, yet we can assume that two sets have the same elements without being equal as objects (i.e. the principle of extensionality does not hold for sets). I am also interested in having extensionality between functions and the reasons for that are two:

---

<sup>13</sup> Clearly models of the above sort exist if one does not require that there should be denotations for all open wffs.

(1) Things get quite complicated if we did not have extensionality on functions, see for instance [SC1].

(2) Extensionality on functions facilitates the identification of properties with classes.

[BE6] has many examples of theories where the principle of extensionality is assumed on functions yet the theory of properties is intensional. Also Aczel's work in [AC6] concludes that properties are propositional functions yet predication is functional application only when the property is basic. The main reason that Aczel gave to defend his thesis was to do with intensionality. Take

$$\text{S-PFT: } \forall x(\text{pred}(P,x) = \Phi) \iff (P = \hat{x}\Phi).$$

This principle which is rejected for reasons of intensionality, asserts that predication and property abstraction are inverses of each other. Aczel also presents an alternative operator  $\text{pred}'$ , where

$$\forall x(\text{pred}'(P,x) = \Phi) \iff (P = \hat{x}\Phi).$$

And this  $\text{pred}'$  is really functional application.

Some might argue here that there is no relation between intensionality and the distinction between predication and functional application. For instance, in [SC2], Scott only wanted the three axioms:

$$(\alpha) \lambda x.t = \lambda y.t[y/x]$$

$$(\beta) (\lambda x.t)(y) = t[y/x]$$

$$(\gamma) \lambda x.t = \lambda x.t' \iff \forall x.(t=t')$$

and did not insist on  $(\epsilon) : t(x) = t'(x) \rightarrow t = t'$ . But in there, even though he did not study intensionality, he made a distinction between predication and application. Also<sup>14</sup> one can build a fine grained meaning algebra of the Carnap-Lewis type starting from extensional primitive entities. This is no reason however to deny that in [AC6], the distinction between predication and functional application was due to

---

<sup>14</sup> This was drawn to my attention by Uwe Mönnich.

intensionality matters.

*A.2.2 Difficulties with such models:* Cantor's Theorem will pose a difficulty to any theory which aims to make functions play the role of objects. According to Cantor's theorem, the cardinality of a function space is bigger than the cardinality of the domain itself. Cantor's argument goes as follows:

*Cantor's theorem:* Given any finite or infinite transfinite cardinal, there exists a greater one. More precisely, if S is any set, then the set PS whose elements are all the subsets of S has a greater cardinality than S. (PS is the power set of S and we know that the power set of any set is isomorphic to the set of all the functions from S into {0,1}).

Proof:

If  $\Phi: S \rightarrow PS$  is bijective then the elements are classified in two categories: The first is when s belongs to  $\Phi(s)$  and the second is when s does not belong to  $\Phi(s)$ . Let  $A = \{s: \neg s \in \Phi(s)\}$ . We have that  $A \in PS$  therefore  $A = \Phi(a)$  for some a. Hence  $a \in A$  iff  $\neg a \in A$ . Contradiction.  $\square$

The above argument does not only apply to the characteristic functions of S. We can also prove that for any V with at least two distinct elements, the set of functions from V to V has greater cardinality than V. The proof is as follows:<sup>15</sup>

Assume that V is isomorphic to  $[V \rightarrow V]$ , i.e. there exists an F from V to  $[V \rightarrow V]$  which is bijective. Let us define G:  $V \rightarrow V$  such that:

$$\begin{aligned} G(x) &= 1 \text{ if } F(x)(x) = 0 \\ &= 0 \text{ if } F(x)(x) = 1. \end{aligned}$$

We have assumed that V contains {0,1} (Actually, we could take any two distinct elements of V). As F is surjective, there exists a v in V such that  $F(v) = G$ . This implies that:  $G(v) = 1$  iff  $G(v) = 0$ , which is absurd.  $\square$

*A.2.3 Existence of models:* The above shows that we are going to have problems

---

<sup>15</sup> It may be sufficient here just to point out that  $2^V \subset V^V$  and this implies that  $\text{card}(V^V) > \text{card}(V)$ .

constructing models of nominalisation - recall that we previously wanted  $P$  to be a subset of  $U$ , but by Cantor's theorem the cardinality of  $P$  is greater than that of  $U$ . In essence, we need to find ways of restricting  $P$  without either lapsing into triviality or running foul of Cantor's theorem. That is, we are looking for interesting restrictions - restrictions which leave us with enough functions for nominalisation. We must break the ties created by the old tradition and build somewhat more original models. In part B, we shall talk about different ways of proving the existence of non-trivial models which are not susceptible to Cantor's argument. Those models will contain denotations for all nominalised items. Scott models and Frege structures both possess this property; but as we shall see, the former have a difficulty regarding quantification, while the latter do not.



*PART B. THE DIFFERENT ATTEMPTS AT A SOLUTION*

We have seen the problems of the semantics of nominalisation from the theoretical side and from the perspective of those issues pertaining to model existence. In this part we meet some of the solutions that have been offered to those problems, and comment briefly on their application to nominalisation.

*B.1. Solution to the theoretical problem*

We said that the theoretical problem is mainly a problem of set theory and of predication theory. The following is a summary of various set theories and their application to the development of theories of nominalisation. This summary looks at these issues from three different angles. The first has to do with the language of the theory, the second is concerned with the axioms and the third deals with logic.

*B.1.1 Notes on set theory:*

*B.1.1.1. Altering the language:* Since Russell's letter to Frege, concerning the inconsistency of Frege's system, there have been many attempts at overcoming the paradox. The first two accounts of avoiding the paradox by restricting the language were due to Russell and Poincaré. They both disallowed impredicative specification: only predicative specification (as will be defined below) was to be permitted. Russell's own solution (in [RU1]) was to adopt the vicious circle principle which can be roughly stated as follows:

"No entity determined by a condition that refers to a certain totality should belong to this totality."

Poincaré (in [PO1]) took refuge in banning "les définitions non prédicatives" which were taken by him to be:<sup>16</sup>

---

<sup>16</sup> Definitions by a relation between the object to be defined and all individuals of a kind of which either the object itself to be defined is supposed to be a part or other things that cannot be themselves defined except by the object to be defined.

"Définitions par une relation entre l'objet à définir et tous les individus d'un genre dont l'objet à définir est supposé faire lui-même partie ou bien dont sont supposés faire parties des être qui ne peuvent être eux-même définis que par l'objet à définir".

So both Russell and Poincaré required only predicative sets to be considered, where  $A = \{x: \Phi(x)\}$  is predicative iff  $\Phi$  contains no variable which can take  $A$  as a value.<sup>17</sup> Russell's and Poincaré's solution was to use predicative comprehension, instances of which start with individuals, then generate sets, then new sets and so on as in the following example: Take 0 at level 0,  $\{0, \{0\}\}$  at level 1,  $\{0, \{\{0\}\}, \{0, \{0\}\}\}$  at level 2 and so on. Russell's simple theory of types in Principia Mathematica applied the vicious circle principle, assuming all the elements of the set before constructing it. This theory obviously overcomes the paradox, but it is rather unsatisfactory, for the following reasons:

1. We need formulas which are not stratified (i.e. where we have impredicativity), and there are many sets we would like to have but cannot be provided within this theory.
2. A class can have members only of uniform type. Also, sets here can neither belong to themselves, nor contain other sets from the same level.
3. There are infinite series of universal classes, one for each level; but no one unique universal set.
4.  $-x$  (the complement of  $x$ ) comprises all members of  $x$  of next lower type than  $x$ ; and not everything that does not belong to  $x$ .
5. There is an infinite number of null classes, one for each level.
6. Boolean algebra is reproduced in each type.
7. Numbers are no longer unique as we have different sets of natural numbers at each level.

---

<sup>17</sup> This helps because it is otherwise very easy to get a vicious circle fallacy if we let the arguments of a certain propositional function (or the elements of a set) presuppose the function (or the set) itself.

Note however, that the theory offered by Russell is quite different from what we know today as the simple theory of types. For Russell, a sentence is to be placed (within a context) into a hierarchy according to  $\epsilon$ , so  $x \in y$  would be acceptable if the variable  $x$  is going to take values of lower range than that of  $y$  and formulae like  $(x \in x)$  or  $((x \in y) \ \& \ (y \in x))$  would be unacceptable. Those sentences that satisfy this requirement of ' $\epsilon$ ' are called stratified and Russell only accepts stratified formulae. This is how the paradox is overcome, for the sentence  $\Phi$  denoting  $\neg(y \in y)$  is not stratified.

Here I should stop to explain the concept of stratification for it is going to form an important step in our discussion and assessment of our theory in terms of the others. There are two types of stratification: homogeneous stratification and heterogeneous stratification. Frege and Russell used stratification in the second sense but Cocchiarella stuck to the first type. A well formed formula  $\Phi$  is said to be heterogeneously stratified if there is a function  $f$  from the variables and constants of  $\Phi$  to the natural numbers such that for each atomic well formed formula  $F(x_1, \dots, x_n)$  of  $\Phi$ ,

$$f(F) = 1 + \max [f(x_i)].$$

$\Phi$  is said to be homogeneously stratified if the function  $f$  is further restricted so that  $f(x_i) = f(x_j)$  for  $0 \leq i, j \leq n$ . As an example of a non-stratified formula we take: It is nice to be nice. We also take John loves Mary and running as an heterogeneously stratified formula and John loves Mary as an homogeneously stratified one.

*B.1.1.2 Altering the axioms:* We can avoid the paradox by altering not the language but the axioms of the theory. The most straightforward such theory is ZF (Zermelo-Fraenkel) where the axioms are made to fit the limitation of size doctrine; that is, sets are not allowed to get too big too quickly. Take the system of first order logic provided in A.1, and alter comprehension to the following:

For each open well formed formula  $\Phi$ ,

$$\exists x \forall y [(y \in x) \leftrightarrow (y \in z) \ \& \ \Phi] \text{ where } x \text{ does not occur in } \Phi.$$

It is exactly this new axiom which is responsible for the elimination of the paradoxes. Take Russell's paradox: to prove the existence of  $\{x: \neg(x \in x)\}$  we need a  $z$  big enough so that  $\{x: \neg(x \in x)\}$  is included in  $z$ . But we cannot show the existence of such a  $z$ .

Russell's paradox is restricted in ZF as follows:

Take  $\Phi(t)$  to be  $\neg(t \in t)$ ,

take  $n = \{t: (t \in x) \& \neg(t \in t)\}$

If  $n \in n \implies (n \in x)$  and  $\neg(n \in n)$  contradiction,

If  $\neg(n \in n) \implies$  if  $n \in x \implies n \in n$  contradiction,

if  $\neg n \in x$  then we are fine.

So the limitation of size doctrine exemplified by the above axiom is how we avoid the paradox.<sup>18</sup>

It is worth pointing out that although very different conceptually, both the simple theory of types and ZF give rise to an iterative concept of set. That is, both require the elements of a set be present before a new set can be constructed. (For a precise formulation of the iterative conception of set, and a proof that ZF is a typical iterative theory, see [BO1])

ZF is not the only axiomatic approach aimed at restricting the paradoxes. In NF (New Foundations), Quine restricts the axiom of comprehension of A.1.2, to obtain the following:

SCP:  $\exists y \forall x [(x \in y) \leftrightarrow \Phi(x)]$  where  $x$  is not free in  $\Phi(x)$

and  $\Phi(x)$  is stratified.

Thus it applies only to stratified formulae and now the only concepts that are allowed to have extensions are the concepts that correspond to these stratified formulae. In ZF, we did not have a universal set whereas in NF we do, for take  $\underline{x=x}$ , this is a

---

<sup>18</sup> To avoid the paradox, we do not accept very comprehensive sets.

stratified formula.

NF has only one universal set, one complement of each set, and one null set. Furthermore, Cantor's theorem does not hold in NF (the universal set is equinumerous to its power set.)<sup>19</sup> However, NF is said to lack motivation because its axiom of comprehension is justified only on technical grounds and one's mental image of set theory does not lead to such an axiom. To overcome some of the difficulties, Quine adopted similar measures to B-G (Bernays-Gödel) set theory. Like B-G, ML contains a bifurcation of classes into elements and non-elements. Sets can enjoy the property of being full objects whereas classes cannot. ML was obtained from NF by replacing SCP by two axioms, one for class existence and one for elementhood. The rule of class existence provides for the existence of the classes of all elements satisfying any condition  $\Phi$ , stratified or not. The rule of elementhood is such as to provide the elementhood of just those classes which exist for NF. Therefore, the two axioms of comprehension of ML are:

*The axiom of comprehension by a set:*

$$(1) \exists y \forall x (x \in y \leftrightarrow \Phi(x)),$$

where  $\Phi(x)$  is a stratified formula with set variables only  
in which  $y$  does not occur free.

*The axiom of impredicative comprehension by a class:*

$$(2) \exists y \forall x (x \in y \leftrightarrow \Phi(x)),$$

where  $\Phi(x)$  is any formula in which  $y$  does not occur free.

ML was liked both for the manipulative convenience we regain in it and the symmetrical universe it furnishes. It was however proved subject to the Burali-Forti paradox: (The well ordered set  $\omega$  of all ordinals has an ordinal which is greater than any member of  $\omega$  and hence is greater than  $\omega$ .)

---

<sup>19</sup> However, NF is weak for mathematical induction and the axiom of choice is not compatible with NF. We cannot prove Peano's axiom  $[s(n) = s(m) \rightarrow n = m]$  in it, unless we assume the existence of a class with  $m+1$  elements.

*Suggestions for making ML consistent:*

(1) Fitch suggests staying within the dependable realism where we are assured of consistency and expanding it as far as possible. Fitch's suggestion comes under non-standard logic which we shall meet in the next paragraph.

(2) The other is Black's suggestion to construct a system similar to ML but in which the contradictions are no longer derivable. Wang's system P fits this program.

Wang keeps to ML except that the axiom (1) is restricted further to the requirement that  $\Phi$  should not only be stratified as in NF and ML, but  $\Phi$  should also be normal, where a normal formula is one in which all bound variables are element variables (so quantification is restricted). Wang claims that P is the system Quine originally intended, and proved P consistent relative to NF.

Note that the axiom of infinity can be proved in P and that everything provable in type theory is also provable in P. Note also that Burali-Forti's paradox is no longer provable in P; this is because, to prove such paradox, non-normal formulae are used, but these are excluded in P.

Our description above of Russell's type theory, ZF set theory and Quine's NF and ML, has been brief, but should suffice to convince the reader of the need to have as many sets as one can. It has been argued by those who favour the iterative conception of set that we do not need self-application (see [BO1]). But we have seen the necessity of type-free theories and the development of many type free systems such as Feferman's (in [FE2] and [FE9]). Kripke's work on the theory of Truth [KR1] is further evidence that we should not rule out self referential statements and that we must look for a theory which allows for it. Gödel's work and especially his proof of the incompleteness theorems, showed that self-referential statements are as legitimate as arithmetic: and is not set theory the domain with which we study such statements?

Natural language is full of self-reference and self-application like: There is nothing more beautiful than beauty. All this points to the need for as many sets as possible, including sets that belong to themselves.<sup>20</sup>

*B.1.1.3. Altering the logic:*

*Rejection of the law of excluded middle:* The paradox we faced was of the form:

$$(x \in x) \leftrightarrow \neg(x \in x).$$

Clearly the paradox can be avoided by dropping the assumption of LEM that any one place predicate either applies to a given object or does not. Fitch offered a system which did just that. Note that here we can stick to two valued logics and that this system is not necessarily intuitionistic. If we go back to the example of impredicative specification given at the beginning of this section, according to this approach we can assume the existence of R, the set of all elements which do not contain themselves. What we cannot do though is assume that we have either  $(R \in R)$  or  $\neg(R \in R)$ .

*Many valued logics:*  $(x \in x) \leftrightarrow \neg(x \in x)$  would not be contradictory if a consistent set of truth values was chosen. Consider as an illustration a three valued logic where the truth values are 0 (truth), 1(false) and u(undefined). The above sentence<sup>21</sup> is not contradictory for we associate with  $(x \in x)$  the value u and we define in the semantics that the negation of u is u. Therefore  $u \leftrightarrow \neg u$  is not contradictory and the paradox is avoided. Note here that there are many three valued interpretations and that the status of u varies from one interpretation to another. For some, u acts as not yet known, for others it is undefined. If we take the view that u is not yet known then we can order our models according to the state of our knowledge. Knowledge is cumulative whereas ignorance is not. What we know up to a stage, will always

---

<sup>20</sup> All the above set theories reject the impredicative specifications and assumptions of classes and class existence, except ML which assumes impredicative clauses due to axiom (2) above. However, both the axiomatic approaches and type theoretic approaches to set theory are in need of a model which is infinite, and we do not know how to construct one in such a way as to avoid the antinomies.

<sup>21</sup> According to some interpretations of  $\leftrightarrow$ , this sentence has no truth value; this is the case in Kleene's 3-valued logic.

remain known after that stage, but we will also know more things. Domains looked at in this way are ordered and the fixed point theorem is applicable; this enables the construction of the limit model which is a model of the limit of our knowledge. Such an ordering of domains is very useful for Artificial Intelligence and Computer Science but is problematic for the semantics of nominalisation. Note in passing that not all the 3-valued interpretations would allow us to have a full comprehension principle.

*Intuitionistic logic:* Intuitionists banish metaphysics from their (mathematical) theories. Although for them all objects are abstract, they are constructive: existence is equated with being creatable by constructive methods. However the demand for constructive evidence is not a sign of limitation, for intuitionists have some secure means to construct not only the finite objects but also the infinite. The domain of objects or of the mathematical properties of these objects is not fixed in advance; things in their universe are incomplete and will remain incompletable (Gödel).<sup>22</sup> So existence is constructibility and the ways of constructions are not known a priori. Objects and their properties are mental constructions. Language is not important; for it is vague and ambiguous - even if it is a formal one.<sup>23</sup> Note that Russell's theory of types is itself constructive if we neglect the axioms of infinity and of reducibility.

According to the intuitionists there are two ways to build sets, either by constructing their elements (species) or by characterising a property of their elements (spreads). We can only admit  $x$  to be an element of  $S$  (spreads or species) if  $x$  has or might have been constructed before  $S$ . In the case of a species  $S$ , an object is a member of  $S$  if it has been or might have been defined before  $S$ , and which satisfies the condition  $S$ .

Intuitionists reject classical mathematics and the law of excluded middle. Their

---

<sup>22</sup> This concept of incompleteness is best illustrated by a quotation from Poincaré (in [PO1]): "Quand je parle de tous les nombres entiers, je veux dire tous les nombres entiers qu'on a inventés et tous ceux que l'on pourra inventer un jour... et c'est ce "que l'on pourra" qui est l'infini". When I speak of whole numbers, I mean all whole numbers already invented and all those that could be invented one day... and it is the "could be" that is infinite.

<sup>23</sup> Language remains important in practical terms, of course: otherwise these mental constructions could not be communicated from mathematician to mathematician.



argument is that classical mathematics is not safe and is subject to the paradoxes. According to the classical mathematician, the meaning of any sentence consists of its truth conditions, and as those truth conditions obtain independently of human knowledge we have only two truth values (true and false). For the intuitionists, truth is no longer bivalent: the truth of any sentence is a proof for it. The meaning of a logical connective can no longer be given as the effect it has on each sentence with this connective as the main one: instead it is given in terms of proofs. As the intuitionists reject the LEM, some strange results, or results which the classical mathematician would not dream of asserting, obtain. For the classical mathematician, there are continuous<sup>24</sup> and non-continuous functions. For the intuitionists, all real-valued functions which are defined over closed bounded intervals are even uniformly continuous. Of course the process is not magical: when the classical mathematician provides an intuitionist with a real valued function defined over a closed interval and which is not continuous according to the classical conception, the intuitionist would answer that this function is not defined (in intuitionistic terms). For the classical mathematician, for each set  $S$  included in  $X$ ,  $X = S \cup (X-S)$ . For the intuitionist this is only true in the case where  $S$  is detachable. That is when for each  $x \in X$ , we have a proof of either  $(x \in S)$  or of  $\neg(x \in S)$ . For the classical mathematician, interpretation is based on set-theoretic and truth-theoretic models whereas for the intuitionist, we can use: topological interpretation,<sup>25</sup> Kripke model interpretation<sup>26</sup> or Heyting algebra.<sup>27</sup>

These are not the only ways to avoid the paradoxes. For instance Hintikka (in [HI2]) avoids them by altering the interpretation. So for CP we have

<sup>24</sup> A function is continuous over a domain, if it is continuous at every point in that domain. A function is continuous at a point  $y$  if whenever we take a point  $x$  which is very close to  $y$ ,  $f(x)$  will be very close to  $f(y)$ .

<sup>25</sup> Where for each  $P$  we assign  $[[P]]$  (an open set of a topology  $\langle X, O \rangle$ ) to be the set all of whose  $p$ -basic neighbourhoods subsets prove  $P$ . Then we use algebraic constructs to interpret the connectives, e.g.  $[[P1 \& P2]] = [[P1]] \cap [[P2]]$ .

<sup>26</sup> This is essentially like the ordered models of the many valued logics; at each stage, knowledge is increased and ignorance reduced.

<sup>27</sup> A Heyting algebra is a structure  $\langle A, \cap, \cup, \Rightarrow, T, \perp \rangle$  such that  $A$  is a lattice with respect to  $\cap, \cup, \perp, T$  and where  $\Rightarrow$  is to be interpreted as implication.

$$\exists y \forall x [(x \in y) \leftrightarrow [\neg(x = y) \rightarrow \Phi(x) ]].$$

Also the tone of our discussion has been concerned only with the logical paradoxes; solutions to the semantical paradoxes consist in the separation of the object language and the metalanguage, but this issue is not our direct concern in this thesis.

*Frege structures:* Frege structures are not only solutions to the problem of model existence, but are also systems of set theory in their own right: they single out that part of Frege's theory which is consistent. Frege structures could be classified as a restriction of logic, and they free Frege's notion of set from the paradox in the following way: the logical constants can apply to any object, but the result will never be a truth value unless the object itself was a proposition. The condition  $x \in x$  is not necessarily a proposition and so

$$(x \in x) \leftrightarrow \neg(x \in x) \text{ is not contradictory.}$$

The logic is weak in this way: the logical constants still apply to any object as with Frege but the result is a truth-value only if the object itself is one. With Frege this was not the case; he had the operator  $--$  (which stands for content) and which gives the content of each object. So  $--A$  is always a truth value whether or not the object  $A$  itself was a truth value. All the other logical constants in Frege's theory were applied to the content of the object and so always resulted in a truth-value. So in particular  $\neg A$  (not  $A$ ) is always a truth value whether or not  $A$  was. Realising this about Frege's theory, Aczel reduced the logic to a weaker one where the logical constants only give truth values for truth values. In Aczel's Frege structures, the axiom  $(\forall b)$  is not rejected. In fact the whole of axiom  $(V)$  is proven as a theorem in Frege structures and does not need to be asserted as an axiom as with Frege. Also, each concept has an extension, and decidable sets (the extensions of decidable concepts) are objects to which LEM applies.<sup>28</sup> In a Frege structure you can prove that a set belongs to itself, (take  $R = \{x: (x = x)\}$ ) and so it seems quite convenient to think of Frege structures as

---

<sup>28</sup> This is actually discussed in detail in chapter 3 under the heading "decidable properties".

models for nominalisation, but I shall leave this matter for the next section. What we have asserted in this section is that Frege structures solve the theoretical (ontological) problem of set theory and so are candidates to be used for the semantics of nominalisation.<sup>29</sup> Before we move to the use of those theories for nominalisation, we give a summary of the work that was carried by Feferman in the foundations of set theory. This is because Feferman's work investigates all of these restrictions (i.e. restricting the axioms, the logic or the language) and plays a crucial role in the area of property theory.

*B.1.1.4 Feferman and the foundational issues:* Feferman, in many of his papers, has worked on the question of the paradoxes and the possible solutions. He investigated for instance in [FE9] the strategies of restricting the axioms, the logic or the language. He also investigated in [FE2] a theory  $T_0$  which I believe is worth more attention than it has received. Feferman's  $T_0$  was a formulation of Bishop's constructive mathematics, as are the theories of Martin-Löf's and Myhill. Yet Martin-Löf's is the theory which had been most used by Computer Scientists because it is more related to notions such as computation, program specifications and constructive proofs. Maybe it is the presence of canonical/noncanonical elements in Martin-Löf's theory and the notion of types which are very attractive to computer scientists. Yet I believe that Feferman's theory is simpler, has notions which are more related to property theories (such as abstraction and application) and it studies classes, properties, comprehension principles and various other notions of interest to a theory of nominalisation.

Of course in this thesis there is no room to discuss either  $T_0$  or any other of Feferman's theories which avoid the paradoxes by various means. We must still however introduce the comprehension principles that Feferman uses in two of his theories.

---

<sup>29</sup> Before closing this section, we mention that the paradox does not occur in free logic. That is if one assumes that not every term denotes, one can have a consistent theory. [TE2] provides a good account of how the paradox is avoided in free logic.

In  $T_0$ , the comprehension principle is restricted to elementary formulas where a formula is elementary if it is both stratified and has no bound class variables. Hence the principle looks like:

$$\text{ECA: } (\exists X)(\{x:\Phi(x,y,z)\} = X \ \& \ \forall x(x \in X \iff \Phi(x,y,z))),$$

where  $\Phi(x,y,z)$  can only be an elementary formula.

$T_0$  was a constructive theory. Feferman, before  $T_0$ , had investigated the use of full classical logic. Yet the paradox is avoided by having positive and negative formulas. The membership relation is now split into two partial predicates  $\epsilon$  and  $\epsilon'$  with the axiom:

$$\text{Dis}(\epsilon, \epsilon') : \neg(x \in \{u/\Phi(u,y_1,\dots,y_n)\} \ \& \ x \in \{u/\Phi(u,y_1,\dots,y_n)\})$$

The comprehension principle is then divided into two comprehension principles: one for the positive formulas and the other is for the negative formulas as follows:

$$\begin{aligned} \text{(CA)}_{(+/-)} \\ x \in \{u/\Phi(u,y_1,\dots,y_n)\} \iff \Phi^+(x, y_1,\dots,y_n) \\ x \in \{u/\Phi(u,y_1,\dots,y_n)\} \iff \Phi^-(x, y_1,\dots,y_n) \end{aligned}$$

Now of course Russell's paradox is avoided here because if we take  $R = \{x/\neg x \in x\}$ , then

$$R \in R \iff (\neg R \in R)^+ = (R \in R)^- = R \in R.$$

These are two of the ways that Feferman uses to avoid the paradoxes. However none of them as we see has a full comprehension principle, whereas Frege structures provide us with a full one.

*B.1.2 Effects of set theory on Nominalisation:* I have said above that nominalisation inherits the same problem as set theory. Therefore, it should inherit the same solution. I shall summarise here the influence that the various approaches to set theory had on the semantics of nominalisation.

*B.1.2.1 Language and nominalisation:* The reform of set theory by following the route of altering the language was based on the vicious circle principle, and resulted in

Russell's theory of types. The language here becomes typed and the ladder of types has to be climbed step by step. Russell's theory of types was made simpler by Church and this is essentially the language used by Montague (in [TH2]) as an application to natural languages. (Montague was the first to apply this approach of set theory to the syntax and semantics of natural languages.) In Chapter 6, we shall find a detailed description of this account and of its inappropriateness to nominalisation. It is worth mentioning here that Montague did not himself deal with nominalisation and that his account is very problematic from the nominalisation point of view. There have been few attempts at dealing with nominalisation within the Montague tradition. Examples are Carlson's work and Parson's floating types (in [CA1] and [PA5]). The main problem with Montague semantics is the typing constraints and the existence of the function  $f$  which has to associate once and for all the syntactic type of each syntactic category. This could be dealt with by changing the function  $f$ , but the approach is cumbersome and leads to difficulties.

*B.1.2.2 Axioms and nominalisation:* We said above that type theory is not adequate to handle nominalisation. What about the solution based on restricting the axioms? Does it help nominalisation? The way to know the answer is to try the various methods we have met of restricting the axioms. If we start with ZF set theory, we will still get a problem. This is because in ZF, we cannot have a set that contains itself. What about systems like NF or ML? We know that they contain sets that belong to themselves, and so they should be promising candidates for the semantics of nominalisation. In fact they have already been applied to this by Cocchiarella who used Quine's approach to both NF and ML and obtained two systems. We illustrate by going back to Cocchiarella's system of non-standard second order logic shown in A.1.

*1. Altering (CP\*)* Here, the paradox is avoided by restricting the formulae in (CP\*) to what is called stratified formulae. A stratified formula is one built up with respect to the vicious circle principle as we explained in the previous section. That is, one cannot

assume the undefined in trying to define it. So a stratified formula is one where the arguments are of lower level than the level of the predicates. This means that in (CP\*), we do not take  $F_n$  to be simply free in  $\Phi$ , but we impose in addition the constraint that the whole bivalence be stratified. To return to our example,  $X(X)$  is not a stratified formula and so the comprehension principle cannot assure us of the existence of the predicate  $F$ . We therefore failed to prove the contradiction  $F(F) \leftrightarrow \neg F(F)$ .

2. *Altering (A\*)* Instead of altering (CP\*), we alter (A3\*) to (A3\*\*) where:

$$(A3**) \quad \forall x \exists y (x = y)$$

We then have to add  $(a = a)$  as an axiom and replace  $\lambda$ -CONV\* to:

$$(E/\lambda\text{-CONV*}) \quad [\lambda x_1, \dots, x_n \Phi] (a_1, \dots, a_n) \leftrightarrow \exists x_1, \dots, x_n ((a_1 = x_1) \& \dots \& (a_n = x_n) \& \Phi)$$

where no  $x_i$  occurs free in any  $a_j$ , for  $1 \leq i, j \leq n$ .

Note here that because of the elimination of (A3\*), we can no longer prove the theorem

$$\forall x \Phi \rightarrow \Phi(a/x).$$

Therefore, we cannot substitute  $F$  for  $x$  in the special instance of (CP\*) and so we cannot derive the paradox.

The option of restricting either (A3\*) or (CP\*) was put forward by Cocchiarella, and the two systems were proved to be equivalent to NF and ML respectively ([CO2]), even though Cocchiarella committed himself to a conceptualistic (naive) conception of set and argued that both NF and ML lack motivation if they are regarded as set theories in the mathematical sense. However, I have two criticisms of Cocchiarella's two systems. The first is that the models are not at all easy to imagine: we have no idea what they look like. The second is that restricting nominalisation to stratified formulas means that not all the desired items can be nominalised. There are expressions we can nominalise in natural languages that this approach does not handle the nominalisation of; e.g. *nice(nice)*. It must be noted of course that this

criticism is basically of his first system, since the second system does allow *nice(nice)*. But even Cocchiarella himself rejects this system because in it the axiom:

$$(IND^*) (\forall X)(\forall Y)(X \simeq Y \iff (\forall x)(X(x) \iff Y(x))).$$

is refutable.

*B.1.2.3 Logic and nominalisation:* The last category is the use of non-standard logics. Take for instance the use of a three-valued logic, rather than the classical two-valued one.  $F(F) \leftrightarrow \neg F(F)$  would not be inconsistent any more, for we can give  $F(F)$  the value  $u$  (undefined) and in the interpretation of  $\neg$  and  $\leftrightarrow$ , we take:  $\neg u \leftrightarrow u$ . This solution has been applied to nominalisation by Ray Turner ([TU2, 3, 4, 5]). Turner used three valued logics and this allowed him to have an untyped language which could deal with nominalisation without falling into the paradox. This approach has been successful as far as predication is concerned, for one can nominalise all formulae. However it has a problem with quantification, since it is only to quantify over ideal elements (i.e. the limits of the finite ones). (It has been claimed that this is so mainly because Scott domains are only suitable for Computer Science applications and not for linguistics. But as it is a question of models, I shall leave the details of the problem of quantification to be briefly discussed in part B.2 and in Chapter 7.) However, anybody who adopts a non-two-valued logic should be able to defend their use of it. Many-valued logics have been criticised by philosophers as being unnatural, and Turner did not offer any justification for using them.

In this section, we have talked about the set theoretical approaches that have been offered. We looked at the theory of types and nominalisation and although we did not claim it was impossible to work out a theory of nominalisation based on Montague's semantics, we did say that it was difficult and cumbersome. We recall here that Russell's theory of types was unsatisfactory and so other theories came into being. The same applies to nominalisation, for Turner's and Cocchiarella's systems are

less problematic than Montague's approach, because systems like NF and ML, or logics which are non-standard, were better attempts to provide a system without paradox than Russell's theory of types. Our criticism of Cocchiarella is that only stratified formulae can be nominalised and that his models are difficult to imagine. In the light of the theoretical problem, we do not find anything against Turner except his use of three valued logics, which have a very controversial status in the literature. However, when it comes to the question of models, we shall find that a problem occurs which we shall describe in B.2. It seems therefore that all the theories of nominalisation that have been worked out so far face some problems. There still are many solutions for set theory that have not hitherto been applied to the semantics of nominalisation, one of these being the notion of Frege structures. It seems at this stage that all the disadvantages of the theories that have been worked out so far can be circumvented by the use of Frege structures. The use of Frege structures will allow us to keep to two-valued logic<sup>30</sup> which is the first advantage over Turner's work; also, we can quantify over all our nominalised items, which is another. Moreover, Frege structures permit us to nominalise all our open well-formed formulae and they are easy to work with, which gives us two clear advantages over Cocchiarella.

*B.1.2.4 The place of logic in the above applications:* Of course one here will wonder why this section did not occur under B.1.2.3. The reason for this is that we are not here trying to study only how the avoidance of the paradoxes by altering the logic was applied to nominalisation, but to study what sort of logic one obtains in the theories of B.1.2.1, B.1.2.2 and B.1.2.3. Although this section could have been accommodated in the three above sections, we decided to single it out on its own to make the comparison more illustrative.

The primary characteristic of most of the theories discussed in B.1.2.1, B.1.2.2 and

---

<sup>30</sup> Here I do not mean that there are only two propositions. I am trying to say that once something is a proposition, it is either true or false.



B.1.2.3 is that they are extensional rather than intensional. Of course extensionality simplifies the theory tremendously (as many terms and propositions will be identified) yet it is not good enough for various reasons. The ideal solution of course would be if we can have a theory where syntactic elements can reduce to each other as much as possible yet the theory does accommodate intensionality. None of the theories explained above does that, yet the one that we shall provide is an intensional theory where the principle of extensionality applies to functions. It may be objected that Turner's theory does not have the axiom of extensionality and hence may be intensional. This is not true however. The loss of extensionality in Turner's thesis is due to the use of partial predicates. Hence Turner's theory is disadvantageous from the points of view we are discussing. It does not have the extensionality axiom and it is not intensional.

Now if we consider the theories of B.1.2.1, we must say about them that they are unattractive. We really would like the syntax to be as expressive as possible, and theories where the syntax is restricted are also restricted for the cases of nominalisation they can consider. Our discussion hence should concentrate on the logics obtained from either altering the axioms or the logic. This is the work of Cocchiarella and Turner. Cocchiarella's two main theories discussed above could be compared to Quine's NF and ML, which should be viewed as theories of classes in the logical sense and not the iterative sense. In fact Cocchiarella argues that NF and ML would lack motivation if they were considered as theories of classes in the iterative sense. Knowing that Cocchiarella's theories try to accommodate Frege's sense of classes, we must now mention that one is based on a system which is proposition free. That is: the following is no longer provable:

$$\forall x\Phi \rightarrow \Phi(a/x).$$

This is unattractive and yields undesirable consequences, such as loss of indiscernability.

The theory of Turner uses three valued logic with Kleene's connectives and this forces the use of partial predicates. It must be noted however that none of the above theories used an intuitionistic logic. This is not an argument that intuitionistic logic should be preferred over a classical one. It is rather an argument that this logic should be investigated.

### *B.2. Solution to model existence*

There were many solutions to the problem shown in A.2. The problem discussed there is not specific to nominalisation. It is the problem of finding models of the  $\lambda$ -calculus. Therefore I shall start by describing some of those models, and then I shall discuss how they have been used for the semantics of nominalisation.

*B.2.1  $\lambda$ -calculus and its models:* We can forget about the formal axiomatisation of the  $\lambda$ -calculus with logic on the top of it and just remember that the  $\lambda$ -calculus with logic is a formal system which has 2 important operations: abstraction and application together with  $\lambda$ -conversion.<sup>31</sup> Until recently, models of the  $\lambda$ -calculus have been problematic: do they really exist, and what are they like? One answer can be that the model itself is a structure which has two operations (abstraction and application); but this is an unsatisfactory answer. First, we could abstract the formula  $\neg P(x)$  and then apply the abstract to itself which would yield Russell's paradox. Second, not every structure which has the two operations can be a model of the  $\lambda$ -calculus. Take for instance any combinatory algebra (which has K, S and '.'). We could prove in a combinatory algebra that the axiom of abstraction

$$(\exists F) (\forall y_1, \dots, (\forall y_n) [F(y_1, \dots, y_n) = A])$$

holds, but that does not mean that the combinatory algebra is a model of the  $\lambda$ -calculus. It will be if we consider the extensional  $\lambda$ -calculus, but in the absence of extensionality we will have many choices for the function F in the axiom of

---

<sup>31</sup> There are other rules like  $\zeta$ -rule, but we ignore them for the moment.

abstraction and so the structure cannot be a model. What we should really require from the model is that if two wffs are equivalent or convertible in the  $\lambda$ -calculus then their values in the model must be the same.

The other problem with defining models of the  $\lambda$ -calculus is that some  $\lambda$ - terms denote functions and so they have to take the elements of the structure  $M$  itself as argument. But again they themselves are terms and must take elements of  $M$  as values. We could take what is known as a term model as a model of the  $\lambda$ -calculus. Term models are just a trivial formulation because all they do is translate the syntax step by step. Two other formulations of models are environment models and combinatory models. The environment models include in them two embedding functions  $P$  and  $\Phi$  which belong to  $D \rightarrow [D \rightarrow D]$  and  $[D \rightarrow D] \rightarrow D$  respectively.

$[D \rightarrow D]$  is not the set of all functions and it usually is the case that certain mathematical properties play a role in choosing  $[D \rightarrow D]$ . Usually,  $[D \rightarrow D]$  is the set of all the continuous functions and is closed under the standard operations (such as composition, abstraction, application,...). The combinatory model is exactly the combinatory algebra we talked about above but with the very important element  $\epsilon$  which obeys some axioms. What  $\epsilon$  does is to single out the functional part of every element. In the presence of extensionality we do not need  $\epsilon$  and that is why in the case of extensionality, combinatory algebras are models of the  $\lambda$ -calculus. Both environment models and combinatory models are equivalent to each other and for a proof of this, the reader is referred to [ME1]. These are not the only kinds of models provided for the  $\lambda$ -calculus. The two kinds of models cited above together with the term models are algebraic, there are others which have a built-in structure. (It is easy to work with such models as one does not get involved with the cumbersome syntax). The two main models that I shall talk about throughout the thesis are: Scott domains and Frege structures. Only the first has been applied to the semantics of nominalisation and in this thesis I discuss the semantics of nominalisation based on



Frege structures. Scott domains are introduced in the appendix, and Frege structures are introduced in the second chapter. I shall however briefly mention some characteristics of Frege structures before I continue as this will enable the reader to understand what we are talking about without having to jump to the second chapter yet.

*B.2.1.1 Frege Structures:* In Chapter 7 we shall meet the application of Scott domains to nominalisation and explain its problem of predication. We shall also show that it is not possible to find a solution to such a problem within semantic domains without logic, therefore semantic domains are not adequate for the semantics of nominalisation. Frege structures are more conclusive than a solution to domain equations and they can be used as models for nominalisation. The remaining question is, do we encounter the same problem as Turner? We show in Chapter 7 that the answer is negative and that all the advantages that Turner obtained by using Scott domains, we obtain within our Frege structures. Scott domains are one possible solution to the problem mentioned in A.2. and have solved it by restricting the functions to the continuous ones. By so restricting the functions, we do not lose any power of interpretation in Computer Science or recursion theory, according to results obtained by Church and Kleene (see [CH6]). However, when it comes to the semantics of natural languages, we have a problem which may come either from continuity or from the ordering on the domains. There does not seem to be any solution for it in Scott domains. In any case, we need to look for another solution to A.2 which holds more promise for nominalisation. No one would want to work with the cumbersome structures of the term models, and we would like a model which we can master with set theoretical or topological techniques as was the case with Scott domains.  $P\omega$  ([SC3]) is such a model, but unfortunately, there is no extra advantage in using it.  $P\omega$  does not have more to offer than Scott domains, as there is an equivalence relation between the two - and Turner's problem is not going to be solved with  $P\omega$ . One other

solution to the problem of A.2. is Aczel's notion of Frege structures. Frege structures are not only a collection of collections of functions (as in the case of  $E_{\infty}$ ), but they also have a certain logic which works on them, and whose availability solves also the problem of A.1. Therefore, Frege structures solve both problems of part A. The solution to the technical problem has been discussed in B.1 and, I shall not discuss Frege structures further in this section as they are the subject of Chapter 2.

*B.2.2 Using those models for nominalisation:* In the previous section, we described two solutions to the problem of model existence of the  $\lambda$ -calculus having in mind that those two solutions are to be assessed as models of nominalisation. In this section, we shall comment briefly on how each solution has been or can be used for the semantics of nominalisation.

*B.2.2.1 Scott domains and nominalisation:* We mentioned in B.1.2.1 that the theory of types was not adequate to the semantics of nominalisation. The typing constraints according to Church's type theory are too restrictive for nominalisation and we need to have functions which can apply to themselves or to items of the same type. Abandoning Church's type theory does not imply getting rid of all the typed theories. We can still keep to typed languages but make the typing adequate to deal with nominalisation. The area of Computer Science and its use of the  $\lambda$ -calculus gives us good examples of typed theories which still allow functions to be applied to themselves. Natural languages seem to make more demands on a semantic theory than computer languages, but the progress in Computer Science could nevertheless lead to useful insights about natural languages. I am not of course claiming that results in Computer Science can always be applied to natural languages; indeed Scott domains are a counterexample. To date, the only result from Computer Science applied to the semantics of nominalisation seems to have been Turner's work (referenced above). However, no one has yet applied Frege structures to the semantics of nominalisation. I intend to work out such an application and to assess its advantages over the use of

Scott domains as models. I shall show in Chapter 7 that the ordering relation on Scott domains makes predication trivial. For, a predicate  $P$  is true of all the objects in the model iff it is true of the bottom element. Also the use of Scott's domains forced us to use three valued logics. All these disadvantages do not occur in our application of Frege structures to nominalisation. From the theoretical point of view, Frege structures are going to have an equal advantage, and their explicit closure embodies in it the abstraction principle (see Chapter 2).

### *CONCLUSION AND COMPARISON WITH COCCHIARELLA*

In the first part of this chapter, we outlined two problems with the semantics of nominalisation. One is a problem of set theory or predication, the second is a problem of models. The fear of Russell's paradox, which obviously threatens a theory of nominalisation, led to questions on the nature of universals and predication. These questions are not new however, and have been the concern of ancient philosophers. What philosophers nowadays take from them is a decision as to which objects are to be subject to predication and which concepts have extensions. This is the theoretical point of view. With respect to model existence, Cantor's diagonal theorem makes us fear the non-existence of models. The division of the problem into two parts does not imply a total independence of both problems. In a way, they are strongly related, for we start from an ontology and build a model which contains that which conforms to our ontology. Yet, separating those two problems makes us concentrate on each independently and then later we consider both as a whole.

The second part of this chapter discussed some of the solutions of set theory to the first problem (A.1) and of the model construction to the second problem (A.2) and briefly described some of the applications of both set theory and model construction to the semantics of nominalisation. Frege structures provide a solution

to both problems but have not been used for the semantics of nominalisation. We have commented that they have all the advantages of the previous applications and more, they do not have any of the previous disadvantages. In the next chapter, we shall introduce in detail Frege structures and the theory that we shall be using together with the semantics. In subsequent chapters, we shall discuss some further advantages of Frege structures in relation to property theory, intensionality, quantifiers and type theory; afterwards, we shall compare our work to others. Before moving to the next chapters however, it would be nice to locate Cocchiarella's proposal discussed in this chapter to the one proposed here and in chapter 6. The location is going to be mainly in terms of the typing system, because whereas I use a type free theory, Cocchiarella uses a second order one. There are however some similarities and differences in these two ways of typing that I would like to illustrate.

According to axiom (9) under A.1.2, we have  $ME_n \subseteq ME_0$  for all  $n > 1$ , where  $ME_n$  are the meaningful expressions of any type  $n$ . For us, we have that  $ME_n \subseteq ME_0$  for any  $n \geq 1$  but the pictures of both approaches are quite different. According to our approach these types are related to each other in a chain like way. That is  $ME_n \subseteq ME_{n-1} \subseteq \dots \subseteq ME_0$ . For Cocchiarella we have that each  $ME_n \subseteq ME_0$  for  $n > 1$ , yet no relation exists between  $ME_n$  and  $ME_m$  for  $n \neq m$ . Also for Cocchiarella, propositions are not included in objects, even though they can be embedded in  $ME_0$  by axiom (8) under the same paragraph. Hence Cocchiarella's whole structure can be understood as a collection of objects, which has a denumerably infinite number of subcollections called functions but where propositions are outside the domain of objects and can be mapped into it. This structure for Cocchiarella is not a structure of types in the sense that we have in the typing structure in Chapter 6. In fact everything that Cocchiarella has so far we have; as will be seen in the next chapter, a Frege structure is  $F_0, \dots, F_n$ . where  $F_0$  is the collection of objects,  $F_k$  is the collection of  $k$ -ary functions and each of these  $F_k$  can be embedded in  $F_0$  by  $\lambda^k$ . What we shall have in

addition is a typing system constructed inside  $F_0$ , which cannot be found in Cocchiarella's theory. Also, our system is first order in that the quantification over objects and functions is the same, whereas Cocchiarella's system is second order.



CHAPTER 2. FREGE STRUCTURES AND NOMINALISATION

In this chapter we introduce the reader to Frege structures (see [AC3]) and set out the theory that we shall be using throughout this work. Afterwards, we give the semantics to be adopted and lay out the proof theory.

PART A. SUMMARY OF FREGE STRUCTURES

Before launching into this section, let us introduce some convenient notation and informal definitions:

If  $f$  is a function of 2 arguments then we will sometimes write  $afb$  for  $f(a,b)$ . For example, we write  $a \& b$  for  $\&(a,b)$ .

Until we give the exact definition of an  $F$ - functional, let us understand it to be a function which takes functions as arguments and returns functions as values.

$F_0^n$  stands for:  $F_0 \times F_0 \times \dots \times F_0$ ,  $n$  times.

*Metalinguage abstraction:* For every expression  $e[x_1, \dots, x_n]$  of the metalanguage built up in the usual way from variables ranging over  $F_0$  and constants ranging over  $\cup_n F_n$ , the expression  $\langle e[x_1, \dots, x_n] / x_1, \dots, x_n \rangle$  denotes the  $n$ -place function  $f: F_0 \times \dots \times F_0 \rightarrow F_0$  such that for each  $a_i$  in  $F_0$ ,  $1 \leq i \leq n$ ,  $f(a_1, \dots, a_n)$  is the value of  $e[a_1, \dots, a_n]$ , the expression  $e$  in which  $x_i$  has been replaced by  $a_i$  for  $i = 1, \dots, n$ . For each expression  $e[\xi_1, \xi_2, \dots, \xi_n]$  of the metalanguage built in the usual way out of variables (ranging over  $F_n$  for  $n \geq 0$ ) and constants (ranging over  $F_n$  for  $n \geq 0$  and over  $F$ - functionals), the expression  $\langle e[\xi_1, \xi_2, \dots, \xi_n] / \xi_1, \xi_2, \dots, \xi_n \rangle$  denotes the  $n$ -place function obtained by abstracting  $\xi_1, \xi_2, \dots, \xi_n$  in  $e$ . The next concept is one that we shall be referring to very often; we therefore introduce it by a named definition, Def\*.

*Def\*:* If  $F$  is a 1-place  $F$ - functional and  $\langle e[x] / x \rangle$  is in the domain of  $F$ , we write  $Fxe[x]$  for  $F(\langle e[x] / x \rangle)$ .

For example,  $V: F_1 \rightarrow F_0$  and  $\lambda: F_1 \rightarrow F_0$  are  $F$ - functionals and we write  $V$

$\langle f(x)/x \rangle$  and  $\lambda \langle f(x)/x \rangle$  as  $\forall x f(x)$  and  $\lambda x f(x)$  respectively.

We understand by a propositional function, a function of the Frege structure which takes propositions as values, i.e.  $f(x)$  is a proposition for every  $x$ .

### *A.1. Informal introduction*

Before we introduce Frege structures formally, we need to introduce the reader to the geography of the field with which we are concerned here. The existing models of the  $\lambda$ -calculus did not deal with logic added on top of the  $\lambda$ -calculus, since once logic is added, consistency might be threatened. Also, if one constructs a theory which will have logic,  $\lambda$ -abstraction and predication, then one has to show the existence of the models of this theory. This is the work we find with Feferman for instance, yet his models are not tidy and clear. Hence one would like to have a clear idea of a model of the  $\lambda$ -calculus with logic on it, and Frege structure is such a model. However, such a construction was not obvious for a long time. It was initiated by Scott in [SC2] yet the work was incomplete and hence such a model was not achieved. Then came the construction of Frege structures where simply the idea is to start from any model of the  $\lambda$ -calculus and build logic on top by inductively constructing two collections (of the possible propositions and the possible truths) and taking the limit of these two collections which actually draw the logic we now have on the top of the initially considered model of the  $\lambda$ -calculus.

As it sounds, the process is quite simple, yet it depends on having a clear idea of the structure and on proving some theorems which will ensure the existence of the various logical connectives in the model considered.

Now that logic has been constructed on the top of a model of the  $\lambda$ -calculus, we can consider the structure only in terms of its objects and functions. The objects include propositions and truths and the functions obey the condition that propositional functions can be projected in the domain of objects (i.e. as sets). Those sets can be

applied to any object (hence we now have not only functional application such as  $f(x)$ , but also the application of one object to another as in  $\text{app}(a,b)$ ), and set application to an object results in a proposition.

This is the simple idea of a Frege structure. Next, the reader finds the various steps used to construct a Frege structure.

A Frege structure consists of a denumerably infinite number of collections  $(F_n)_{n \geq 0}$  such that:<sup>32</sup>

1.  $F_0$  is a collection of objects which has three very important subcollections **PROP**, **TRUTH** and **SET** where,

**PROP** is a subcollection of  $F_0$  which can be thought of as the collection of propositions and

**TRUTH** is a subcollection of **PROP** which can be thought of as the collection of true propositions.

**SET** is a subcollection of  $F_0$  which can be thought of as the collection of objects which are nominals of propositional functions.

2. For each  $n > 0$ ,  $F_n$  is a collection of  $n$ -ary functions which take all their arguments in  $F_0$ .

3. There is a set of  $F$ -functionals that operate over  $(F_n)_{n \geq 0}$  and which ensure important closure properties on  $(F_n)_{n \geq 0}$ . For example:

$\forall : F_1 \rightarrow F_0$  is a functional such that:

If  $f$  in  $F_1$  is a propositional function

then  $\forall f$  is in **PROP** and

$\forall f$  is in **TRUTH** iff  $f(a)$  is in **TRUTH** for each  $a$  in  $F_0$

$\lambda : F_1 \rightarrow F_0$  and  $\text{app} : F_0 \times F_0 \rightarrow F_0$  are two other functionals which possess the very important property:  $\text{app}(\lambda f, a) = f(a)$  for every  $a$  in  $F_0$  and every  $f$  in  $F_1$ .

4.  $(F_n)_{n \geq 0}$  is super explicitly closed: i.e. for each expression  $e[\xi_1, \xi_2, \dots, \xi_n]$  of the

<sup>32</sup> There are variables and constants that range over  $F_n$ , for  $n \geq 0$ .

metalinguage built in the usual way out of variables (ranging over  $F_n$  for  $n \geq 0$ ) and constants (ranging over  $F_n$  for  $n \geq 0$  and over  $F$ - functionals), the  $n$ -place function denoted by  $\langle e[\xi_1, \xi_2, \dots, \xi_n] / \xi_1, \xi_2, \dots, \xi_n \rangle$  is an  $F$ - functional. This means that Frege structures are closed under composition, projection, etc.<sup>33</sup>

Now that we have some idea of the structures' form, let us try to give an intuitive picture. A Frege structure is a collection of both objects and functions (which are distinct) where we can map any function  $f$  into an object  $a$  and this object will preserve some of the properties of the function. For instance if the function  $f$  is a propositional function then the nominal of the function,  $\lambda f$ , is an object<sup>34</sup> which belongs to the category **SET**. Moreover **SET** contains only those objects which are nominals of propositional functions. Thus, if  $a$  is in **SET** then there must be a  $k$ -ary propositional function  $f$  such that  $a = \lambda_0^n f$ , where:  $\lambda_0^1$  is  $\lambda$  and maps 1-ary functions into objects (i.e. into  $F_0$ );  $\lambda_0^2$  maps 2-ary functions into objects;...  $\lambda_0^n$  maps  $n$ -ary functions into objects. By induction, we can define  $\lambda_m^n$  which maps  $n$ -ary functions into  $F_m$ .

It is natural to ask whether the intersection of **SET** and **PROP** is empty or not; some elements of **PROP** are elements of elements of **SET**, yet the intersection between **SET** and **PROP** is not certain to be empty.<sup>35</sup> Independently of whether **SET** and **PROP** are disjoint, there is an important relation between them which is the following;<sup>36</sup> they both have strong links with propositional functions. Let us

<sup>33</sup> Properties 1-4 are only informally presented here and there are many concepts that were introduced above but were not quite explained (e.g.  $F$ - functionals). This will be done next however. The above introduction is intended to be as simplified as possible to allow the reader to imagine the structure of the model first before putting all the details in front of his eyes.

<sup>34</sup>  $(\lambda f)$  is not an extensional object and even though we write sometimes  $(\lambda f) = \{x: f(x)\}$  this does not imply extensionality.

<sup>35</sup> Take for example an element  $a$  of **PROP** and consider  $b$  to be the set  $\{x: (x=a)\}$ . Obviously  $b$  is in **SET** because  $\langle x=a \rangle / x$  is a propositional function, so we have  $a$  is in  $b$ .

<sup>36</sup> **SET** and **PROP** are not necessarily disjoint. Take for example,  $a$  in **PROP** and assume the following principle:

$\forall x(\text{app}(t,x) = \text{app}(t',x)) \rightarrow t=t'$ .

$\lambda a l = a$  can then be seen as follows:

$\forall x, \text{app}(\lambda a l, x) = l a(x) = \text{app}(a, x)$ .

Therefore  $\forall x \text{app}(\lambda a l, x) = \text{app}(a, x)$  and hence  $a = \lambda a l$ .

Now, if  $l a$  is a propositional function, then **SET**  $\cap$  **PROP** is not empty. The question here is whether  $l a$  is a propositional function when  $a$  is a proposition. We do not need to answer this question here and we leave it to Chapter 5.

consider 1-ary functions to illustrate the argument and take a propositional function  $f$ . For any object  $a$ ,  $f(a)$  is a proposition (i.e. is in **PROP**).  $\lambda f$  is a set and  $\mathbf{app}(\lambda f, a) = f(a)$ . We can always jump from propositional functions to sets (and from sets to propositions). But we can also jump from sets to propositional functions. Take the operation  $| |_1$  defined as: For each object  $a$  of the Frege structure,  $|\lambda f|_1 = \langle \mathbf{app}(a, x)/x \rangle$ . Obviously for each  $a$ ,  $|\lambda f|_1$  is in  $F_1$  and if, in particular, we take  $a$  to be in **SET** (say  $a$  is  $\lambda f$ ) then we have that  $|\lambda f|_1 = \lambda f|_1 = f$ . Therefore we have an equivalence between sets and propositional functions; each set corresponds to a propositional function and each propositional function corresponds to a set. This is important and it is this strong link that I am trying to emphasise between **SET** and propositional functions.<sup>37</sup>

So in a Frege structure, we can take any function into an object and we can preserve some properties of the function and use them for establishing facts about the function or its nominal. In short, we do not lose information by mapping the function into an object. We can switch back from objects to functions using  $| |_n$ , the inverse operator of  $\lambda_0^n$  where we have the following theorem:  $|\lambda_0^n f|_n = f$  for any  $n$ -ary propositional function  $f$ .

The ability to switch back and forth between objects and functions is not the only important aspect of the program; the presence of **PROP**, **TRUTH** and of a logic in a Frege structure is also crucial. The logic is built in a way that allows us to talk about truths and propositions without falling into any contradictions. We have classified Scott domains as inadequate because they do not have any logic in them - and when one tries to build a logic on them, one faces problems with quantifiers. In a Frege

<sup>37</sup> Note that for each  $n$ , this bivalent path holds between  $P F_n$  and **SET**, through  $\lambda_0^n$  and  $| |_n$  where again we have  $\mathbf{app}_n(\lambda_0^n f, \check{a}) = f(\check{a})$ , for  $\check{a}$  in  $F_0^n$ , and  $f$  in  $F_n$ . The functionals  $\lambda_0^n$ ,  $\mathbf{app}_n$  and the operation  $| |_n$  could be defined recursively as follows: Take  $|\lambda f|_1 = \langle \mathbf{app}_1(a, x)/x \rangle$  and  $\lambda_0^1 f(\check{a}) = \lambda(\langle f(\check{a}, x)/x \rangle)$ , and assume  $\lambda_0^{n+m} f$  has been defined. Then take  $\lambda_0^{n+m+1} f = \lambda_0^{n+1}(\lambda_{n+1}^{n+m+1} f)$ .  $\mathbf{app}_n$  is also defined by recursion where:  
 $\mathbf{app}_1 = \mathbf{app}$  and assume we have defined up to  $\mathbf{app}_n$ . Then  
 $\mathbf{app}_{n+1}(a, b) = \mathbf{app}_n(\mathbf{app}(a, b), b)$ .  
 One can prove that  $\mathbf{app}_n(\lambda_0^n f, \check{a}) = f(\check{a})$  for each  $n$  in  $\omega$  and  $\check{a}$  in  $F_0^n$ .

structure, we have combined both the elegance of a simple structure (objects and functions) together with the presence of a consistent logic (and therefore the ability to talk about semantics and truths in a philosophically sound way). Before we can move further in this thesis, we need to describe the formal details of a Frege structure.

A.2. The models

Having in the previous section informally introduced Frege structures, I shall fill in all the technical details in this section and show that Frege structures exist.

Consider  $F_0, F_1, \dots, F_n, \dots$  a family  $F$  of collections where  $F_0$  is a collection of objects, and

$$(\forall n > 0) [ F_n \text{ is a collection of } n\text{-ary functions from } F_0^n \text{ to } F_0 ]$$

*Defl:* An explicitly closed family: We say that a family  $F$  as above is explicitly closed iff: For every expression  $e[x_1, \dots, x_n]$  of the metalanguage built up in the usual way from variables ranging over  $F_0$  and constants ranging over  $\cup_n F_n$ , the  $n$ -place function denoted by  $\langle e[x_1, \dots, x_n] / x_1, \dots, x_n \rangle$  is in  $F_n$ .

More formally,  $F$  is explicitly closed iff 1, 2 and 3 below hold:

- 1. Closure under constant functions: For each  $a$  in  $F_0$ , the function  $f_a$  is in  $F_1$ , where  $(\forall x) [f_a(x) = a]$ .
- 2. Closure under composition: For each  $f$  in  $F_m$ , for each  $g_1, \dots, g_m$  in  $F_k$ ,  $f(g_1, \dots, g_m)$  is in  $F_k$  where  $(f(g_1, \dots, g_m))(x_1, \dots, x_k) = f(g_1(x_1, \dots, x_k), \dots, g_m(x_1, \dots, x_k))$ .
- 3. Closure under projection: For each  $n, i \geq 1$ ,  $P_i^n$  is in  $F_n$  where  $P_i^n(a_1, \dots, a_n) = a_i$  for each  $a_i$  in  $F_0$  and  $1 \leq i \leq n$ .

For example, if  $f$  and  $g$  are unary functions of  $F$  and  $h$  is a binary function of  $F$ , then the following function  $\langle f \circ g(h(x_1, x_2)) / x_1, x_2 \rangle$ <sup>38</sup> is a 2-ary function (i.e. in  $F_2$ ).

In what follows, we assume such a closed family and call it  $F$ .

<sup>38</sup> I.e. the function which takes any  $(a_1, a_2)$  into  $f(g(h(a_1, a_2)))$ , that is fogoh.

*Def2.* F- functional: A function  $D: F_{n_1} x_1 \dots x_{n_k} F_{n_k} \dashrightarrow F_0$  is an F- functional with respect to the explicitly closed family  $F$ , iff:

$$(\forall m \geq 0) (\forall f_1 \text{ in } F_{m+n_1}) \dots (\forall f_k \text{ in } F_{m+n_k}) \\ [ \langle D(\langle f_1(\check{y}, \check{x}_1) / \check{x}_1 \rangle, \dots, \langle f_k(\check{y}, \check{x}_k) / \check{x}_k \rangle) / \check{y} \rangle \text{ is in } F_m ] .$$

where  $\check{y}$  is a list of  $m$ -variables and  $\check{x}_i$  is a list of  $n_i$  variables, for  $i = 1, \dots, k$ . Note that if  $f_1, \dots, f_k$  are 1-place functions and  $D: F_1 x_1 \dots x_{n_k} F_{n_k} \dashrightarrow F_0$  then  $D(f_1, \dots, f_k)$  is in  $F_0$ . What is the intuitive meaning of F- functionals? We know that an F- functional is a functional, so that it operates on functions. But once we include functionals in the structure, we need to ensure that any expression which contains functionals should actually be in the structure. Assume for the sake of argument that  $D: F_{n_1} x_1 \dots x_{n_k} F_{n_k} \dashrightarrow F_0$  is an F- functional. Assume also that for some  $m \geq 0$ ,  $f_i$  is in  $F_{m+n_i}$  for  $i = 1, \dots, k$ . We know that according to the explicit closure, if  $\check{y}$  is a list of  $m$ -variables ranging over  $F_0$  and for each  $i$ ,  $\check{x}_i$  is a list of  $n_i$  variables ranging over  $F_0$ , then  $\langle f_i(\check{y}, \check{x}_i) / \check{x}_i \rangle$  is an element of  $F_{n_i}$  for each  $i$ . Therefore it makes sense to talk of the expression  $D(\langle f_1(\check{y}, \check{x}_1) / \check{x}_1 \rangle, \dots, \langle f_k(\check{y}, \check{x}_k) / \check{x}_k \rangle)$ . This expression however is open in  $\check{y}$  and if we abstract over  $\check{y}$  in this expression we are going to obtain an element of  $F_m$ ? Nothing so far in the structure ensures that this is the case, and we must therefore impose the constraint that these functionals should have such a property. A functional which has this property is called an F- functional and now if  $D$  is an F- functional then

$$[ \langle D(\langle f_1(\check{y}, \check{x}_1) / \check{x}_1 \rangle, \dots, \langle f_k(\check{y}, \check{x}_k) / \check{x}_k \rangle) / \check{y} \rangle \text{ is in } F_m ] .$$

Now we extend the definition of explicit closure to the following:

*Def3.* A super explicitly closed family: Taking a family as above, we say that this family is super explicitly closed iff for every expression  $e[\xi_1, \dots, \xi_m]$  of the metalanguage, built up in the usual way from variables ranging over  $\bigcup_n F_n$  and constants ranging over  $\bigcup_n F_n$  and over F- functionals, the  $m$ -place function denoted by  $\langle e[\xi_1, \dots, \xi_m] / \xi_1, \dots, \xi_m \rangle$  is an F- functional.<sup>39</sup>

<sup>39</sup> This notion of explicit closure is going to provide us with the full comprehension principle we have

*Theorem:* Any explicitly closed family which has variables for functions and objects, constants for objects, functions and F- functionals, is a super explicitly closed family.

The proof is by an easy induction.  $\square$

As an example of an explicitly closed family, consider  $P\omega$  as described in Chapter 1. Define  $F_0$  to be the set of all subsets of  $\omega$  (i.e.  $P\omega$ ). Define, for each  $n \geq 0$ ,  $F_n$  to be the set of all continuous functions from  $F_0^n \rightarrow F_0$ . We have demonstrated that the constant functions, the projection functions, etc are continuous. We have also shown that continuity is closed under composition and that any combination  $e[x_1, \dots, x_n]$  of variables for objects and constants for both functions and objects results in the function denoted by  $\langle e[x_1, \dots, x_n]/x_1, \dots, x_n \rangle$  being an element of  $F_n$ . Therefore the family  $(F_n)_n$  just obtained from  $P\omega$  (call it FE), is an explicitly closed family. Furthermore, FE is super explicitly closed as it can be proven not only that  $\langle e[x_1, \dots, x_n]/x_1, \dots, x_n \rangle$  denotes a continuous function but also that for any expression  $e[\xi_1, \dots, \xi_n]$  built in the usual way out of variables ranging over  $\cup_n F_n$  and constants ranging over both  $\cup_n F_n$  and F- functionals,  $\langle e[\xi_1, \dots, \xi_n]/\xi_1, \dots, \xi_n \rangle$  denotes a continuous function.

So far, we have only explicit closure on our structure. But that is not enough to give a logic on the structure - something we have been arguing is necessary. In what follows, we see how to obtain such a logic.

Assume an explicitly closed family  $F$  and a list of logical constants which are the following F- functionals:

$$\neg : F_0 \rightarrow F_0$$

$$\vee, \&, \rightarrow, = : F_0 \times F_0 \rightarrow F_0$$

$$\forall, \exists : F_1 \rightarrow F_0$$

*Defn. Logical system:* A logical system on a super explicitly closed family  $F$ , relative

---

been promising.



to a set of logical constants as above, is the set of two collections of objects  $\langle \mathbf{PROP}, \mathbf{TRUTH} \rangle$  such that  $\mathbf{TRUTH} \subseteq \mathbf{PROP}$ . These two collections are closed under an adopted logical schemata for each logical constant. The logical schemata corresponds to the external logic and tells us, for each logical constant from the list, how to build new propositions out of other ones using the logical constant. It also gives the conditions of truth for the resulting proposition.

*THE LOGICAL SCHEMATA:*

NEGATION

If  $a$  is in  $\mathbf{PROP}$  then  $\neg a$  is in  $\mathbf{PROP}$  and  $\neg a$  is in  $\mathbf{TRUTH}$  iff  $a$  is not in  $\mathbf{TRUTH}$ .

CONJUNCTION

If  $a, b$  are in  $\mathbf{PROP}$  then  $(a \ \& \ b)$  is in  $\mathbf{PROP}$  and  $(a \ \& \ b)$  is in  $\mathbf{TRUTH}$  iff  $a$  is in  $\mathbf{TRUTH}$  and  $b$  is in  $\mathbf{TRUTH}$ .

DISJUNCTION

If  $a, b$  are in  $\mathbf{PROP}$  then  $(a \ \vee \ b)$  is in  $\mathbf{PROP}$  and  $(a \ \vee \ b)$  is in  $\mathbf{TRUTH}$  iff  $a$  is in  $\mathbf{TRUTH}$  or  $b$  is in  $\mathbf{TRUTH}$ .

IMPLICATION

If  $a$  is in  $\mathbf{PROP}$  and the object  $b$  is in  $\mathbf{PROP}$  provided that  $a$  is in  $\mathbf{TRUTH}$  then  $(a \rightarrow b)$  is in  $\mathbf{PROP}$  and  $(a \rightarrow b)$  is in  $\mathbf{TRUTH}$  iff  $a$  is in  $\mathbf{TRUTH}$  implies  $b$  is in  $\mathbf{TRUTH}$ .

UNIVERSAL QUANTIFICATION

If  $f$  is a propositional function in  $F_1$  then  $\forall f$  is in  $\mathbf{PROP}$  and  $\forall f$  is in  $\mathbf{TRUTH}$  iff  $f(a)$  is in  $\mathbf{TRUTH}$  for all objects  $a$ .

EXISTENTIAL QUANTIFICATION

If  $f$  is a propositional function in  $F_1$  then  $\exists f$  is in  $\mathbf{PROP}$  and  $\exists f$  is in  $\mathbf{TRUTH}$  iff  $f(a)$  is in  $\mathbf{TRUTH}$  for some object  $a$ .

EQUALITY

If  $a, b$  are objects then  $(a = b)$  is in  $\mathbf{PROP}$  and  $(a = b)$  is in  $\mathbf{TRUTH}$  iff  $a=b$ .

EXTENDED CONJUNCTION

If  $a$  is in  $\mathbf{PROP}$  and the object  $b$  is in  $\mathbf{PROP}$  provided that  $a$  is in  $\mathbf{TRUTH}$  then  $(a \ \& \rightarrow b)$  is in  $\mathbf{PROP}$

and  $(a \&\rightarrow b)$  is in **TRUTH** iff  $a$  is in **TRUTH** and  $b$  is in **TRUTH**.

BI-IMPLICATION

If  $a, b$  are in **PROP** then  $(a \equiv b)$  is in **PROP**  
and  $(a \equiv b)$  is in **TRUTH** iff  $(a$  is in **TRUTH** iff  $b$  is in **TRUTH**).

In short, a logical system builds a logic on our structure.<sup>40</sup> But something is still missing: predication and abstraction. We do not want to gain logic yet lose the bijection between objects and functions. Therefore, our structure must have more in it. The next definition will tell us what.

*Def5, λ-system:* A λ-system on an explicitly closed family **F** is a pair of functionals  $\langle \lambda, \text{app} \rangle$  such that:

$$\lambda : F_1 \dashrightarrow F_0 \text{ and } \text{app} : F_0 \times F_0 \dashrightarrow F_0 \text{ satisfy:}$$
$$\text{app} (\lambda x f(x), a) = f(a), \text{ for each } f \text{ in } F \text{ and } a \text{ in } F_0.$$

If<sup>41</sup> we take the system FE<sup>42</sup> given above, and

if we define  $\lambda : F_1 \dashrightarrow F_0$  as  $\lambda f = \{(n,m) : m \text{ is in } f(e_n)\}$

where we take  $(n,m)$  to be  $1/2(n+m)(n+m+1)+m$ <sup>43</sup>

and define  $\text{app} : F_0 \times F_0 \dashrightarrow F_0$  as  $\text{app} (a,b) = \{m : e_n \subseteq b \text{ for some } n, (n,m) \text{ is in } a\}$ ;

then  $(\lambda, \text{app})$  forms a λ-system for FE.

Proof:  $\text{app} (\lambda f, a) = \{m : e_n \subseteq a \text{ for some } n \text{ and } (n,m) \text{ is in } \lambda f\}$

$$= \{m : e_n \subseteq a \text{ for some } n \text{ and } m \text{ is in } f(e_n)\}$$
$$= \{m \text{ in } f(e_n) : e_n \subseteq a\}$$
$$= f(a) \text{ by continuity. } \square$$

Therefore  $(\lambda, \text{app})$  is a λ-system for FE. Actually, FE contains  $\lambda$  and  $\text{app}$  and so it is a λ-structure, but we leave this to the next definition.

<sup>40</sup> From now on, we shall use *a is true* for *a is in TRUTH*, *a is a proposition* for *a is in PROP* and *a is a set* for *a is in SET*.

<sup>41</sup> Note that the λ-system here is only  $\lambda$  and  $\text{app}$ .

<sup>42</sup> I have tried to choose the simplest example and the one I give here is the simplest (apart from the trivial case where the set is one element only). It is a well known result that there are no finite (non trivial) models of the λ-calculus.

<sup>43</sup> Recall that the topology on  $P\omega$  was defined in Chapter 1; the reader may wish to refer back for some notations.

*Def6. λ-structure:* A λ-structure is an explicitly closed family **F** which has a λ-system.

Note that the λ-structure contains λ and **app** and that it is an explicitly closed family. Now take the example of the λ-system on FE given above. FE is also a λ-structure having ( λ, **app** ) as λ-system, because both λ and **app** are in FE, as FE is explicitly closed.

*Def7. Frege structures:* A Frege structure is a logical system relative to a list of logical constants on an explicitly closed family **F**, together with a λ-system.

As an example of a Frege structure, take the λ-structure FE given above and which has a λ-system ( λ, **app** ). Aczel (in [AC3])<sup>44</sup> showed that each λ-structure can be extended to a Frege structure. Therefore we now have an example of a Frege structure. Let us sketch the proof of how our particular λ-structure FE can be extended to a Frege structure. This will make the reader understand the notion of Frege structure, and get him used to working with it. Before proceeding, however, we must define two missing notions: that of an independent family of **F**-functionals and of a primitive **F**-functional. We say that a family of **F**-functionals is independent iff for any two **F**-functionals in the family, the range of values of those **F**-functionals are disjoint. This implies that if **F** and **G** belong to an independent family of **F**-functionals, then for any  $\check{f}$  and  $\check{g}$  such that  $F(\check{f}) = G(\check{g})$ , we should definitely have  $\check{f} = \check{g}$ . From independence only we cannot conclude that  $\check{f} = \check{g}$ . For this we need primitivity and this is the next notion we define.

We say that an **F**-functional  $F: F_{n_1} \times \dots \times F_{n_k} \rightarrow F_0$  is primitive iff there exists a projection  $P_i$  in  $F_{n_i+1}$  for each  $1 \leq i \leq k$  such that  $P_i(F(\check{f}), \check{a}) = f_i(\check{a})$  where  $\check{f} = f_1, \dots, f_k$  is in  $F_{n_1} \times \dots \times F_{n_k}$  and  $\check{a}$  is in  $F_0^i$ . The aim of primitive **F**-functionals is similar to injectivity; if we have  $F(\check{f}) = F(\check{g})$  then we should be able to deduce  $\check{f} = \check{g}$ . It can be easily checked from the definition of **F**-primitivity that this is the case.

<sup>44</sup> Dana Scott has found similar results in his Combinators and Classes paper (see [SC2]). However he used 3-valued logic and did not fully complete his account.

The proof that we can extend any  $\lambda$ -structure into a Frege structure is based on two theorems. The first is one which asserts the existence of an independent family of primitive F- functionals on the  $\lambda$ -structure, which include the logical constants,  $\&$ ,  $\vee$  etc. It simply states that if for each natural number  $m$  we let  $(v_{m_1}, \dots, v_{m_k})$  be a finite sequence of natural numbers, then there is an independent family of primitive F- functionals:  $\Phi_m: F_{v_{m_1}} \times \dots \times F_{v_{m_k}} \rightarrow F_0$ , for  $m = 0, 1, 2, \dots$ . The second is the well known fixed point theorem which applies to monotonic operators and helps us to find the logical schema of these logical constants. This theorem simply states the following: if  $A$  is a partially ordered collection of objects such that every chain in  $A$  has a least upper bound<sup>45</sup> then any monotonic<sup>46</sup> operator  $Y$  from  $A$  to  $A$  has a fixed point. That is  $(\exists a \in A) [Y(a) = a]$ . Let us apply those two theorems to our FE and obtain out of it a Frege structure. Up to here, we know that the  $\lambda$ -structure FE exists and the first theorem enables us to find all the logical constants needed. What remains to turn it into a Frege structure is to find a logical system for the logical constants. This is the task of the second theorem. The idea is to associate with each logical constant two predicates which will ultimately (after we get to the fixed point) give all the propositions obtained from the logical constant and all the truths respectively. The construction is well known mathematically and is similar to the one followed by Kripke in [KR1].<sup>47</sup> Now consider our  $\lambda$ -structure FE. We can be sure from theorem 1 that we have a list of F- functionals which includes:

$$\neg : F_0 \rightarrow F_0,$$

$$\&, \vee, \rightarrow, \equiv, = : F_0 \times F_0 \rightarrow F_0$$

$$\forall, \exists : F_1 \rightarrow F_0.$$

But we still need to make sure that they satisfy the closure properties we want to impose on them.

<sup>45</sup> See appendix for these notions.

<sup>46</sup>  $Y$  is monotonic  $\iff (\forall x, y \text{ in } A) [x \leq y \implies Y(x) \leq Y(y)]$  where  $\leq$  is the partial order.

<sup>47</sup> Please remember the independence property of the F- functionals. This is a very important property and without it we cannot prove the existence of Frege structures.

I shall here try to make the construction a little easier than that described by Aczel (in [AC3]). To construct a logical schema for each constant, i.e. to define the whole logical system, we follow Aczel's intended construction but will carry an example with us at all times. The logical system is defined inductively. As the basis of the induction, we start with a pair  $\chi_0 = (\chi_{0p}, \chi_{0t})$  such that  $\chi_{0t} \subseteq \chi_{0p}$ . Intuitively,  $\chi_{0p}$  is the set of propositions at stage 0 and  $\chi_{0t}$  is the set of truths at stage 0.

Example 1

Let  $\chi_0 = (\chi_{0p}, \chi_{0t}) = (\{0,1\}, \{1\})$ . Note that both  $\{0,1\}$  and  $\{1\}$  are in  $P_\omega$ .

Before proceeding to the induction step, we must define a couple of auxiliary predicates which ensure that the logical constants map their arguments into appropriate values. That is, for each logical constant  $F$ , there is one predicate  $\Phi_F$  which tests whether a particular tuple of arguments has the correct status of propositionhood, and a second predicate  $\Psi_F$  which states the conditions under which the tuple will be mapped into **TRUTH** by  $F$ . To see why we need this, recall the logical schema for negation that we presented under NEGATION above:

(1)

If  $a$  is in **PROP** then  $\neg a$  is in **PROP**, and  $\neg a$  is in **TRUTH** iff  $a$  is not in **TRUTH**.

This is an instance of a general logical schema for those functionals  $F$  in a Frege structure which correspond to truth-functional connectives:

(2)

If  $\check{f}$  is in  $F_{n_1 x \dots x n_k}$  and  $C(F, \check{f})$ , then  $F(\check{f})$  is in **PROP**; and  $F(\check{f})$  is in **TRUTH** iff  $C(F, \check{f})$ , where  $C$  expresses  $F$ 's truth conditions and  $C'$  expresses  $F$ 's propositionhood.<sup>48</sup>

Now it is  $\Phi_F$  which tests that the arguments  $\check{f}$  are in **PROP**, while  $\Psi_F$  does the work of  $C$  in (2).

---

<sup>48</sup> Actually this principle is divided into two parts in (3) below.

Example 2

$\Phi_{\neg}$  and  $\Psi_{\neg}$  take arguments in  $(\cup \chi_i) \times F_0$  and

$\Phi_{\neg}(\chi_0, x)$  is:  $x$  is in  $\chi_{0p}$

$\Psi_{\neg}(\chi_0, x)$  is:  $x$  is not in  $\chi_{0t}$ .

Thus,  $\Phi_{\neg}(\chi_0, x)$  is true of the set  $\chi_{0p} = \{0, 1\}$ , and  $\Psi_{\neg}(\chi_0, x)$  is true of all elements in  $F_0 - \chi_{0t}$ , i.e. everything except the element 1.

In order to carry out the induction step of the construction, we introduce a principle which determines how the propositions and truths at stage  $i+1$  are built from the propositions and truths at stage  $i$ . The principle has two parts:

(3)

(i)  $\chi_{i+1p}$  is the collection of those  $F(\check{f})$  where  $F$  is a logical constant and  $\Phi_F(\chi_i, \check{f})$ .

(ii)  $\chi_{i+1t}$  is the collection of those objects  $F(\check{f})$  where  $F$  is a logical constant and both  $\Phi_F(\chi_i, \check{f})$  and  $\Psi_F(\chi_i, \check{f})$ .

In other words, given the pair  $(\chi_{ip}, \chi_{it})$ , we construct  $(\chi_{i+1p}, \chi_{i+1t})$  in the following way: first,  $\chi_{i+1p}$  has to contain all and only those elements  $F(\check{f})$  such that  $\check{f}$  belongs to the propositions at stage  $i$ , i.e. it is in  $\chi_{ip}$  according to  $\Phi_F(\chi_i, \check{f})$ ; and second,  $\chi_{i+1t}$  must contain all and only those elements  $F(\check{f})$  such that  $\check{f}$  belongs to both the propositions and the truths at stage  $i$ , i.e. it is in  $\chi_{ip}$  and  $\chi_{it}$  according to  $\Phi_F(\chi_i, \check{f})$  and  $\Psi_F(\chi_i, \check{f})$ . Notice that the principle guarantees that  $\chi_{(i+1)t} \subseteq \chi_{(i+1)p}$ .

Example 3

We wish to build  $\chi_1 = (\chi_{1p}, \chi_{1t})$  from  $(\chi_{0p}, \chi_{0t}) = (\{0, 1\}, \{1\})$ . By (3i),  $\chi_{1p}$  is the set of objects  $\neg x$  such that  $\Phi_{\neg}(\chi_0, x)$ , i.e. it is the set  $\{\neg 0, \neg 1\}$ . By (3ii),  $\chi_{1t}$  is the set of objects  $\neg x$  such that  $\Phi_{\neg}(\chi_0, x)$  and  $\Psi_{\neg}(\chi_0, x)$ , i.e. such that  $x$  belongs to  $\chi_{0p}$  but does not belong to  $\chi_{0t}$ . The only thing which satisfies both these conditions is 0, so  $\chi_{1t} = \{\neg 0\}$ .

Example 4

$\Phi_{\&}$  and  $\Psi_{\&}$  take arguments in  $(\cup \chi_i)_x (F_0 \times F_0)$  and

$\Phi_{\&}(\chi_0(x,y))$  is: x and y are in  $\chi_{0p}$

$\Psi_{\&}(\chi_0(x,y))$  is: x and y are in  $\chi_{0t}$ .

Thus, we can supplement the  $\chi_{1p}$  of the previous example with the set of objects  $\&(x,y)$  such that  $(x,y) \subseteq_{\chi_{0p}} \chi_{0p}$ , i.e. the set  $\{0 \& 0, 0 \& 1, 1 \& 0, \dots\}$ . Similarly, we add to  $\chi_{1t}$  the set of objects  $\&(x,y)$  such that  $(x,y) \subseteq_{\chi_{0t}} \chi_{0t}$ , i.e. the set  $\{1 \& 1\}$ . Note that according to our example, the collection of objects in **TRUTH** at stage 1 is  $\{1 \& 1, \neg 0\}$ .

Note also that  $\neg 0, 1 \& 1, 1 \vee 0$  are distinct objects, even though they are all in **TRUTH** and all have the same truth value in Frege's terms. If we wish, we could reconstruct Frege's notion of the True and the False by forming the relevant equivalence classes, but Frege structures give us an intensional ontology. This is justified on the grounds that objects with the same truth value, e.g.  $\neg 0$  and  $1 \& 1$  are equivalent in truth value but distinct. We will return to questions of intensionality in Chapter 5.

We see that the pair is being enlarged at each step starting from the first step where we take  $\chi_{0p} = \{0,1\}$  and  $\chi_{0t} = \{1\}$ , with the property that for each  $i$  we have:  $\chi_{it} \subseteq \chi_{ip}$ . Note that we are not imposing the condition that  $\chi_{it} \subseteq \chi_{(i+1)t}$  or  $\chi_{ip} \subseteq \chi_{(i+1)p}$ ; in fact our construction is monotonic in another sense which we shall see below. The aim is now to keep going up to a certain level  $\alpha$  where  $\chi_{\alpha} = (\chi_{\alpha p}, \chi_{\alpha t})$  is a logical system, because it is obvious that  $\chi_i$  at the levels we met so far are not logical systems. Take for example  $\chi_0$  in our example above based on FE. Then  $\chi_0$  is not a logical system, as can be seen by taking the logical schema for  $\neg$  :

If  $a$  is a proposition then  $\neg a$  is a proposition such that  $\neg a$  is true iff  $a$  is not true.

$\chi_0$  is not a logical system because  $1$  is in  $\chi_{0p}$  (supposed to represent propositions) but

$\neg 1$  is not in  $\chi_{0p}$ . Nor is  $\chi_1$  a logical system because  $\neg 1$  is in  $\chi_{1p}$  but  $\neg \neg 1$  is not in  $\chi_{1p}$  and so on. To solve this problem, let us consider the fixed point (if it exists) of this construction. It may be that the fixed point is a logical system and if so, we have succeeded. Before we prove that the fixed point is a logical system, let us remind ourselves again of the construction. The construction is built through an operator  $Y$  which takes us from level  $i$  to level  $i+1$  in such a way that  $Y(\chi_i) = \chi_{i+1}$ , where  $\chi_i = (\chi_{ip}, \chi_{it})$ ,  $\chi_{i+1} = (\chi_{i+1p}, \chi_{i+1t})$ ,  $\chi_{it} \subseteq \chi_{ip}$ ,  $\chi_{i+1t} \subseteq \chi_{i+1p}$ . Moreover  $\chi_{i+1p}$  and  $\chi_{i+1t}$  are obtained as follows:

For any  $F$ - functional  $F$ ,  $\chi_{i+1p}$  is the collection of those  $F(\check{f})$  where  $F$  is a logical constant and  $\Phi_F(\chi_i, \check{f})$  and  $\chi_{i+1t}$  is the collection of those objects  $F(\check{f})$  where  $F$  is a logical constant and both  $\Phi_F(\chi_i, \check{f})$  and  $\Psi_F(\chi_i, \check{f})$ .

Now we prove that any  $\chi$  such that  $\chi = Y(\chi)$  is a logical system. To show that, we have to prove that for each logical constant  $F$ , the logical schemata of  $F$  holds in  $\chi$ . Let  $F$  be a logical constant whose logical schema is as follows:

If  $\check{f}$  is in  $F_{n_1} x \dots x F_{n_k}$  and  $\Phi_F(\chi, \check{f})$ , then  $F(\check{f})$  is in  $\chi_p$ ; and  $F(\check{f})$  is in  $\chi_t$  iff  $\Psi_F(\chi, \check{f})$ .

Let us prove that this schema holds in  $\chi$  where  $\chi$  is a fixed point,  $\chi = (\chi_p, \chi_t)$  and  $Y(\chi) = (\chi'_p, \chi'_t)$ . Let  $\check{f}$  be in  $F_{n_1} x \dots x F_{n_k}$  where  $\Phi_F(\chi, \check{f})$ . As  $\Phi_F(\chi, \check{f})$  then  $F(\check{f})$  is in  $\chi'_p$  by definition, but  $\chi'_p = \chi_p$  (because  $\chi = Y(\chi)$ ), therefore  $F(\check{f})$  is in  $\chi_p$ . Now let us prove that  $F(\check{f})$  is in  $\chi_t$  iff  $\Psi_F(\chi, \check{f})$ .

( $\Rightarrow$ ) If  $F(\check{f})$  is in  $\chi_t$  then  $F(\check{f})$  is in  $\chi'_t$ . As  $F(\check{f})$  is in  $\chi'_t$  then there exists an  $F$ - functional  $G$  and a sequence  $\check{g}$  in  $F_{n_1} x \dots x F_{n_k}$  such that  $F(\check{f}) = G(\check{g})$  and  $\Phi_G(\chi, \check{g})$  and  $\Psi_G(\chi, \check{g})$  by definition. But the logical constants are independent. Therefore  $F = G$  and as the family is primitive,  $\check{f} = \check{g}$ . Therefore we have from  $\Psi_G(\chi, \check{g})$  that  $\Psi_F(\chi, \check{f})$ .

( $\Leftarrow$ ) Suppose  $\Psi_F(\chi, \check{f})$ , since also  $\Phi_F(\chi, \check{f})$  then  $F(\check{f})$  is in  $\chi'_t$ ; but  $\chi'_t = \chi_t$ , therefore  $F(\check{f})$  is in  $\chi_t$ .



This implies that the logical schema of  $F$  holds in  $\chi$ . Now we know that if there exists a fixed point  $\chi$  then this  $\chi$  is a logical system. Let us find a fixed point.

We define an ordering  $\leq$  on  $(\chi_i)_i$  as follows:

$$\chi_i \leq \chi_{i+1} \text{ if}$$

$$(i) \chi_{ip} \subseteq \chi_{i+1p}, \text{ and}$$

$$(ii) \text{ if } x \text{ is in } \chi_{ip}, \text{ then } x \text{ is in } \chi_{it} \text{ iff } x \text{ is in } \chi_{i+1t}.$$

With this ordering we can show that  $Y$  is monotonic.<sup>49</sup> Note that the levels can be any ordinal even a transfinite one, for if we are at a finite ordinal  $i$  we define  $Y(\chi_i) := \chi_{i+1}$  as above. If we are at a limit ordinal  $j$ , we define  $Y(\chi_j) = \cup \chi_i$  for  $i < j$ . Applying the fixed point theorem we get a fixed point of  $Y$ . The reason for this is of course the monotonicity of the operator  $Y$ , as we know that the ordering relation  $\leq$  is a partial ordering on all those pairs.

---

<sup>49</sup> This is due to the fact that for each  $i \geq 0$ , we follow the logical schemata for each  $F$ -functional to go from level  $i$  to level  $i+1$ , and monotonicity is hidden in those schemata together with the fact that we have an independent family of  $F$ -functionals.

*A.3. Frege structures as models and comparison with Scott domains*

$\lambda$ -structures are models of the  $\lambda$ -calculus in an obvious way. For just take the interpretation of terms as follows over a defined Frege structure  $F$ , where  $g$  is an assignment function which takes variables into objects of  $F_0$  :

$$\begin{aligned} [[x]]_{g, F} &= g(x) \\ [[MN]]_{g, F} &= \text{app} ([[M]]_{g, F}, [[N]]_{g, F}) \\ [[\lambda xM]]_{g, F} &= \lambda < [[M]]_{g[a/x], F} / a >^{50} \end{aligned}$$

Now it is easy to show that this interpretation has the property that:

$$\lambda \vdash M=N \implies [[M]]_{g, F} = [[N]]_{g, F} .^{51}$$

Therefore, Frege structures are models of the  $\lambda$ -calculus and in turn we know that they solve the second problem. For the remainder of this section, we shall concentrate on the comparison between both Scott domains and Frege structures as models, and hence help justify our claim that Frege structures are better candidates for the semantics of natural languages than Scott domains.

On Scott domains, one has a topology (Scott topology based on a partial ordering relation) and two special elements Top and Bottom. (Bottom is less than all the other elements and Top is greater than all of them.) We shall see in Chapter 7 that this ordering relation, together with the existence of Bottom and the requirement that the functions be continuous, make Scott domains problematic for the semantics of natural languages. On Frege structures, however, we have no ordering and no requirement on the continuity of functions. What we have in a Frege structure is a collection of objects  $F_0$  together with, for each  $n$ , a collection  $F_n$  of  $n$ -ary functions which take elements of  $F_0$  as arguments and return elements of  $F_0$  as values. But although we do not consider all possible functions to be elements of the Frege structure, we still consider only structures which are explicitly closed. This explicit

<sup>50</sup> Note that the second  $\lambda$  above is the  $\lambda$ -structure one whereas the first one is the formal language one.

<sup>51</sup>  $\lambda \vdash M=N$  means that  $M=N$  is derivable in the  $\lambda$ -theory.

closure imposes the existence of some necessary functions such as projections, constants, etc, and requires the closure of our structure under some important functional operations such as composition. We have both constants for functions and variables for functions, but the functionality on a Frege structure does not stop at those first order functions; we also have functionals. However, whereas for functions our language contains both variables and constants, for functionals it only contains constants.

One should bear in mind that none of the collections **PROP**, **TRUTH** or **SET** is internally definable. Intuitively, we say that a collection  $\chi$  of objects is internally definable if we can talk about it through the object language and just not the metalanguage. An example of a collection which is not internally definable is the collection of truths in a theory which contains names for its wffs. If this collection was internally definable, then there must be a predicate  $T$  such that for any object  $a$ ,  $T(a)$  is true iff  $a$  is true.<sup>52</sup> But according to Tarski, a theory cannot contain its own truth predicate (in the object language) without falling into inconsistency and therefore  $T$  is a predicate of the metalanguage. Now if we want to talk about truth in this metalanguage then again we have to have a truth predicate  $T'$  in the meta-metalanguage and this process iterates. Just as  $T$  is not an element of the object language in Tarski's approach, so inside a Frege structure the collection of truths is not internally definable. Aczel gives a more formal definition of internal definability and considers a collection  $\chi$  of objects in  $F_0$  to be internally definable in the Frege structure iff there exists a propositional function  $C$  in  $F_1$  such that the following holds:

(\*\*) For any object  $a$  in  $F_0$ ,  $C(a)$  is in **TRUTH** iff  $a$  is in  $\chi$ .

It might be clearer if we set **FALSE** = **PROP** - **TRUTH**, and then replace (\*\*) be the following:

---

<sup>52</sup> Note that we do not restrict this condition to every wff but range it over all objects.

(\*\*\*) For any object  $a$  in  $F_0$ ,  $C(a)$  is in **TRUTH** iff  $a$  is in  $\chi$  and  $C(a)$  is in **FALSE** otherwise.

Some might find it easier to draw a contrast with the following schematic definition, where  $C$  is not a propositional function:

(\*\*\*\*) For any object  $a$  in  $F_0$ ,  $C(a)$  is in **TRUTH** iff  $a$  is in  $\chi$  and  $C(a)$  is in  $F_0$  otherwise.

(\*\*\*) makes  $\chi$  decidable, while (\*\*\*\*) only makes semi-decidable.

It may seem unfortunate that the collection of truths is not internally definable, but it is essentially this that provides Frege structures with consistency. Notice that since elements of **SET** are the nominalisations of propositional functions, we have no way of talking about the nominalised items internally and **SET** is not internally definable. Moreover it may also seem that we will encounter a problem in defining second order quantifiers. I hope that it will become clear throughout the work that the inability to internally define quantifiers does not have any serious effects. On the contrary, we keep to simplicity while being able to formalise many concepts within the theory.

The undefinability of **PROP** and of **SET** is due to the undefinability of **TRUTH**. The collection of propositions is not internally definable, for if it were (through a predicate  $P$ ) we would find that **TRUTH** is also internally definable (through the propositional function  $\langle P(x) \ \&\rightarrow\ x/x \rangle$ , which stands for a function in  $F_1$ ). That **PROP** is not internally definable implies that **SET** is not either. This is because if  $S$  were a propositional function in  $F_1$  internally defining **SET** then  $\langle S(\{y/x\})/x \rangle$  is a propositional function in  $F_1$  internally defining the collection of propositions. Note also that, for each  $n$ ,  $P F_n$  (the collection of  $n$ -ary propositional functions) is not internally definable. For if it were, we get that the collection of propositions is also. The proof here needs an extension of the definition of internal definability so that instead of having a function we have a functional.

Let us return to the comparison of Frege structures with Scott domains. Frege structures do not have any ordering or continuity problems and their restricted logic<sup>53</sup> would allow us to solve the problems of Scott domains (and of Cocchiarella). But of course the solving part is not going to be easy. We have to do something about the non-internal definability of SET. There are a few ways to go here: we have to either see how the function domains (as with Scott) could be built inside Frege structures, or else show that we do not need second level quantifiers and therefore the problem does not arise. Now the word inside brings an uncomfortable feeling - especially after we pointed out that all the interesting collections are not internally definable. I assure the reader however that this difficulty is only temporary and that we can always find solutions to the problem.<sup>54</sup> It is important for the reader to know that a Frege structure can be built on the top of a model where continuity and ordering play a very important role (such as  $E_\infty$ ). However the way quantifiers are constructed on a Frege structure using the fixed point, is not based on the ordering relation, and so the problem that faced Turner in his work based on  $E_\infty$  (where quantifiers depended on the ordering relation - see Chapter 7) is not faced by the quantifier treatment on a Frege structure.

The fact that functions, but not functionals, can be mapped into  $F_0$  in a Frege structure is not a disadvantage, indeed it may even be seen as a virtue, since there appears to be no justification in NL semantics for nominalizing expressions - for example determiners - which would require a formalisation as functionals. Also, in Frege structures we have more possible elements than we do in Scott domains. We have propositions, truths and sets which are all legitimate elements of the Frege structure. We could not talk about them internally but that is how it should be. Tarski's undefinability of Truth and Gödel's famous result<sup>55</sup> make it impossible for

---

<sup>53</sup> i.e.  $\neg$ ,  $\&$ , etc do not necessarily have to apply to propositions only but can be applied to any objects and the result will be a proposition only in case the objects themselves are.

<sup>54</sup> Such details are again examined in Chapter 4.

<sup>55</sup> According to Gödel's theorem, we can only give a proof relative to some other system. The two theorems of Gödel are:

us to be able to internally define any of these collections. So, our inability to internally define any of these collections is not a weakness in comparison with Scott domains; Scott domains could not talk about them at all, and therefore can not be adequate for NL semantics. If we try to extend Scott domains in a way that will allow us to talk about truths and propositions, we obtain Frege structures.

---

(1) For any formal system consistent and strong, sufficient for arithmetic, there exists a sentence  $\Phi$  formalisable in that system which is true but not provable.

(2) No consistent formal system which is strong enough for arithmetic is capable of proving its own consistency.

(2)  $\equiv$   $\neg$ (1), for  $\Phi$  is taken to be " $\Phi$  is true but not provable".

*PART B. A THEORY OF PROPERTIES, ITS SEMANTICS AND PROOF THEORY*

In this part, I introduce the theory to be used throughout the rest of the thesis. This theory is first order, intensional and type free. There have been many arguments for both type freeness (e.g. Feferman's work [FE1], [FE2] and [FE9]) and intensionality: further justifications for adopting these features will not be given here. But why use a first order theory? The reason for using a first order theory is due to the very nature of Frege structures; this does not, however, imply that one cannot interpret higher order languages with Frege structures; that would be incorrect. For instance, the highly typed language of Martin-Löf could be interpreted with Frege structures. Nonetheless, first order languages are easy to work with, and it is well known that higher order languages can be reduced to first order languages having extra predicates to simulate the types.<sup>56</sup> People turned to higher order languages for many reasons, two of them being the issue of expressive power and the paradoxes. On the question of expressive power, we can talk about second and higher order quantifiers because they can be reduced to first order ones (we defined in Part B, clause (14'),  $[[\forall X \Psi]]_g = \forall \langle [[\Psi]]_{g[|a|/X]} / a \rangle$ ).<sup>57</sup> On the question of the paradoxes, we see that people are returning to first order theories (e.g. Feferman, Turner).

I assume full extensionality of functions, in the sense that the following principle holds:

$$(EE) \quad \forall x [f(x) = g(x)] \rightarrow f=g, \text{ or}$$

"if two propositional functions are true of the same arguments, then they are identical"

It is important to distinguish (EE) from a principle of extensionality of properties which might be formulated as:

$$(EP) \quad \forall x [f(x) \text{ is true iff } g(x) \text{ is true}] \rightarrow f= g, \text{ or}$$

<sup>56</sup> Neither am I implying that types and higher order are the same thing.

<sup>57</sup> Barwise and Cooper ([BA3]) argued that first order languages cannot be used to define some quantifiers like Most; but we are concerned with expressivity here and not definability.

"if two propositional functions are true of the same arguments, then they are identical"<sup>58</sup>

(EP) is rejected in the current framework. Put briefly, equivalence does not imply equality; in fact, if it did, a version of Russell's paradox could be constructed. If we had taken  $f$  and  $g$  to be propositional functions which denoted truth values, then (EE) and (EP) would of course collapse. The point to be emphasised is that in a Frege structure, it can be the case that  $f(x)$  and  $g(x)$  are both in **TRUTH**, yet  $f(x) \neq g(x)$ , since they are distinct propositions.

Now let us put forward the theory  $T_{\Omega}$  and its semantics. This is a theory of properties and propositions and for any term  $t$ ,  $\Omega t$  is to be understood as 't is a proposition'.<sup>59</sup>

*B.1. The theory  $T_{\Omega}$*

*Syntactic categories and items*

Let us consider the following categories and items:

$x, y, z, x_1, y_1, \dots$  range over the category of individual variables which is denumerably infinite.

$c, c', c_0, c_1, \dots$  range over the category of individual constants which is denumerably infinite.

$t, t', t'', t_1, t_2, \dots$  range over terms.

We have the following operators and logical constants:  $=, \Omega, \vee, \&, \rightarrow, \downarrow, \forall, \lambda, \text{app}$ .

*Syntactic clauses for  $T_{\Omega}$ :*

$$t := x \mid c \mid \text{app}(t_1, t_2) \mid t_1 = t_2 \mid t_1 \& t_2 \mid t_1 \vee t_2 \mid t_1 \rightarrow t_2 \mid \Omega t_1 \mid \lambda x. t_1 \mid \forall x t_1 \mid \downarrow x t_1.$$

We define  $\perp =_{df} c_0 = c_1$ <sup>60</sup> and define three more logical constants ' $\neg$ ', ' $\equiv$ ' and ' $\&\rightarrow$ '

<sup>58</sup> Of course (EE) and (EP) collapse if we take the semantic value of  $f(x)$  (and  $g(x)$ ) to be a truth value.

<sup>59</sup> In Chapter 3 we define properties in terms of propositions.

<sup>60</sup> Where  $c_0$  and  $c_1$  are two distinct constants.



out of the previous ones as follows:

$$\neg t =_{df} t \rightarrow \perp$$

$$t_1 \equiv t_2 =_{df} (t_1 \rightarrow t_2) \& (t_2 \rightarrow t_1)$$

$$t_1 \& \rightarrow t_2 =_{df} t_1 \& (t_1 \rightarrow t_2).$$

Bound/free variables and substitution are defined as usual; in  $t_1[t_2/x]$  the bound variables of  $t_1$  are changed to avoid collision.

As can be seen from the above, we only have terms (which are defined recursively using the logical constants  $\vee, \&, \rightarrow, \perp, \forall, =$  and the three important operators  $\Omega, \lambda$  and  $\text{app}$ ). It is the tradition to define inductively both terms and wffs; by contrast everything here is a term and the logical constants operate not only on the propositions but on all terms.

*Axioms and Rules:*

$$(\alpha) \lambda x.t = \lambda y.t[y/x] \text{ where } y \text{ is not free in } t$$

$$(\beta) \text{app}((\lambda x.t), t') = t[t'/x]$$

$$(\gamma) \frac{t_1 = t_2 \quad t'_1 = t'_2}{\text{app}(t_1, t'_1) = \text{app}(t_2, t'_2)}$$

$$(\delta) \frac{t = t' \quad t = t''}{t' = t''}$$

$$(\epsilon) \frac{\text{app}(t, x) = \text{app}(t', x)}{t = t'} \text{ where } x \text{ is not free in } t, t' \text{ or any open assumptions}$$

From the axioms so far, we can deduce the following theorems:

*Theorem :*

$$(i) = \text{ is reflexive: i.e. (r) } t = t \text{ for any term } t.$$

$$(ii) = \text{ is symmetric: i.e. (s) } \frac{t = t'}{t' = t}$$

$$(iii) = \text{is transitive: i.e. } (t) \frac{t = t' \quad t' = t''}{t = t''}$$

Proof:

$$\text{For (i), } \frac{\text{app}(\lambda x.t, x) = t \quad \text{app}(\lambda x.t, x) = t}{t = t} \quad \begin{array}{l} \text{By } (\beta) \\ \text{By } (\delta) \end{array}$$

(ii) is now easy to deduce from both ( $\delta$ ) and reflexivity as follows:

$$\frac{t = t'}{t = t' \quad t = t \quad (\text{reflexivity})} \\ \frac{}{t' = t \quad \text{By } (\delta)}$$

Also (iii) is easy to deduce as:

$$\frac{t = t' \quad t' = t''}{t' = t \quad t' = t''} \quad (\text{symmetry}) \\ \frac{}{t = t''} \quad (\delta)$$

□

*Theorem* : From the above rules we can deduce

$$(\zeta) \frac{t = t'}{\lambda x.t = \lambda x.t'}$$

Proof:

$$\frac{t = t'}{t[x/x] = t'[x/x]} \\ \frac{\text{app}(\lambda x.t, x) = \text{app}(\lambda x.t', x)}{\lambda x.t = \lambda x.t'} \quad \begin{array}{l} (\beta) \text{ and transitivity} \\ (\epsilon) \end{array}$$

and therefore ( $\zeta$ ) is a theorem. □

*Theorem* : We can also deduce from above that ( $\eta$ ) is a theorem, where ( $\eta$ ) is:

$$(\eta) \quad (\lambda y.\text{app}(u, y)) = u \text{ for } y \text{ not free in } u.$$

Proof:

$$\frac{\text{app}(u, y)[x/y] = \text{app}(u, x) \quad \text{as } y \text{ is not free in } u}{\text{app}(\lambda y.\text{app}(u, y), x) = \text{app}(u, x) \quad \text{as } x \text{ is not free in } u} \quad (\epsilon) \\ \lambda y.\text{app}(u, y) = u$$

□

Note, however, that from (ε) above we have been able to deduce both (ζ) and (η), but from (ζ) alone we cannot deduce (ε) as we will also need (η) in the derivation.

This is because if we start the proof:

$$\frac{\text{app}(t,x) = \text{app}(t',x)}{\lambda x.\text{app}(t,x) = \lambda x.\text{app}(t',x)} \quad (\zeta)$$

Then we get to a stage where we need (η) to be able to deduce  $t=t'$ . Note also that the rules

$$\frac{\text{app}(t_1,t'_1) = \text{app}(t_2,t'_2)}{t_1 = t_2}$$

$$\frac{\text{app}(t_1,t'_1) = \text{app}(t_2,t'_2)}{t'_1 = t'_2}$$

are not valid; this means that the converse of (γ) does not hold. Take for example,  $t_1$  and  $t_2$  to be  $\lambda x.x=c'$  and  $\lambda x.c=x$  respectively, where  $c, c'$  are two distinct constants; take also  $t'_1$  and  $t'_2$  to be  $c'$  and  $c$  respectively, then we have that  $\text{app}(t_1,t'_1) = \text{app}(t_2,t'_2)$ , but we do not have  $t_1 = t_2$  or  $t'_1 = t'_2$ .<sup>61</sup>

(α)-(ε) are just axioms and rules of the lambda calculus with extensionality; we still need a logic and we therefore add the following:

$$(VI) \quad \frac{t \quad \Omega t'}{tVt'} \quad \frac{t' \quad \Omega t}{tVt'}$$

$$(VE) \quad \frac{tVt' \quad \{t\}_s \quad \{t'\}_s}{s}$$

<sup>61</sup> This will force us to introduce in the third chapter a relation called pred which makes the two derivations above valid. 'pred' is highly intensional in that if  $\text{pred}(t_1,t_2) = \text{pred}(t'_1,t'_2)$  then we have  $t_1 = t'_1$  and  $t_2 = t'_2$ .

$$(\Omega V) \frac{\Omega t \quad \Omega t'}{\Omega(tVt')} \quad \frac{\Omega(tVt')}{\Omega t} \quad \frac{\Omega(tVt')}{\Omega t'}$$

$$(\&I) \frac{t \quad t'}{(t\&t')}$$

$$(\&E) \frac{t\&t' \quad t\&t'}{t \quad t'}$$

$$(\Omega\&) \frac{\Omega t \quad \Omega t'}{\Omega(t\&t')} \quad \frac{\Omega(t\&t')}{\Omega t} \quad \frac{\Omega(t\&t')}{\Omega t'}$$

$$(\rightarrow I) \frac{\Omega t \quad \{t\} \quad t'}{t \rightarrow t'}$$

$$(\rightarrow E) \frac{t \quad t \rightarrow t'}{t'}$$

$$(\Omega \rightarrow) \frac{\Omega t \quad \{t\} \quad \Omega t'}{\Omega(t \rightarrow t')}$$

$$(\Omega \perp) \quad \Omega \perp \text{ } ^{62}$$

$$(\perp) \frac{\perp}{t}$$

$$(\exists I) \frac{t[t']}{\exists t \text{ } ^{63}}$$

<sup>62</sup> Note that this axiom is redundant, as other axioms may be.

<sup>63</sup> We mean  $\exists x t[x]$ ; and hence we have to impose the condition that  $t'$  is free for  $x$  in  $t$ .

$$(\exists E) \frac{\exists x \{t[x]\}}{s} \quad \text{provided } x \text{ is not free in } t, s \text{ or any open assumption}$$

$$(\Omega\exists) \frac{\Omega(t[x])}{\Omega(\exists x t)} \quad x \text{ not free in } t \text{ or any open assumption}$$

$$(\forall I) \frac{t[x] \quad x \text{ not free in } t \text{ or any open assumption}}{\forall x t}$$

$$(\forall E) \frac{\forall x t}{t[t']}$$

$$(\Omega\forall) \frac{\Omega(t[x])}{\Omega(\forall x t)^{64}} \quad x \text{ not free in } t \text{ or any open assumption}$$

$$(\Omega) \frac{t}{\Omega t}$$

$$(\Omega=) \quad \Omega(t=t')$$

$$(\Omega_{\text{sub}}) \frac{t=t' \quad \Omega(t''[t])}{\Omega(t''[t'])}$$

$$(T_{\text{sub}}) \frac{t=t' \quad t''[t]}{t''[t']}$$

---

64

Where  $\frac{a}{b}$  means both  $\frac{a}{b}$  and  $\frac{b}{a}$

B.2. The metatheory of  $T_\Omega$

We write  $\vdash t$  if  $t$  is a theorem of  $T_\Omega$  and  $\Gamma \vdash t$  if  $t$  is deducible from  $\text{Axioms}(T_\Omega) \cup \Gamma$ .

*Theorem:* We can prove from our rules above that:

$$(T1) \frac{t = t' \quad t}{t'}$$

$$(T2) \frac{t_i = t'_i \text{ for } i=0, \dots, n \quad t_0[t_1/x_1, \dots, t_n/x_n]}{t'_0[t'_1/x_1, \dots, t'_n/x_n]}$$

$$(T3) \frac{t = t' \quad \Omega t}{\Omega t'}$$

Proof:

(T1) is deducible from  $(T_{\text{sub}})$

(T2) is deducible by induction on the way terms are constructed.

(T3) is deducible from  $(\Omega_{\text{sub}})$ .

Now before completing the metatheory of  $T_\Omega$  we stop mention with other theories that were offered as theories of Frege structures. In [AC4] Aczel offered a language of Frege structures but he made negation primitive (not defined). Here, I use  $\perp$  instead.

Flagg and Myhill (in [FL1] and [FL2]) offered a theory based on the  $\lambda$ -calculus. This implied that where we had to choose between substitution and application in our axioms and rules, they could use application only. In [SM1], Smith offers a theory of Frege structures with the aim of interpreting Martin-Löf type theory. I must also mention Mönnich in [MU1] who used Frege structures, but the theory was derived from [AC4]. Also Beeson in [BE4] offered an axiomatic theory of Frege structures.

Now we proceed with the metatheory of  $T_\Omega$

(T4) If  $\vdash t \& t'$  then  $\vdash t' \& t$

(T5) If  $\vdash t \vee t'$  then  $\vdash t' \vee t$

- (T6) If  $\vdash t$  then  $\vdash \neg\neg t$ <sup>65</sup>
- (T7)  $\{\Omega t\} \vdash t \equiv t$
- (T8) If  $\vdash t \equiv t'$  and  $\vdash t' \equiv t''$  then  $\{\Omega t, \Omega t'\} \vdash t \equiv t''$
- (T9) If  $\vdash t \equiv t'$  then  $\vdash t' \equiv t$
- (T10) If  $\vdash \forall t$  then  $\vdash \neg ] (\neg t)$
- (T11)  $\{\Omega(t[x])\} \vdash \forall t \rightarrow \neg ] (\neg t)$   $x$  not free in  $t$   
or any open assumption<sup>66</sup>
- (T12) If  $\vdash t$  then  $\vdash \forall t$
- (T13) If  $\Gamma \vdash t$  then  $\Gamma \vdash \forall x t$  for  $x$  not free in  $\Gamma$ .
- (T14) If  $\Gamma \vdash \forall t$  then  $\Gamma \vdash t$
- (T15)  $\vdash ] x. x$
- (T16)  $\vdash ] x. \neg x$
- (T17) If  $t \in \Gamma$  then  $\Gamma \vdash t$
- (T18) From  $\Gamma \vdash t$  and  $\Gamma \vdash t \rightarrow t'$  we deduce  $\Gamma \vdash t'$
- (T19) If  $\Gamma \cup \{t\} \vdash t'$  then  $\Gamma \cup \{\Omega t\} \vdash t \rightarrow t'$ <sup>67</sup>
- (T20)  $\{\Omega t\} \vdash t \rightarrow (\neg t \rightarrow t')$
- (T21)  $\{\Omega t\} \vdash t \rightarrow \neg\neg t$ <sup>68</sup>
- (T22)  $\{\Omega t\} \vdash \neg t \equiv \neg\neg\neg t$
- (T23)  $\{\Omega t, \Omega t'\} \vdash \neg(t \vee t') \equiv \neg t \& \neg t'$ <sup>69</sup>
- (T24) If  $\{\Omega t, \Omega t'\} \vdash t \equiv t'$  then  $\{\Omega t, \Omega t'\} \vdash \neg t \equiv \neg t'$
- (T25) If  $\Gamma \vdash \perp$  then  $\Gamma \vdash t$
- (T26) If  $\Gamma \vdash t$  then  $\Gamma \vdash ] t$
- (T27) If  $\Gamma \vdash t$  then  $\Gamma \cup \Lambda \vdash t$
- (T28) If  $\Gamma \vdash ] x t[x]$  and  $\Lambda \cup \{t[y]\} \vdash t'$  then  $\Gamma \cup \Lambda \vdash t'$  for  $y$  not free in  $\Lambda$

<sup>65</sup> The other direction does not necessarily hold.

<sup>66</sup> The other side does not necessarily hold.

<sup>67</sup> This is known as the deduction theorem; please note the insertion of  $\{\Omega t\}$ . This is important as without it we would get Curry's paradox as is explained in Chapter 3.

<sup>68</sup> The other direction does not necessarily hold.

<sup>69</sup> But not necessarily  $\{\Omega t, \Omega t'\} \vdash \neg(t \& t') \equiv \neg t \vee \neg t'$

(T29) If  $\Gamma \vdash t$  and  $\Lambda \vdash t'$  then  $\Gamma \cup \Lambda \vdash t \& t'$

(T30) If  $\Gamma \vdash t \forall t'$ ,  $\Lambda \cup \{t\} \vdash t_1$  and  $\chi \cup \{t'\} \vdash t_1$  then  $\Gamma \cup \Lambda \cup \chi \vdash t_1$

(T31) If  $\Omega a$  then  $\neg(a \& \neg a)$  i.e.  $\{\Omega a\} \vdash \neg(a \& \neg a)$

(T32)  $\{\Omega t, \Omega t'\} \vdash t \rightarrow (t' \rightarrow t)$

(T33)  $\{\Omega t, \Omega t'\} \vdash (t \rightarrow t') \rightarrow ((t \rightarrow \neg t') \rightarrow \neg t)$

(T34)  $\{\Omega t, \Omega t'\} \vdash t \rightarrow t \forall t'$

(T35)  $\{\Omega t, \Omega t'\} \vdash t \rightarrow t' \forall t$

(T36)  $\{\Omega t, \Omega t'\} \vdash t \rightarrow (t' \rightarrow t \& t')$

(T37)  $\{\Omega t, \Omega t', \Omega t'', t \equiv t'\} \vdash t''[t \equiv t']$

(T38) If  $\vdash t$  and  $\vdash t \rightarrow t'$  then  $\vdash t'$ <sup>70</sup>

Proof:

(T4) holds because from  $t \& t'$  you can deduce  $t'$  and from  $t \& t'$  you can deduce  $t$ ; but from  $t'$  and  $t$  we deduce  $t' \& t$ .

(T5) holds because if  $\vdash t \forall t'$  then  $\Omega(t \forall t')$  and so  $\Omega t$  and  $\Omega t'$ . Hence, if we assume  $t$  we get  $t' \forall t$  as  $\Omega t'$  and if we assume  $t'$  we get  $t' \forall t$  as  $\Omega t$ ; this implies that  $t' \forall t$  due to (VE).

To prove (T6) it is enough to say that from  $t$  we deduce  $\Omega t$  and from the assumption  $\neg t$  we get  $\perp$  since  $t$  is true; hence by ( $\neg I$ ) we get  $\neg t \rightarrow \perp$ , i.e.  $\neg \neg t$ .

The following is a proof of (T7):

$$\begin{array}{l} \{t\} \\ \Omega t \quad t \\ \hline t \rightarrow t \end{array} \quad \text{By } (\rightarrow I)$$

From  $t \rightarrow t$  and  $t \rightarrow t$  we get  $t \equiv t$ .

For (T8), the proof is as follows:

If  $\vdash t \equiv t'$  and  $t' \equiv t''$ , then  $\vdash t \rightarrow t'$  and  $\vdash t' \rightarrow t''$ ; hence  $\{\Omega t\} \vdash t \rightarrow t''$ .

In the same way we can show that  $\{\Omega t''\} \vdash t'' \rightarrow t$ .

Therefore  $\{\Omega t, \Omega t''\} \vdash t = t''$  and so (T8) is a theorem.

The proof of (T9) goes as follows:

<sup>70</sup> We refer to this as Modus Ponens.



If  $\vdash t \equiv t'$ , then  $\vdash (t \rightarrow t') \& (t' \rightarrow t)$ ;

hence  $\vdash (t' \rightarrow t) \& (t \rightarrow t')$  and so  $\vdash t' \equiv t$ .

For (T10) we have to show that  $\vdash \neg \exists (\neg t)$ , i.e. that  $\vdash \exists (\neg t) \rightarrow \perp$ ;

But  $\Omega(\forall t)$ , hence  $\Omega(t[x])$  and so  $\Omega(T[x])$  for  $x$  not free in  $t$  or any open assumption.

$\Omega(\neg t[x])$  implies  $\Omega(\exists (\neg t))$  by  $(\Omega \exists)$ , as  $x$  is not free in  $t$  or any open assumption.

If we assume that  $\exists (\neg t)$  then from  $\neg t[x]$  where  $x$  not free in  $\perp$ , i.e.  $t[x] \rightarrow \perp$  we get  $\perp$  as  $\vdash \forall t$ . Hence from the assumption  $\exists (\neg t)$  we get  $\perp$ .

Now applying  $(\Omega)$  we have  $\Omega(\exists (\neg t))$  and from assumption  $\exists \neg t$  we get  $\perp$ , then  $\exists (\neg t) \rightarrow \perp$  and so  $\vdash \neg \exists (\neg t)$ .

For (T11), the proof goes as follows:

$\{\Omega(t[x])\} \vdash \Omega(\forall t)$  by  $(\Omega V)$ .

But by (T10), if  $\vdash \forall t$ , then  $\vdash \neg \exists (\neg t)$ ,

hence  $\{\Omega(t[x])\} \vdash \forall t \rightarrow \neg \exists (\neg t)$ .

In (T13), the condition that  $x$  not free in  $\Gamma$  is there to enable us to apply VI which imposes that  $x$  be not free in  $t$  or any open assumptions.

(T14) is a consequence of (VE).

We know that  $=$  is reflexive, hence  $c=c$  and so  $x[c=c]$  is true; this implies by  $(\exists I)$  that  $\exists x.x$ , hence (T15).

We know that  $\Omega \perp$  and that from  $\perp$  we deduce  $\perp$ , hence  $\perp \rightarrow \perp$  and so  $\neg \perp$ . If we take  $t$  to be  $\neg x$ , then  $t[\perp]$  is  $\neg \perp$  and is true. By  $(\exists I)$  we get  $\exists x.\neg x$  and so (T16) is a theorem.

(T17) is true due to our definition of  $\Gamma \vdash t$ .

(T18) is a consequence of  $(\rightarrow E)$ .

The proof of (T19) is very straightforward due to the addition of  $\Omega t$ ; without that, we could not have proved it.

The proof of (T20) goes as follows:

$$\begin{array}{l}
 \{\Omega t\}1 \\
 \quad \{t\}2 \\
 \quad \quad \{\neg t\}3 \\
 \quad \quad \quad \underline{\quad} \text{By } (\rightarrow E) \\
 \text{-----} \text{By } (\underline{\quad}). \\
 \quad t'4 \\
 \Omega(\neg t) \quad \text{By 1 and } (\Omega\neg) \\
 \text{-----} \\
 \neg t \rightarrow t' \text{ By 3, 4 and } (\rightarrow I) \\
 \text{-----} \\
 t \rightarrow (\neg t \rightarrow t')
 \end{array}$$

Hence  $\{\Omega t\} \vdash t \rightarrow (\neg t \rightarrow t')$ .

The proof of (T21) goes as follows:

$$\begin{array}{l}
 \{\Omega t\}1 \\
 \quad \{t\}2 \\
 \quad \quad \Omega(\neg t)3 \text{ from 1} \\
 \quad \quad \quad \{\neg t\}4 \\
 \text{-----} (\rightarrow E) \\
 \quad \underline{\quad} \\
 \text{-----} (\rightarrow I) \\
 \quad \neg t \rightarrow \underline{\quad} \\
 \text{-----} (\rightarrow I) \\
 t \rightarrow \neg \neg t
 \end{array}$$

Hence  $\{\Omega t\} \vdash t \rightarrow \neg \neg t$ .

The proof of (T22) goes as follows:

From (T21), we have that  $\{\Omega t\} \vdash \neg t \rightarrow \neg \neg \neg t$ ,

if we assume  $\neg \neg \neg t$ , then if we assume  $t$ , we get  $\neg \neg t$  by (T21);

but  $\neg \neg \neg t$  is  $\neg \neg t \rightarrow \underline{\quad}$  and as we have  $\neg \neg t$ , hence  $\underline{\quad}$ .

Hence  $\neg t$  and so from the assumption  $\neg \neg \neg t$  we get  $\neg t$ .

As  $\Omega(\neg \neg \neg t)$ , then  $\{\Omega t\} \vdash \neg \neg \neg t \rightarrow \neg t$ .

Combining both results we get  $\{\Omega t\} \vdash \neg \neg \neg t \equiv \neg t$ .

The proof of (T23) goes as follows:

A.  $\Omega t, \Omega t'$  then  $\Omega(tVt')$  and so  $\Omega(\neg(tVt'))$ .

If we assume  $\neg(tVt')$  then if we assume  $t$  then  $tVt'$  and so  $\underline{\quad}$ .

hence  $\neg t$ ; i.e.  $\neg(tVt') \rightarrow \neg t$ . Also  $\neg(tVt') \rightarrow \neg t'$ .

Hence  $\neg(tVt') \rightarrow (\neg t \& \neg t')$ .

B.  $\Omega t, \Omega t'$  then  $\Omega(\neg t)$  and  $\Omega(\neg t')$ ; hence  $\Omega(\neg t \& \neg t')$ .

If we assume  $\neg t \& \neg t'$  then  $\neg t$  and  $\neg t'$ ;

and if we assume  $tVt'$ , we get from the assumption  $t, \perp$  by  $\neg t$  and  $(\rightarrow E)$ .

Also from the assumption  $t'$  we get  $\perp$  by  $\neg t'$  and  $(\rightarrow E)$ ;

and from the assumption  $tVt'$  we get  $\perp$  if we already assume  $\neg t \& \neg t'$ .

Hence the assumption  $\neg t \& \neg t'$  implies  $\neg(tVt')$ .

A and B imply  $\{\Omega t, \Omega t'\} \vdash \neg(tVt') \equiv (\neg t \& \neg t')$ .

The proof of (T24) goes as follows:

$\{\Omega t, \Omega t'\} \vdash (t \rightarrow t') \& (t' \rightarrow t)$ .

Now  $\Omega t, \Omega t'$  then  $\Omega(\neg t)$  and  $\Omega \neg t'$ .

If we assume  $\neg t$  then as  $\Omega t'$  and  $t' \rightarrow t$  we get  $t' \rightarrow \perp$ ; i.e.  $\neg t'$ .

Hence  $\{\Omega t, \Omega t'\} \vdash \neg t \rightarrow \neg t'$ .

The same method enables us to get  $\{\Omega t, \Omega t'\} \vdash \neg t' \rightarrow \neg t$ .

Hence  $\{\neg t, \neg t'\} \vdash \neg t \equiv \neg t'$ .

(T25) results from  $(\perp)$ .

(T26) results from  $(\exists I)$ .

For (T27) it is enough to say that if  $t$  is a theorem of  $\text{Axioms}(T_\Omega) \cup \Gamma$  then  $t$  is a theorem of  $\text{Axioms}(T_\Omega) \cup \Gamma \cup \Lambda$ .

The proof of (T28) follows the usual procedure.

(T29) holds because  $\Gamma \cup \Lambda \vdash t$  and  $\Gamma \cup \Lambda \vdash t'$ , and so  $\Gamma \cup \Lambda \vdash t \& t'$ .

(T30) is due to  $(\forall E)$ .

For (T31) it is enough to say that if  $\Omega a$  then  $\Omega(\neg a)$ , hence  $\Omega(a \& \neg a)$ . Assume  $a \& \neg a$ , then  $a$  and  $\neg a$ ; by  $(\rightarrow E)$  we get  $\perp$  and by  $(\perp)$  we get anything and in particular  $\neg(a \& \neg a)$ .

For (T32) the proof goes as follows:

$\Omega t'$  and from the assumption  $t$  and  $t'$  we get  $t$ ,

hence  $t' \rightarrow t$  from assumption  $t$ , but  $\Omega t$  and so

$$\{\Omega t, \Omega t'\} \vdash t \rightarrow (t' \rightarrow t).$$

The proof of (T33) goes as follows:

We have  $\Omega(t \rightarrow t')$  as  $\Omega t$  and  $\Omega t'$ .

We also have  $\Omega(t \rightarrow \neg t')$  as  $\Omega t$  and  $\Omega(\neg t')$ .

If we assume  $t$ , we get  $\neg t'$  and  $t'$ , hence  $\perp$  and so  $\neg t$ .

Therefore  $(t \rightarrow \neg t') \rightarrow \neg t$  from assumption  $t \rightarrow t'$  and so

$$(t \rightarrow t') \rightarrow ((t \rightarrow \neg t') \rightarrow \neg t) \text{ from assumption } \Omega t, \Omega t' \text{ and so (T33).}$$

The proof of (T34) goes as follows:

If  $\Omega t, \Omega t'$  then  $\Omega(t \vee t')$ .

If we assume  $t$  then  $t \vee t'$  by (VI) and so  $t \rightarrow (t \vee t')$  from assumption  $\Omega t, \Omega t'$ .

This means (T34) is a theorem.

The same proof can be followed for (T35).

(T36) and (T38) are obvious and (T37) is done by induction on the way expressions are constructed.  $\square$

Our theory is intuitionistic; if we add either of  $t \vee \neg t$  or  $\neg \neg t \rightarrow t$ , then we obtain a classical theory; this is why the axiom

$$\frac{\frac{\Omega t \quad \perp}{(\perp)'} \quad \{\neg t\}}{t}$$

leads to a classical theory and this is why we ruled it out.

### B.3. The semantics of $T_\Omega$

We define a model to be a pair  $M = \langle F, C \rangle$  where  $F$  is a Frege structure and  $C: \text{CON} \rightarrow \cup F_n$ , such that for each constant  $c_i$ ,  $C(c_i) \in F_0$ .

The semantics with respect to an assignment function  $g: \text{VAR} \rightarrow F_0$  is as follows:

$$(1) \llbracket [x] \rrbracket_g = g(x)$$

- (2)  $[[c]]_g = C(c)$
- (3)  $[[\lambda x.t]]_g = \lambda a. [[t]]_{g[a/x]}$
- (4)  $[[t=t']]_g = ([[t]]_g = [[t']]_g)$
- (5)  $[[t \vee t']]_g = ([[t]]_g \vee [[t']]_g)$
- (6)  $[[t \& t']]_g = ([[t]]_g \& [[t']]_g)$
- (7)  $[[t \rightarrow t']]_g = ([[t]]_g \rightarrow [[t']]_g)$
- (8)  $[[\exists t]]_g = \exists a. [[t]]_{g[a/x]}$
- (9)  $[[\forall t]]_g = \forall a. [[t]]_{g[a/x]}$
- (10)  $[[\Omega t]]_g = \Omega[[t]]_g$
- (11)  $[[app(t,t')]]_g = \mathbf{app} ([[t]]_g, [[t']]_g)$

*Theorem* : We can prove from the above that if we assume  $C$  to be the constant function which maps each constant into itself, then for each expression  $A$  built in the usual way and open in  $x, y_1, \dots, y_n$ ,

$$[[\lambda x.A[x, y_1, \dots, y_n]]]_g = \lambda a. A(a, g(y_1), \dots, g(y_n)),$$

where  $A$  is built exactly like  $A$  except that functionals inside are replaced by the corresponding  $F$ - functionals of the Frege structure.

E.g.  $[[\lambda x.(x=y)]]_g = \lambda a.(a = b)$  where  $g(y) = b$ ;

also,  $[[\lambda x.app(y \& z, x)]]_g = \lambda a. \mathbf{app}(b \& c, a)$  where  $g(y) = b$  and  $g(z) = c$ .

*Theorem* : If we take a model  $\langle F, C \rangle$  as above, then it is a model of the theory  $T_\Omega$

Proof: Let  $g$  be an assignment of the variables into  $F_0$ , then:

(\*) For each term  $t$ ,  $[[t]]_{M,g}$  is in  $F_0$ .

The proof of (\*) is by induction on  $t$ :

It is trivial for variables or constants.

If  $t$  is  $\lambda x.t'$  then  $[[t]]_{M,g} = \lambda a[[t']]_{M,g[a/x]}$ . But by an easy induction on terms one can prove that  $\langle [[t']]_{M,g[a/x]} / a \rangle$  is in  $F_1$ , therefore,  $\lambda \langle [[t']]_{M,g[a/x]} / a \rangle = \lambda a[[t']]_{M,g[a/x]}$  is in  $F_0$ .

For the case where we have to apply semantic clauses (4)-(7) to  $t$ , the proof is trivial

as the logical constants are elements of the Frege structure  $\mathbf{F}$  and the proof is done by induction.

(8)-(9) are proved as for the case where  $t$  is  $\lambda x.t'$ .

(10) is trivial as  $\Omega$  is that  $\mathbf{F}$ - functional of the Frege structure which has the following logical schemata:

If  $a$  is an object then  $\Omega a$  is an object such that  $\Omega(a)$  is true iff  $a$  is a proposition.<sup>71</sup>

(11) is trivial as  $\mathbf{app}$  is an  $\mathbf{F}$ - functional of the Frege structure and the proof by induction on  $t$ .

What about the axioms and rules? It is tedious to check each of them one by one, but they all hold in the model. The proof is illustrated with the following:

(&I):

If  $[[t]]_{M,g}$  is in **TRUTH** and  $[[t']]_{M,g}$  is in **TRUTH**, then  $[[t\&t']]_{M,g}$  - which by definition is  $[[t]]_{M,g} \& [[t']]_{M,g}$  - is in **TRUTH**, due to the logical schema of  $\&$  in a Frege structure.

( $\Omega\&$ ):

If  $[[\Omega t]]_{M,g}$  is in **TRUTH** and  $[[\Omega t']]_{M,g}$  is in **TRUTH**  
then ( $[[t]]_{M,g}$  is a proposition) and ( $[[t']]_{M,g}$  is a proposition).  
So ( $[[t\&t']]_{M,g}$  is a proposition) from the logical schema of  $\&$ .  
Therefore  $[[\Omega(t\&t)]]_{M,g}$  is true.  $\square$

#### *B.4. Soundness and completeness*

After having made the reader follow the above proof, we now tell her that this was unnecessary as it is obvious that the theory we put forward is a theory of the Frege structure; consistency of this theory is assured from the model construction we gave in part A of this chapter. We shall not repeat the construction but the reader

<sup>71</sup> Note that  $\Omega$  does not internally define propositions; since it is not itself a propositional function.

should always remember that we construct our  $\Omega$ ,  $\forall$ ,  $\&$ , etc, so that they are independent and primitive. This means that we can never have  $a \& b = a \forall c$  and if  $a \& b = a' \& b'$  then  $a = a'$  and  $b = b'$ .

The reader may still not be persuaded that we have proved the consistency of the language  $T_\Omega$ . The route usually followed to prove consistency is the construction of a model of the theory and we shall show how this can be done. We understand by a consistent set of  $T_\Omega$  a set  $\Gamma$  of closed expressions of  $T_\Omega$  such that no contradiction can be deduced from the assumption  $\Omega e$  for  $e$  in  $\Gamma$ ; otherwise  $\Gamma$  is inconsistent. A maximal consistent set  $\Gamma$  of closed expressions is such that if  $M$  is any closed expression not in  $\Gamma$  then  $\{\Gamma, M\}$  is inconsistent. We say that  $\Gamma$  is satisfiable with respect to a model  $M$  iff there is an assignment function  $g$  such that  $[[\Phi]]_{M,g}$  is true for every  $\Phi$  in  $\Gamma$ . An expression  $t$  is valid iff for every model  $F$  which is a Frege structure, for every assignment function  $g$ ,  $[[t]]_{F,g}$  is in **TRUTH**.

Now to prove consistency, we have to show that for any consistent set  $\Gamma$  of closed expressions from  $T_\Omega$  there exists a model (which is a Frege structure) for  $T_\Omega^*$  which satisfies the set of closed expressions  $\Gamma$ , where  $T_\Omega^*$  is the extension of  $T_\Omega$  obtained from  $T_\Omega$  by adding a countable number of constants.

It must be noted here that the proofs found in the next few pages are only outline of how things should be followed. For instance the construction of the model below is not followed to the end. The main idea there should be to assume the existence of a  $\lambda$ -model and to construct the various expressions (which contain the logical connectives) inductively in a way that the process remains monotonic and then to take the fixed point. However, as mentioned before we provide only the outline and how the construction of the model should be initiated.

*Theorem 1:* If  $\Lambda$  is any consistent set of closed expressions of  $T_\Omega$  then there exists a model (which is a Frege structure) with respect to which  $\Lambda$  is satisfiable.

*Proof:* This theorem should be proved in three parts;

Part 1: The construction of the extension  $T_{\Omega}^*$  of  $T_{\Omega}$  and then the construction of a maximal consistent set  $\Gamma$  of  $T_{\Omega}^*$  which contains  $\Lambda$ ,

Part 2: the construction of a model  $M$  of  $\Gamma$ ,

Part 3: the proof that  $M$  is a Frege structure.

In this theorem, we shall only outline the proof; mainly because parts 1 and 2 are standard proofs that can be found in any relevant book (e.g. [BE6]); also Mönningh has shown (in [MO1]) a theory not very distinct from the one offered here to be complete. Also, part 3 is long and tedious, hence it will not be worked out in full detail. I shall also make one simplification, that is reduce the work to objects and unary functions rather than work with  $n$ -ary functions for  $n \geq 1$ .

Part 1:  $T_{\Omega}^*$  is constructed by adding denumerably many primitive constants.  $\Lambda$  is a set of closed expressions from  $T_{\Omega}$ ; we can therefore extend  $\Lambda$  to a maximal set  $\Gamma$  of closed expressions in  $T_{\Omega}^*$ .  $\Gamma$  can be constructed so that it possesses the following properties:

- (i)  $\Gamma$  is a maximal consistent set.
- (ii)  $\Gamma$  contains  $\Lambda$ .
- (iii) We cannot have both  $\Phi$  and  $\neg \Phi$  being deducible from  $\Gamma$ .<sup>72</sup>
- (iv)  $\Gamma$  contains  $\forall x t[x]$  iff  $\Gamma$  contains  $t[c]$  for every new constant  $c$ ;
- (v)  $\Gamma$  contains  $\exists x t[x]$  iff  $\Gamma$  contains  $t[c]$  for some new constant  $c$ ;
- (vi)  $\Gamma$  contains  $\lambda x. t[x]$  iff there is a new constant  $c$  such that  $\Gamma$  contains  $c = \lambda x. t[x]$ .
- (vii) For any expression  $t$  containing an occurrence of a new constant  $c$ ,  $\exists c'$  such that  $t[c] = \text{app}(c', c)$ .  $\square$ <sup>73</sup>

Part 2: Now we have to search for a model which satisfies  $\Gamma$ ; this model will obviously satisfy  $\Lambda$  and we want it to be a Frege structure. In this part we construct such a model and in the next part we show it to be a Frege structure.

<sup>72</sup> Because, if we did, then we would have  $\perp$  deducible from  $\Gamma$  and so  $\Gamma$  inconsistent.

<sup>73</sup> Again the reader is referred to [BE6] and [MO1].



The model  $F$  is built out of  $\Gamma$  as follows:

We define an equivalence relation  $\cong$  on  $T_{\Omega}^*$  as follows:

$$t \cong t' \text{ iff } t=t' \text{ is deducible from } \Gamma;$$

$\cong$  is obviously an equivalence relation. We take  $F_0$  to be the set of equivalence classes of all the closed expressions of  $T_{\Omega}^*$  with respect to the relation  $\cong$  above.

What about propositions and truths? For any  $t$  of  $T_{\Omega}^*$ ,  $[t]$  (the equivalence class of  $t$  according to the equivalence relation above) is not a proposition if there exists a  $t'$  such that we have both  $t \cong t'$  and  $t \cong \neg t'$ . For example,  $\text{app}(\lambda x. \neg \text{app}(x,x), \lambda x. \neg \text{app}(x,x))$  is not a proposition. To see this take  $t$  and  $t'$  to be  $\neg \text{app}(\lambda x. \neg \text{app}(x,x), \lambda x. \neg \text{app}(x,x))$ , then we can deduce both  $t=t'$  and  $t = \neg t'$  from  $\Gamma$ ; therefore we have  $t \cong t'$  and  $t \cong \neg t'$ . Truths are all the equivalence classes of all the closed expressions  $t$  which are in  $\Gamma$  and for which  $\Omega t$  is deducible. E.g.  $t=t$  is true where  $t$  is a closed expression.

Now we construct  $\&, \vee, \dots$  as follows:

$$[a] \& [b] = [a\&b],$$

$$[a] \vee [b] = [a\vee b],$$

$$[a] \rightarrow [b] = [a \rightarrow b],$$

$$\forall x[a] = [\forall xa]$$

$$\exists x[a] = [\exists xa]$$

$$\lambda. [a] = [\lambda.a]$$

$$\text{app}([a],[b]) = [\text{app}(a,b)].$$

We take  $F_1$  to be the collection of all the expressions  $t$  open in one variable, chosen in the enumeration such that there is no expression  $t'$  which precedes  $t$  in the enumeration and where  $t'[c] = t[c]$  can be deduced in  $\Gamma$  for any new constant  $c$ .

Having led the reader up to here, she can now practise trying to build an function  $g$  in the usual way. She can also define the semantic function  $[[ \ ]]$  relative to the assignment function  $g$  in the usual way.<sup>74</sup> Now it is easily provable that this model

<sup>74</sup> For help she can refer to [BE6].

under the assignment function  $g$  is a model of  $T_{\Omega}^*$  which satisfies  $\Gamma$ .  $\square$

Part 3: Here, one has to show that the model constructed above is a Frege structure. I.e. one has to show that the structure is explicitly closed, that  $(\lambda, \text{app})$  form a  $\lambda$ -system and that the structure above is a logical system relative to the set of logical constants. As I said above, the proof is long and tedious and hence it will not be done here.  $\square$

Consequently any consistent set of closed expressions of  $T_{\Omega}$  is satisfiable by a model (which is a Frege structure).

Of course from consistency, one obtains soundness as seen below:

*Soundness theorem* : If an expression  $t$  of  $T_{\Omega}$  is deducible, then  $t$  is valid.

Proof:

This is done by checking that each axiom of  $T_{\Omega}$  is valid and that each rule preserves validity which is easy to do. Now, as  $t$  is deducible, this means that there is a finite proof tree with bottom element  $t$ , where at each stage the formula is either an axiom (which is valid) or obtained from previous formulae by application of the rules (which preserve validity).  $\square$

Now we come to complement our proof of soundness above by a proof of completeness. Some people might sacrifice completeness for other results as in the case of substitutional/referential interpretation. The substitutional interpretation of first order languages is not strongly complete but it is claimed that it provides some philosophical and ontological advantages over the referential interpretation [DU1].<sup>75</sup> All our interpretations are referential and so we have no reason yet to ignore completeness; moreover, it is argued that completeness is to be aimed at in our theories [CH3].

*Theorem 2*:  $\Gamma \vdash t$  iff  $\Gamma \cup \{\neg t\}$  is inconsistent.<sup>76</sup>

Proof:

---

<sup>75</sup> See Chapter 1 for the meaning of Substitutional/Referential interpretation.

<sup>76</sup> This theorem together with the consistency theorem gives completeness.

$\Gamma \vdash t$  then  $\Gamma \cup \{\neg t\} \vdash t$  from (T27).  $\Gamma \cup \{\neg t\} \vdash \neg t$  from (T17). Hence  $\Gamma \cup \{\neg t\} \vdash t \& \neg t$  from (T30). therefore  $\Gamma \cup \{\neg t\} \vdash \perp$  which means that  $\Gamma \cup \{\neg t\}$  is inconsistent.  $\square$

*The completeness theorem* : If an expression  $t$  of  $T_\Omega$  is valid, then  $t$  is deducible.

Proof: Any first order theory interpreted using standard semantics is complete. Higher order languages are incomplete under standard semantics but there is a procedure to make them complete; this procedure consists in using Henkin's techniques which we shall describe below. A second order language interpreted under standard semantics is incomplete but could be made complete á la Henkin; also the theory of types is incomplete under standard semantics, but could be made complete á la Henkin.<sup>77</sup>  $T_\Omega$  is complete with respect to the interpretation we gave; this is seen from both Theorem 1 and Theorem 2.

If  $t$  is valid then  $\{\neg t\}$  is inconsistent, for if it was consistent then there would exist a model  $M$  in which  $\neg t$  was satisfiable. As  $\neg t$  is satisfiable in  $M$  then  $[[\neg t]]_{M,g}$  is true with respect to an assignment function  $g$ . But  $[[t]]_{M,g}$  is true as  $t$  is valid. Hence contradiction and so  $\neg t$  is inconsistent. From Theorem 2, we get that  $\vdash t$ . Hence completeness.  $\square$

<sup>77</sup> Henkin in [HE2] proposes to make the simple type theory as formalised by Church [CH5], complete.

The crucial clause in the standard semantics to Church's type theory, is the following:  
 (1) For each type  $a, b$ ,  $D_a^D_b$  is the set of all functions from  $D_b$  to  $D_a$  and it is to provide denotations for wffs of type  $ab$ . In (1) we considered the set of all functions and this is the key to the problem. Looking back at Church's paper, he proved that his logic satisfies Peano's arithmetic. But according to the theorem which says that any two sets that satisfy Peano's arithmetic in their higher order must be isomorphic, Church's TRA must be isomorphic to  $\langle \mathbb{N}, 0, +, x, s \rangle$ . If so, they must then be enumerable. But Church could not enumerate his TRA and therefore his system is incomplete. The problem comes from the standard interpretation given above, together with letting the valuations of the functions of order  $n$  be the set of all functions of  $n$ -ordered tuples of individuals. The solution consists in reinterpreting sets, functions and the definition of validity. The values of functions should not be elements of the set of all the functions as above but only a certain class of functions. So validity of a sentence  $Q$  is obtained iff  $[[Q]]_{M,V}$  is true for all  $M, V$  where the domain of values of functions is reinterpreted as above. The question is of course whether such models exist. For, if one takes an arbitrary class above,  $[[f]]_{M,V}$  may not be in any of the domains. Henkin imposes the condition that this arbitrary choice of class must have denotations for all the functions. Under this interpretation, he proves that the theory is complete. I.e. " $Ao$  is valid iff  $\vdash Ao$  in the general sense". Note that the formulation of the simple theory of types is essentially the language used by Montague. However, Montague semanticists tend to interpret it using standard techniques, so it is not surprising that the semantics becomes incomplete. It could be made complete using Henkin's procedure which I just described, and for such a proof the reader is referred to [GA1].

Before closing this chapter it is interesting to see what happens to the compactness theorem; this is because our deduction theorem (T19) has a different form, that is: If  $t_1 \vdash t_2$  then  $\{\Omega t_1\} \vdash t_1 \rightarrow t_2$ .

*The compactness theorem* : A set S of sentences of  $T_\Omega$  is satisfiable iff every finite subset of S is satisfiable.

To prove this theorem, one needs prove one direction only. That is:

If every finite subset of S is satisfiable, then S is satisfiable.

If we do the standard proof here, then we get to a stage from which we cannot proceed, as is shown below.

If S is not satisfiable then S is inconsistent. Next we must prove that if S is inconsistent then there is a finite subset of S which is not satisfiable. As S is inconsistent, then anything is deducible from S; and in particular  $\perp$ . Therefore there must be a finite proof of  $\perp$  from  $t_1, \dots, t_n$  in S. Hence, by our version of the deduction theorem,  $\{\Omega t_1, \dots, \Omega t_n\} \vdash t_1 \rightarrow (\dots (t_n \rightarrow \perp) \dots)$ .

From here, we cannot deduce that  $t_1 \rightarrow (\dots (t_n \rightarrow \perp) \dots)$  is valid and hence that  $\{t_1, \dots, t_n\}$  is not satisfiable. This means that we cannot use the standard method to prove the compactness theorem. Another way of proving the theorem consists in applying the Tychonoff's theorem on product spaces. This is done topologically where the compactness property is used and where a space  $(X, T)$  is compact iff

$$(\forall \{O_i\}_{i \in I} \text{ of } T\text{-open sets such that } \cup O_i = X) [(\exists J \text{ finite subset of } I): X = \cup_{j \in J} O_j].$$

The two versions of the Tychonoff theorem are:

*Tychonoff product theorem* : If  $(X_i, O_i)_{i \in I}$  is a non-empty family of non-empty compact spaces, then the product space  $\Pi(X_i, O_i)$  is also compact.

*Tychonoff theorem* :  $\Pi X_i$  is the product space of a countable family of non-empty spaces  $(X_i, O_i)_{i \in I}$ . Then  $\Pi X_i$  is compact iff each  $(X_i, O_i)$  is compact.

It is interesting to prove compactness using this theorem, especially given that Frege

structures themselves can be built over a topological space; we shall leave this however to another occasion.  $\square$

CHAPTER 3. A THEORY OF PROPERTIES AND THEORIES OF TRUTH

In this chapter, we study some basic characteristics of the theory offered in Chapter 2. We will be concerned here with the logical operations on properties and the various theories of truths that could be obtained.

PART A. A THEORY OF PROPERTIES

First we start with some definitions. We introduce in our language  $T_{\Omega}$  the operator  $\Delta$ , understanding  $\Delta P$  to mean that  $P$  is a property.  $\Delta$  is defined as follows:

$$\Delta P =_{df} \forall x \Omega(\text{app}(P,x)).$$

That is, something is a property iff whenever it applies to an object, the result is a proposition; e.g.  $\lambda x. \neg(x=x)$ .

Note that any element of **SET** is a property, because if  $P$  is in **SET** then  $\text{app}(P,x)$  is a proposition for any  $x$  and therefore  $\Omega(\text{app}(P,x))$  is true. Hence  $\Delta P$  is true. Note that we introduce a  $\Delta$  in the model for  $\Delta$  in the formal language (in the same way as we did for  $\Omega$ ).<sup>78</sup>

A.1. Closure conditions on properties

Having defined properties in  $T_{\Omega}$  let us now look at their closure conditions to see whether they "behave properly". We can construct properties in the following way:

1.  $P \cup P' = \lambda x. (\text{app}(P,x) \vee \text{app}(P',x))$
2.  $P \cap P' = \lambda x. (\text{app}(P,x) \ \& \ \text{app}(P',x))$
3.  $P^c = \lambda x. \neg \text{app}(P,x)$
4.  $P \multimap P' = \lambda x. [\forall y (\text{app}(P,y) \rightarrow \text{app}(P',\text{app}(x,y)))]$

<sup>78</sup>  $\Delta$  does not internally define properties; this is because if  $P$  is not a property then  $\forall x \Omega(\text{app}(P,x))$  is not a proposition.

$$5. \Theta = \lambda x.(x=x)$$

$$6. \nabla = \lambda x.\neg(x=x)$$

(1) - (3) give us boolean combinations of properties, using join, meet and complement. (4) gives us function space, and (5), (6) give us the universal and the empty property, respectively. Now we can prove the following theorem:

*Theorem 1:*  $\Delta\Theta$ ,  $\Delta\nabla$  and if  $\Delta P$  and  $\Delta P'$  then  $\Delta(P \cup P')$ ,  $\Delta(P \cap P')$ ,  $\Delta P^c$ ,  $\Delta(P \rightarrow P')$ .

Proof:

We shall only prove that  $\Delta(P \rightarrow P')$ , as the others are similar.

We have to show that  $\forall x \Omega(\forall z(\text{app}(P,z) \rightarrow \text{app}(P',\text{app}(x,z))))$ .

If  $\Delta P'$  then  $\forall x \Omega(\text{app}(P',x))$ ,

hence  $\Omega(\text{app}(P',\text{app}(x,z)))$ ; but  $\Omega(\text{app}(P,z))$  as  $\Delta P$ .

Therefore  $\Omega(\text{app}(P,z) \rightarrow \text{app}(P',\text{app}(x,z)))$ .

Hence  $\Omega(\forall z(\text{app}(P,z) \rightarrow \text{app}(P',\text{app}(x,z))))$  and so

$\forall x \Omega(\forall z(\text{app}(P,z) \rightarrow \text{app}(P',\text{app}(x,z))))$ ;

hence  $\Delta(P \rightarrow P')$ .  $\square$

$\Theta$  stands for the universal property,  $\nabla$  stands for the empty property, and, of course, if  $P$ ,  $P'$  are properties, then so are their disjunction and conjunction. Also, the complement of any property is a property. This theorem implies that our domain of properties satisfies some important closure conditions; note especially that if  $P$  and  $P'$  are properties then  $P \rightarrow P'$  is also a property. It is well known that this would not hold if the notion of property was more comprehensive. For instance, for Turner in [TU9] (and Feferman in [FE2]), if  $P$ ,  $P'$  are properties (classes) then  $P \rightarrow P'$  is not necessarily a property (resp. class) because according to their approach, there were more properties (or classes) and propositions than there is according to the approach put forward here.

We understood  $\Delta P$  to be  $P$  is a property, however some people understand by this  $P$  is a class. Both interpretations work in parallel and to illustrate this point we

introduce  $\in$  by the following definition:

$a \in P =_{df} \text{app}(P,a)$ , and we understand by it:  $a$  belongs to the class  $P$ .

We can now prove the following:

$$(i) \frac{P=P' \quad \Delta P}{\Delta P'}$$

$$(ii) \frac{\Delta P}{\Omega(t \in P)}$$

$$(iii) \frac{\Omega(\text{app}(t,x))}{\Delta(\lambda x.\text{app}(t,x))} \quad \text{where no assumption depends on } x.$$

*Theorem 2:*

$$(i) a \in P \cap P' = ((a \in P) \& (a \in P'))$$

$$(ii) a \in P \cup P' = ((a \in P) \vee (a \in P'))$$

*Proof:*

We only prove (ii) here and leave (i) to the reader.

$$\begin{aligned} a \in P \cup P' &= \text{app}(P \cup P', a) = \text{app}(\lambda x. (\text{app}(P, x) \vee \text{app}(P', x)), a) \\ &= \text{app}(P, a) \vee \text{app}(P', a) \\ &= (a \in P) \vee (a \in P') \quad \square \end{aligned}$$

The above theorem shows that properties are closed under union and intersection.

That is, if John is either a lawyer or a doctor then either John is a lawyer or John is a doctor; also if John is a clever man, then he is both a man and clever. Our latter example creates a few problems, since it only works for a restricted set of adjectives.

For instance, from Mary is a beautiful dancer, one should not deduce that Mary is beautiful. How can one accommodate this in the above framework? The solution here would be in not identifying beautiful dancer with beautiful  $\cap$  dancer; yet still identifying clever man with clever  $\cap$  man.

Operators such as  $\cup$ ,  $\cap$  and  $^c$  are just ways of building new properties (or classes)



out of old ones. We have not yet defined any relations between properties (those relations may not be properties). Here we take the first step and define the following between properties:

$$P \subseteq P' = (\forall x)(\text{app}(P, x) \rightarrow \text{app}(P', x))$$

We understand  $P \subseteq P'$  to be P is a subproperty of P'.

We also define the following operation on properties, which we have not included with the previous ones because of its distinctive status - a status which will become clear below.

$$\Pi P = \lambda x. (\forall y (\text{app}(P, y) \rightarrow \text{app}(y, x)))$$

$\Pi P$  is the collection of subproperties of P. It is obvious that we should not deduce from  $\Delta P'$  and  $P \subseteq P'$  that  $\Delta P$ ; but if  $\Delta P$ , do we then have  $\Delta(\Pi P)$ ? Well, we need to add another condition, namely,  $\forall y (\text{app}(P, y) \rightarrow \Delta y)$ . With this new condition, things fit;

*Theorem 3:* If  $\Delta P$  and  $\forall y (\text{app}(P, y) \rightarrow \Delta y)$  then  $\Delta(\Pi P)$ .

Proof:

$\text{app}(\Pi P, x) = \forall y (\text{app}(P, y) \rightarrow \text{app}(y, x))$ ; and we can show by  $(\Omega \rightarrow)$  of Chapter 2,  $\Omega(\text{app}(P, y) \rightarrow \text{app}(y, x))$  if we can show both that

(i)  $\Omega(\text{app}(P, y))$  is deducible, and that

(ii)  $\Omega(\text{app}(y, x))$  is deducible from assumption that  $\text{app}(P, y)$ .

(i) follows from  $\Delta P$  and (ii) follows from  $\text{app}(P, y)$  and  $\forall y (\text{app}(P, y) \rightarrow \Delta y)$ .

Hence  $\Delta(\Pi P)$ .  $\square$

Now we start by listing some characteristics of our domain of properties. We have already seen two of these characteristics in Theorem 2, but we shall be working with  $\text{app}$  instead of  $\in$  from now on. With the following theorem we reveal more of our domain of properties,

*Theorem 4:*

(i)  $\text{app}(\lambda x. \Phi, t) \ \& \ \text{app}(\lambda x. \Psi, t) = \text{app}(\lambda x. (\Phi \ \& \ \Psi), t)$

(ii)  $\text{app}(\lambda x. \neg \Phi, t) = \neg \text{app}(\lambda x. \Phi, t)$

- (iii)  $\text{app}(\lambda x. \Phi, t) \vee \text{app}(\lambda x. \Psi, t) = \text{app}(\lambda x. (\Phi \vee \Psi), t)$
- (iv)  $\text{app}(P^c, t) = \neg \text{app}(P, t)$
- (v)  $\text{app}(P \cap P', t) = \text{app}(P, t) \& \text{app}(P', t)$
- (vi)  $\text{app}(P \cup P', t) = \text{app}(P, t) \vee \text{app}(P', t)$
- (vii)  $\text{app}((P^c)^c, t) = \neg \neg \text{app}(P, t)$
- (viii)  $\{\Delta P, \Delta P'\} \vdash \text{app}((P \cup P')^c, t) \equiv \text{app}(P^c, t) \& \text{app}(P'^c, t)$ <sup>79</sup>
- (ix)  $\{\Delta P, \Delta P'\} \vdash \text{app}(P^c \cup P'^c, t) = \text{app}(P^c, t) \vee \text{app}(P'^c, t)$
- (x)  $\{\Delta P, \Delta P'\} \vdash \text{app}(P^c \cap P'^c, t) \equiv \text{app}((P \cup P')^c, t)$ <sup>80</sup>
- (xi)  $\{\text{app}(P, t)\} \vdash \text{app}((P^c)^c, t)$ <sup>81</sup>
- (xii) If  $\Omega t$  then  $\forall y \text{app}(\lambda x. t, t') \rightarrow \text{app}(\lambda x. \forall y t, t')$
- (xiii) If  $\Omega t$  then  $\exists y \text{app}(\lambda x. t, t') \rightarrow \text{app}(\lambda x. \exists y t, t')$

Proof: We only prove (x) as (i)-(vii) are similar cases of  $\beta$ -conversion, (viii) comes from (x) and (v), (ix) is a particular case of (vi) and (xi) comes from (vii) and the fact that from  $a$  we deduce  $\neg \neg a$ . Also, (xii) and (xiii) are easy to prove.

$$\begin{aligned} \text{app}(P^c \cap P'^c, t) &= \text{app}(\lambda x. (\text{app}(P^c, x) \& \text{app}(P'^c, x)), t) \\ &= \text{app}(P^c, t) \& \text{app}(P'^c, t) \\ &= \neg \text{app}(P, t) \& \neg \text{app}(P', t) \\ &\equiv \neg (\text{app}(P, t) \vee \text{app}(P', t)) \end{aligned}$$
<sup>82</sup>

$$\begin{aligned} \text{and } \text{app}((P \cup P')^c, t) &= \text{app}(\lambda x. \neg \text{app}(P \cup P', x), t) \\ &= \neg \text{app}(P \cup P', t) \\ &= \neg (\text{app}(P, t) \vee \text{app}(P', t)) \end{aligned}$$

Hence  $\text{app}(P^c \cap P'^c, t) \equiv \text{app}((P \cup P')^c, t)$ .  $\square$

Now we come to the predication relation; if we allow app to define the predication relation then we will face some problems related to intensionality. The problem will

<sup>79</sup> But not necessarily  $\{\Delta P, \Delta P'\} \vdash \text{app}((P \cap P')^c, t) \equiv \text{app}(P^c, t) \vee \text{app}(P'^c, t)$

<sup>80</sup> But not necessarily:  $\text{app}(P^c \cup P'^c, t) = \text{app}((P \cap P')^c, t)$

<sup>81</sup> Not necessarily  $\text{app}((P^c)^c, t) \vdash \text{app}(P, t)$ .

<sup>82</sup> Due to (T23) of Chapter 2 and the fact that  $P$  and  $P'$  are properties.

be illustrated in Chapter 5. But here I shall try to accommodate Aczel's solution to the intensionality problem within our framework: we know that app is really functional application as we have  $\text{app}(\lambda f, a) = f(a)$ . So we need a distinct predication relation which I introduce as follows:  $\text{pred} : F_0 \times F_0 \dashrightarrow F_0$ , such that pred satisfies the following axioms:

$$(P1) \frac{\text{pred}(a,b) \quad \text{app}(a,b) \quad \Omega(\text{app}(a,b)) \quad \Omega(\text{pred}(a,b))}{\text{app}(a,b) \quad \text{pred}(a,b) \quad \Omega(\text{pred}(a,b)) \quad \Omega(\text{app}(a,b))}$$

$$(P2) \forall x(\text{pred}(P,x) = \text{pred}(Q,x)) \rightarrow P = Q$$

$$(P3) \text{pred}(P,a) = \text{pred}(Q,b) \rightarrow (P = Q \ \& \ a = b).$$

Now of course we have to make sure that pred belongs to the Frege structure; how can pred be built such that this holds? pred is built like any primitive independent F- functional and  $\Phi_{\text{pred}}, \Psi_{\text{pred}}$  are defined as follows:

$$\Phi_{\text{pred}}(\chi_0, x, y) \text{ is: } \text{app}(x, y) \text{ is in } \chi_{0p}$$

$$\Psi_{\text{pred}}(\chi_0, x, y) \text{ is: } \text{app}(x, y) \text{ is in } \chi_{1p}$$

The logical schema of pred is:

If a, b are objects where app (a,b) is a proposition then pred (a,b) is in **PROP** and pred (a,b) is in **TRUTH** iff app (a,b) is in **TRUTH**.

Now it is obvious that pred as defined here enables (P1), (P2) and (P3) to hold. (P1) and (P2) hold because of the characteristics of app; (P3) holds because pred is built like any other logical constant and it is primitive independent.

After introducing pred, we have to extend our terms to embody the additional condition: If t, t' are terms then pred(t,t') is a term. Now  $T_\Omega$  is extended to  $T_{\Omega_P}$  where the terms are those of  $T_\Omega$  together with the terms obtained from the new condition; axioms are those of  $T_\Omega$  together with (P1)-(P3). For the remainder of this chapter, we assume that we are working inside  $T_{\Omega_P}$ .

In any theory of predication we would like to have that (PRED) below is valid:

$$(PRED) (\exists z)(\forall x)(\forall y)(\{\Omega(\text{app}(x,y))\} \vdash \text{app}_2(z,x,y) \equiv \text{app}(x,y)).^{83}$$

*Theorem 5:* (PRED) is valid.

Proof:

This is seen by taking  $[[z]]$  to be  $\lambda_0^2(\text{pred})$ , which is an object of the Frege structure.  $\square^{84}$

The following shows that the application of a property to an object is equivalent in truth value to the predication of that property of the object. This does not however say anything about equality.

*Theorem 6:* If  $\Delta P$  then  $\forall x(\text{app}(P,x) \equiv \text{pred}(P,x))$ .

Proof:

We have to show that

$$\forall x(\text{app}(P,x) \rightarrow \text{pred}(P,x)) \text{ and } \forall x(\text{pred}(P,x) \rightarrow \text{app}(P,x)).$$

$$\Delta P \implies \text{app}(P,x);$$

by (P1), if we assume  $\text{app}(P,x)$  then we get  $\text{pred}(P,x)$ ;

hence  $\text{app}(P,x) \rightarrow \text{pred}(P,x)$ .

If  $\Omega(\text{app}(P,x))$  then  $\Omega(\text{pred}(P,x))$  by (P1);

again by (P1),  $\text{pred}(P,x) \rightarrow \text{app}(P,x)$ .

Hence  $\text{app}(P,x) \equiv \text{pred}(P,x)$ .

This is for all  $x$  and so  $\forall x(\text{app}(P,x) \equiv \text{pred}(P,x))$ .  $\square$

Now it is interesting to see what would happen to the closure of our properties if we understand the predication relation to be given in terms of pred and not app.

We start from our definition of  $\Delta$  above. We see that it does not make any difference if we replace app by pred. I.e.  $\Delta^*P =_{df} \forall x \Omega(\text{pred}(P,x))$  does not give anything new; this is because pred and app are equivalent when they result in propositions.

Suppose, however, that we introduce a relation  $\epsilon^*$  such that  $a \epsilon^* P =_{df} \text{pred}(P,a)$ .

What would happen to theorems 1-4 if we replace app by pred and  $\epsilon$  by  $\epsilon^*$ ? For

<sup>83</sup>  $\text{app}_2$  is to be understood as binary application. It applies the first argument to the pair consisting of the second and third arguments. In particular,  $\text{app}_2(\lambda f,a,b) = f(a,b)$ .

<sup>84</sup> It is also obvious that as the axiom (PRED) does not have any reference to pred then the validity of the axiom can be proven without the functional pred.

theorems 1 and 3, nothing new results, since if  $\Delta P$  then  $\text{pred}(P,x) \equiv \text{app}(P,x)$  for any  $x$ .

In Theorems 2 and 4, let us replace any occurrences of  $\in$  by  $\in^*$ ,  $=$  by  $\equiv$  and  $\text{app}$  by  $\text{pred}$ . We combine the theorems that work for pred in one theorem, Theorem 7, and we add the condition that  $\Delta P$  and  $\Delta P'$ :

*Theorem 7:* If  $\Delta P, \Delta P'$  then the following holds,

- (i)  $\text{pred}(P, t) \ \& \ \text{pred}(P', t) \equiv \text{pred}(P \cap P', t)$
- (ii)  $\text{pred}(P^c \cup P'^c, t) \equiv \text{pred}(P^c, t) \vee \text{pred}(P'^c, t)$
- (iii)  $\text{pred}(P^c \cap P'^c, t) \equiv \text{pred}((P \cup P')^c, t)$ <sup>85</sup>
- (iv)  $\text{pred}(P, t) \rightarrow \text{pred}((P^c)^c, t)$ <sup>86</sup>
- (v)  $\text{pred}(P^c, t) \equiv \neg \text{pred}(P, t)$

Proof:

(i)

If  $\Delta P, \Delta P'$  then  $\Delta(P \cap P')$ .

Therefore  $\Omega(\text{pred}(P \cap P', t)), \Omega(\text{pred}(P, t))$  and  $\Omega(\text{pred}(P', t))$ .

But  $\text{pred}(P \cap P', t) = \text{pred}(\lambda x. (\text{app}(P,x) \ \& \ \text{app}(P',x)), t)$

$$\equiv \text{app}(P, t) \ \& \ \text{app}(P', t), \text{ as } \Delta(P \cap P').$$

Since  $\text{pred}(P, t) \equiv \text{app}(P, t)$  and  $\text{pred}(P', t) \equiv \text{app}(P', t)$  then

$$\text{pred}(P, t) \ \& \ \text{pred}(P', t) \equiv \text{app}(P, t) \ \& \ \text{app}(P', t). \text{ Hence (i) is a theorem.}$$

(ii)

$$\Delta P \implies \Delta P^c \implies \text{pred}(P^c, t) \equiv \text{app}(P^c, t).$$

$$\Delta P' \implies \Delta P'^c \implies \text{pred}(P'^c, t) \equiv \text{app}(P'^c, t).$$

$$\Delta P^c \text{ and } \Delta P'^c \implies \Delta P^c \cup P'^c \implies$$

$$\text{pred}(P^c \cup P'^c, t) \equiv \text{app}(P^c \cup P'^c, t).$$

<sup>85</sup> Not necessarily  $\text{pred}(P^c \cup P'^c, t) \equiv \text{pred}((P \cap P')^c, t)$ , as we have:  $\{\Omega t, \Omega t'\} \vdash \neg(t \vee t') \equiv \neg t \ \& \ \neg t'$  but not:  $\{\Omega t, \Omega t'\} \vdash \neg(t \ \& \ t') \equiv \neg t \vee \neg t'$ .

<sup>86</sup> But not necessarily:  $\text{pred}((P^c)^c, t) \rightarrow \text{pred}(P, t)$ ; this will only be the case if  $\Delta P$  where  $\Delta P$  will be defined below.

But by Theorem 4 (vi),  $\text{app}(P^c \cup P'^c, t) = \text{app}(P^c, t) \vee \text{app}(P'^c, t)$ ,

$$\begin{aligned} \text{hence } \text{pred}(P^c \cup P'^c, t) &\equiv \text{app}(P^c, t) \vee \text{app}(P'^c, t) \\ &\equiv \text{pred}(P^c, t) \vee \text{pred}(P'^c, t). \end{aligned}$$

(iii)

$$\Delta P \implies \Delta P^c$$

$$\Delta P' \implies \Delta P'^c$$

$$\Delta P^c \text{ and } \Delta P'^c \implies \Delta(P^c \cap P'^c) \implies$$

$$\text{pred}(P^c \cap P'^c, t) \equiv \text{app}(P^c \cap P'^c, t).$$

$$\Delta P \text{ and } \Delta P' \implies \Delta(P \cup P') \implies \Delta((P \cup P')^c) \implies$$

$$\text{pred}((P \cup P')^c, t) \equiv \text{app}((P \cup P')^c, t).$$

But by Theorem 4, (V),  $\text{app}(P^c \cap P'^c, t) = \text{app}(P^c, t) \& \text{app}(P'^c, t)$

and by Theorem 4, (Viii),  $\text{app}((P \cup P')^c, t) \equiv \text{app}(P^c, t) \& \text{app}(P'^c, t)$ .

Hence  $\text{app}(P^c \cap P'^c, t) \equiv \text{app}((P \cup P')^c, t)$

and so  $\text{pred}(P^c \cap P'^c, t) \equiv \text{pred}((P \cup P')^c, t)$ .

(iv)

$$\Delta P \implies \Omega(\text{pred}(P, t))$$

$$\Delta P \implies \Delta P^c \implies \Delta(P^c)^c.$$

But by Theorem 4, (Vii),  $\text{app}((P^c)^c, t) = \neg \neg \text{app}(P, t)$

$$\{\text{pred}(P, t)\}$$

$$\text{app}(P, t)$$

$$\neg \neg \text{app}(P, t)$$

$$\text{app}((P^c)^c, t)$$

$$\Omega(\text{pred}(P, t)) \quad \text{pred}((P^c)^c, t)$$

-----

$$\text{pred}(P, t) \rightarrow \text{pred}((P^c)^c, t)$$

(v)

$\text{pred}(P^c, t) \equiv \text{app}(P^c, t)$  when  $\Delta P$ .

$$\text{app}(P^c, t) = \neg \text{app}(P, t);$$

$$\text{hence } \text{pred}(P^c, t) \equiv \neg \text{app}(P, t).$$

But  $\text{app}(P, t) \equiv \text{pred}(P, t)$ ; hence by (T24),  $\neg \text{app}(P, t) \equiv \neg \text{pred}(P, t)$ .

Therefore,  $\text{pred}(P^c, t) \equiv \neg \text{pred}(P, t)$ .  $\square$

If  $\Delta P$  and  $\Delta P'$  are not assumed then the version of Theorem 7 is as follows:

*Theorem 8:* The following holds in  $T_{\Omega P'}$

$$(i) \{ \text{pred}(P \cap P', t) \} \vdash \text{pred}(P, t) \ \& \ \text{pred}(P', t)$$

$$(ii) \{ \text{pred}(P, t) \ \& \ \text{pred}(P', t) \} \vdash \text{pred}(P \cap P', t)$$

$$(iii) \{ \text{pred}(P^c, t) \} \vdash \neg \text{pred}(P, t)$$

$$(iv) \{ \neg \text{pred}(P, t) \} \vdash \text{pred}(P^c, t).$$

Proof:

(i)

If we assume  $\text{pred}(P \cap P', t)$  then  $\Omega(\text{pred}(P \cap P', t))$ ,

hence  $\Omega(\text{app}(P \cap P', t))$  and so  $\Omega(\text{app}(P, t))$  and  $\Omega(\text{app}(P', t))$ .

This means that  $\Omega(\text{pred}(P, t))$  and  $\Omega(\text{pred}(P', t))$ .

But  $\text{app}(P, t) \equiv \text{pred}(P, t)$ ,  $\text{app}(P', t) \equiv \text{pred}(P', t)$ ,

$$\text{app}(P \cap P', t) \equiv \text{pred}(P \cap P', t) \text{ and } \text{app}(P \cap P', t) = (\text{app}(P, t) \ \& \ \text{app}(P', t)).$$

Hence  $\text{pred}(P \cap P', t) \equiv (\text{pred}(P, t) \ \& \ \text{pred}(P', t))$ .

Therefore the assumption  $\text{pred}(P \cap P', t)$  implies  $\text{pred}(P, t) \ \& \ \text{pred}(P', t)$ ;

i.e.  $\text{pred}(P \cap P', t) \vdash \text{pred}(P, t) \ \& \ \text{pred}(P', t)$ .

Now (ii), (iii) and (iv) are easy.  $\square$

### A.2. Decidable properties

Now, even if  $\Delta P$ , we still do not have that  $\text{pred}(P, c) \vee \neg \text{pred}(P, c)$ ; we therefore define a property to be decidable as follows:

$$DP =_{df} \forall x (\text{pred}(P, x) \vee \neg \text{pred}(P, x))$$

E.g.  $D\Theta$ ; this is because  $\forall x (\text{pred}(\Theta, x) \vee \neg \text{pred}(\Theta, x))$  is true as it is equivalent (in

terms of  $\equiv$ ) to  $\forall x((x=x) \vee \neg(x=x))$ . We know that  $x=x$  is always true, therefore  $(x=x) \vee \neg(x=x)$  is always true and so  $\forall x((x=x) \vee \neg(x=x))$  is true.

For  $\nabla$  we know that  $\text{pred}(\nabla x) \equiv \neg(x=x)$  and so for any  $x$ ,  $\neg \text{pred}(\nabla x) \equiv (x=x)$  which is true; therefore,  $\forall x(\text{pred}(\nabla x) \vee \neg \text{pred}(\nabla x))$  is true and so  $D\nabla$

As an example of an undecidable property, take:  $P_a = \lambda x.(x=a)$ ;  $P_a$  is undecidable for take  $\text{pred}(P_a, x) \equiv (x=a)$  and  $\neg \text{pred}(P_a, x) \equiv \neg(x=a)$ . Therefore,  $\text{pred}(P_a, x) \vee \neg \text{pred}(P_a, x) \equiv (x=a) \vee \neg(x=a)$  which we do not have a proof for and so we do not have that  $P_a$  is decidable.<sup>87</sup>

*Theorem 9:* Let  $P$  be a property such that  $DP$ . Then for any  $t$ ,

$$\text{pred}(P, t) \equiv \neg \neg \text{pred}(P, t).$$

Proof:

( $\Rightarrow$ ) We always have  $\text{pred}(P, t) \rightarrow \neg \neg \text{pred}(P, t)$  for any property  $P$ .

( $\Leftarrow$ )

(1)  $\Omega(\neg \neg \text{pred}(P, t))$  because  $\Omega(\text{pred}(P, t))$ .

(2)  $\text{pred}(P, t) \vee \neg \text{pred}(P, t)$  because  $DP$ .

$$\{\neg \neg \text{pred}(P, t)\}$$

$$\{\text{pred}(P, t)\} \quad \{\neg \text{pred}(P, t)\}$$

$\perp$

$$\Omega(\neg \neg \text{pred}(P, t)) \quad \text{pred}(P, t) \vee \neg \text{pred}(P, t) \quad \text{pred}(P, t) \quad \text{pred}(P, t)$$

(3) -----

$$\neg \neg \text{pred}(P, t) \rightarrow \text{pred}(P, t)$$

Therefore the theorem.  $\square$

The above theorem shows that the domain of decidable properties obeys classical logic; the following theorem shows that this domain is closed under  $\cup$ ,  $\cap$  and  $^c$ .

*Theorem 10:* If  $DP$  and  $DP'$  then  $D(P \cup P')$ ,  $DP^c$ ,  $D(P \cap P')$ .

<sup>87</sup> This is mainly because equality is not decidable in the  $\lambda$ -calculus; and we are using an intuitionistic theory.



Proof:

For  $D(P \cup P')$ , it is enough to say that:

$$\begin{aligned} & \text{pred}(P \cup P', x) \vee \neg \text{pred}(P \cup P', x) \\ & \equiv \text{pred}(P, x) \vee \text{pred}(P', x) \vee \neg (\text{pred}(P, x) \vee \text{pred}(P', x)) \\ & \equiv \text{pred}(P, x) \vee \text{pred}(P', x) \vee (\neg \text{pred}(P, x) \ \& \ \neg \text{pred}(P', x)) \\ & \equiv (\text{pred}(P, x) \vee \text{pred}(P', x) \vee \neg \text{pred}(P, x)) \ \& \ (\text{pred}(P, x) \vee \text{pred}(P', x) \vee \neg \text{pred}(P', x)) \end{aligned}$$

For  $P^c$ , we use the above theorem.

The case of  $P \cap P'$  is done by saying that

$$\begin{array}{l} a \vee \neg a, b \vee \neg b \\ \hline (a \ \& \ b) \vee \neg (a \ \& \ b) \end{array} \quad \square$$

The following theorem pushes negation inside pred in the definition of DP and shows that for any object we cannot predicate both a property and its complement to that object.

*Theorem 11:*

- (i) For any P such that  $\Delta P$ ,  $DP \equiv \forall x(\text{pred}(P, x) \vee \text{pred}(P^c, x))$
- (ii)  $\forall x$ , if  $\Delta x$  then  $[\forall y[\neg[\text{pred}(x, y) \ \& \ \text{pred}(x^c, y)]]]$

Proof:

- (i) If  $\Delta P$  then  $\text{pred}(P^c, x) \equiv \text{app}(P^c, x)$  and  $\text{pred}(P, x) \equiv \text{app}(P, x)$ ;  
but  $\text{app}(P^c, x) = \neg \text{app}(P, x)$  and  $\text{app}(P, x) \equiv \text{pred}(P, x)$ ,  
hence  $\text{pred}(P^c, x) \equiv \neg \text{pred}(P, x)$ .

Therefore  $\forall x(\text{pred}(P, x) \vee \neg \text{pred}(P, x)) \equiv \forall x(\text{pred}(P, x) \vee \text{pred}(P^c, x))$ ;  
and so  $DP \equiv \forall x(\text{pred}(P, x) \vee \text{pred}(P^c, x))$ .

- (ii) If  $\Delta x$  then  $\text{pred}(x^c, y) \equiv \neg \text{pred}(x, y)$ , from above.

But  $\neg(\text{pred}(x, y) \ \& \ \text{pred}(x^c, y)) \equiv \neg(\text{pred}(x, y) \ \& \ \neg \text{pred}(x, y))$  and  
we always have  $\neg(\text{pred}(x, y) \ \& \ \neg \text{pred}(x, y))$ .  $\square$

Now before we move to our next step, we need to lay out some theorems of the theory, the first of which is concerned with the conjunction of complements of

properties.

*Theorem 12:* For any properties P and P', if DP and DP' then we can derive the following in  $T_{\Omega_P}$ :

- (i)  $\text{app}(P^c \cup P'^c, t) \equiv \text{app}((P \cap P')^c, t)$
- (ii)  $\text{pred}(P^c \cup P'^c, t) \equiv \text{pred}((P \cap P')^c, t)$
- (iii)  $\text{app}((P^c)^c, t) \rightarrow \text{app}(P, t)$
- (iv)  $\text{pred}((P^c)^c, t) \rightarrow \text{pred}(P, t)$

Proof:

(i) ( $\Rightarrow$ ) We have to show that

$$(\text{app}(P, t) \vee \neg \text{app}(P, t)) (\text{app}(P', t) \vee \neg \text{app}(P', t)) (\neg \text{app}(P, t) \vee \neg \text{app}(P', t))$$

-----

$$\neg(\text{app}(P, t) \& \text{app}(P', t))$$

In other words, we need to prove that

$$(a \vee \neg a) (b \vee \neg b) (\neg a \vee \neg b)$$

-----

$$\neg(a \& b)$$

This is done as follows:

$$(a \vee \neg a) (b \vee \neg b) (\neg a \vee \neg b)$$

$$\{a \& b\} (1)$$

$$a \quad b \quad \{\neg a\} \quad \{\neg b\}$$

$$\underline{\quad} \quad \underline{\quad}$$

----- (VE) and  $\neg a \vee \neg b$

$$\underline{\quad}$$

----- By discharging (1)

$$\neg(a \& b)$$

( $\Leftarrow$ ) is similar.

(ii) Now it is enough to say that

$$\text{pred}(P^c \cup P'^c, t) \equiv \text{app}(P^c \cup P'^c, t) \text{ and}$$

$$\text{pred}((P \cap P')^c, t) \equiv \text{app}((P \cap P')^c, t).$$

(iii) We proved in Theorem 9 that if  $P$  is a property such that  $DP$  then  $\text{pred}(P, t) \equiv \neg \neg \text{pred}(P, t)$ . We also proved in Theorem 7 that if  $P$  is a property then  $\text{pred}(P^c, t) \equiv \neg \text{pred}(P, t)$ . Hence as  $P^c$  is a property,  $\text{pred}((P^c)^c, t) \equiv \neg \text{pred}(P^c, t) \equiv \neg \neg \text{pred}(P, t)$ . Therefore  $\text{pred}((P^c)^c, t) \equiv \neg \neg \text{pred}(P, t)$ .

As  $\text{app}((P^c)^c, t) \equiv \text{pred}((P^c)^c, t)$  and

$$\text{app}(P, t) \equiv \text{pred}(P, t) \text{ then}$$

$$\text{app}((P^c)^c, t) \equiv \neg \neg \text{app}(P, t) \text{ and so}$$

$$\text{app}((P^c)^c, t) \rightarrow \text{app}(P, t).$$

(iv) is a consequence from the proof of (iii) above.  $\square$

Now we define  $\text{pr}\ddot{\text{e}}d(P, x) =_{\text{df}} \text{pred}(P^c, x)$ .

*Theorem 13:*

(i) If  $DP$  then we have  $\text{pred}(P, x) \vee \text{pr}\ddot{\text{e}}d(P, x)$  for any  $x$ .

(ii) If  $\Omega A$  then  $\text{pr}\ddot{\text{e}}d(\lambda x.A, t) \equiv \neg \text{pred}(\lambda x.A, t)$

(iii) For  $P$  a property,  $\forall x \neg (\text{pred}(P, x) \& \text{pr}\ddot{\text{e}}d(P, x))$ .<sup>88</sup>

*Proof:*

(i)  $DP =_{\text{df}} \forall x \text{pred}(P, x) \vee \neg \text{pred}(P, x)$ .

But  $\text{pred}(P^c, x) \equiv \neg \text{pred}(P, x)$  for any property  $P$ ,

hence if  $DP$  then  $\text{pred}(P, x) \vee \text{pred}(P^c, x)$  for any  $x$ .

(ii)  $\text{pr}\ddot{\text{e}}d(\lambda x.A, t) = \text{pred}((\lambda x.A)^c, t)$ .

If  $\Omega A$  then  $\Delta \lambda x.A$  and so

$$\text{pred}((\lambda x.A)^c, t) \equiv \neg \text{pred}(\lambda x.A, t).$$

$$\text{Hence } \text{pr}\ddot{\text{e}}d((\lambda x.A), t) \equiv \neg \text{pred}(\lambda x.A, t).$$

(iii) If  $\Delta P$  then  $\text{pred}(P^c, x) \equiv \neg \text{pred}(P, x)$ ;

as we have  $\forall x \neg (\text{pred}(P, x) \& \neg \text{pred}(P, x))$ ,

---

<sup>88</sup> We introduced  $\text{pr}\ddot{\text{e}}d$  for those who are interested in comparing the theory that is presented here with those theories presented elsewhere such as Feferman's and Turner's.

then  $\forall x \neg (\text{pred}(P, x) \ \& \ \check{\text{pred}}(P, x))$ .  $\square$

One of the basic characteristics of the theory of property offered here is the full (even though weak) comprehension principle. This principle says that:

(CP) For  $f$  a propositional function, we have:

$$\text{app}(\lambda x.f(x), t) \text{ is true iff } f(t) \text{ is true.}$$

As mentioned in our discussion in Chapter 1, this full comprehension principle would lead to inconsistency if the notion of property was strengthened. This is why the work of Turner, Feferman and others focussed on restricting the principle. The fullness of the principle is very useful to have because, as we see, it relates the internal logic to the external one. E.g. because from  $\Phi[t] \rightarrow \Psi[t]$  and  $\Phi[t]$  we can derive  $\Psi[t]$  (if we are inside **PROP**) and because of (CP), we have that from  $\text{pred}(\lambda x.\Phi, t)$  and  $\text{pred}(\lambda x.\Phi \rightarrow \Psi, t)$  we can derive  $\text{pred}(\lambda x.\Psi, t)$ . Having (CP), one can make do with just the axioms of first order logic. As we have seen, things are not so easy for Turner; he had to provide another set of axioms for the internal logic after he got rid of (CP). According to our theory, if we are inside **PROP** then we could have the following;

$$\text{pred}(\lambda x.\Phi, t) \ \& \ \text{pred}(\lambda x.\Psi, t) \equiv \text{pred}(\lambda x.\Phi \ \& \ \Psi, t) \text{ and:}$$

$$\text{pred}(\lambda x.\neg \Phi, t) \equiv \neg \Phi(t).$$

Now if we want a more general version of the comprehension principle, we can introduce the following:

$$\text{For any } \Psi \text{ a propositional wff open in } x, (\exists P)(\forall t)(\text{pred}(P, t) \equiv \Psi[t/x]);$$

the above principle is valid. Also we of course have extensionality:

$$(\forall x)(\forall y)((\forall z)(\text{app}(x, z) = \text{app}(y, z)) \rightarrow x=y)$$

Now we come to our next step, that is, the truth theory of  $T_{\Omega}$ . Before presenting this however we summarise what we have done so far: we built a predication relation which is equivalent to  $\text{app}$  (inside **SET**). But outside **SET**, the

behaviour of pred is not known. That is we can not say what  $\text{pred}(P, t)$  is when  $P$  is not a property. Take  $P$  to be  $\lambda x. \neg \text{app}(x, x)$  for example, we find that  $\text{app}(P, P) = \neg \text{app}(P, P)$  by simple  $\beta$ -conversion, but there is no way for us to say what  $\text{pred}(P, P)$  is. This is because we did not allow  $\beta$ -conversion to take place inside pred as it did inside app. This is acceptable if we follow the view that properties are elements of **SET** and that predication is restricted to those elements. One might question here why it is that we introduce pred where we had app. It was not only because pred solved the problem of Rajneeshee and Fondalee, but also because pred is a primitive independent **F**- functional. Thus pred provides properties with the characteristics of high intensionality; that is, if  $\text{pred}(P, a) = \text{pred}(Q, b)$  then  $P = Q$  and  $a = b$ . We then introduced the notion of a decidable property; that is, a property which at any moment you can decide whether it holds of some object or not.

### *PART B. THEORIES OF TRUTH*

Now we come to build a truth operator **T** on our structure. The first question that will be asked here is: why introduce **T** when if you can deduce  $A$  then you know that  $A$  is true? But as Kripke (in [KR2]) and other authors on theories of truth have stressed, self-reference occurs frequently in natural language and often this self-reference involves the attribution of truth.

Within the present context of Frege structures the natural question to initiate our discussion is:

What is the logical schema governing the truth predicate?

We obviously cannot take:

(1) If  $A$  is an object then  $T(A)$  is a proposition such that  $T(A)$  is true iff  $A$  is true

as our logical schema for **T**. This is because the above internally defines **TRUTH**.

Also, if we take

(2) If  $A$  is an object then  $T(A)$  is an object such that  $T(A)$  is true iff  $A$  is true,

This will not help as it will give the identity function. What about if we take the following?

(3) If  $A$  is a proposition then  $T(A)$  is a proposition such that  $T(A)$  is true iff  $A$  is true.

Will this help? The two operators  $\Phi_T$  and  $\Psi_T$  should then be defined as:

$\Phi_T(X, x)$  is  $x$  is in  $X_p$

$\Psi_T(X, x)$  is  $x$  is in  $X_t$

All this gives the impression that the way to obtain a truth theory here is to introduce a new operator  $T$  and build  $T$  in the model. Assuming this to be the case, we then extend the formal language  $T_{\Omega_p}$  to  $T_{\Omega_{pt}}$  by adding a new constant  $T$  such that if  $t$  is a term then  $T(t)$  is a term. What axioms should  $T$  have? The obvious one is  $T(t) \equiv t$ ; this however would tie us to the condition that we are already in. That is, the introduction of  $T$  is unnecessary and it is enough to deduce  $t$ . However even though according to our account the liar sentence is not a proposition and therefore cannot be a truth, the liar sentence disjoined with its negation is true in any classical theory and therefore we feel the loss of something which is independent of whether the theory is classical or intuitionistic. Therefore, what one needs is something more than the set **TRUTH** that we have. Before we discuss how truth could be implemented in a Frege structure, we must stress that the present account is very elementary and is meant only to lay the foundation for the development of theories of truth within the context of Frege structures. I do not claim to have provided the final analysis. Second, I am trying to fit already existing theories of truth on the top of Frege structures and do not wish to defend the axiom system that I present below.<sup>89</sup> I only use it to highlight the problems that a theory of truth faces: namely,

---

<sup>89</sup> This axiom system is lifted from Turner's paper [TU9].

no theory can support the full Tarski biconditional schema:

$$T(t) \leftrightarrow t$$

We would like the predicate  $T$  to be such that from the assumption  $T(t)$  we deduce  $t$ ; but assuming  $t$  should not imply  $T(t)$ . This observation is captured in (TR1) below. Also we want to have only two levels,  $t$  and  $T(t)$  and to obtain that from the assumption  $T(t)$  we can deduce  $T(T(t))$ ; this results in (TR2) below. Again, if  $T(t)$  and  $T(t \rightarrow t')$  then  $T(t')$ , which results in (TR3). (TR3) is the first instance where the behaviour of  $T$  is dictated by the behaviour of the logical constants (here  $\rightarrow$ ). (TR4) is the case for  $\forall$  and  $\exists$  and (TR5) is the case for  $\&$ . If we let  $T(\neg t) \equiv \neg T(t)$  hold then we will get the paradox, yet (TR6) can capture negation without falling into inconsistency. Finally (TR7) could be understood as stating that if you have good grounds for asserting  $t$  (that is  $t$  is a theorem of  $T_{\Omega t}$ ) then  $T(t)$  must be a theorem of the theory of truth  $T_{\Omega t}$ .

$$(TR1) \frac{T(t)}{t}$$

$$(TR2) \frac{T(t)}{T(T(t))}$$

$$(TR3) \frac{T(t) \& T(t \rightarrow t')}{T(t')}$$

$$(TR4) \frac{\forall x T(t)}{T(\forall x t)} \qquad \frac{\exists x T(t)}{T(\exists x t)}$$

$$(TR5) \frac{T(t \& t')}{T(t)} \qquad \frac{T(t \& t')}{T(t')}$$

$$(TR6) T(\neg T(t)) \equiv T(T(\neg t))$$

$$(TR7) \text{ If } \vdash t \text{ then } T_{\Omega t} \vdash T(t)$$

*B.1. Various truth theories*

If we are inside **PROP**, no gain is obtained as we get not only (TR1)-(TR7) - and in particular (TR1) - but also the converse of (TR1): i.e. from  $t$  we deduce  $T(t)$ . This of course will imply that the whole introduction of  $T$  inside  $T_{\Omega p}$  is a trivial matter, for not even a better expressivity is gained - this is because we obtain that  $\vdash a$  is the same thing as  $\vdash T(a)$  and vice versa. But the matter will not be as trivial if we are interested in discussing the truth of many sentences, even though we know that they are compounded out of sentences which we deny to be propositions in our theory. Let us see how various theories of truth could be built on the top of a Frege structure.

*I. Frege structures with Tarski's notion of Truth:* This is essentially the notion of Truth that we have so far in our Frege structure. According to this notion you can only talk about the truth of a sentence in the metalanguage. Also, if you want to talk about the truth of sentences in the metalanguage, then you have to go into a higher metalanguage and so on... This of course leads to a hierarchy of metalanguages each of which talks about the truth of its predecessor. Here if we want to talk externally about truth in our structure then we introduce  $T$ , an external operator such that  $T(\#a)$  is<sup>90</sup> true iff  $a$  is in **TRUTH**.

This notion of truth is the weakest notion because we need self-reference and sentences that talk about their own truth or falsity. The first account to allow for this was Kripke's in [KR1]; we shall see whether this account can be used to extend Frege structures to where they can have a theory of truth equivalent to that of Kripke's - but first, let us provide a completely predicative theory of truth which is between Tarski's and Kripke's.

*II. A predicative theory of Truth:*

---

<sup>90</sup>  $\#a$  is the code number of  $a$  if we assume a certain numbering.



We start with  $T_0 = \text{TRUTH}$

$$P_0 = \text{PROP}$$

We take  $T_1 = T_0 \cup \{ T(x) : x \text{ is in } T_0 \}$

$$P_1 = P_0 \cup \{ T(x) : x \text{ is in } P_0 \}$$

$$T_2 = T_1 \cup \{ T(x) : x \text{ is in } T_1 \}$$

$$P_2 = P_1 \cup \{ T(x) : x \text{ is in } P_1 \}$$

:  
:  
:

$$T_n = T_{n-1} \cup \{ T(x) : x \text{ is in } T_{n-1} \}$$

$$P_n = P_{n-1} \cup \{ T(x) : x \text{ is in } P_{n-1} \}$$

:  
:

For transfinite level  $\omega$ , we take  $T_\omega = \bigcup_{n \in \omega} T_n$  and  $P_\omega = \bigcup_{n \in \omega} P_n$ ; then for  $\omega+1$ ,  $\omega+2, \dots$  we repeat the above process. For ordinal  $\alpha$  we have two cases:

either  $\alpha$  is a successor ordinal, i.e.  $\alpha = \beta+1$  and so

$$T_\alpha = T_\beta \cup \{ T(x) : x \text{ is in } T_\beta \}$$

$$P_\alpha = P_\beta \cup \{ T(x) : x \text{ is in } P_\beta \}$$

or  $\alpha$  is a limit ordinal then

$$T_\alpha = \bigcup_{\beta < \alpha} T_\beta$$

$$P_\alpha = \bigcup_{\beta < \alpha} P_\beta$$

Theorem: If  $\alpha < \beta$  then  $T_\alpha \subseteq T_\beta$  and  $P_\alpha \subseteq P_\beta$ .

Proof: Trivial.  $\square$

Now to use the fixed point theorem we define a monotonic operator  $J$  as follows:

$$J: \{(T_i, P_i)\}_i \rightarrow \{(T_i, P_i)\}_i$$

such that

$$J((T_i, P_i)) = (T_{i+1}, P_{i+1}) \text{ if } i \text{ is a successor ordinal}$$

$$= (\bigcup_{\alpha < i} T_\alpha, \bigcup_{\alpha < i} P_\alpha) \text{ if } i \text{ is a limit ordinal.}$$

We define an ordering relation  $\leq$  on  $\{(T_i, P_i)\}_i$  as follows:

$$(T_i, P_i) \leq (T_j, P_j) \text{ iff } T_i \subseteq T_j \text{ and } P_i \subseteq P_j$$

*Lemma:* J is monotonic; i.e. if  $(T_i, P_i) \leq (T_j, P_j)$  then  $J((T_i, P_i)) \leq J((T_j, P_j))$ .

Proof: Easy.  $\square$

Applying the fixed point theorem to the operator J above we obtain a pair  $(T^\infty, P^\infty) = J(T^\infty, P^\infty)$ .

The theory of Truth that we obtain is stronger than Tarski's and weaker than Kripke's. It is stronger than Tarski's in that we can talk about the truth of a sentence in the language itself so we can say: snow is white is true, it is true that snow is white is true. It is weaker than Kripke's in that we remain totally predicative and use only propositions as arguments of meaningful assertions of truths. For instance the liar sentence: This sentence is not true (which was taken by Kripke to be ungrounded), does not lead to any problem for us only because we refuse to assign it a truth value as we exclude it from **PROP**. By doing so, we will be subject to criticism: why should we rule out those sentences as propositions and lose the ability to discuss their truth value? This is not the right place to defend either taking those sentences to be propositions or refusing to do so. If one denies the status of these sentences as propositions then the above predicative theory of truth covers them; for even though it can continue ad infinitum discussing the truth or falsity of sentences that are already known to be true or false (e.g. The snow is white), it cannot however discuss the truth or falsity of sentences which are not propositions such as: This sentence is not true. For those who prefer to go beyond predicativity to self-referential sentences involving the concept of truth in a more embedded way rather than the above predicative simple way, we show them briefly the problem below. If we make the paradoxical sentences legitimate subjects of the truth predicate, we will face the possibility of becoming inconsistent. Of course T is a predicate which applies to any expression and so is defined for any expression, but the information it gives is only significant when this expression is itself a proposition. However, our notion of

proposition is rather restricted and defined by induction on expressions involving identity and the logical constants. To be able to build a theory of truth which gives insight into these paradoxical sentences, and which could be compared with other existing truth theories, we have to force the truth operator to tell us something about some sentences (especially the paradoxical ones) which are not propositions. To do so we start as follows:

We start first from a language  $L_0$  which has abstraction and application but no logic. We then construct logic on top in the same way that we did for a Frege structure, obtaining  $L_1 = L_0(\Omega_0, T_0)$ , where  $\Omega_0$  and  $T_0$  are those of the Frege structure.<sup>91</sup>

We now take  $\Omega_1 a = \Omega_0 a \vee \Omega_0(\Omega_0 a)$  and  $T_1 a = T_0 a \vee T_0(T_0 a)$ . Hence for any  $a$  so far, if we can prove  $T_0 a$  then  $a$  is true, if we can prove  $\Omega_1 a$  and  $\neg T_0 a$  then  $a$  is false.

We may assume we can continue in this way the construction of the various  $L_n$ ,  $\Omega_n$ ,  $T_n$ , and take the fixed point. However, take the Russell sentence  $a$ , then

$$\Omega_0 a = u \text{ and } T_0 a = u. \text{ However, } a \in \text{PROP}_1 \text{ and } a \in \text{TRUTH}_1, \Omega_1 a = 1 \text{ and } T_1 a = 1.$$

So it seems that we are getting the Russell sentence to be true in every stage after stage 1. This is something we wouldn't want to have and the above construction is not acceptable. It seems hence that the Kripke construction is not straightforward.

There is however another account which uses the Gupta/Herzberger construction of a theory of truth (in [GU1] and [HE4]), where the limit is obtained at a stabilisation ordinal rather than at a fixed point. We shall follow this account next and construct on the top of a Frege structure a theory of truth which is equivalent to the Gupta/Herzberger theory of truth.

*III. The Gupta/Herzberger notion of Truth:* By constructing a truth operator as above, we obtain expressivity; the sets **PROP** and **TRUTH** are extended to a stage where they contain statements about the truth of already existing statements. Our use above of monotonicity to build the truth theory was paralleled by the way all the logical

---

<sup>91</sup> Note that here we use propositions and truths whereas Kripke uses the extensions and antiextensions. These two approaches however are equivalent.

constants were built inside Frege structures. Let us provide another theory of truth where the process depends on revision rather than monotonicity; that is where the concept of truth is being revised at each step.

Above, when building a weaker theory of truth than that of Kripke, we kept things predicative so that at each stage  $\alpha$  such that  $\alpha = \beta+1$ , we have

$$T_{\alpha} = T_{\beta} \cup \{ T(x) : x \text{ is in } T_{\beta} \}. \text{ Hence } T_{\beta} \subseteq T_{\alpha}. \square$$

Next, when we tried to construct a Kripke truth theory, we counted on monotonicity. Here, we shall lose both predicativity and monotonicity in favour of the following revision process. The crucial point is that we do not follow the Kripke construction, and hence our external logic does not depend on Kleene's connectives but on the fully classical ones.

We start as above with a Frege structure **F** which has **TRUTH** and **PROP**. We let

$$\mathbf{TRUTH}_0 = \mathbf{TRUTH}$$

$$\mathbf{PROP}_0 = \mathbf{PROP}$$

$$\mathbf{TRUTH}_1 = \mathbf{J}(\mathbf{TRUTH}_0)^{92}$$

$$\mathbf{PROP}_1 = \mathbf{J}(\mathbf{PROP}_0)$$

Where:

$$T(a) \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0) \text{ iff } a \text{ is in } \mathbf{TRUTH}_0$$

$$a \ \& \ b \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0) \text{ iff } a \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0) \text{ and } b \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0)$$

$$a \ \vee \ b \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0) \text{ iff } a \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0) \text{ or } b \text{ is in } \mathbf{J}(\mathbf{TRUTH}_0)$$

$$\text{and so on ... } T(a) \text{ is in } \mathbf{J}(\mathbf{PROP}_0) \text{ iff } a \text{ is in } \mathbf{PROP}_0 \dots$$

Note<sup>93</sup> here that we shall differ from the above (Kripke's truth) in that if  $a$  is in  $\mathbf{TRUTH}_0$  then this does not imply that  $a$  is in  $\mathbf{J}(\mathbf{TRUTH}_0)$ . This means that  $\mathbf{J}$  is not monotonic and so we cannot apply the fixed point theorem to find the limit. There is

---

<sup>92</sup>  $\mathbf{J}$  is built below in a way that  $\mathbf{J}(S)$  contains  $T(a)$  if a certain condition holds, contains  $a \ \& \ b$  if another condition holds and so on.

<sup>93</sup> We can prove here that truth is closed under the logical connectives. E.g.  $T(a \ \& \ b)$  is in  $T_{\alpha}$  iff  $T(a)$  is in  $T_{\alpha}$  and  $T(b)$  is in  $T_{\alpha}$ . We shall leave this until the construction is given more formally.

however another theorem we can use to find the limit of such a construction; it is the following theorem of Herzberger (in [HE4]):

*Theorem:* For any model  $M$  there is an ordinal  $\sigma$  in the revision process based on  $M$ , such that  $M_\sigma$  is a stabilisation ordinal.

We still have to explain what a stabilisation ordinal is. A stabilisation ordinal  $\sigma$  is an ordinal such that any element is positively stable iff that element is in  $\text{TRUTH}_\sigma$ ; where an element  $t$  is positively stable iff  $(\exists \alpha)(\forall \beta \geq \alpha)(t \text{ is in } \text{TRUTH}_\beta)$ .

The above is what should be done if the only way to obtain a theory of truth in  $T_{\Omega_p}$  is by extending it to  $T_{\Omega_{pt}}$ . This however is not the case as we can introduce  $T$  in  $T_{\Omega_p}$  as follows:  $T(t) =_{df} \text{pred}(\lambda x.t, x)$ . This satisfies (TR1)-(TR7) if we are inside **PROP**, as the following theorem shows:

*Theorem:* If  $\Omega t$  and  $\Omega t'$  then (TR1)-(TR7) are theorems.

Proof: Easy.<sup>94</sup>  $\square$

---

<sup>94</sup> The only thing worth mentioning here is that  $\forall x \text{pred}(\lambda y.t, t') \rightarrow \text{pred}(\lambda y. \forall x t, t')$ .

CHAPTER 4. DETERMINERS AND QUANTIFIERS IN A FREGE STRUCTURE

We have in the previous chapter offered a few theorems about closures of properties and classes and discussed the possible truth theories that could be constructed within our framework. Here, we shall concentrate on both determiners and quantifiers and prove some relevant theorems about them.

One of Montague's main achievement in PTQ (see [TH2]) was to show how a logically adequate treatment of quantifier phrases could be systematically incorporated into a fragment of English. A further round of investigation into the characteristics of quantifiers and determiners was inaugurated by Barwise and Cooper's paper [BA3], which explored the way in which mathematical results in the area of generalised quantifiers could be applied to natural language. Since then, there has been a copious discussion of this topic - van Benthem provides a good summary of the main results (in [BE1] and [BE7]). In this chapter, we inquire how natural language quantifiers might be incorporated into the framework of Frege structures. Although we will have to leave a number of problems unsolved, we are nevertheless able to prove some relevant theorems.

In a Montague treatment, a sentence like Every boy runs receives a translation of the following form:

$$(1) \text{ (every'('boy'))(run').}$$

Within the framework of [BA3], we say that every'('boy') is a quantifier - interpreted as a set of sets (or, intensionally, as a second order property of properties), and that every' is a determiner - interpreted as a function from sets to quantifiers. An alternative analysis, adopted by van Benthem, treats determiners as relations between sets. For example, (2) denotes an instance of the schema (3):

$$(2) \text{ every'('boy', run')}$$

$$(3) \text{ D(A,B)}$$

As we will see later, (3) provides a convenient notation for expressing interesting characteristics of determiners.

Introducing  $D$  in (3) above prepares us for the important concept of a determiner relation, also known as the characteristic property of the determiner. A characteristic property of a determiner is that particular set theoretical relation which characterises this determiner set theoretically; e.g. for every', it is  $\subseteq$  and for a' it is  $\cap^1$ . We shall see below what  $\subseteq$  and  $\cap^1$  are.

#### PART A. TWO EXAMPLES OF DETERMINERS

We start first by defining the two determiners every' and a' in our framework. Let

$$\text{every}' =_{df} \lambda x. \lambda y. \forall z (\text{app}(x, z) \rightarrow \text{app}(y, z))$$

$$a' =_{df} \lambda x. \lambda y. \exists z (\text{app}(x, z) \ \& \ \text{app}(y, z))$$

The meanings of every', a' are not classes but we can prove some important theorems about them. We need however to introduce the characteristic properties of these determiners. We have also to show that these characteristic properties (or for that matter the determiners themselves) behave properly; that is when we combine things together in the right way we get a proposition. This is shown to be the case in the following few definitions, lemmas and theorems. The characteristic property of every', namely  $\subseteq$  has already been defined as follows:

If  $P_1, P_2$  are properties,

$$P_1 \subseteq P_2 =_{df} \forall x (\text{app}(P_1, x) \rightarrow \text{app}(P_2, x))$$

*Lemma 1:*  $\subseteq$  is a transitive, reflexive relation on properties.

Proof: Obvious.  $\square$

Now what about symmetry or antisymmetry? The relation  $\subseteq$  cannot be antisymmetric (we do not want it to be). As far as symmetry is concerned, all we get is:

$$\text{If } P_1 \subseteq P_2 \text{ and } P_2 \subseteq P_1 \text{ then } \forall x (\text{app}(P_1, x) \equiv \text{app}(P_2, x))$$

We call this equisymmetry.

*Lemma 2:*  $\subseteq$  is equisymmetric on properties.

Proof: Easy.  $\square$

*Theorem 1:* If  $P_1$  and  $P_2$  are properties then

- (i)  $\text{app}_2(\text{every}', P_1, P_2) = P_1 \subseteq P_2$  and
- (ii)  $\Omega(\text{app}_2(\text{every}', P_1, P_2))$ .

Proof:

- (i)  $\text{app}_2(\text{every}', P_1, P_2) = \forall z (\text{app}(P_1, z) \rightarrow \text{app}(P_2, z)) = P_1 \subseteq P_2$ .
- (ii) If  $P_1$  and  $P_2$  are properties then  $\Omega(\text{app}(P_1, x))$  and  $\Omega(\text{app}(P_2, x))$ , hence  $\Omega(\text{app}(P_1, x) \rightarrow \text{app}(P_2, x))$  and so  $\Omega(\forall x (\text{app}(P_1, x) \rightarrow \text{app}(P_2, x)))$ .

Therefore  $\Omega(\text{app}_2(\text{every}', P_1, P_2))$ .  $\square$

We define  $P_1 \cap^1 P_2 =_{\text{df}} \exists z (\text{app}(P_1, z) \ \& \ \text{app}(P_2, z))$ .

It is obvious that  $P_1 \cap^1 P_2$  is a proposition when both  $P_1$  and  $P_2$  are properties.

Another concept that we introduce here is that of an empty property. We say that a property  $P$  is empty and write  $\emptyset P$  iff  $\forall z (\neg \text{app}(P, z))$ . E.g.  $\nabla$  is an empty property, as can be seen from the following proof;

$$\begin{array}{c}
 x=x \\
 \hline
 \neg \neg (x=x) \\
 \hline
 \forall x \neg \neg (x=x) \\
 \hline
 \forall x \neg \text{app}(\nabla x) \\
 \hline
 \emptyset P.
 \end{array}$$

*Theorem 2:* If  $P$ ,  $P_1$  and  $P_2$  are properties then the following holds:

- (i) If  $\neg \emptyset P$  then  $\neg \emptyset(P \cup P)$
- (ii) If  $\neg \emptyset(P_1 \cup P_2)$  then  $\neg \emptyset(P_2 \cup P_1)$

Proof:

- (i) is trivial to prove since when  $P$  is a property,  $\text{app}(P \cup P, z) \equiv \text{app}(P, z)$  and so  $\neg \emptyset P \equiv \neg \emptyset(P \cup P)$ .



(ii) is also trivial as when  $P_1, P_2$  are properties,

then  $\text{app}(P_1 \cup P_2, z) \equiv \text{app}(P_2 \cup P_1, z)$ .  $\square$

*Theorem 3:* If  $P_1$  and  $P_2$  are properties then

$\text{app}_2(a', P_1, P_2) = P_1 \cap^1 P_2$  and

$\Omega(\text{app}_2(a', P_1, P_2))$

*Proof:*

$\text{app}_2(a', P_1, P_2) = \} z (\text{app}(P_1, z) \& \text{app}(P_2, z))$ . By  $(\beta)$

$= P_1 \cap^1 P_2$ .

If  $\Delta P_1$  and  $\Delta P_2$  then  $\Omega(\text{app}(P_1, x))$  and  $\Omega(\text{app}(P_2, x))$ .

Hence  $\Omega(\text{app}(P_1, x) \& \text{app}(P_2, x))$  and so  $\Omega(\text{app}_2(a', P_1, P_2))$ .  $\square$

### PART B. NON INTERNAL DEFINABILITY

Outside<sup>95</sup> SET we cannot draw useful conclusions about every' because we cannot decide the propositionhood of an arbitrary formula in which  $\rightarrow$  is the main connective.<sup>96</sup> This is not a disadvantage as we only want every' to have meaning when we are inside SET. What we cannot do, however, is to define the type of every' or of determiners inside Frege structures. Suppose we have the following definitions:

$\text{Quant}(t) =_{\text{df}} \forall x (\Delta x \rightarrow \Omega(\text{app}(t, x)))$

$\text{Det}(t) =_{\text{df}} \forall x (\Delta x \rightarrow \text{Quant}(\text{app}(t, x)))$ <sup>97</sup>

$\text{Quant} =_{\text{df}} \langle [\forall x (\Delta x \rightarrow \Omega(\text{app}(t, x)))] / t \rangle$

$\text{Det} =_{\text{df}} \langle [\forall x (\Delta x \rightarrow \text{Quant}(\text{app}(t, x)))] / t \rangle$

Det and Quant do not internally define determiners and quantifiers because

<sup>95</sup> Here and thereafter, we shall consider Frege structures where  $\text{PROP} \cap \text{SET}$  is empty. In Chapter 5, we shall see how such Frege structures could be constructed.

<sup>96</sup> The reader is reminded again that  $a \dashv$  is a proposition in the case where  $a$  is a proposition and  $b$  is a proposition assuming  $a$  is true.

<sup>97</sup> Note that we could have defined it as:  $\text{Det}(t) = \forall xy ((\Delta x \& \Delta y) \rightarrow \Omega(\text{app}_2(t, x, y)))$  which is closer to van Benthem's approach in [BE1] and [BE7].

$\forall x (\Delta x \rightarrow \text{Quant}(\text{app}(t, x)))$  and

$\forall x (\Delta x \rightarrow \Omega(\text{app}(t, x)))$

are not propositions for any  $t$ . In fact even if  $t$  is a property, we still do not have a guarantee that  $\text{Det}(t)$  and  $\text{Quant}(t)$  are propositions.<sup>98</sup> We can explain the problem differently; assume we define new domains out of old ones in a Frege structure as follows:

$F_0$ , **Prop** and **SET** are three basic domains. For any two of these domains  $A$  and  $B$  we let

$A \rightarrow B = \{a \text{ in } A : \text{for every } x \text{ in } A, \text{app}(a, x) \text{ is in } B\}$

The type of quantifiers should be **QUANT** = **SET**  $\rightarrow$  **PROP**, and that of determiners is **DET** = **SET**  $\rightarrow$  (**SET**  $\rightarrow$  **PROP**) = **SET**  $\rightarrow$  **QUANT**. **QUANT** is a non-empty subset of **SET**, yet **DET** is empty. This is because if  $a$  is in **DET** then  $a$  is in **SET** and for every  $b$  in **SET**,  $\text{app}(a, b)$  is in **QUANT**. Since  $a$  is in **SET** then  $\text{app}(a, b)$  is in **PROP**. But as  $\text{app}(a, b)$  is in **QUANT** then  $\text{app}(a, b)$  is in **SET**. Hence  $\text{app}(a, b)$  is in **PROP**  $\cap$  **SET**, which is empty. Absurd.

This creates the first complication. **DET** should be constructed on the top of  $F_0$  even though terms, verbs, etc.. could be inside  $F_0$ . The second complication comes from the fact since **SET** is not internally definable, **QUANT** is also not internally definable, because **QUANT**  $\subseteq$  **SET**.<sup>99</sup>

All the above is not serious as there is no particular reason for wanting determiners and quantifiers to be internally definable.<sup>100</sup> As everything fits together properly, and we can prove many desirable features of our determiners, why insist on

<sup>98</sup> This is because  $\Delta x$  is not a proposition.

<sup>99</sup> Actually as it is here **QUANT** = **SET** can be easily proven. One might question the acceptability of this, yet I have nothing to say about it.

<sup>100</sup> Sets are not closed under function space: for take  $A$  and  $B$  to be sets, then if we define  $B^A$  to be  $\{f: f \subseteq A \times B \ \& \ (\forall x \in A)(\exists! y \in B)(\langle x, y \rangle \in f)\}$ , we then cannot show that  $B^A$  is a set. This can be seen as follows: If we spell out the definition of the function space we get:  $\{f: f \subseteq A \times B \ \& \ (\forall x \in A)(\exists y \in B)(\langle x, y \rangle \in f \ \& \ (\forall y' \in B)(\langle x, y' \rangle \in f \rightarrow y = y'))\}$ . Now if  $A$  and  $B$  are sets then to have  $B^A$  a set we must restrict  $f$  in  $A \times B$  to be a set. This is related to the problem that if we have that  $\Delta P$  and  $p \sqsubset P'$  in the sense that  $(\forall x)(\text{app}(P, x) \rightarrow \text{app}(P', x))$ , then we do not necessarily have that  $\Delta P'$  as it was shown in Chapter 3.

internal definability? The following lemma proves inside the theory that combining a determiner and a property results in a quantifier.

*Lemma 3:*  $\{\text{Det}(Q), \Delta P\} \vdash \text{Quant}(\text{app}(Q,P))$

Proof:

$$\begin{array}{c} \{\Delta P\} \\ \{\text{Det}(Q)\} \\ \forall x (\Delta x \rightarrow \text{Quant}(\text{app}(Q, x))) \quad \text{From DetQ} \\ \hline \Delta P \rightarrow \text{Quant}(\text{app}(Q,P)) \quad \text{By (VE)} \\ \hline \Delta P \quad \Delta P \rightarrow \text{Quant}(\text{app}(Q,P)) \\ \hline \text{Quant}(\text{app}(Q,P)). \quad \text{By (}\rightarrow\text{E)} \end{array}$$

Hence the lemma.  $\square$

PART C. THE DETERMINER "the"

The reader may now wonder why it is we only discussed every and a. This should not give the impression that the remaining determiners are definable in terms of the above two. In fact, we now come to the. This determiner might sound problematic at first, as it deals with definite descriptions and we have a problem talking about definite descriptions inside Frege structures. Not only should the be a functional which operates on<sup>101</sup>  $F_0$  (even though most categories take denotations inside  $F_0$ ); but also the has the problem that definite descriptions have in any Frege structure. every, a had denotations in  $DET = SET \rightarrow (SET \rightarrow PROP)$ ; the however should have a denotation in  $SSET \rightarrow (SET \rightarrow PROP) \subseteq DET$  where  $SSET$  is the collection of all singleton sets. Hence it appears that with the we will face more problems than with other determiners. This is because, not only is  $SET$  neither decidable nor internally definable, but also  $SSET$  is neither decidable nor internally definable.

Let us take the usual translation of the:

$$the' = \lambda u \lambda v [ \ ] y [ \forall x (app(u, x) \equiv (x=y)) \ \& \ app(v,y) ] ]$$

Let us define for any property  $P$ ,  $SN(P) =_{df} [ \ ] y [ \forall x (app(P, x) \equiv (x=y)) ]$ .  $SN$  is not a propositional function outside  $SET$  and so  $SN$  does not internally define singleton properties. Some properties are obviously singleton properties, e.g.  $\lambda x.(x=a)$ . For some others however we can only tell they are single if there is some information to the effect. We leave open the question of how this problem should be dealt with.

As mentioned before each determiner is associated with its characteristic property. For each determiner we introduce, we have to show that if we apply this determiner to two properties we get a proposition. As the following theorem shows, the' has this characteristic.

---

<sup>101</sup> Every determiner has this characteristic.

*Theorem 4:* If  $P_1$  and  $P_2$  are properties then

- (1)  $\Omega(\text{app}_2(\text{the}', P_1, P_2))$  and
- (2)  $\Omega(\text{app}_2(\text{the}'' , P_1, P_2))$ .

Proof:

$$(1) \text{app}_2(\text{the}', P_1, P_2) = \exists y [\forall x (\text{app}(P_1, x) \equiv (x=y)) \& \text{app}(P_2, y)].$$

If  $\Delta P_1$  and  $\Delta P_2$  then  $\Omega(\text{app}_2(\text{the}', P_1, P_2))$ .

$$(2) \text{app}(\text{the}'' , P_1, P_2) = \text{SN}(P_1) \& P_1 \subseteq P_2.$$

As  $\Delta P_1$  and  $\Delta P_2$  then  $\Omega(\text{SN}(P_1))$  and  $\Omega(P_1 \subseteq P_2)$  and hence  $\Omega(\text{app}_2(\text{the}'' , P_1, P_2))$ .  $\square^{102}$

#### *PART D. HOW TO SHOW SOMETHING IS A DETERMINER*

We shall need to define all remaining determiners that haven't yet been defined (e.g. few), as none is definable in terms of the others; but now the whole method for doing so should be obvious and the lack of internal definability no longer worrying.

Having determiners such as every', a' and the' is one thing; being able to deduce that every', a' and the' are determiners is something else. I.e. we introduced every', a', etc.. by equations but can we prove that  $\text{Det}(\text{every}')$ ,  $\text{Det}(\text{a}')$ , etc..? Take the formula for every':

$$\lambda x. \lambda y. \forall z [\text{app}(x, z) \rightarrow \text{app}(y, z)];$$

To show that  $\text{Det}(\text{every}')$  we have to show that

$$\forall x (\Delta x \rightarrow \forall y (\Delta y \rightarrow \Omega(\text{app}_2(\text{every}', x, y)))).$$

But to be able to show the implication we need to have  $\Omega(\Delta x)$ , and  $\Omega(\Delta y)$ , which we cannot assume. For this we need an extension for implication as follows:

We always have that if  $\{a\} \vdash b$  then  $\{\Omega a\} \vdash a \rightarrow b$  (our version of the deduction theorem). We need that if  $\{\Omega a\} \vdash b$  then  $\vdash \Omega a \rightarrow b$ . Can we assert this rule? That is:

$$(*) \text{ If } \{\Omega a\} \vdash b \text{ then } \vdash \Omega a \rightarrow b.$$

<sup>102</sup> However things are not as smooth as may seem. When we come to measure theory, some things may not be provable inside the theory and some extra devices may be needed. The following for instance, is not provable in our theory:

If  $P_1$  and  $P_2$  are properties,  $P_1 \subseteq P_2$  and  $\text{SN}(P_2)$  and  $\neg \text{OP}_1$  then  $\text{SN}(P_1)$ .

It may be claimed here that this rule leads to an inconsistency similar to Curry's paradox because if  $a$  is  $\lambda x(\Omega_{\text{app}}(x,x) \rightarrow \perp)$ , then  $a$  is a well-formed expression. However it is not the case that we will get Curry's paradox, for take the following chain of deductions:

$\text{app}(a,a) = \Omega_{\text{app}}(a,a) \rightarrow \perp$  by  $\beta$ -conversion

$\text{app}(a,a) \vdash \Omega_{\text{app}}(a,a) \rightarrow \perp$  from above

$\text{app}(a,a) \vdash \Omega_{\text{app}}(a,a)$  obvious

$\text{app}(a,a) \vdash \perp$  by MP

$\Omega_{\text{app}}(a,a) \vdash \text{app}(a,a) \rightarrow \perp$  by DT

But now applying (\*) we get:  $\vdash \Omega_{\text{app}}(a,a) \rightarrow (\text{app}(a,a) \rightarrow \perp)$

which is not contradictory.

Note that we should not always deduce from  $\{a\} \vdash b$  that  $\vdash a \rightarrow b$ ; because if we did then we get Curry's paradox as explained in the previous chapter. However, I am not sure whether the deduction from  $\{\Omega\} \vdash b$  to  $\vdash \Omega \rightarrow b$  is harmless and hence the following theorem that every', a' and the' are determiners can only hold if we conjecture that (\*) holds.

Theorem 5: Det(every'), Det(a'), Det(the'), if (\*) is consistent.

Proof:

For every': We have to prove that  $\forall x (\Delta x \rightarrow \forall y (\Delta y \rightarrow \Omega(\text{app}_2(\text{every}', x, y))))$ .

$\{\Delta x, \Delta y\} \vdash \Omega(\text{app}_2(\text{every}', x, y))$  according to Theorem 1, (ii) above.

$\{\Delta x\} \vdash \Omega(\text{app}(y, z)) \rightarrow \Omega(\text{app}_2(\text{every}', x, y))$  according to (\*).

From this we have:  $\{\Delta x\} \vdash \forall z [\Omega(\text{app}(y, z)) \rightarrow \Omega(\text{app}_2(\text{every}', x, y))]$

$\{\Delta x\} \vdash [\forall z \Omega(\text{app}(y, z))] \rightarrow \Omega(\text{app}_2(\text{every}', x, y))$

$\{\Delta x\} \vdash \Delta y \rightarrow \Omega(\text{app}_2(\text{every}', x, y))$

$\{\Delta x\} \vdash \forall y (\Delta y \rightarrow \Omega(\text{app}_2(\text{every}', x, y)))$ .

Repeating the same process, we get:

$\vdash \Delta x \rightarrow \forall y (\Delta y \rightarrow \Omega(\text{app}_2(\text{every}', x, y)))$

$\vdash \forall x (\Delta x \rightarrow \forall y (\Delta y \rightarrow \Omega(\text{app}_2(\text{every}', x, y))))$ .

The proof of a' and the' is similar to that of every'.  $\square$

From here we see that the theory can be strengthened in many ways: our framework is powerful yet flexible.

*PART E. CHARACTERISTICS OF DETERMINERS AND QUANTIFIERS*

Here we are concerned with some characteristics of determiners that could be proven in our theory. We start with the first theorem that asserts that the result of applying a quantifier to a property results in a proposition.

Theorem 6:  $\{\text{Quant}(Q), \Delta P\} \vdash \Omega(\text{app}(Q, P))$

Proof:

We have to prove that:  $\Omega(\text{app}(Q, P))$  from assumptions:  $\forall x (\Delta x \rightarrow \Omega(\text{app}(Q, x)))$  and  $\Delta P$ .

But  $\{\forall x (\Delta x \rightarrow \Omega(\text{app}(Q, x))), \Delta P\} \vdash \Delta P, \Delta P \rightarrow \Omega(\text{app}(Q, P))$

and  $\{\Delta P, \Delta P \rightarrow \Omega(\text{app}(Q, P))\} \vdash \Omega(\text{app}(Q, P))$

Hence the theorem.  $\square$

We still cannot prove:  $\{\text{Quant}(Q), \Delta P, \Delta P'\} \vdash (\text{app}(Q, P) \ \& \ P \subseteq P') \rightarrow \text{app}(Q, P')$ , but this is not worrying as it should not always hold. The following theorem is to show that the domain of quantifiers is closed under  $\cup$ ,  $\cap$  and  $^c$ .

*Theorem 7:*

(i)  $\{\text{Quant}(a), \text{Quant}(b)\} \vdash \text{Quant}(a \cap b)^{103}$

(ii)  $\{\text{Quant}(a), \text{Quant}(b)\} \vdash \text{Quant}(a \cup b)^{104}$

(iii)  $\{\text{Quant}(a)\} \vdash \text{Quant}(a^c)^{105}$

Proof: We illustrate only (i):

<sup>103</sup> E.g. Every man and some women.

<sup>104</sup> E.g. Every man or some women.

<sup>105</sup> E.g. Not every man.

$$\frac{\forall x (\Delta x \rightarrow \Omega(\text{app}(a,x))) \quad \forall x (\Delta x \rightarrow \Omega(\text{app}(b,x)))}{\forall x (\Delta x \rightarrow \Omega(\text{app}(a,x)) \& \Omega(\text{app}(b,x)))}$$

Hence  $\forall x (\Delta x \rightarrow \Omega(\text{app}(a,x) \& \text{app}(b,x)))$ .  $\square$

Also the following theorem is concerned with the closure on the domain of determiners. Now closure is in term of  $\cap_1, \cup_1, c^1$ .<sup>106</sup>

*Theorem 8:*

(i)  $\{\text{Det}(a), \text{Det}(b)\} \vdash \text{Det}(a \cap_1 b)$ <sup>107</sup>

(ii)  $\{\text{Det}(a), \text{Det}(b)\} \vdash \text{Det}(a \cup_1 b)$ <sup>108</sup>

(iii)  $\{\text{Det}(a)\} \vdash \text{Det}(a^{c^1})$ <sup>109</sup>

Where

$$a \cap_1 b = \lambda x. (\text{app}(a,x) \cap \text{app}(b,x))$$

$$a \cup_1 b = \lambda x. (\text{app}(a,x) \cup \text{app}(b,x))$$

$$a^{c^1} = \lambda x. (\text{app}(a,x))^c$$

Proof: We illustrate only (iii).

$$\text{Det}(a) = \forall x (\Delta x \rightarrow \text{Quant}(\text{app}(a,x)))$$

$$\equiv \forall x (\Delta x \rightarrow \text{Quant}(\text{app}(a,x))^c) \text{ By Theorem 7, (iii).}$$

Hence  $\text{Det}(a^{c^1})$ .  $\square$

We would be interested in proving something in general about these determiner relations. Let us consider monotonicity. We have two kinds of monotonicity: *upwards monotonicity* and *downwards monotonicity* (see [BE1], [BE7] and [BA3]). These are defined as follows, where C is a property of sets:

(upwards) If  $A \subseteq A'$  and  $C(A)$  then  $C(A')$

(downwards) If  $A' \subseteq A$  and  $C(A)$  then  $C(A')$

<sup>106</sup> We introduce the subscript '1' to make the distinction between the intersections considered.

<sup>107</sup> E.g. many and many

<sup>108</sup> E.g. Some or few.

<sup>109</sup> E.g. not all, not every.



As an example of an upwards monotone determiner, we take a'. a' is monotone in both arguments. E.g. a boy who sings walks entails a boy walks. Also a boy sings and dances entails a boy sings.

Hence to show that a' is upwards monotone in both arguments we need to show that

$$(i) \text{ app}_2(a', P_1, P_2) \ \& \ P_1 \sqsubseteq P'_1 \rightarrow \text{app}_2(a', P'_1, P_2) \text{ and}$$

$$(ii) \text{ app}_2(a', P_1, P_2) \ \& \ P_2 \sqsubseteq P'_2 \rightarrow \text{app}_2(a', P_1, P'_2).$$

Both (i) and (ii) can be shown as follows:

For (i) :

$$\begin{aligned} & \text{app}_2(a', P_1, P_2) \ \& \ P_1 \sqsubseteq P'_1 = \\ & \}z(\text{app}(P_1, z) \ \& \ \text{app}(P_2, z)) \ \& \ P_1 \sqsubseteq P'_1 \\ & \rightarrow \}z(\text{app}(P'_1, z) \ \& \ \text{app}(P_2, z)). \end{aligned}$$

The proof of (ii) is similar.

every' can also be shown monotone in the right argument but not in the left.<sup>110</sup>

We now consider another property of determiner relations; that is *conservativity*. We say that a property of sets C is conservative if

$$(CONS) \quad D(P_1, P_2) \equiv D(P_1, P_2 \cap P_1), \text{ where } D \text{ is the determiner relation of } C.$$

As an example, a' and every' are conservative. E.g. a boy walks entails a boy is both a boy and he walks. Also every man runs entails every man is both a man and he runs.

Now to show that a' and every' are conservative, we have to show that for any  $P_1$  and  $P_2$  properties,

$$(a'CONS) \quad P_1 \cap^1 P_2 \equiv P_1 \cap^1 (P_2 \cap P_1).$$

$$(every'CONS) \quad P_1 \sqsubseteq P_2 \equiv P_1 \sqsubseteq (P_2 \cap P_1).$$

This is shown by the following two theorems:

*Theorem 9:* If  $P_1$  and  $P_2$  are properties then  $P_1 \cap^1 P_2 \equiv P_1 \cap^1 (P_2 \cap P_1)$ .

Proof: The only thing worth mentioning here is that  $\text{app}(P_2 \cap P_1, z) = \text{app}(P_2, z) \ \&$

<sup>110</sup> For a clear discussion of such characteristics of determiners, the reader is referred to [BE7].

$\text{app}(P_1, z)$ .  $\square$

*Theorem 10:*  $P_1 \subseteq P_2 \equiv P_1 \subseteq (P_2 \cap P_1)$

Proof: Trivial.  $\square$

Of course we would like the conservativity condition to hold of any determiner we define and we would be happy if we could prove conservativity for determiner relations as a special type of their own. It is not obvious how to do so and we must be satisfied with proving properties about each determiner relation individually.

Now we can take the definition of properties of concepts which is given in [BE7; page 459] to be:

"determiners only dependent upon the intersection of their arguments; that is if  
 $C \cap D = A \cap B$  then  $D(C, D) \equiv D(A, B)$ ".

Now we can prove that  $\underline{a}'$  has such a property. This is because the determiner relation for  $\underline{a}'$  is  $\cap^1$  and we can prove that if  $C \cap D = A \cap B$  then  $C \cap^1 D \equiv A \cap^1 B$ .

Before closing this section, we give the following theorem which shows that every' is a transitive relation:

*Theorem 11:*  $\{\text{app}_2(\text{every}', P_1, P_2), \text{app}_2(\text{every}', P_2, P_3)\} \vdash \text{app}_2(\text{every}', P_1 \cap P_3)$

Proof:

$\text{app}_2(\text{every}', P_1, P_2) \ \& \ \text{app}_2(\text{every}', P_2, P_3) \equiv$   
 $\forall z (\text{app}(P_1, z) \rightarrow \text{app}(P_2, z)) \ \& \ \forall z (\text{app}(P_2, z) \rightarrow \text{app}(P_3, z)).$   
Hence  $\forall z (\text{app}(P_1, z) \rightarrow \text{app}(P_3, z)). \ \square$

Transitivity does not hold for  $\underline{a}'$ .

#### *PART F. NO LOSS OF QUANTIFICATION WITH FIRST ORDER THEORIES*

We have said that we do not need more than two levels of quantification, namely quantification over objects and quantification over predicates. An obvious

question, however, is whether we need these two levels of quantification. In this section we shall discuss this issue and show that the semantics that we have been following does not result in any loss of quantification. The crucial point here is the following:

if we had two levels of quantification, one over individual variables and the other over predicate variables then the universal quantifier clause is defined as:

$$(1) \llbracket [\forall X \Phi(X)] \rrbracket_g = \forall \langle \llbracket [\Phi(X)] \rrbracket_{g[\lambda a/X]} / a \rangle$$

However, if we were able in a Frege structure to have higher order quantifiers, then we would replace (1) by (2):

$$(2) \llbracket [\forall X \Phi(X)] \rrbracket_g = \forall \langle \llbracket [\Phi(X)] \rrbracket_{g[f/X]} / f \rangle$$

However, our inability to do so is not problematic, since  $A$  is isomorphic to  $F_1$ , where  $A$  is the collection of  $\lambda a$  such that  $a$  is in  $F_0$ .

Proof:

Take the identity function from  $A$  to  $F_1$ .

This function is injective, obviously.

It is surjective because for all  $f$  in  $F_1$ ,  $\lambda f$  is in  $A$  and  $\lambda f = f$ .  $\square$

Hence we are fine up to here. As we have shown, quantification over objects coincides with quantification over functions. What about if we considered

$$(3) \llbracket [\forall X \Phi(X)] \rrbracket_g = \forall \langle \llbracket [\Phi(X)] \rrbracket_{g[a/X]} / a \rangle$$

Do we get any loss of quantification? And if we do not, is this equivalent to (4) below?

$$(4) \forall \langle \llbracket [\Phi(X)] \rrbracket_{g[\lambda a/X]} / a \rangle$$

Regarding loss of quantification, we are fine as  $F_0$  is isomorphic to  $A$ .

Take  $|| : F_0 \dashrightarrow A$

$||$  is injective for if  $\lambda a = \lambda b$  then  $a=b$  by  $(\epsilon)$ .

$||$  is surjective, obvious.

About equivalence, if  $\text{app}(a,x) = \lambda a (x)$  then we do have equivalence. However,

always  $\text{app}(a,x) = |a| (x)$  and therefore the two definitions are equivalent.

In summary what this chapter shows is that our theory works well for both quantifiers and determiners; it is also simple, tidy and flexible. We have demonstrated that even though determiners and quantifiers are not internally definable, we can prove very useful things about them - things such as monotonicity, symmetry, and equisymmetry that are the main concerns of workers in this area. What we have not investigated is the cardinality characteristics of determiners and quantifiers. We will pursue such questions in future work.

CHAPTER 5. INTENSIONALITY AND EXTENSIONALITY USING A FREGE STRUCTURE

PART A. AN INTENSIONAL SOLUTION TO PROPOSITIONAL ATTITUDES

Problems of intensionality have been central to much research in natural language semantic, at least since the time of Frege. Kripke's possible world semantics for modal logic has been extremely influential, and plays a major role in Montague's extensive treatment of intensional constructions in his *Universal Grammar and Proper Treatment of Quantification* - see [TH2]. Although Montague made enormous progress in this area, the analysis of propositional attitudes has remained intractable, for reasons that we shall briefly review.

On a Possible worlds approach, intensions are functions from worlds to extensions. For example, the intension of a sentence is a function from worlds to truth values. Consequently, two intensions are identical if they yield the same value for each possible world. We call this *weak* intensionality. One notorious consequence of weak intensionality is that any two logically necessary propositions have the same intension, namely the function that yields True at each world. While this may be acceptable if we only restrict our semantics to alethic modalities, it leads to the well-known problem of *logical omniscience* (cf [HI3]) when one considers verbs of propositional attitude. That is, we have the result that if John believes that  $2+2 = 4$ , then this entails that John believes that  $p$ , where  $p$  is any truth of arithmetic. This consequence seems inevitable if we adopt weak intensionality together with the principles of compositionality and substitutivity of co-extensive expressions. Rather than abandoning either of the latter, we would prefer to solve the problem by using a stronger notion of intensionality. Before showing how Frege structures can help us, let us explain compositionality and substitutivity, and say why they create difficulties in weakly intensional frameworks.

*Compositionality* is the principle that says that the meaning of any complex expression

is a function of the meaning of its parts; for example, the meaning of John runs is a function of the meaning of John and of the meaning of runs and this function is specified in advance.

*Substitutivity* is the process that allows us to replace a by b in  $\Phi$  where  $a=b$ ; so that if the Morning star = the evening star and John dreams of the morning star then John dreams of the evening star.<sup>111</sup>

To illustrate a problem that having both substitutivity and compositionality creates in extensional/weak intensional logics, consider the following example:

Let  $\Phi$  be John is aware that the least integer greater than  $x$  is greater than 5000.

Let  $\Phi[x/t_1]$  be the result of replacing  $x$  in  $\Phi$  by  $t_1 =$  the sum of the first 100 positive integers.

(i.e.  $\Phi[x/t_1]$  is John is aware that the least integer greater than the sum of the first 100 positive integers is greater than 5000).

Let  $\Phi[x/t_2]$  be the result of replacing  $x$  in  $\Phi$  by  $t_2 = 5050$ .

Clearly  $t_1$  and  $t_2$  have the same extension, and therefore according to one version of substitutivity, they should be interchangeable *salva veritate*. However, since  $x$  occurs in an opaque context, we might adopt a stronger version of substitutivity according to which two terms are only interchangeable if they have the same intension. However, this will not help us here, since  $t_1$  and  $t_2$  have the same extension in every possible world, and thus have the same (weak) intension. Consequently, if we assume that  $\Phi[x/t_1]$  is a compositional function of the intensions of  $\Phi$  and  $t_1$  we are still forced to the conclusion that  $\Phi[x/t_1]$  has the same semantic value as  $\Phi[x/t_2]$ .

To illustrate with an easier example, take

(1a) All oculist are doctors

(1b) All eye-doctors are doctors.

---

<sup>111</sup> In this example, I used definite descriptions, which need a different treatment from proper names. In this chapter I rarely touch on proper names.

According to both extensional and weakly intensional frameworks, both (1a) and (1b) are true and hence John believes (1a) iff John believes (1b). This is again unacceptable. Extensional approaches cannot deal with belief sentences because equality of functions there coincides with co-extensionality. Montague's IL cannot deal with belief sentences because equality of functions in that approach coincides with co-extensionality in all possible worlds. Montague followed Frege in assigning to each sentence both an intension and an extension (Frege's terms were sense and reference). Before moving on to provide a better treatment for propositional attitudes, allow me to discuss Frege's sense and reference as we will be using these terms quite often.

According to Frege (see [FR3]), every expression has both a reference and a sense; the expression is said to designate its reference and to express its sense. The reference of a sentence is its truth value, while its sense is the thought which it expresses. Frege argued, however, that when expressions occur in certain contexts - for example, in indirect quotation - this view has to be modified: expressions do not have their customary reference, but have an *indirect* reference, which coincides with their sense. In particular, the reference of a subordinate clause, such as that John is nice, is not a truth value, but its customary sense, namely the thought that John is nice.<sup>112</sup> The question that is asked is: how are we to know when to use the sense or the reference of an expression?

The key to this problem comes from realizing that the reference of a sentence is not always a function of the reference of its parts. E.g. The morning star has the same reference as The evening star, but ( $\square$  The morning star = The evening star) is false because there is a possible state of affairs in which the evening star is not the same as

---

<sup>112</sup> The area of sense/reference, intension/extension is a rich one. We have only mentioned Frege (the father of the subject) and Montague. Other scholars provided interesting theories which deal with the subject. Church, for instance developed a whole axiomatic system (in CH7) in which  $\lambda$  was used to give the intension of a function; for both Montague and Church, intensions play the role of Frege's sense and extensions play the role of reference. The extension of a property like nice is a set of individuals (or a function from individuals to truth-values). The intension of nice, on the other hand, is a function from indices to extensions. For Montague, the indices range over  $W \times T$  (the cartesian product of the set of possible worlds and possible times). One must not also forget Carnap's work in the area - see [CA2].

the morning star. Frege did not abandon functionality for reference but held the view that the reference of  $\eta$  depends on the *syntactic context* in which it occurs. If context is ordinary then the reference of  $\eta$  is  $\text{ref}(\eta)$ , if oblique then it is  $\text{sen}(\eta)$ . For instance if we go back to our example of oculist and eye-doctor, Frege's solution consists in applying believe to the sense of (1a) and (1b) rather than to their reference. Even though  $\text{ref}(1a) = \text{ref}(1b)$ ,  $\text{sen}(1a)$  is not the same as  $\text{sen}(1b)$ . Hence  $\text{Bel}(j, \text{sen}(1a))$  does not necessarily imply  $\text{Bel}(j, \text{sen}(1b))$ .

Montague attempted to solve the problem along the lines of Frege, yet he built a weak notion of intensionality which implicated that two expressions have the same sense if they have the same reference in every possible world, and this led to difficulties mentioned above. Another problem that should be mentioned concerning intensionality in weakly intensional frameworks is the following which was given by Bealer and discussed by Aczel in [AC6]:

Rajneeshee =  $\lambda x. \text{follows}(x, \text{Rajneesh})$   
Fondalee =  $\lambda x. \text{follows}(\text{Jane Fonda}, x)$   
 $\text{app}(\text{Rajneeshee}, \text{Jane Fonda}) = \text{follows}(\text{Jane Fonda}, \text{Rajneesh})$   
 $\text{app}(\text{Fondalee}, \text{Rajneesh}) = \text{follows}(\text{Jane Fonda}, \text{Rajneesh})$   
Therefore  $\text{app}(\text{Rajneeshee}, \text{Jane Fonda}) = \text{app}(\text{Fondalee}, \text{Rajneesh})$

This conclusion might be questioned since someone could believe that Rajneeshee holds of Fonda, without believing that Fondalee holds of Rajneesh. What is the solution? Aczel's approach consists in taking properties to be propositional functions while not making the predication relation be functional application except when the property is basic (e.g. green). This really amounts to adopting Leibnitz's law in belief contexts while rejecting the following:

$B\Phi \ \& \ (\Phi \equiv \Psi) \rightarrow B\Psi$ ; where B is a belief operator.

Now the problem of intensionality mentioned above is solved if we replace app by pred where pred is the relation defined in Chapter 3. This is because:



$\text{pred}(\text{Rajneeshee, Jane Fonda}) \equiv \text{follows}(\text{Jane Fonda, Rajneeshee})$  and

$\text{pred}(\text{Fondalee, Rajneeshee}) \equiv \text{follows}(\text{Jane Fonda, Rajneeshee})$ .

Hence all we obtain is that  $\text{pred}(\text{Rajneeshee, Jane Fonda}) \equiv \text{pred}(\text{Fondalee, Rajneeshee})$  and believing one will not imply believing the other.

Before we introduce the problem of building intensionality out of extensionality, let us give an informal account of how problems like the first two examples above could be solved according to our approach. The point to our solution of propositional attitudes is that the meaning of each sentence is a strongly intensional proposition and that we look for the truth value only when we are interested in it. Two propositions might have the same truth value in every possible situation but still not be equal as objects and hence not be interchangeable in any belief context. We illustrate with our doctor example:

Under our interpretation, we take:  $\forall x(O(x) \rightarrow D(x))$  and  $\forall x(ED(x) \rightarrow D(x))$  to be the respective representations of the sentences (1a) and (1b). Note that those two representations are already propositions and that when we adjoin that to both sentences it will not add anything new.

Now the truth values of  $\forall x(O(x) \rightarrow D(x))$  and  $\forall x(ED(x) \rightarrow D(x))$  are the same in any possible situation, but as objects they are distinct. Therefore, to believe one proposition is not necessarily to believe the other.

Because of the strong intensionality in a Frege structure, we face a slight problem which, however, we can mend easily. We know that  $\Phi \& \Psi$  as a proposition is equivalent to  $\Psi \& \Phi$  but not equal to it. Therefore, if someone knows  $\Phi \& \Psi$ , it will not follow that he knows  $\Psi \& \Phi$ . This could be mended by postulating some axioms about the functions know and believe, for example:  $K(\Phi \& \Psi) \equiv K(\Psi \& \Phi)$ .

Since on our approach the meaning of any sentence  $\Phi$  is a strongly intensional proposition, if we want to find the truth value of  $\Phi$ , then we have to unpack the truth content of the proposition that it denotes. This chapter is intended to work out in

detail the intensionality/extensionality problem and to discuss if one can build from the strongly intensional structure that we have so far an extensional one which helps in finding the truth value of any sentence.

As we have now achieved a stronger notion of intensionality, we may ask what has happened to the possible worlds of Montague's approach? They can be reconstructed in many ways:

(a) As maximal sets of propositions.

(b) As a set of models

(c) By defining *Necessary*, and *Possible* in terms of the equality relation that already exists in the Frege structure, and studying the modal logic one obtains.

In Part D, we shall very briefly discuss how each of these approaches might be accommodated in our framework.

In Chapter 3, we defined the operator T and said that if we can deduce T(t) then we know that t is true. Also, we know that T satisfies the following:

$$T(A\&B) \equiv T(A)\&T(B)$$

$$T(A\vee B) \equiv T(A)\vee T(B)$$

$$T(\forall xA) \equiv \forall xT(A)$$

:

:

With these equivalences one would wonder if it is possible to construct an extensional structure out of the intensional one and to see the connection between the intensional truth operator and the extensional one. However the answer is no and this chapter is intended to explain this puzzle of intensionality; that is can one accommodate inside a highly intensional structure an extensional one? Before we start, let me briefly summarise the points that will be tackled:

We try to build an extensional structure out of an intensional one. This is done by defining an equivalence relation  $\simeq$  on the Frege structure together with both the sense

and the reference of any expression. This equivalence relation is the weakest that one requires to identify two extensionally equivalent propositions. It will be shown however that not even this weak notion of extensionality can be added to a Frege structure as the resulting extensional structure of propositions will either collapse in a trivial one element or is itself inconsistent. Before we explain this puzzle, we define the  $\simeq$ -relation so that the *reference* of any expression (object, function or proposition) is the  $\simeq$ -equivalence class of that expression. The *sense* of any equivalence class is a particular way of computing a representative of that class. The relation  $\simeq$  allows us to place the denotations of groundhog and woodchuck in the same equivalence class and so the reference of groundhog is equal to the reference of woodchuck, even though the senses are not the same. Application and abstraction are defined on the extensional structure, and it is shown that<sup>113</sup>  $\text{PROP}/\simeq$  - the collection of all the equivalence classes of elements of  $\text{PROP}$  - is a boolean algebra. It will also be shown that  $\text{PROP}/\simeq$  collapses into a trivial one element set. Of course the reference of any proposition is True if either that proposition or any one  $\simeq$ -equivalent to it is in **TRUTH**. However we shall show that this cannot be done without resulting in an inconsistency.

Let me emphasize that we are assuming the principle of extensionality on functions (we build the Frege structure by starting from a model which allows for this extensionality, e.g.  $E_\infty$  as opposed to  $P_\omega$ ); in fact in the theory  $T_\Omega$  we assumed the principle of extensionality in the following form:

$$(\epsilon) \quad \frac{(\text{app}(a,x) = \text{app}(b,x))}{a = b}$$

Assuming this extensionality would allow us to concentrate on  $F_0$  (the collection of objects) and to ignore the remaining  $F_1, F_2, \dots$ . This is due to the following lemma:

*Lemma 1:* In a Frege structure, if  $(\epsilon)$  is assumed then we always have  $a = \lambda |a|$ .

<sup>113</sup> Of course this is provided extensionalisation can be patched up. I.e. provided we do not face the problem will be explained below.

Proof:  $\forall x, \text{app}(\lambda a, x) = \langle \text{app}(a, x) / x \rangle(x) = \text{app}(a, x)$ . Using  $(\epsilon)$ , we get that  $\lambda a = a$ .  $\square$

### *PART B. AN EXTENSIONAL STRUCTURE*

It may be questioned here why we are going to construct an extensional structure when the main aim is to show that building extensionality on the top will result in inconsistency. This extensional structure however, is built just to illustrate the problem. It could be viewed hence as an example of the inconsistency of adding extensionality to a Frege structure.

We need to construct a Frege structure  $F = F_0, F_1, \dots, F_n, \dots$  built on the top of  $E_\infty$  and having certain properties; namely that  $\text{PROP} \cap \text{SET} = \emptyset$ . In  $E_\infty$  extensionality holds on functions, and to give the reader a clear idea of how our Frege structure can be constructed following the techniques explained in Chapter 2, basing things on top of  $E_\infty$  which is constructed in the appendix, we include the following summary.

(1) Build  $E_\infty$  following the steps of the appendix.

(2) Define  $\text{app}$  and  $\lambda$  on the top of  $E_\infty$  in the usual way.

Now steps (1)- (2) above still do not guarantee that  $\text{PROP} \cap \text{SET} = \emptyset$  because we still have not decided what  $\text{PROP}$  and  $\text{SET}$  should be built on using the logical constants. To guarantee that we have  $\text{PROP} \cap \text{SET} = \emptyset$ , we proceed as follows:

We take  $B_p = \{1, 0\} \subseteq F_0 = E_\infty$  and  $B_s = \{ \lambda_0^n f : f \text{ is in } E_\infty^n \rightarrow E_\infty \text{ and for all } x, f(x) \text{ is in } B_p \}$ .<sup>114</sup>

Obviously, from this construction,  $B_p \cap B_s = \emptyset$ . We then take our construction of the logical constants so that  $\text{PROP}$  is the smallest set containing  $B_p$  and closed under

1. The logical constants.
2. For all n-ary functions  $f$  such that for all  $\hat{o}$  in  $F_0^n$ ,  $f(\hat{o})$  is in  $\text{PROP}$ , the

---

<sup>114</sup> 1 and 0 are the elements of  $E_\infty$  as in the appendix and are distinct from 1 and 0 of  $\text{PROP}/\square$  below.

following are in **PROP**:  $\forall f, \lambda f$ .

We then take **SET** =  $\{\lambda_0^n f: f \text{ is any } n\text{-ary propositional function}\}$ .

Because  $B_p \cap B_s = \emptyset$ , then **PROP**  $\cap$  **SET** =  $\emptyset$ . Hence the following lemma.

*Lemma 2:* In a Frege structure built as above, **PROP**  $\cap$  **SET** =  $\emptyset$ .

The above lemma guarantees that no element is both the result of predication (of a propositional function to an object) and of  $\lambda$ -abstraction (of a propositional function).

Let us define an equivalence relation on **F** which associates with each object which is intensional another object (which is extensional), and which to each function also associates the extensional part of that function. We will need this equivalence relation to talk about the intension and extension of an expression. For instance, one feature of this equivalence relation should be:

$${}^u\alpha = {}^u\beta \text{ iff } \alpha \simeq \beta, \text{ where } {}^u\alpha \text{ stands for the extension of } \alpha.$$

Now, we start by defining an equivalence relation on **F**<sub>0</sub>:

Let  $a \simeq_0 b$  be defined as follows:

$$a \simeq_0 b =_{df}$$

$$(i) (a=b) \text{ or}$$

$$(ii) (\Omega a \ \& \ \Omega b \ \& \ a \equiv b) \text{ or}$$

$$(iii) (\Delta a \ \& \ \Delta b \ \& \ (\forall x(\text{app}(a,x) \equiv \text{app}(b,x)))) \text{ or}$$

$$(iv) \Omega a \ \& \ \Omega b \ \& \ [ \ ] a', a'', b', b'' /$$

$$a = \text{app}(a', a'') \ \& \ b = \text{app}(b', b'') \ \& \ (\forall x(\text{app}(a', x) \equiv \text{app}(b', x))) \ \& \ (a'' \simeq_0 b'' \ \forall b'' \simeq_0 a'').$$

Note that  $a \simeq_0 b$  does not imply  $a \equiv b$ . In fact  $a$  and  $b$  might be sets.

*Lemma 3:*  $\simeq_0$  is both reflexive and symmetric.

Proof:

$\simeq_0$  is reflexive because  $a=a$  for any  $a$ , hence  $a \simeq_0 a$  for any  $a$ .

$\simeq_0$  is symmetric because if  $a \simeq_0 b$  then

if (i) then  $b \simeq_0 a$ ,

if (ii) then  $\Omega b \ \& \ \Omega a \ \& \ (b \equiv a)$ , hence  $b \simeq_0 a$ ,

if (iii) then  $\Delta b \ \& \ \Delta a \ \& \ \forall x(\text{app}(b,x) \approx \text{app}(a,x))$  and hence  $b \approx_0 a$

if (iV) then  $\Omega b \ \& \ \Omega a \ \& \ [ \ ] \ b', b'', a', a'' /$

$b = \text{app}(b', b'') \ \& \ a = \text{app}(a', a'') \ \& \ (\forall x(\text{app}(a', x) \equiv \text{app}(b', x))) \ \& \ (b'' \approx_0 a'' \ \forall a'' \approx_0 b'')]$ ,

hence  $b \approx_0 a$ .  $\square$

It is more difficult to establish transitivity, and so we define the following by induction:

$$\approx_1 = \approx_0 \cup \{ \langle a, b \rangle : (\exists c)[a \approx_0 c \ \& \ c \approx_0 b] \}$$

$$\approx_{n+1} = \approx_n \cup \{ \langle a, b \rangle : (\exists c)[a \approx_n c \ \& \ c \approx_n b] \}$$

$$\approx = \bigcup_n \approx_n.$$

That  $\approx$  is reflexive and symmetric is obvious, as  $\approx$  is the transitive closure of  $\approx_0$ .

The following lemma establishes transitivity.

*Lemma 4:*  $\approx$  is transitive.

Proof:

If  $a \approx b$  and  $b \approx c$  then  $\exists n, m \ a \approx_n b$  and  $b \approx_m c$ . If  $n < m$  then  $a \approx_m b$  and  $b \approx_m c$ , therefore,  $a \approx_{m+1} c$  and so  $a \approx c$ .  $\square$

Note that (iV) is the cause of non-transitivity of  $\approx_0$  and this is what causes us to build the transitive closure of  $\approx_0$ . To see this clearly, take  $a \approx_0 b$  and  $b \approx_0 c$  where (iV) holds for both. Then when we get to  $b = \text{app}(b', b'') = \text{app}(b'_1, b''_1)$ , no relation at all can be deduced between  $b'$  and  $b'_1$  or  $b''$  and  $b''_1$ .

It is extremely interesting, however, to discover what sort of extensionality one gets from this equivalence relation; below we show the extensional behaviour of  $\approx$ . The following theorem for instance, states that if  $f$  is a propositional function and if  $a \approx b$  then  $f(a) \approx f(b)$ . For example

the morning star  $\approx$  the evening star, hence

the morning star is the star that rises at 4.00 am  $\approx$

the evening star is the star that rises at 4.00 am.

Theorem 1: If  $a \simeq b$  then  $f(a) \simeq f(b)$ , for any propositional function  $f$ .

Proof:  $f(a) = \text{app}(\lambda f, a)$ ,  $f(b) = \text{app}(\lambda f, b)$  and

$[\forall x(\text{app}(\lambda f, x) \equiv \text{app}(\lambda f, x)) \text{ and } a \simeq b]$ .

We conclude that  $\text{app}(\lambda f, a) \simeq \text{app}(\lambda f, b)$  by (iv) since  $f \simeq f$  and  $a \simeq b$ .

Hence,  $f(a) \simeq f(b)$ .  $\square$

The careful reader might produce an example which fells our aspirations concerning intensionality. The following is such an example:

If we take  $a$  to be groundhog(t) and  $b$  to be woodchuck(t) where  $t$  denotes a particular individual; and if we have that  $\forall x \Omega(\text{groundhog}(x))$ ,  $\forall x \Omega(\text{woodchuck}(x))$  and  $\forall x(\text{groundhog}(x) \equiv \text{woodchuck}(x))$  then we get that  $\text{groundhog}(t) \simeq \text{woodchuck}(t)$ . Now take  $f$  to be  $\langle \text{Bel}(j, x) / x \rangle$  and assume that  $f$  is a propositional function. Then according to the above theorem we get the undesirable consequence that  $\text{Bel}(j, \text{groundhog}(t)) \simeq \text{Bel}(j, \text{woodchuck}(t))$ .

However we assumed above that  $f$  is a propositional function and this assumption was crucial in obtaining the undesirable consequence. One way to avoid this is to prevent  $\langle \text{Bel}(j, x) / x \rangle$  from being a propositional function. Note however that even though we then reject  $\forall x \Omega(\text{Bel}(j, x))$ , we would still hope to have:

$$(PB) \quad \forall x(\Omega x \rightarrow \Omega(\text{Bel}(j, x))).$$

One way of achieving this would be by taking  $\langle \text{Bel}(j, x) / x \rangle$  to be a function which takes objects and returns objects with the special restriction that when one argument object is a proposition, the value object is a proposition too. This is a common construction for functions in a Frege structure and hence (PB) should be postulated as an axiom in the formal theory.

It is now time to define equivalence inside  $F_n$ ,  $n > 0$ . We define  $f \simeq^1 g$  as follows:

$$f \simeq^1 g =_{df} (\forall x(f(x) = g(x))) \vee (\forall x(\Omega(f(x)) \ \& \ \Omega(g(x)) \ \& \ (f(x) \equiv g(x))))$$

It is seen from this definition that nothing has been mentioned about (iV) or (iV') and hence our above definition of  $\simeq^1$  is suitable for a framework in which either

equivalence relation is assumed. Also the following two theorems show that equivalence in  $F_1$  is isomorphic to equivalence in  $F_0$  (for both senses of equivalence).

*Theorem 2:*  $f \simeq^1 g$  iff  $\lambda f \simeq \lambda g$ ; for any  $f, g$  propositional functions.

Proof:

( $\Rightarrow$ )

Assume  $f \simeq^1 g$ . If  $\forall x(f(x) = g(x))$

then  $\forall x(\text{app}(\lambda f, x) = \text{app}(\lambda g, x))$

then  $\lambda f = \lambda g$  and so  $\lambda f \simeq \lambda g$

If  $\forall x(\Omega(f(x)) \& \Omega(g(x)) \& (f(x) \equiv g(x)))$

then  $\Delta(\lambda f) \& \Delta(\lambda g) \& \forall x(\text{app}(\lambda f, x) \equiv \text{app}(\lambda g, x))$

Therefore  $\lambda f \simeq \lambda g$ .

( $\Leftarrow$ )

If  $\lambda f = \lambda g$  then  $f \simeq^1 g$ .

We cannot have  $\Omega(\lambda f)$  or  $\Omega(\lambda g)$  as **PROP** and **SET** are disjoint.

If (iii) then  $\forall x(\Omega(f(x)) \& \Omega(g(x)) \& (f(x) \equiv g(x)))$  and so  $f \simeq^1 g$

(iV) is impossible because  $\lambda f$  is a set and so we cannot have  $\Omega(\lambda f)$ .  $\square$

*Theorem 3:*  $f \simeq^1 g$  iff  $\lambda f \simeq \lambda g$ ; for any  $f, g$  propositional functions.

The proof is the same as above.  $\square$

Now we restrict our attention to  $F_0$  and having built an equivalence relation on  $F_0$  we see what extensional structure one can obtain on  $F_0$ , according to this equivalence relation.

Let  $\mathbf{1} = \{a: \Omega a \& a \simeq \top\}$  and  $\mathbf{0} = \{a: \Omega a \& a \simeq \perp\}$  where  $\top = \forall x.x=x$  and  $\perp = (c_0=c_1)$ ;<sup>115</sup> then we have the following theorem:

*Theorem 4:*  $\text{PROP}/\equiv = \{\mathbf{0}, \mathbf{1}\}$  where  $\equiv$  is the equivalence class according to  $\simeq$ <sup>116</sup>

Proof:

<sup>115</sup> Where  $c_0$  and  $c_1$  are distinct.

<sup>116</sup> Having an equivalence relation  $\simeq$  on **PROP**, say, one writes  $\text{PROP}/\equiv$  the quotient.  $\text{PROP}/\equiv = \{[a] : a \in \text{PROP}\}$ , where  $[a] = \{a' : a' \in \text{PROP} \text{ and } a' \simeq a\}$  is the equivalence class of  $a$ .



$0 = [\perp]$  and is in  $\text{PROP}/[\ ]$  as  $\perp$  is in  $\text{PROP}$ . For the same reason  $1$  is in  $\text{PROP}/[\ ]$ , hence  $\{0, 1\} \subseteq \text{PROP}/[\ ]$ .

Take  $a$  in  $\text{PROP}/[\ ]$ ;  $a = [b]$  where  $b$  is in  $\text{PROP}$ . Now if  $b$  is in  $\text{TRUTH}$  then  $b \equiv \top$  and so  $[b] = [\top] = 1$ , i.e  $a = 1$ . If  $b$  is not in  $\text{TRUTH}$  then as  $\Omega b$ ,  $b \equiv \perp$  and so  $[b] = [\perp] = 0$ , i.e  $a = 0$ . Therefore  $\text{PROP}/[\ ] = \{0, 1\}$ .  $\square^{117}$

Note here that we are working inside the model (Frege structure) and not inside the formal theory. This is why our non intuitionistic argument is allowed - we are working in the model set theoretically. If we define the boolean operators  $\#, \vee, \supset, \dots$ <sup>118</sup> in  $\text{PROP}/[\ ]$  as usual then we have the following theorem,

*Theorem 5:* For any  $a, b$  in  $\text{PROP}$ , the following holds;

- (1)  $[a \ \& \ b] = [a] \# [b]$ ,
- (2)  $[a \ \vee \ b] = [a] \vee [b]$ ,
- (3)  $[a] \vee 1 = 1$ ,
- (4)  $[a \equiv b] = ([a] = [b])$

and so on.

*Proof:*

If  $a \ \& \ b$  in  $\text{PROP}$  then  $[a \ \& \ b] = 1$  or  $0$ . If  $[a \ \& \ b] = 1$  then  $a \ \& \ b$  is in  $\text{TRUTH}$ , then  $a$  is in  $\text{TRUTH}$  and  $b$  is in  $\text{TRUTH}$ , therefore  $[a] = [b] = 1$  and so  $([a] \# [b]) = 1$ . For the remaining cases, the proof is similar.  $\square$

Now we know that  $F/[\ ]$ <sup>119</sup> has a logic on it where  $\text{PROP}/[\ ]$  has a boolean structure, what about application? If we characterize application in the quotient structure as:

$$(**) \quad [f]([a]) = [f(a)],$$

then is this well-defined? I.e. if  $[f] = [g]$  and  $[a] = [b]$  then is  $[f]([a]) = [g]([b])$ ?

Before we attempt to show that application is well-defined in the above sense, we

<sup>117</sup> Note that this theorem is not proved constructively. We used the law of excluded middle in arguing whether  $b$  is in  $\text{TRUTH}$  or not.

<sup>118</sup> Conjunction, disjunction and implication respectively.

<sup>119</sup> Again  $F/[\ ]$  is the collection of all the equivalence classes according to  $\approx$

need a reminder that our construction of a Frege structure made use of a fixed point, and that for each logical operator  $F$  we had both  $\Phi_F$  and  $\Psi_F$ . Since we need to use  $\equiv$  which is defined in terms of  $\rightarrow$  in the following sense:  $a \equiv b$  is  $(a \rightarrow b) \& (b \rightarrow a)$ , the relevant logical constant is  $\rightarrow$ . Recall that

$a \rightarrow b$  is true iff  $a$  is in **PROP**, and  $b$  is in **TRUTH** if  $a$  is in **TRUTH**.

$\Phi_{\rightarrow}(X, x, y)$  is:

"  $x$  is in  $X_0$  and  $y$  is in  $X_0$  provided that  $x$  is in  $X_1$  "

where  $X_0$  is the collection of propositions and  $X_1$  is the collection of true propositions with  $X_1 \subseteq X_0$ .

This observation enables us to deduce that if  $a \equiv b$  and  $\Omega a$  then  $\Omega b$ . It can be seen as follows:

If  $a \equiv b$  then  $(a \rightarrow b) \& (b \rightarrow a)$ ; hence from  $\Phi_{\rightarrow}(\text{PROP}, \text{TRUTH}, a, b)$  and  $\Phi_{\rightarrow}(\text{PROP}, \text{TRUTH}, b, a)$  we get that  $\Omega a$  and  $\Omega b$ . In fact we did not need the condition that  $\Omega a$ .

The following lemmas are needed to show that application is defined.

*Lemma 5:* If  $\Omega a$ ,  $\Omega b$ ,  $a \equiv a'$ ,  $b \equiv b'$  and  $a \simeq b$  then  $a' \simeq b'$ .

Proof:

$\Omega a$  and  $a \equiv a'$  hence  $\Omega a'$  and so  $a \simeq a'$ . In the same way we get that  $b \simeq b'$ . But  $a \simeq b$ , hence  $a' \simeq b'$  by transitivity and symmetry of  $\simeq$   $\square$

*Lemma 6:* If  $\Delta a$ ,  $\Delta b$ ,  $a \simeq b$ ,  $\forall x(\text{app}(a, x) \equiv \text{app}(a', x))$ ,  $\forall x(\text{app}(b, x) \equiv \text{app}(b', x))$  then  $a' \simeq b'$ .

Proof:

If  $\Delta a$ , then  $\forall x \Omega(\text{app}(a, x))$ , hence as  $\text{app}(a', x) \equiv \text{app}(a, x)$ , we get  $\forall x \Omega(\text{app}(a', x))$ , hence  $\Delta a'$ . From  $\Delta a$ ,  $\Delta a'$  and  $\forall x(\text{app}(a, x) \equiv \text{app}(a', x))$  we get that  $a \simeq a'$ . We follow the same procedure to prove that  $b \simeq b'$ . From  $a \simeq a'$ ,  $b \simeq b'$ ,  $a \simeq b$  we get that  $a' \simeq b'$ .  $\square$

*Theorem 6:* If  $\Delta a$ ,  $\Delta b$ ,  $a \simeq b$ ,  $c \simeq d$  then  $\text{app}(a, c) \simeq \text{app}(b, d)$ .

Proof:

As  $\Delta a, \Delta b, a \simeq b$ , then  $\forall x(\text{app}(a,x) \equiv \text{app}(b,x))$ .

Also  $c \simeq d$  hence by (iV)  $\text{app}(a,c) \simeq \text{app}(b,d)$ .  $\square$

*Theorem 7:* If  $f, g$  are propositional functions, and if  $f \simeq^1 g$  and  $a \simeq b$  then  $f(a) \simeq g(b)$ .

Proof:

Since  $f \simeq^1 g$ , we get by Theorem 2 that  $\lambda f \simeq \lambda g$ . Hence by Theorem 6,  $\text{app}(\lambda f, a) \simeq \text{app}(\lambda g, b)$ , hence  $f(a) \simeq g(b)$ .  $\square$

This theorem shows that application is well-defined with respect to  $\simeq$  on propositional functions. One wonders whether application is well-defined everywhere and not solely on propositional functions. It is not obvious how to show this due to clause (iV). But we do not care about application outside propositional functions and can be satisfied with this position.

*Theorem 8:* If  $\Delta a, \Delta b$  and  $a \simeq b$  then  $\text{pred}(a,x) \simeq \text{pred}(b,x)$ .

Proof:

By Theorem 6,  $\text{app}(a,x) \simeq \text{app}(b,x)$ .

Also by Lemma 5,  $\text{pred}(a,x) \simeq \text{pred}(b,x)$ , this is because:

$$\begin{aligned} \Omega(\text{app}(a,x)), \Omega(\text{app}(b,x)), \text{app}(a,x) \equiv \text{pred}(a,x), \text{app}(b,x) \equiv \text{pred}(b,x) \\ \text{and } \text{app}(a,x) \simeq \text{app}(b,x). \quad \square \end{aligned}$$

*Theorem 9:* For any  $a, b, c$  sets such that  $a \simeq b$ , we have:

$$\{\text{app}(\text{every}'(a),c)\} \vdash \text{app}(\text{every}'(b),c).$$

Proof:

$$\begin{aligned} \text{app}(\text{every}'(a),c) &= \\ \forall x(\text{app}(a,x) \rightarrow \text{app}(c,x)) & \\ \equiv \forall x(\text{app}(b,x) \rightarrow \text{app}(c,x)) & \\ = \text{app}(\text{every}'(b),c). & \quad \square \end{aligned}$$

Note also that if we mix our structures in the following way:

$$(*) \quad [f](a) = [f(a)],$$

then this is well-defined. I.e. the following holds;

*Lemma 7:* If  $[f] = [g]$  then for any  $a$ ,  $[f](a) = [g](a)$ .

*Proof:*

Assume  $[f] = [g]$ , then for any  $a$ , either

(i)  $f(a) = g(a)$  or

(ii)  $\Omega(f(a)) \ \& \ \Omega(g(a)) \ \& \ f(a) \equiv g(a)$ .

If  $f(a) = g(a)$  then  $[f(a)] = [g(a)]$ ,

if (ii) then  $f(a) \simeq g(a)$ , i.e.  $[f(a)] = [g(a)]$ .  $\square$

Note here that we mixed the types of  $f$  and  $a$ . This needs attention and anyone who would like to use this method has to ensure that the typing fits well.

In the extensional structure, we defined functional application as  $[f]([a]) = [f(a)]$ .

What about abstraction? We define  $\lambda^e[f] = [\lambda f]$ .

*Theorem 10:*  $\lambda^e$  is well-defined on propositional functions. I.e. if  $[f] = [g]$  then  $\lambda^e[f] = \lambda^e[g]$  for  $f$  and  $g$  propositional functions.

*Proof:* If  $[f] = [g]$ , i.e.  $f \simeq^1 g$  then  $\lambda f \simeq \lambda g$  by Theorem 2, and so  $[\lambda f] = [\lambda g]$ . Hence  $\lambda^e[f] = \lambda^e[g]$ .  $\square$

We should also build application of an object to another. We define  $\text{app}^e([a],[b]) = [\text{app}(a,b)]$ .

*Theorem 11:*  $\text{app}^e$  is well-defined on classes of properties. That is, if  $a, b$  are properties and if  $[a] = [b]$  and  $[c] = [d]$  then  $\text{app}^e([a],[c]) = \text{app}^e([b],[d])$ .

*Proof:* This is due to Theorem 6.  $\square$

In building the boolean connectives of the extensional structure above we did not mention anything about universal quantification; it can be shown however that once it has been defined, we can obtain the following theorem,

*Theorem 12:*  $[\forall x f(x)] = \Pi x [f(x)]$  where  $\Pi$  is the universal quantifier in the extensional structure.

*Proof:* Easy.  $\square$

So far it is clear that the extensional structure is explicitly closed. However what we have not shown yet is that the addition of the extensionality axioms to the system (by  $\approx_0$ ) results in either of the following two conclusions:

1. The whole **PROP**/ $\square$  collapses into one trivial element.
2. This can be seen as an inconsistency in the theory.<sup>120</sup>

Now this means that one cannot build extensionality on the top of the intensional structure. In fact this has its background in the literature: Gordeev has shown (see [BE4], page 235) that one cannot add extensionality without making the system inconsistent. In fact the inconsistency can be shown in our structure above by taking the following example:

Take  $R = \lambda x(x \in x = \perp)$ ,

and  $a = \text{app}(\lambda x(x = \perp), \perp)$ ,

and  $b = \text{app}(\lambda x(x = \perp), R \in R)$ .

Then  $a = (\perp = \perp) \in \text{TRUTH}$ ,

and  $b = (R \in R) = \perp \in \text{PROP -TRUTH}$ .

But  $\perp \approx_0 (R \in R)$  because  $R \in R \equiv \perp$ .

Hence  $a \approx b$

This implies that the whole **PROP**/ $\square$  collapses into a trivial one element algebra, because  $\mathbf{1} = [a] = [b] = \mathbf{0}$ .

---

<sup>120</sup> Thanks to Uwe Mönnich who drew my attention to this point.

*PART C. THOMASON'S APPROACH*

Another issue of interest concerns the relationship between the present theory and that of Thomason in [TH4]. Thomason takes propositions to be intensional and has an extensional truth-finding operator  $^u$ :  $\text{PROP} \rightarrow \{0,1\}$ , which unpacks intensionality and satisfies the usual boolean connectives.<sup>121</sup>  $^u$  is like  $^u$  in MG but Thomason does not introduce  $\hat{\phantom{x}}$ , though it could be constructed when possible worlds are introduced.

With our approach, the extensional structure is built out of the intensional one and it satisfies some nice closure properties. Even though,  $\sim, \supset$  and other connectives had to be introduced as new constants by Thomason, and meaning postulates had to be provided to ensure an homomorphism between propositions and truths, Thomason did not provide a way to build a model which satisfies these meaning postulates. Similarly, the relation between intensional quantifiers and extensional ones is introduced by meaning postulates with Thomason, whereas for us, we have shown that these axioms can be satisfied in a model where extensionality is built on the top of intensionality (see theorems 5 and 12). However, because there is no model in which extensionality and intensionality occur together in this strong sense, the extensional and intensional models have to be separate and joined by homomorphic functions. That is, the following two meaning postulates of Thomason

$$^u(\forall x\phi) = \forall x^u\phi$$

$$^u(a_1 \equiv a_2) = (^u a_1 = ^u a_2)$$

have to be imposed.

Not only does Thomason introduce two constants for each logical constant and quantifier (for the intensional and extensional structures) - which is something I also do here - but he also introduces for each verb or common noun two different names,

---

<sup>121</sup> Thomason uses a 2-valued classical logic, where everything is constructed out of  $\neg, \&$ ; and type freeness is not insisted upon although all the work can be done using constructive and type free theories.

one standing for the intensional interpretation and another standing for the extensional one. For instance, walk+ below is intensional and walk# is extensional. The connection between these two verbs is again stipulated by means of a meaning postulate:

$$\forall x^e [ {}^u\text{walk+}(x) = \text{walk\#}(x)].$$

According to the approach adopted here, if we take walk+ to be of type  $\langle e,p \rangle$ , then we have the result that  $[\text{walk+}]$  can be considered to be of type  $F_0/\square \cup F_0 \rightarrow \text{PROP}/\square$ , which enables us to apply (\*) and obtain the following:

$$\forall x^e [ [\text{walk+}(x)] = [\text{walk+}](x)].$$

As well as introducing walk+ of type  $\langle e,p \rangle$  and walk# of type  $\langle e,t \rangle$ , Thomason also postulates a third constant walk' of type  $\langle \langle \langle e,p \rangle, p \rangle, p \rangle$ . Having the following constants, walk', walk+, walk#, John', John+, (where John' is of type  $\langle \langle e,p \rangle, p \rangle$  and John+ is of type  $e$ ) creates a number of possibilities for the translation of John walks in Thomason's approach. It can be any of:

- (1) walk'(John')
- (2) walk+(John+)
- (3) walk#(John+)
- (4) John'(walk+)

(3) is of type  $t$  and so is ruled out. The remaining three should have the same truth value and Thomason ensures this by meaning postulates; something like:

- (5)  $\forall y^{\langle e,p \rangle} [ {}^u\text{John}'(y) = {}^u y(\text{John+})]$
- (6)  $\forall y^{\langle \langle \langle e,p \rangle, p \rangle, p \rangle} [ {}^u\text{walk}'(y) = {}^u y(\text{walk+})]$
- (7)  $\forall x^e [ {}^u\text{walk+}(x) = \text{walk\#}(x)]$

These enable him to derive  ${}^u\text{walk}'(\text{John}') = {}^u\text{walk+}(\text{John+})$ . What about walk'(John') and John'(walk+)? To move freely between the 4 formulae, Thomason still needs two postulates:

- (8)  $\forall y^{\langle e,p \rangle} [ \text{John}'(y) = y(\text{John+})]$

$$(9) \forall y \langle \langle e, P \rangle, P \rangle [\text{walk}'(y) = y(\text{walk}+)]$$

Using the above, Thomason can prove:  ${}^u\text{walk}'(\text{John}') = {}^u\text{John}'(\lambda v_4 {}^e\text{walk}'(\lambda z \langle e, P \rangle .z(v_4)))$ . So they have the same truth value. Note that even though formula (7) (or its equivalent) is not added as a meaning postulate but is a consequence of (\*) in our approach, it is still to be seen how formulae (5)-(6) could be obtained here. (5)-(6) are needed because they lift (or lower) the types of walk+ and John+ (or walk' and John'). If we could write John' as  $\lambda P.P(\text{John}+)$  and walk' as  $\lambda P.P(\text{walk}+)$  then (5) and (6) would no longer be needed due to  $\lambda$ -conversion. It is a common feature of both the approach here and Thomason's approach that John', walk', etc. cannot be written as straightforward  $\lambda$ -expressions in terms of their corresponding elements of a lower level. This is because this actually reduces the intensionality behaviour of the various constituents. We do not want for instance that  $\text{John}'(y) = y(\text{John}+)$  but want their truth-values to be the same for every  $y$ . With the approach that I put forward, we can find elements in the model that could be written using  $\lambda$ -expressions without affecting the internal definability of any type. This will be discussed in detail in Chapter 6.

#### *PART D. POSSIBLE WORLDS*

Now we come to the issue of possible worlds and modality. There are many ways we could deal with this issue, but let us see how to accommodate some of them within our framework. The three we want to consider are:

- A. Take possible worlds to be maximal sets of propositions and define  $\Box$  in terms of these possible worlds.
- B. Take a collection of Frege structures and define  $\Box$  in terms of this collection; that is, each element stands for a possible world.
- C. Define  $\Box$  in terms of the equality relation using one Frege structure only.



Now, (A) seems to be quite hard to do while staying inside the Frege structure. The reason for this is that maximal sets of propositions are not internally definable.

Again (C) is problematic for the following reasons:

If we take  $\Box\alpha =_{df} \alpha=(\alpha=\alpha)$ , we can not prove things like:  $\Box(\underline{\text{white is white}})$ . This is because even though  $(\underline{\text{white is white}}) = (\underline{\text{white is white}})$ , there is nothing which guarantees the equality between  $(\underline{\text{white is white}})$  and  $(\underline{\text{white is white}} = \underline{\text{white is white}})$ . They are equivalent as propositions but not necessarily equal. We could of course solve this problem by positing some axioms to this effect; but this is ad hoc.

Solution (B) appears the most convenient within our framework. We just take a collection of Frege structures, each of them standing for a possible world and then interpret  $\Box$  according to the usual techniques. By dealing with modality and possible worlds in this way, we can dispense with our earlier definitions of  $\hat{\quad}$  and  $\cup$  and redefine them in terms of possible worlds. This is again straightforward and very common in the literature.

We will not say more on possible worlds in this thesis but we finish by summarizing what this chapter was concerned with. We started by defining two equivalence relations which helped us illustrate that one cannot add extensionality axioms on the top of a Frege structure. We then compared our work with Thomason and discussed possible worlds.

*CHAPTER 6. TYPE THEORY AND THE MONTAGUE FRAGMENT IN A FREGE STRUCTURE*

*PART A. MONTAGUE'S IL*

The originality of Montague's approach lies in the axiomatisation of a procedure which maps English terms and expressions into the logical language IL. In doing ordinary symbolic logic we intuitively translate from English to the formal language. With Montague, we have two formalised steps: the syntactic step which translates English into IL, (the language of typed intensional logic that Montague used), and the semantic step which gives a semantic interpretation of IL. The interpretation thus obtained is also an interpretation of the English fragment that was translated into IL.

The translation procedure is axiomatised in the following way: first translate the English basic categories into IL, then with each syntactic rule of the English fragment (English is axiomatised) associate a translation rule which translates the output of the syntactic rule into IL. This procedure is set up so that if an expression is assigned by a syntactic rule to a certain syntactic category, then it is mapped by the translation rule into a logical expression of the corresponding logical type. Each syntactic category corresponds to one and only one logical type, though we can have two different categories associated with the same type. IL employs Russell's type theory, and thus can be classified under the approach of restricting the language to avoid the paradoxes. Since Montague offered his approach, most of the subsequent approaches to natural language semantics seem to have been Montagovian. This is unfortunate - though not because we think that Montague's approach is not worthy of attention: on the contrary, it has made tremendous impact on the study of the semantics of natural languages. However, in so far as it utilises type theory the approach is problematic. We have already seen that the Montagovian approach consists of two main components : type theory and the translation procedure. Since PTQ (see [TH2]) was developed, semanticists and linguists have been facing different kinds of problems

with it, which is not surprising. Some of them would perhaps claim that the defects lie in the translation procedure - which is something I disagree with. The most problematic issues lie in the theory of types, and that is where semanticists should start. The translation procedure is elegant and novel and we cannot think of any other procedure which would work better. Thus the issue first should be to elaborate a logical theory which works better than type theory, and then to ameliorate any difficulties with the translation procedure. Fine - but what are the other alternatives to type theory? We have seen most of them in Chapter 1 when we studied the theoretical problem of nominalisation. Russell's type theory was merely the first and the weakest solution offered: as we saw there are many others. It is true that we were talking in abstract terms in Chapter 1 and that we did not give many linguistic examples as to why we think our Frege structures would be better suited for natural languages; we hope that semanticists understand that as type theory was the weakest theory that could work for mathematics, it is unlikely to be powerful enough for the semantics of natural languages either. (This is not only because it is weak mathematically but because it has been claimed by linguists that type theory does not cope with certain issues.) It is straightforward to list many defects of type theory that mathematicians complained about years ago, and to demonstrate that they generate corresponding problems for linguistics. Take again the example of a set that contains itself: mathematicians realised years ago that this concept was impossible in type theory - and this is one of the reasons which led to ZF set theory. Linguists only lately<sup>122</sup> recognised their analogous problem: namely that they could not predicate a property of itself. As another example, consider quantification. In type theory, there is a different set of natural numbers at each level, and the quantifier ranged over each level separately. The analogous linguistic problem is that there is no way to say everything has a property but only: everything of level n has

---

<sup>122</sup> The reader is referred to Parsons work in [PA5] and Turner's work in [TU7].

a property of level  $n+1$ . These and many other problems of type theory for mathematics are also problems of natural language semantics based on type theory. This does not mean, of course, that we should follow mathematics blindly - for this might make us lose philosophical insights. For instance, it is enough for analysis to use B-G (Bernays-Gödel) set theory - but in Natural Language semantics we are also looking for philosophical insights from the theories we use. Frege structures, embodying as they do Frege's ideas, have philosophical motivations.

This chapter is concerned with showing why type theory in Montague's sense is problematic for nominalisation, and then building a type theory similar to Montague's inside our framework. The result will be that we avoid the disadvantages of Montague's system and yet retain all the good things that type theory has to offer. We start first by showing the inadequacy of Montague's typing in IL.

In IL, we can have a function of type  $\langle a, b \rangle$  applied to an element of type  $a$  but we can never apply a function to itself or to any other function of the same type. The typing of any item is fixed in advance, by the syntactic and translation rules. As mentioned already, to each category of the English fragment there corresponds a type in IL such that all the expressions of that category are translated into logical expressions of that type. In Montague's approach, categories are defined recursively. A complex category of the form  $X/Y$  labels an expression that takes expressions of category  $Y$  as arguments and yields an expression of category  $X$ . As an example, the category  $IAV$  is defined to be  $IV/IV$  and it takes expressions of type  $IV$ , returning expressions of type  $IV$ . Similarly,  $T=t/IV$  takes an intransitive verb and gives a sentence. Types are also defined recursively in Montague's IL and there is a homomorphism from categories to types. The important point to make here is that the function  $f$  which maps categories into types always makes sure that the type of the category built out of two old ones is higher than the type of its input; so for example, the type of  $T$  is higher than the type of  $IV$ . It is essentially this typing

constraint that creates a problem for nominalisation. We illustrate this by the following few examples.

(1a). John runs

(1b). To run is fun

(1c). John is fun

(1d). fun is fun.

Let us assume that in each case the predicate denotes a function which applies to the denotation of the subject. Thus, in (1a), John is of type  $e$  and runs is of type  $\langle e, t \rangle$ . Let us assume that in (1b), to run (which is syntactically built out of run) is of type  $\langle e, t \rangle$ . In (1c), it is obvious that is fun is of type  $\langle e, t \rangle$  while in (1b), it must be of type equal or higher than  $\langle \langle e, t \rangle, t \rangle$ . This is a problem; we seem to have two different types for is fun. Now if we take (1d), we see that according to the Montagovian approach, one fun must be of higher level than the other. Just this simple sentence on its own creates an infinite number of fun's in the syntax. For assume we say fun<sub>0</sub> is fun<sub>1</sub>, then we also want to say fun<sub>1</sub> is fun<sub>2</sub> and again fun<sub>2</sub> is fun<sub>3</sub> and so on. We do not have this problem with Frege structures, since nominalised forms there take interpretations in SET, IV's are interpreted as propositional functions, and terms as elements of  $F_0$  - and everything fits well together because of the isomorphism between sets and propositional functions. However, for those people who like Montague's IL, and especially its type theory, we introduce the following typing in our framework.

### *PART B. A TYPE THEORY $T_{pol}$*

Let us assume that  $F_0$ , SET and PROP constitute three basic intensional domains where  $F_0$  is the domain of objects, SET is the domain of properties and PROP is the domain of propositions together with the conditions that  $SET \subseteq F_0$  and  $PROP \subseteq F_0$ .

Then we define other intensional domains out of those ones as follows:

$A \rightarrow B = \{a \text{ in } A : \text{for every } x \text{ in } A, \text{app}(a,x) \text{ is in } B\}$

As a special case  $F_0 \rightarrow \text{PROP}$  is the collection of all the unary propositional functions (actually the nominals of these propositional functions). In fact, in a Frege structure,  $(F_0 \rightarrow \text{PROP}) = \text{SET}$  as is shown in the following lemma:

*Lemma 1:* In a Frege structure,  $\text{SET} = F_0 \rightarrow \text{PROP}$ .

Proof:

(i) If  $a$  is in  $\text{SET} \subseteq F_0$ , then for every  $x$  in  $F_0$ ,  $\text{app}(a,x)$  is in  $\text{PROP}$ , and hence  $a$  is in  $F_0 \rightarrow \text{PROP}$ .

(ii) If  $a$  is in  $F_0 \rightarrow \text{PROP}$ , then  $a$  is in  $F_0$  and for every  $x$  in  $F_0$ ,  $\text{app}(a,x)$  is in  $\text{PROP}$ , and hence  $a$  is in  $\text{SET}$ .

Hence  $\text{SET} = F_0 \rightarrow \text{PROP}$ .  $\square$

In what follows we assume we are working with Frege structures where  $\text{PROP} \cap \text{SET}$  is empty. Note that  $\text{PROP} \rightarrow \text{PROP}$  need not be empty even in the case where  $\text{PROP} \cap \text{SET}$  is empty. If  $a$  is in  $\text{PROP} \rightarrow \text{PROP}$ , then  $a$  is in  $\text{PROP}$  and for every  $x$  in  $\text{PROP}$ ,  $\text{app}(a,x)$  is in  $\text{PROP}$ . But  $a$  is not in  $\text{SET}$ , since we have no guarantee that  $\text{app}(a,x)$  is in  $\text{PROP}$  for arbitrary  $x$  in  $F_0$ .

Even though  $\text{PROP} \rightarrow \text{PROP}$  may not be empty, we do not allow  $\neg$ ,  $\&$ , etc to be objects of  $F_0$ . They are in the Frege structure, but as functionals rather than objects. Another non-empty domain in a Frege structure is  $(F_0 \rightarrow \text{PROP}) \rightarrow \text{PROP} = \{a \text{ in } (F_0 \rightarrow \text{PROP}) : \text{for every } x \text{ in } F_0 \rightarrow \text{PROP}, \text{app}(a,x) \text{ is in } \text{PROP}\}$ . It is non-empty because it contains  $\langle x=b/x \rangle$ . The above domain is the domain of properties of sets and is similar to Montague's denotation of terms.

Note that the domains defined above have the property that if  $A$  and  $B$  are domains then  $(A \rightarrow B) \subseteq A$ . This is the fact which will enable us to interpret nominalisation. These domains also have the properties given by the following lemma:

*Lemma 2:* If  $A$  and  $B$  are domains built as above then:

(1) If  $A \subseteq A'$  then  $(A \rightarrow B) \supseteq (A' \rightarrow B)$ .

(2) If  $B \subseteq B'$  then  $(A \rightarrow B) \subseteq (A \rightarrow B')$ .

Proof: Easy.  $\square$

Types are defined recursively as follows:

$p, e$  are fixed objects.

1.  $p$  is a non-empty intensional type.<sup>123</sup>

2.  $e$  is a non-empty intensional type.

3. If  $a, b$  are intensional types then  $\langle a, b \rangle$  is an intensional type.<sup>124</sup>

A. The basic expressions of  $T_{pol}$  are as follows:

1. For each type  $a$ , there is a denumerably infinite number of constants;  $Con_a$  is the collection of all non-logical constants of type  $a$ .

2. For each type  $a$ , there is a denumerably infinite number of variables;  $Var_a$  is the collection of all variables of type  $a$ .

3. There is also a set of functionals which take arguments in a particular type and return values in particular types. For instance the function which takes elements  $u$  in  $\langle e, p \rangle$  and returns  $\lambda v \langle \langle e, p \rangle, p \rangle_{app}(v, u)$  is a functional which takes arguments of type  $\langle e, p \rangle$  and returns values of type  $\langle \langle \langle e, p \rangle, p \rangle, p \rangle$ . If  $G$  is such a functional, we denote it by  $G_{\langle e, p \rangle} \langle \langle e, p \rangle, p \rangle$  and we denote its type by  $ME_{\langle \langle e, p \rangle, \langle \langle e, p \rangle, p \rangle \rangle}$ . We do not have variables over functionals but we have constants over them. Functionals are going to provide interpretations for determiners, verb phrase adverbs, etc. This is acceptable because, we have only a denumerably infinite number of determiners, verb phrase adverbs and infinitive complement verbs and their translation will be given in Part C. Here is a list of some of the functionals that we assume to be in our language:

that' is the functional such that for any  $u$  of type  $p$ ,

$$that'(u) = \lambda v \langle e, p \rangle_{app}(v, u).$$

<sup>123</sup> The notion of empty type does not occur in Montague. Yet I introduce it here as it makes things more elegant.

to' is the functional such that  $u$  of type  $\langle e, p \rangle$ ,

$$to'(u) = \lambda v \langle \langle e, p \rangle, p \rangle_{app}(v, u).$$

ing' is the functional such that  $u$  of type  $\langle e, p \rangle$ ,

$$ing'(u) = \lambda v \langle \langle e, p \rangle, p \rangle_{app}(v, u).$$

every' is the functional such that for any  $u$  of type  $e$ ,

$$every'(u) = \lambda v \langle e, p \rangle \forall w^e (app(u, w) \rightarrow app(v, w)).$$

the' is the functional such that for any  $u$  of type  $e$ ,

$$the'(u) = \lambda v \langle e, p \rangle \exists w^e \forall o^e (app(u, o) \equiv (o = w)) \& app(v, w).$$

a' is the functional such that for any  $u$  of type  $e$ ,

$$a'(u) = \lambda u \langle e, p \rangle \lambda v \langle e, p \rangle \exists w^e (app(u, w) \& app(v, w)).$$

We also have a set of empty types, i.e. types  $\langle b, a \rangle$  where

- (1)  $a = b = \langle e, p \rangle$
- (2)  $a = \langle \langle e, p \rangle, p \rangle$  and  $b = \langle e, p \rangle$  or
- (3)  $a = \langle e, p \rangle$  and  $b = \langle \langle e, p \rangle, p \rangle$ .

Note that if we have a type  $\langle a, b \rangle$  which is empty, then the type  $\langle alb \rangle$  need not be empty.<sup>125</sup> We have types where the syntactic categories such as terms, verbs, common nouns, nominals, etc. will take translations. We also have functionals which operate on those types. For instance every gets translated as a functional which takes arguments from the type of common nouns and returns arguments of type terms. The idea of restricting the type hierarchy to three layers (objects, functions and functionals) is not novel - see for instance [CH3], page 77.

### B. The syntactic rules of $T_{pot}$ :

$ME_a$ , the collection of meaningful expressions of type  $a$ , is defined recursively as follows:

*B1. Intensional expressions:* If  $a$  and  $b$  are intensional non-empty types then

1. Each variable of type  $a$  is in  $ME_a$ .

<sup>125</sup> Note that one could do away with empty types and use a free logic instead.



2. Each constant of type  $a$  is in  $ME_a$ .
3. If  $\alpha$  is in  $ME_a$  and  $u$  is a variable of type  $b$  and  $\langle b, a \rangle$  is a non-empty type then  $\lambda u. \alpha$  is in  $ME_{\langle b, a \rangle}$ .
4. If  $\alpha$  is in  $ME_{\langle a, b \rangle}$  and  $\beta$  is in  $ME_a$ , then  $app(\alpha, \beta)$  is in  $ME_b$  and  $pred(\alpha, \beta)$  is in  $ME_e$ .
5. If  $\alpha, \beta$  are in  $ME_a$  then  $\alpha = \beta$  is in  $ME_p$ .
- 6-10 If  $\Phi, \Psi$  are in  $ME_p$  then the following are in  $ME_p$ ;
  6.  $\neg \Phi$
  7.  $\Phi \vee \Psi$
  8.  $\Phi \& \Psi$
  9.  $\Phi \rightarrow \Psi$
  10.  $\Phi \equiv \Psi$
11. If  $\Phi$  is in  $ME_p$  and  $u$  is a variable of any intensional type then  $\forall u \Phi$  is in  $ME_p$ .
12. If  $\Phi$  is in  $ME_p$  and  $u$  is a variable of any intensional type then  $\exists u \Phi$  is in  $ME_p$ .
13. If  $\alpha$  is in  $ME_e$  then  $\Omega \alpha$  is in  $ME_e$ .
14. If  $\alpha$  is in  $ME_a$  then  $\alpha$  is in  $ME_e$ .
15. If  $\alpha$  is in  $ME_{\langle a, b \rangle}$  then  $\alpha$  is in  $ME_a$ .
16. If  $G_a^b$  is a functional in  $ME_{\langle a, b \rangle}$  and  $\alpha$  is in  $ME_a$  then  $G_a^b(\alpha)$  is in  $ME_b$ .

If  $\alpha$  is in  $ME_{\langle a, \langle b, c \rangle \rangle}$ ,  $\beta$  is in  $ME_a$  and  $\delta$  is in  $ME_b$ , then we write  $app_2(\alpha, \beta, \delta)$  for  $app(app(\alpha, \beta), \delta)$  which is in  $ME_c$ .

*The semantics of  $T_{pot}$ :*

A model structure is a Frege structure where the constants  $j, m, w$ , etc. in  $Con_e$  - which correspond to proper names in English - are not propositions nor sets nor composed out of other objects using app or pred.

The set of denotations of type  $a$  is defined as follows:<sup>126</sup>

<sup>126</sup> Notice here that we have nothing which corresponds to Montague's individual concepts, i.e. of type

$$D_e = F_0$$

$$D_p = \text{PROP}$$

$$D_{\langle a,b \rangle} = (D_a \rightarrow D_b) \text{ where } \langle a,b \rangle \text{ is non-empty and } a, b \text{ are intensional.}$$

An assignment function  $g$  is a function which assigns an element of  $D_a$  to each variable  $u$  of intensional type  $a$ .

We also need a function  $C$  which assigns an element of  $D_a$  to each constant of type  $a$ .

Also  $C$  assigns an F-functional of the Frege structure  $F$  to each constant functional in

$T_{\text{pol}}$ . Hence a *model*  $M$  is a 2-tuple  $\langle F, C \rangle$ . Now we move to the semantic clauses of

$T_{\text{pol}}$ :

1. If  $\alpha$  is a non-logical constant then  $[[\alpha]]^{M,g} = C(\alpha)$ .

2. If  $\alpha$  is a variable, then  $[[\alpha]]^{M,g} = g(\alpha)$ .

3. If  $\alpha$  is in  $ME_a$ ,  $u$  is a variable of type  $b$ ,  $a$  and  $b$  are intensional and  $\langle a,b \rangle$  is non-empty, then  $[[\lambda u.\alpha]]^{M,g} =$  an element  $h$  of  $D_b$  such that for every  $x$  in  $D_b$ ,  $\text{app}(h,x) = [[\alpha]]^{M,g[x/u]}$ .<sup>127</sup>

In the previous chapter, we defined  $\lambda^e[f]$  to be  $[\lambda f]$ .

4. If  $\alpha$  is in  $ME_{\langle a,b \rangle}$  and  $\beta$  is in  $ME_a$  and  $a$  and  $b$  are intensional then

$$[[\text{app}(\alpha,\beta)]]^{M,g} = \text{app}([[\alpha]]^{M,g},[[\beta]]^{M,g}).$$

4'. If  $C_a^b$  is in  $ME_{\langle a,b \rangle}$  and  $\beta$  is in  $ME_a$  then

$$[[C_a^b(\beta)]]^{M,g} = [[C_a^b]]^{M,g}([[\beta]]^{M,g}).$$
<sup>128</sup>

5. If  $\alpha, \beta$  are in  $ME_a$  and  $a$  is an intensional type, then  $[[\alpha = \beta]]^{M,g} = ([[ \alpha ] ]^{M,g} = [[ \beta ] ]^{M,g})$ .

6. If  $\Phi$  is in  $ME_p$  then  $[[\neg \Phi]]^{M,g} = \neg [[\Phi]]^{M,g}$ .

7. If  $\Phi$  and  $\Psi$  are in  $ME_p$  then  $[[\Phi \& \Psi]]^{M,g} = [[\Phi]]^{M,g} \& [[\Psi]]^{M,g}$ .

$\langle s,e \rangle$ ; e.g. take the pope to denote a function in  $D_e^{I \times J}$  which picks out a different individual at each moment of time. However, there are a number of articles in the literature which dispute the utility of this type, see for instance [DO1].

<sup>127</sup> Note that we impose on our models the very important property needed for clause 3, namely, that the element  $h$  exists. We will have something to say about that in Part C.

<sup>128</sup> As remarked earlier, this enables us to deal with determiners, verb phrase adverbs and infinitive complement verbs whose denotations are not inside  $F_0$  but are functionals which operate on  $F_0$ .

8. If  $\Phi$  and  $\Psi$  are in  $ME_p$  then  $[[\Phi \vee \Psi]]^{M,g} = [[\Phi]]^{M,g} \vee [[\Psi]]^{M,g}$ .
9. If  $\Phi$  and  $\Psi$  are in  $ME_p$  then  $[[\Phi \rightarrow \Psi]]^{M,g} = [[\Phi]]^{M,g} \rightarrow [[\Psi]]^{M,g}$ .
10. If  $\Phi$  and  $\Psi$  are in  $ME_p$  then  $[[\Phi \equiv \Psi]]^{M,g} = [[\Phi]]^{M,g} \equiv [[\Psi]]^{M,g}$ .
11. If  $\Phi$  is in  $ME_p$  and  $u$  is a variable then  $[[\forall u \Phi]]^{M,g} = \forall \langle [[\Phi]]^{M,g[x/u]} / x \rangle$ .
12. If  $\Phi$  is in  $ME_p$  and  $u$  is a variable then  $[[\exists u \Phi]]^{M,g} = \exists \langle [[\Phi]]^{M,g[x/u]} / x \rangle$ .

The following theorem shows that the above semantics is well defined.

*Theorem 4:* If  $\alpha$  is in  $ME_a$  then  $[[\alpha]]^{M,g}$  is in  $D_a$ . Also if  $C_a^b$  is in  $ME_{\langle a,b \rangle}$  and  $\beta$  is in  $ME_a$  then  $[[C_a^b(\beta)]]^{M,g}$  is in  $D_b$ .

*Proof:* We prove this theorem by induction on  $\alpha$ .

Clauses 1 and 2 are obvious due to the definitions of  $g$  and  $C$ .

Clause 3: If  $\alpha$  is in  $ME_a$  and  $a, b$  are intensional then  $[[\alpha]]^{M,g[x/u]}$  is in  $D_a$  for any  $x$  in  $D_b$  by induction. But  $[[\lambda u. \alpha]]^{M,g} = h$  in  $D_b$  such that for every  $x$  in  $D_b$ ,  $\mathbf{app}(h, x) = [[\alpha]]^{M,g[x/u]}$  which is in  $D_a$ . Hence  $[[\lambda u. \alpha]]^{M,g}$  is in  $D_b \rightarrow D_a$ .

The proof of clause 4 is as follows:

If  $\alpha$  is in  $ME_{\langle a,b \rangle}$  and  $\beta$  is in  $ME_a$  and  $a, b$  are intensional, then  $[[\mathbf{app}(\alpha, \beta)]]^{M,g} = \mathbf{app}([[\alpha]]^{M,g}, [[\beta]]^{M,g})$ , with the condition (from induction) that

$$[[\alpha]]^{M,g} \text{ is in } D_{\langle a,b \rangle} = D_a \rightarrow D_b \text{ and}$$

$$[[\beta]]^{M,g} \text{ is in } D_a.$$

Hence  $\mathbf{app}([[\alpha]]^{M,g}, [[\beta]]^{M,g})$  is in  $D_b$  by definition of  $D_a \rightarrow D_b$ .

The proof of clauses 5-12 are obvious from the logical schemas of the connectives.  $\square$

Now with our type theory  $T_{pol}$  and our typed domains inside  $F_0$ , let us see how Montague's approach could be accommodated. There are two routes one could follow for this purpose:

- (1) Interpret the PTQ fragment of Montague inside  $T_{pol}$ .
- (2) Interpret IL inside  $T_{pol}$ .

We shall describe in detail the first route in the next section and avoid commenting on the second route as we have seen in the previous chapter the difficulty of defining sense and reference inside the Frege structure using the equivalence relation. There may be other ways to do so but we shall not have anything to say about it here.

*PART C. Interpreting PTQ in  $T_{pol}$*

Now that we have the type theory and the semantics, let us interpret an extension of the PTQ fragment of Montague inside  $T_{pol}$ . This extension contains nominalisation, present tense, and deals better with intensionality. Consider the following syntactic categories, their translation types and basic expressions:

Syntactic Category	Category Definition	Translation Type	Basic expressions
e	primitive	e	None
p	primitive	p	None
T	p/IV	$\langle\langle e,p \rangle,p \rangle$	John,Mary,Bill,Wally,ninety,he <sub>0</sub> ,fun
IVInf	p//IV	$\langle\langle e,p \rangle,p \rangle$	None
IVGer	p///IV	$\langle\langle e,p \rangle,p \rangle$	None
IV	p/e	$\langle e,p \rangle$	run,walk,talk,rise,change
VP	p//e	$\langle e,p \rangle$	None
AP	p///e	$\langle e,p \rangle$	fun,nice
Tlp	-	$\langle p   \langle\langle e,p \rangle,p \rangle \rangle$	that
CN	p///e	$\langle e,p \rangle$	man,woman,park,fish,pen,unicorn,
-	-	-	centaur,woodchuck,hedgehog
TV	IV/e	$\langle e, \langle e,p \rangle \rangle$	find,lose,eat,love,seek,believe,assert
DET	T CN	$\langle\langle e,p \rangle   \langle\langle e,p \rangle,p \rangle \rangle$	every,the,a(n).
IVInfIV	-	$\langle\langle e,p \rangle   \langle\langle e,p \rangle,p \rangle \rangle$	to

IVGerIV	-	$\langle\langle e,p \rangle \langle\langle e,p \rangle, p \rangle\rangle$	ing
ele	-	$\langle e,e \rangle$	be

Note that here we are not dealing with prepositions. Note also that the words to, ing and that are expressions in the syntactic categories IVInflIV, IVGerIV and T|p which are introduced to deal with the nominalisation of verbs and sentences.

The domains

$$D_e = F_0, D_p = \text{PROP}, D_{IV} = \text{SET}, D_T = \text{SET} \rightarrow \text{PROP}, D_{CN} = \text{SET},$$

$$D_{p/p} = \text{PROP} \rightarrow \text{PROP}, \text{ and } D_{IV/p} = \text{PROP} \rightarrow \text{SET},$$

are all subsets of  $F_0$ . By contrast, if we had postulated the domains

$$D_{IV||IV} = \text{SET} \rightarrow \text{SET}, D_{DET} = \text{SET} \rightarrow (\text{SET} \rightarrow \text{PROP}),$$

then they would be empty. This is seen from the following theorem:

*Theorem 6:* The following domains are empty in a Frege structure where  $\text{PROP} \cap \text{SET}$  is empty:<sup>129</sup>

- (1)  $\text{SET} \rightarrow \text{SET}$ ,
- (2)  $\text{SET} \rightarrow (\text{SET} \rightarrow \text{PROP})$ ,
- (3)  $(\text{SET} \rightarrow \text{PROP}) \rightarrow \text{SET}$  and
- (4) Every type built recursively out of the above three types using  $\rightarrow$ .

Proof:

(1) is empty because if  $a$  is in  $\text{SET}$  then for every  $x$  in  $\text{SET}$ ,  $\text{app}(a, x)$  is in  $\text{SET}$ . But,  $\text{app}(a, x)$  is also in  $\text{PROP}$ , as  $a$  is a set. Hence  $\text{PROP} \cap \text{SET}$  is not empty. Contradiction.

(2) is empty because if  $a$  is in  $\text{SET} \rightarrow \text{PROP}$  and therefore in  $\text{SET}$ , then for every  $b$  in  $\text{SET}$ ,  $\text{app}(a, b)$  is in  $\text{SET}$ , but since  $a$  is in  $\text{SET}$ ,

<sup>129</sup> Perhaps here one can prove a more general theorem; that is:

$X \rightarrow Y$  is empty for  $X, Y \leq \text{SET}$ , where  $X \leq Y$  is defined inductively as follows:

- (i)  $X = \text{SET}$  or
- (ii)  $X = X' \rightarrow Y$  and  $X' \leq \text{SET}$ .

$\text{app}(a,b)$  is also in **PROP**. Hence  $\text{PROP} \cap \text{SET}$  is not empty. Contradiction.

(3) is empty because if  $a$  is in  $(\text{SET} \rightarrow \text{PROP})$  such that for every  $b$  in  $(\text{SET} \rightarrow \text{PROP})$ ,  $\text{app}(a,b)$  is in **SET**, then as  $a$  is in **SET**, we have that  $\text{app}(a,b)$  is in **PROP**. I.e.  $\text{app}(a,b)$  is in  $\text{SET} \cap \text{PROP}$ . Absurd.

(4) is obvious.  $\square$

This implies that verb phrase adverbs, and determiners should be given denotations outside  $F_0$ . They actually will be treated as functionals which operate over  $F_0$ . Transitive verbs however will be given denotations in  $F_0 \rightarrow \text{SET}$ .

Now we start by translating our basic expressions of PTQ into  $T_{\text{pol}}$  and see how the type-raising of various items could be accommodated here. We take John first. John translates to John' of type  $\langle \langle e,p \rangle, p \rangle$  where  $\text{John}' = \lambda u \langle e,p \rangle \text{app}(u,j)$ . Now we have to make sure that for any model  $M$  and assignment function  $g$ ,  $[[\text{John}']]^{M,g}$  is in the model. I.e. we have to show that there exists an element  $h$  of  $D_{\langle \langle e,p \rangle, p \rangle}$  such that for every  $x$  in  $D_{\langle e,p \rangle}$   $\text{app}(h,x) = [[\text{app}(u,j)]]^{M,g[x/u]} = \text{app}(x,j)$ . This is seen as follows:

For any object  $a$  in  $F_0$ , we are going to construct another object  $t(a)$  in  $\text{SET} \rightarrow \text{PROP}$  which also belongs to the structure such that for any set  $b$ ,  $\text{app}(t(a),b) = \text{app}(b,a)$ .

This is done as follows:

Assume  $a$ , and let  $f_a = \langle \text{app}(x,a)/x \rangle$ .

Take  $f'_a(x)$  to be the conditional proposition: If  $x$  is in **SET**, then  $f_a(x)$  else  $\perp$ .

$f'_a$  is a propositional function because:

If  $x$  is in **SET** then  $f'_a(x) = \text{app}(x,a)$  is in **PROP**,

else  $f'_a(x) = \perp$  is in **PROP**.

Now if we take  $t(a)$  to be  $\lambda f'_a$ , then  $t(a)$  is in  $\text{SET} \rightarrow \text{PROP}$

Proof:  $t(a)$  is obviously in **SET** because  $f'_a$  is a propositional function.

Moreover, for every  $x$  in **SET**,  $\text{app}(t(a),x)$  is in **PROP**.

$t(a)$  also has the property that  $\text{app}(t(a),b) = f'_a(b) = \text{app}(b,a)$  for any  $b$  in **SET**.

Now take  $[[\text{John}'] ]^{M,g}$  which is  $[[\lambda u. \text{app}(u, j) ] ]^{M,g}$  to be  $t ([[j] ]^{M,g})$ . We have to show that  $[[\text{John}'] ]^{M,g}$  is an element of  $D \langle \langle e, p \rangle, p \rangle$  and also that it satisfies the conditions of semantic clause 3 which defined  $[[\lambda u. \alpha ] ]^{M,g}$ . This is seen as follows:

$$\begin{aligned}
 & [[\text{John}'] ]^{M,g} = t ([[j] ]^{M,g}), \\
 & \text{app} ( t ([[j] ]^{M,g}), x ) = \text{app} ( x, [[j] ]^{M,g} ) \text{ for any } x \text{ in } \text{SET} \text{ by definition of } t. \\
 & [[\text{app}(u, j) ] ]^{M,g[x/u]} = \text{app} ([[u] ]^{M,g[x/u]}, [[j] ]^{M,g[x/u]}) = \\
 & \text{app} ( x, [[j] ]^{M,g[x/u]}) = \text{app} ( x, [[j] ]^{M,g} ). \text{ Hence } h \text{ of clause 3 is } t ([[j] ]^{M,g}).^{130}
 \end{aligned}$$

Of course we also translate Mary and Bill to Mary' and Bill' in a similar way to John', and follow the same procedure to build  $[[\text{Mary}'] ]^{M,g}$  and  $[[\text{Bill}'] ]^{M,g}$ .

Now we come to translate our basic expressions which belong to the syntactic category IV. We will take IV to be the category of untensed verb phrases. We start with run, which translates as run' of type  $\langle e, p \rangle$ . In the model, we assume that a primitive propositional function run belongs to  $F_1$  and that  $[[ \text{run}' ] ]^{M,g} = \lambda \text{run} = \text{run}'$ . Due to issues of type raising, which are a major concern of linguists, it is interesting to ask what is the semantic effect of type raising run'. If in  $T_{\text{pol}}$  we define run" to be  $\lambda u \langle \langle e, p \rangle, p \rangle \text{app}(u, \text{run}')$ , then what is  $[[\text{run}'' ] ]^{M,g}$  going to be?

run' =  $\lambda \text{run}$  is in SET, and we construct run" in  $((\text{SET} \rightarrow \text{PROP}) \rightarrow \text{PROP})$  such that  $\text{app}(\text{John}', \text{run}') = \text{app}(\text{run}'', \text{John}')$ , as follows:

we take  $f_{\text{run}'} = \langle \text{app}(x, \text{run}') / x \rangle$  and then consider  $f'_{\text{run}'}(x)$  to be: if  $x$  is in SET then  $f_{\text{run}'}(x)$  else  $\perp$ .<sup>131</sup>

run" =  $\lambda f'_{\text{run}'}$  is in SET and for every  $x$  in SET,  $\text{app}(\text{run}'', x)$  is in PROP.

It is easy to show that run" is in  $((\text{SET} \rightarrow \text{PROP}) \rightarrow \text{PROP})$  and that for every set  $x$ ,  $\text{app}(\text{run}'', x) = \text{app}(x, \text{run}')$ .

Before finishing with run we introduce its extension run# =  $[\lambda \text{run}]$  is in SET/[ ]

This same process is followed for the intransitive verbs walk, talk, rise and change.

<sup>130</sup> One still has to show that  $h$  is unique. For this one has to extend the notion of extensionality so that it applies to app inside each subtype.

<sup>131</sup> This is the same process we followed for John'.

It is now the turn of common nouns. We start with man, which translates into man' of type  $\langle e, p \rangle$ , and in the model we assume a primitive propositional function man and that  $[[\text{man}']]^{M, g} = \lambda \text{man}$ . We apply the same process for woman, park, fish, pen and unicorn. Also here, the process of constructing  $[[\text{man}']]^{M, g}$  is similar to that of constructing  $[[\text{run}']]^{M, g}$  above, where  $\text{man}'' = \lambda u. \text{app}(u, \text{man}')$ .

Now we come to transitive verbs. We start with find. We assume in our model the existence of a primitive binary propositional function find. find translates into  $\text{Tr}(\text{find})$  of type  $\langle e, \langle e, p \rangle \rangle$ , where  $\text{Tr}(\text{find}) = \lambda u^e \lambda v^e \text{app}_2(\text{find}', u, v)$ .  $[[\text{Tr}(\text{find})]]^{M, g}$  is in  $F_0 \rightarrow \text{SET}$ , because find is a binary propositional function in the Frege structure, hence,  $\lambda_0^2 \text{find}$  is in  $\text{SET}$  and it is also in  $F_0 \rightarrow \text{SET}$ . This is because for every  $x$  in  $F_0$ ,  $\text{app}(\lambda_0^2 \text{find}, x)$  is in  $\text{SET}$ , as for every  $y$  in  $F_0$ ,  $\text{app}(\text{app}(\lambda_0^2 \text{find}, x), y) = \text{app}_2(\lambda_0^2 \text{find}, x, y)$ . Now we can show that  $[[\text{Tr}(\text{find})]]^{M, g} = \lambda_0^2 \text{find}$  satisfies semantic clause 3.

In the same way we get the translations of lose, eat, love, seek. We will leave be for now and work through some examples.

We have already introduced the category IV of untensed verb phrases. Tense verb phrases are assigned to a distinct category VP. For some discussion, see [CH3].

*Tense Rule  $S_{tns}$ :*

If  $\alpha$  is in  $B_{IV}$ , then  $F_t(\alpha)$  is the present tense third person singular form of  $\alpha$  and is in  $ME_{VP}$ .

*Translation rule  $T_{tns}$ :*

If  $\alpha$  is in  $B_{IV}$  and  $\alpha$  translates to  $\alpha'$  then  $F_t(\alpha)$  translates to  $\lambda x \text{pred}(\alpha', x)$ .

Hence walk in  $B_{IV}$  translates as walk' and walks translates as  $\lambda x^e \text{pred}(\text{walk}', x)$ .

Next we need the following rule of functional application:

*Subject-predicate rule  $S_4$ :*

If  $\alpha$  is in  $P_T$  or in  $P_{IVInf}$  or in  $P_{IVGer}$  and  $\delta$  is in  $P_{VP}$  then  $F_4(\alpha, \delta)$  is in  $P_p$  where  $F_4(\alpha, \delta) = \alpha \delta$ .<sup>132</sup>

<sup>132</sup> Of course in a more serious fragment we would find a way to avoid this disjunction.



*Translation rule T<sub>4</sub>:*

If  $\alpha$  in  $P_T$  or in  $P_{IVInf}$  or in  $P_{IVGer}$  and  $\delta$  in  $P_{VP}$  and  $\alpha, \delta$  translate into  $\alpha', \delta'$  respectively, then  $F_4(\alpha, \delta)$  translates into  $\text{app}(\alpha', \delta')$ .

For example, John walks translates into  $\text{app}(\text{John}', \lambda \text{upred}(\text{walk}', u)) = \text{app}(\lambda \text{uapp}(u, j), \lambda \text{upred}(\text{walk}', u)) = \text{app}(\lambda \text{upred}(\text{walk}', u), j) = \text{pred}(\text{walk}', j)$  and

$[[\text{pred}(\text{walk}', j)]]^{M, g} = \text{pred}(\lambda \text{walk}, j)$ . In order to determine the reference of John

walks, we have  $[[^u \text{pred}(\text{walk}', j)]]^{M, g} = [[\text{pred}(\lambda \text{walk}, j)]]^{M, g}$

$= [\text{pred}(\lambda \text{walk}, j)] = [\text{app}(\lambda \text{walk}, j)]$ , as  $\text{app}(\lambda \text{walk}, j) \simeq \text{pred}(\lambda \text{walk}, j)$

$= \text{app}^e([\lambda \text{walk}], [j])$

$= \text{app}^e(\text{walk} \#, j)$

with the assumption that the representative of  $[j]$  is  $j$ .

If we wanted to look for  $[[^u \text{pred}(\text{walk}', j)]]^{M, g}$ , it is

$\text{Sen}([[^u \text{pred}(\text{walk}', j)]]^{M, g}) = \text{Sen}(\text{app}(\text{walk} \#, j))$  and we cannot go any further.

Now we give the following rule:

*Determiner-noun rule S<sub>2</sub>:*

If  $\delta$  is in  $P_{TICN}$  and  $\xi$  is in  $P_{CN}$ , then  $F_2(\delta, \xi)$  is in  $P_T$ , where  $F_2(\delta, \xi) = \delta' \xi$ ,

and  $\delta'$  is  $\delta$  except if  $\delta$  is  $a$  and the first word in  $\xi$  begins with a vowel, then  $\delta'$

is  $an$ .

*Translation rule T<sub>2</sub>:*

If  $\delta$  is in  $P_{TICN}$  and  $\xi$  is in  $P_{CN}$ , and  $\delta, \xi$  translate as  $\delta', \xi'$  respectively, then

$F_2(\delta, \xi)$  translates into  $\delta'(\xi')$ .

Before we illustrate with examples, we need to give the translations of the determiners. We start with every:

every translates to every' and

$[[\text{every}']]^{M, g}$  is that F-functional such that for any  $a$  in  $F_0$ ,

$[[\text{every}']]^{M, g}(a) = [[\lambda v \langle e, p \rangle \forall w^e (\text{app}(u, w) \rightarrow \text{app}(v, w))]]^{M, g}[u/a]$

which is an F-functional according to the explicit closure condition on a Frege

structure. Now  $[[\lambda v \langle e, p \rangle \forall w^e(\text{app}(u, w) \rightarrow \text{app}(v, w))]]^{M, g[u/x]}$  is an  $h_x$  in  $D \langle \langle e, p \rangle, p \rangle$  such that for every  $y$  in  $D \langle e, p \rangle$

$$\text{app}(h_x, y) = [[\forall w^e(\text{app}(u, w) \rightarrow \text{app}(v, w))]]^{M, g[u/x][v/y]} = \mathbf{V} \langle \text{app}(x, z) \rightarrow \text{app}(y, z) / z \rangle$$

Similarly,  $\underline{a}$  translates to  $\underline{a}'$  and

$[[a']]^{M, g}$  is that F-functional such that for any  $a$  in  $F_0$

$$[[a']]^{M, g}(a) = [[\lambda v \langle e, p \rangle w^e(\text{app}(u, w) \& \text{app}(v, w))]]^{M, g[u/x] / x}$$

Now we come to the, which translates to the' and

$[[the']]^{M, g}$  is that F-functional such that for any  $a$  in  $F_0$

$$[[the']]^{M, g}(a) = [[\lambda v \langle e, p \rangle w^e[\forall o^e(\text{app}(u, o) \equiv (o = w)) \& \text{app}(v, w)]]]^{M, g[u/a]}.$$

*Lemma 3:*  $[[every']]^{M, g}$ ,  $[[a']]^{M, g}$  and  $[[the']]^{M, g}$  are F-functionals in the Frege structure which when given elements in **SET** return elements in **SET**  $\rightarrow$  **PROP**.

Proof: That they are F-functionals is obvious.

Take  $a$  in **SET**,  $[[every']]^{M, g}(a) = [[\lambda v \langle e, p \rangle \forall w^e(\text{app}(u, w) \rightarrow \text{app}(v, w))]]^{M, g[u/a]}$

$= h_a$  in  $D \langle \langle e, p \rangle, p \rangle$  such that for every  $y$  in  $D \langle e, p \rangle$   $\text{app}(h_a, y) =$

$\mathbf{V}z(\text{app}(a, z) \rightarrow \text{app}(y, z))$ . If  $y$  is in  $D \langle e, p \rangle = \mathbf{SET}$  then  $\text{app}(y, z)$  is in **PROP** and as  $\text{app}(a, z)$  is in **PROP** then  $\mathbf{V}z(\text{app}(a, z) \rightarrow \text{app}(y, z))$  is in **PROP**. Hence,  $[[every']]^{M, g}$

takes elements in **SET** and returns elements in **SET**  $\rightarrow$  **PROP**. The same proof applies to  $[[a']]^{M, g}$  and  $[[the']]^{M, g}$ .  $\square$

For example: every man translates to

$$\text{every}'(\text{man}') = \lambda v \langle e, p \rangle \forall w^e(\text{app}(\text{man}', w) \rightarrow \text{app}(v, w)).$$

$[[every'(\text{man}')]]^{M, g} = [[every']]^{M, g}([[\text{man}']]^{M, g}) = h_{\text{man}}$  such that for every  $y$  in

$D \langle e, p \rangle$   $\text{app}(h_{\text{man}}, y) = \mathbf{V}z(\text{app}(\text{man}, z) \rightarrow \text{app}(y, z))$ .

every man talks translates as

$$\text{app}(\lambda v \langle e, p \rangle \forall w^e(\text{app}(\text{man}', w) \rightarrow \text{app}(v, w)), \lambda u \text{pred}(\text{talk}', u)) =$$

$$\mathbf{V} w^e(\text{app}(\text{man}', w) \rightarrow \text{app}(\lambda u \text{pred}(\text{talk}', u), w)) =$$

$$\mathbf{V} w^e(\text{app}(\text{man}', w) \rightarrow \text{pred}(\text{talk}', w))$$

$$[[every\ man\ talks]]^{M, g} = \mathbf{V}x(\text{man}(x) \rightarrow \text{pred}(\lambda \text{talk}(x))).$$

The treatment of conjunction is the same as with Montague, hence we omit discussion of it.

*Transitive verb rule S<sub>5</sub>:*

If  $\delta$  is in  $P_{TV}$  and  $\beta$  is in  $P_T$ , then  $F_5(\delta, \beta)$  is in  $P_{IV}$  where  $F_5(\delta, \beta) = \delta\beta$  if  $\beta$  does not have the form  $he_n$ , and  $F_5(\delta, he_n) = \delta him_n$ .

*Translation rule T<sub>5</sub>:*

If  $\delta$  is in  $P_{TV}$  and  $\beta$  is in  $P_T$ , and if  $\delta$  translates to  $\delta'$  and  $\beta$  translates to  $\beta'$  then  $F_5(\delta, \beta)$  translates to  $app(\delta', \beta')$ .

We illustrate here how transitive verbs combine with other constituents to result in sentences. For instance: Mary finds John. First, find John translates to:

$$app(\lambda u^e \lambda v^e app_2(\text{find}', u, v), \text{John}') = \lambda v^e app_2(\text{find}', v, \text{John}')$$

Mary finds John translates as:

$$\begin{aligned} & app(\text{Mary}', \lambda u^e pred(\lambda v^e app_2(\text{find}', v, \text{John}'), u)) = \\ & app(\lambda u^e pred(\lambda v^e app_2(\text{find}', v, \text{John}'), u), m) = \\ & pred(\lambda v^e app_2(\text{find}', v, \text{John}'), m) \equiv \\ & app_2(\text{find}', m, \text{John}') \end{aligned}$$

Mary finds John gets in the model the denotation:  $app_2(\text{find}, m, [[\text{John}']]^{M, g})$

Note that even though the formula  $Tr(\text{Mary finds John})$  was reduced to contain only m instead of Mary, this formula could not be reduced so that John' is replaced by j.

Following Montague, this is what will enable us to distinguish between de-re/de-dicto readings of sentences. Take the sentence Mary finds a unicorn.

a unicorn translates as :  $a'(\text{unicorn}') =$

$$\begin{aligned} & \lambda v \langle e, p \rangle ] w^e (app(\text{unicorn}', w) \& app(v, w)). \\ & \neq \lambda v \langle e, p \rangle ] w^e (app(\text{centaur}', w) \& app(v, w)) \text{ which is the translation of } \underline{a} \\ & \underline{\text{centaur}}. \end{aligned}$$

The extension of the above two expressions are empty sets for Montague. According to our approach, a unicorn walks entails there is a unicorn. This is because a unicorn

walks translates as

$$\text{app}(\lambda v \langle e, p \rangle ] w^e (\text{app}(\text{unicorn}', w) \& \text{app}(v, w)), \lambda u \text{pred}(\text{walk}', u)) = \\ ] w^e (\text{app}(\text{unicorn}', w) \& \text{app}(v, w)), \text{pred}(\text{walk}', u)).$$

We would like to make sure that this inference is blocked in the case of John seeks a unicorn, yet goes through in the case of John finds a unicorn.

Here is how this is done: We invoke Montague's meaning postulate (4), p163 of PTQ (in [TH2]). In our notation, Montague's MP(4) looks like:

$$(MP4) ] S \langle e, \langle e, p \rangle \rangle \forall x^e \forall y \langle \langle e, p \rangle, p \rangle ] \text{app}_2(\alpha, x, y) \equiv \text{app}(y, \text{app}(S, x)),$$

where  $\alpha \langle e, \langle e, p \rangle \rangle$  translates as Tr(find), Tr(eat), Tr(kiss), etc. And if  $\alpha$  is  $\text{Tr}(\beta)$  then S is written as  $\beta^*$ .<sup>133</sup>

This should combine with the translation of Mary finds John to yield the equivalent  $\text{app}(\text{app}(\text{find}^*, m), j) = \text{app}_2(\text{find}^*, m, j)$ .

We will still owe some explanation of the typing of the constituents of MP4.  $\text{app}(\alpha, x)$  is of type  $\langle e, p \rangle$  and hence can be applied to  $y$  of type  $\langle \langle e, p \rangle, p \rangle$  (which is also of type  $e$ ).  $\text{app}(\text{app}(\alpha, x), y) = \text{app}_2(\alpha, x, y)$  is of type  $p$ . Also,  $\text{app}(S, x)$  is of type  $\langle e, p \rangle$  and  $\text{app}(y, \text{app}(S, x))$  is of type  $p$ .

For the sake of uniformity, we treat nominalised verb phrases and sentences as having the same type as term phrases of type  $\langle \langle e, p \rangle, p \rangle$ . This has the consequence that they can occur in subject position without requiring any change to the type of tensed verb phrases.

We need the following rule:

*In fnitive rule  $S_{nomI}$ :*

$$\text{If } \alpha \text{ is in } ME_{IV}, \text{ then } F_{nom1}(\alpha) = \underline{\text{to}} \alpha \text{ is in } ME_{IVInf}$$

*Translation rule  $T_{nomI}$ :*

$$\text{If } \alpha \text{ is in } ME_{IV} \text{ and } \alpha \text{ translates as } \alpha' \text{ then } F_{nom1}(\alpha) \text{ translates as } \text{to}'(\alpha').$$

to translates as to'.

<sup>133</sup> Note that one still needs to show that there exists a model for (MP4).

Hence to run translates as:  $\lambda v \langle \langle e, p \rangle, p \rangle \text{app}(v, \text{run}')$ .

Now we deal with gerunds:

*Gerundive rule  $S_{nom2}$ :*

If  $\alpha$  is in  $B_{IV}$ , then  $F_{nom2}(\alpha) = \alpha \text{ing}$  is in  $ME_{IVGer}$ .

*Translation rule  $T_{nom2}$ :*

If  $\alpha$  is in  $B_{IV}$  and  $\alpha$  translates as  $\alpha'$  then  $F_{nom2}(\alpha)$  translates as  $\lambda v \langle \langle e, p \rangle, p \rangle \text{app}(v, \alpha')$ .

Now it seems that to and ing have the same effect. Yet the difference is that  $S_{nom2}$  applies to  $B_{IV}$  only, whereas  $S_{nom1}$  accepts any IV. For instance, one can say:

To run and talk is tiring, which is the same as To run and to talk is tiring.

Yet for gerunds we have only one way of saying it. That is:

Running and talking is tiring.

Now we come to self-application. Let's say that fun belongs to two categories, namely T and AP (adjective phrase), and that be belongs to category e/e.

If fun is in  $B_T$ , then  $\text{Tr}(\text{fun}) = \lambda x \langle \langle e, p \rangle \rangle \text{app}(x, \text{fun}')$ <sup>134</sup>

If be is in  $B_{e/e}$ , then  $\text{Tr}(\text{be}) = \lambda x^e x$

If fun is in  $B_{AP}$ ,  $\text{Tr}(\text{fun}) = \text{fun}'$  of type  $\langle e, p \rangle$ .

be fun in  $P_{IV}$  translates as  $\text{app}(\lambda x x, \text{fun}')$  of type  $\langle e, p \rangle$ .<sup>135</sup>

to be fun in  $P_{IVInf}$  translates  $\lambda x \langle \langle e, p \rangle \rangle \text{app}(x, \text{fun}')$

is fun in  $P_{VP}$  translates as  $\lambda z^e \text{pred}(\text{app}(\lambda x x, \text{fun}'), z) = \lambda z^e \text{pred}(\text{fun}', z)$  of type  $\langle e, p \rangle$ .<sup>136</sup>

to be fun is fun translates as:

$$\text{app}(\lambda x \langle \langle e, p \rangle, p \rangle \text{app}(x, \text{fun}'), \lambda z^e \text{pred}(\text{fun}', z)) = \\ \text{app}(\lambda z \text{pred}(\text{fun}', z), \text{fun}') =$$

<sup>134</sup> This is acceptable because  $\text{fun}'$  is in  $\langle e, p \rangle$ , and thus  $\text{fun}'$  is in e according to B1,16.

<sup>135</sup> Note that we could apply be to fun of type  $\langle e, p \rangle$  as fun is also of type e. The result of  $\text{app}(\lambda x x, \text{fun}')$  is  $\text{fun}'$  of type  $\langle e, p \rangle$ .

<sup>136</sup> Note that we are dealing with tensed and untensed verbs. Basic untensed verb phrases (e.g. to be happy) and Gerundive (untensed) verb phrases (e.g. being happy). The tensed verb phrases are: is happy, are happy etc..

pred(fun',fun').

As it is seen from above, we have succeeded in getting nominalisation and self-application to work. Also, being fun and fun (as a member of  $P_T$ ) get the same translation as to be fun, we get the same translation when they occur as subjects of the predication is fun.

Now we come to the translation of that:

*Sentence nominalisation  $S_{that}$ :*

If  $\alpha$  is in  $ME_p$  then  $F_{that}(\alpha) = \text{that } \alpha \text{ is in } ME_{\langle e,p \rangle, p}$ .

*Translation rule  $T_{that}$ :*

If  $\alpha$  is in  $ME_p$  and  $\alpha$  translates as  $\alpha'$  and that translates as that' then that  $\alpha$  translates as that' ( $\alpha'$ ).

That translates as the functional that'.

Note that since  $p \subseteq e$ , we do not have any problem with typing. For instance, John runs translates as  $\text{pred}(\text{run}',j)$  and that John runs translates as:  $\lambda v^{\langle e,p \rangle} \text{app}(v, \text{pred}(\text{run}',j))$ .

Now,  $\text{pred}(\text{run}',j)$  is of type  $p$ , hence it is of type  $e$  and so we can apply  $v$  of type  $\langle e,p \rangle$  to  $\text{pred}(\text{run}',j)$ .

Mary believes that John runs translates as

$$\begin{aligned} & \text{app}_2(\text{believe}',m, \lambda v^{\langle e,p \rangle} \text{app}(v, \text{pred}(\text{run}',j))) = \\ & \text{app}(\text{app}(\text{believe}',m), \lambda v^{\langle e,p \rangle} \text{app}(v, \text{pred}(\text{run}',j))) = \text{by (MP4)} \\ & \text{app}(\text{app}(\text{believe}',m), \text{pred}(\text{run}',j)) = \\ & \text{app}_2(\text{believe}',m, \text{pred}(\text{run}',j)). \end{aligned}$$

The following section illustrates few more examples of translating English sentences into  $T_{pol}$ .

woodchuck translates as woodchuck' of type  $\langle e,p \rangle$ ,

a woodchuck translates as

$$\lambda v^{\langle e,p \rangle} ]x[\text{app}(\text{woodchuck}',x) \& \text{app}(p,x)] \text{ of type } \langle \langle e,p \rangle, p \rangle \text{ or}$$

be a woodchuck translates as

$$\lambda v \langle e, p \rangle ]_x [\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)] \text{ of type } \langle \langle e, p \rangle, p \rangle$$

is a woodchuck translates as

$$\lambda z^e \text{pred}(\lambda v \langle e, p \rangle ]_x [\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)], z) \text{ of type } \langle \langle e, p \rangle, p \rangle$$

wally translates to wally' of type  $\langle \langle e, p \rangle, p \rangle$ .

wally is a woodchuck translates as

$$\text{pred}(\lambda v \langle e, p \rangle ]_x [\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)], w')$$

that wally is a woodchuck translates as

$$\lambda u \langle e, p \rangle \text{app}(u, \text{pred}(\lambda v \langle e, p \rangle ]_x [\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)], w'))$$

John believes that wally is a woodchuck translates as

$$\text{app}_2(\text{believe}', j, \lambda u \langle e, p \rangle \text{app}(u, \text{pred}(\lambda v \langle e, p \rangle ]_x [\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)], w'))) \equiv \\ \text{app}_2(\text{believe}', j, \text{pred}(\lambda v \langle e, p \rangle ]_x [\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)], w')).$$

Mary finds a unicorn translates as

$$\text{pred}(\lambda u^e \text{app}_2(\text{find}', u, \lambda v \langle e, p \rangle ]_w^e (\text{app}(\text{unicorn}', w) \& \text{app}(v, w))), m) \equiv \\ \text{app}_2(\text{find}', m, \lambda v \langle e, p \rangle ]_w^e (\text{app}(\text{unicorn}', w) \& \text{app}(v, w))) \equiv \text{by (MP4)} \\ \text{app}(\lambda v \langle e, p \rangle ]_w^e (\text{app}(\text{unicorn}', w) \& \text{app}(v, w)), \text{app}(\text{find}', m)) = \\ ]_w^e (\text{app}(\text{unicorn}', w) \& \text{app}(\text{app}(\text{find}', m), w)) = \\ ]_w^e (\text{app}(\text{unicorn}', w) \& \text{app}_2(\text{find}', m, w)).$$

Now we come to discuss why believe is of type  $\langle e, \langle e, p \rangle \rangle$ . This is to enable us to deal with all the following sentences:

- (1) Mary believes John.
- (2) Mary believes John runs.
- (3) Mary believes that John runs.

Both John and that John runs are of type  $ME_T$ , hence the typing of believe works for those two cases. believe also works for John runs because the type of the latter is  $p$  and hence it is  $e$ .

*Validity of Leibniz' law:* We cannot deduce that if

$\text{app}_2(\text{believe}', j, \text{pred}(\lambda v \langle e, p \rangle \}x[\text{app}(\text{woodchuck}', x) \& \text{app}(p, x)], w'))$ , then

$\text{app}_2(\text{believe}', j, \text{pred}(\lambda v \langle e, p \rangle \}x[\text{app}(\text{groundhog}', x) \& \text{app}(p, x)], w'))$ .

This is because  $\text{app}(\text{groundhog}', x)$  is not equal to  $\text{app}(\text{woodchuck}', x)$ , even though they both have the same truth value. Hence we haven't invalidated Leibniz' law.<sup>137</sup>

---

<sup>137</sup> We have omitted discussion of the validity of existential generalisation out of opaque contexts. That is, we would like to see whether from John believes  $f(a)$ , we can deduce that there exists an  $x$  such that  $x=a$  and John believes  $f(x)$ . In other words:

$J \text{ believes } f(a) \vdash x=a \ \& \ \}x(J \text{ believes } f(x))$ .

We shall say something about this in future work.



*CHAPTER 7. SUMMARY AND COMPARISON*

We have provided in this thesis a new approach to semantics using Frege structures. The first chapter explained two problems of nominalisation which, along with intensionality, provided the motivation for the thesis.

The oscillation of previous accounts between restricted comprehension and many valued logics, and the problems concealed by this oscillation (full quantification and property existence problems), awakened considerable curiosity as to what properties we could have: what should we quantify over and what should be nominalised? Similarly the large number of attempts at solving the problems of the propositional attitudes aroused curiosity as to how much intensionality one should have. How could intensionality be accommodated and how could extensionality be restored from such intensionality if the latter was to be considered basic? Finally the interesting idea of accommodating types (intensional and extensional ones) in this type free highly intensional language could not be missed.

There are many other interesting topics that could be accommodated within the present framework - topics I intend to address in future work. Among them are the study of temporal logics in Frege structures. It certainly seems that accommodating temporal logics based on events (such as Kamp's in [KA1]) is most convenient in a framework using Frege structures. Also it is straightforward to implement other systems of time (such as McDermott's work in [MC3]) in Frege structures if one considers a collection of Frege structures, each of them corresponding to a point in time. However, I do not have any comments on how to implement a theory of times based on intervals (such as Allen's in [AL2] and [AL3]) as I am not sure how this could be done as even though points could be gathered into sets, we know that those sets may not be defined.

Leaving temporal logics and coming to non-monotonic logics, Frege structures

from the point of view of self-reference seem to have some advantages over other non-monotonic approaches which have to refer to metalanguages to deal with self-referentiality. All these fascinating areas could be studied within the present framework and we shall do so in future work. For now however, let us end this thesis by showing the difference between the treatment of quantifiers in Scott domains and Frege structures and finally summarizing a few advantages of the present framework.

### *PART A. QUANTIFICATION*

Both semanticists and computer scientists share an interest in quantification. I have referred to the topic quite often throughout this thesis and would now like to show the foundational difference between quantification in a semantics using Frege structures and that in a semantics using Scott domains. This point is a major issue for those interested in the semantics of either computer or natural languages and who base their work on Scott domains. The quantification problem that faced Turner (in [TU2]) can be described as follows: Assume a language which has both objects and functions and assume that wffs are built out of other ones using  $\&$ ,  $\forall$ ,  $\exists$ ,  $\rightarrow$ , ... If the model is a Scott domain  $E_\infty$  then there is no problem interpreting anything which is not a quantified sentence, as the interpretations of all such things are continuous functions and hence belong to the model. Let us choose the following interpretation for the quantifiers  $\forall$  and  $\exists$

$$\begin{aligned} [[\forall x\phi]]_{\text{gwt}} = & \\ & 1 \text{ if for each } d \text{ in } D, [[\phi]]_{\text{g}[d/x]\text{wt}} = 1 \\ & 0 \text{ if for some } d \text{ in } D, [[\phi]]_{\text{g}[d/x]\text{wt}} = 0 \\ & \perp \text{ otherwise} \end{aligned}$$

$$\begin{aligned} [[\exists x\phi]]_{\text{gwt}} = & \\ & 1 \text{ if for some } d \text{ in } D, [[\phi]]_{\text{g}[d/x]\text{wt}} = 1 \end{aligned}$$

0 if for each  $d$  in  $D$ ,  $[[\phi]]_{g[d/x]wt} = 0$

1 otherwise.

Then the following is a proof of the continuity of the quantifier clause for  $\forall$ . Assume by induction that we have  $[[\phi]]$  is continuous where  $\phi$  does not involve quantifiers. To prove the continuity of  $[[\forall x\phi]]$  (i.e. to prove it in  $[ASG \rightarrow S \rightarrow EXT]$  where ASG is the collection of assignment functions, S is the collection of states consisting of worlds and times and EXT is the extensional domain of values), we prove it continuous separately in each of its arguments, according to a theorem we proved in appendix I. Let us prove the continuity of  $[[\forall x\phi]]$  for  $g$  in ASG. Take an  $\omega$ -sequence  $(g_n)_n$  and prove that:

$$[[\forall x\phi]]_{\cup g_n wt} = \cup [[\forall x\phi]]_{g_n wt}$$

$$\text{Assume } [[\forall x\phi]]_{\cup g_n wt} = 0 \iff \text{by definition,}$$

$$(\exists d \in D)([[\phi]]_{\cup g_n [d/x]wt} = 0) \iff \text{by induction,}$$

$$(\exists d \in D)(\cup [[\phi]]_{g_n [d/x]wt} = 0) \iff \text{by the structure of BOOL,}$$

$$(\exists d \in D)(\exists n \in \omega)([[\phi]]_{g_n [d/x]wt} = 0) \iff \text{by logical laws,}$$

$$(\exists n \in \omega)(\exists d \in D)([[\phi]]_{g_n [d/x]wt} = 0) \iff \text{by definition,}$$

$$(\exists n \in \omega)([[\forall x\phi]]_{g_n wt} = 0) \iff \text{by the structure of BOOL,}$$

$$\cup [[\forall x\phi]]_{g_n wt} = 0.$$

$$\text{Assume } [[\forall x\phi]]_{\cup g_n wt} = 1 \iff \text{by definition,}$$

$$(\forall d \in D)([[\phi]]_{\cup g_n [d/x]wt} = 1) \iff \text{by induction,}$$

$$(\forall d \in D)(\cup [[\phi]]_{g_n [d/x]wt} = 1) \iff \text{by the structure of BOOL,}$$

$$(\forall d \in D)(\exists n \in \omega)([[\phi]]_{g_n [d/x]wt} = 1) \iff \text{u} \subseteq d \text{ and monotonicity,}^{138}$$

$$(\exists n \in \omega)([[\phi]]_{g_n [u/x]wt} = 1) \iff \text{monotonicity,}$$

$$(\exists n \in \omega)(\forall d \in D)([[\phi]]_{g_n [d/x]wt} = 1) \iff \text{by definition,}$$

$$(\exists n \in \omega)([[\forall x\phi]]_{g_n wt} = 1) \iff \text{by the structure of BOOL,}$$

$$\cup [[\forall x\phi]]_{g_n wt} = 1.$$

<sup>138</sup> u is the undefined.

Therefore  $[[\forall x\phi]]$  is continuous.  $\square$

Note that this interpretation of quantifiers is abandoned later by Turner (in [TU4] and [TU5]) and he decided to adopt the following clauses instead:

$$\begin{aligned} [[\forall x\phi]]_{\text{gwt}} = & \\ & 1 \text{ iff for each } d \in E_{\infty}^- \cup E_n, [[\phi]]_{\text{g}[d/x]\text{wt}} = 1 \\ & 0 \text{ iff for some } d \in E_{\infty}^- \cup E_n, [[\phi]]_{\text{g}[d/x]\text{wt}} = 0 \\ & \perp \text{ otherwise} \end{aligned}$$

$$\begin{aligned} [[\exists x\phi]]_{\text{gwt}} = & \\ & 1 \text{ iff for some } a \in E_{\infty}^- \cup E_n, [[\phi]]_{\text{g}[a/x]\text{wt}} = 1 \\ & 0 \text{ iff for each } a \in E_{\infty}^- \cup E_n, [[\phi]]_{\text{g}[a/x]\text{wt}} = 0 \\ & \perp \text{ otherwise} \end{aligned}$$

Of course working with Scott domains, you have always to check for continuity and this is the case with the new clauses. It can easily be proved that continuity does in fact hold and so we can still think of Scott domains as models.

We now describe the problem which made Turner move from the first definition of quantifiers to the second one. By adopting the first definition, we had:

$$[[\forall x\phi]]_{\text{gwt}} = 1 \text{ iff } (\forall d \text{ in } D)([[\phi]]_{\text{g}[d/x]\text{wt}} = 1)$$

As  $[[\phi]]$  is continuous, therefore monotonic and as  $u \subseteq d$  (where, as noted above,  $u$  is the undefined) for each  $d$  in  $D$  then we get:

$$(\forall d \text{ in } D)([[\phi]]_{\text{g}[d/x]\text{wt}} = 1) \text{ iff } [[\phi]]_{\text{g}[u/x]\text{wt}} = 1.$$

This clause has serious consequences. I shall illustrate this by taking in the formal language an element  $u'$  which names  $u$ . I.e.  $[[u']]_{\text{gwt}} = u$  always. Now see what happens if we take  $\phi$  to be:  $x=u'$ . Applying the above clause we get:

$$[[x=u']]_{\text{g}[u/x]\text{wt}} = 1 \text{ iff } (\forall d \text{ in } D)([[x=u']]_{\text{g}[d/x]\text{wt}} = 1)$$

which implies:

$$u=u \text{ iff } (\forall d \in D)(d=u).$$

That is absurd. We have to do something about this and the first solution that one thinks of is to exclude the undefined element from the quantifier clause. Therefore, instead of letting  $d$  range over all of  $D$ , we let it range over  $D^*$  (i.e.  $D - \{u\}$ ). But now Scott domains can no longer be models under this interpretation, for we no longer have  $[[\forall x\phi]]$  is continuous. If we go back to the proof of continuity given above, we see that we had to use the undefined element in order to prove continuity. Turner, realising this, exploits an important aspect of the structure of Scott domains. We explained in Chapter 1 the existence of finite and infinite elements in  $E_\infty$  and said that for each element  $d$  of  $E_\infty$   $d$  is the limit of  $(e_n)_n$  where  $e_n$  belongs to  $E_n$  and each  $E_n$  is the domain of finite elements. The infinite (or ideal elements) are those which are in  $E_\infty - \cup E_n$ . By restricting the quantification over these ideal elements only, we can prove again the closure of Scott models. Although Turner's trick is very clever, unfortunately it does not work. By so restricting quantification, only infinite elements can be quantified over and finite elements are ignored. This is unsatisfactory as is illustrated in the following example:

Take the sequence given by Turner,

(i) John is fun

(ii) John runs

(iii) To run is fun.

We can agree with Turner that is fun and runs should both be infinite elements, to be able to apply them to everything in the domain especially their nominals. But if we take  $\phi$  to be  $x$  is fun and apply Turner's quantifier clause for  $\forall$  where  $[[\forall x\phi]] = 1$ , we get: for each ideal element  $d$ ,  $\text{fun}'(d)$  which means that we can only quantify over those ideal elements. But John, Mary, one's Table and so on are finite elements - how can we quantify over them? In [TU3], Turner makes the domain of individuals  $\underline{A}$  a basic domain and builds all other domains out of superclasses of  $\underline{A}$ . This restriction of quantification to the ideal elements makes us lose the power of ascribing properties to

our individuals and prevents quantifiers like every from having any of their usual natural language interpretation.

Having seen that this solution to quantification in Scott domains is unsatisfactory, why is it that Frege structures do not have this problem, even though they themselves can be built on top of a Scott domain? It is because Scott domains themselves do not have any logic on them. Turner tried to incorporate a logic (and in consequence an interpretation of all the connectives) by attaching Kleene's three-valued logic to a Scott domain. In a Frege structure on the other hand quantifiers and other connectives are built inductively step by step so that at the fixed point one gets all these logical constants.

#### *PART B. FURTHER ADVANTAGES*

Now we assess further the advantages one obtains with Frege structures. We start with type freeness and the fact that **SET** is isomorphic to Propositional functions  $F_0 \rightarrow \text{PROP}$  and that  $\text{SET} \subseteq F_0$ . Also, we have the two following functionals in a Scott domain:

$$||_1 : \text{SET} \rightarrow \text{PF1}$$

$$\lambda : \text{PF1} \rightarrow \text{SET}.$$

If we assume that the interpretation of verbs takes place in  $F_i$  for  $i \geq 1$  and thus that  $[[\text{walk}]]$  is in  $F_1$ , then we get:

$$[[\text{to walk}]]_g = \lambda. [[\text{walk}]]_g$$

Now it is straightforward to interpret things like to walk hurts, for:

$$\begin{aligned} [[\text{to walk hurts}]]_g &= [[\text{hurt}]]_g ([[ \text{to walk} ] ]_g) \\ &= [[\text{hurt}]]_g (\lambda. [[\text{walk}]]_g). \end{aligned}$$

The advantage of what we just offered lies in the elegance of classifying the denotation of our items. With Montague's and Turner's approaches, one has always to check whether the denotation of an item is in the right domain. With our approach,

we do not need to check whether  $[[\text{to walk}]]_g$  is in  $F_0$  or not using some confusing domain equations. All we had to say was that  $[[\text{walk}]]_g$  is in  $F_1$ ; therefore  $\lambda[[\text{walk}]]_g$  is in  $F_0$ . This actually seems to be an encouraging advantage about Frege structures: nominalisation is a natural process inside the Frege structure. It also seems that we have real application, unlike in Scott domains where application is only through the isomorphic embedding. This is because instead of interpreting things as above into  $F_i$ , for  $i \geq 0$ , we can restrict everything to  $F_0$ . We did this in the previous chapter and obtained that

$$[[\text{fun is fun}]]_g = \text{pred} ([[fun]]_g, [[fun]]_g)$$

Therefore it seems that by using Frege structures we get the following advantages over Scott domains,

- (1) Real self application
- (2) No cumbersome checking for the right typing
- (3) No redundant semantic types
- (4) Nominalisation seems to flow naturally
- (5) Quantification

For the sake of completeness, we mention a new approach to a theory of properties proposed by Turner (in [TU9]) which abandons completely the use of Scott domains. Turner's new theory is one which starts from Frege's comprehension principle and restricts it in such a way that the paradox is no longer derivable. Turner starts with a first order theory which has a pairing system and adds to this theory a new operator  $p$  (to serve as the predication operator) together with the lambda operator. Then in this case, if one assumes full classical logic and Frege's comprehension principle, one will certainly derive the paradox;

for, take  $a = \lambda x. \neg p(x, x)$ ,

$p(a, a) \leftrightarrow \neg p(x, x)[a/x]$

$\leftrightarrow \neg p(a, a)$ . Contradiction.

Of course, the problem does not come from contraction, i.e.  $p(\lambda x.A, t) \rightarrow A(t, x)$  is always true. But the converse implication (i.e. expansion) is problematic. This is due to negation, i.e. if  $A$  is atomic then we can accept  $A(t, x) \rightarrow p(\lambda x.A, t)$ . But we cannot accept it when  $A$  is like Russell's property, an atomic term preceded by a negation sign. This is exactly what guides Turner in setting his theory. For the theory now will have the following axioms replacing Frege's comprehension principle:

(E1)  $A(t, x) \rightarrow p(\lambda x.A, t)$  when  $A$  is atomic.

(R)  $p(\lambda x.A, t) \rightarrow A(t, x)$ .

(I)  $p(\lambda x.p(\lambda y.A, t), u) \leftrightarrow p(\lambda y.p(\lambda x.A, u), t)$

Now the abandonment of Frege's full comprehension axiom will impose the use of two logics, one inside the predication operator in addition to the usual one for wffs. This is due to the fact that breaking the equivalence between  $p(\lambda x.A, t)$  and  $A(t, x)$  will disconnect the reasoning about wffs and properties. To build models for  $T$  above, one uses the fixed point operator to turn an ordinary model of the first order theory into a model which will validate in it as many instances of the comprehension axiom as possible. It will of course validate only the safe instances whereas the paradoxical ones will oscillate in truth-values. The inductive step to build the model should be obvious. As an example, one can start with the first order model, and an operator  $PI$  which is empty at the beginning. Then at the next step, extend  $PI$  to also contain the pairs  $\langle [[\lambda x.A]], [[t]]_{gM} \rangle$  such that  $[[A]]_{g[[t]]_g/x} = 1$  and so on until one gets a limit ordinal  $\xi$  where  $PI$  then is to have in it all the pairs  $\langle e, d \rangle$  such that for some ordinal smaller than this  $\xi$ ,  $\langle e, d \rangle$  belongs to all the intermediate  $PI$ 's. Now we no longer have a full comprehension principle and we cannot do with properties what we can do with formulae. But there are still a great deal of things that one can identify between properties and wffs; for example, from  $P(\lambda x.A, t)$  and  $P(\lambda x.B, t)$  one can derive  $p(\lambda x.A \& B, t)$ . Turner showed however that theories of Frege structures are weaker than his theory of properties which is a fact that may stand to our advantage for the



following reasons. Firstly, Turner can prove at least as much in his theory as one can in a theory based on Frege structures. Secondly, Turner is paying a price for the strength of his theory - mainly his use of two logics (internal and external) rather than one only. On balance it seems better to use a theory based on Frege structures for properties. Doing so gains the advantages of Turner without the complications.

APPENDIX I. FOUNDATIONS OF DOMAIN THEORY

In the main chapters of the thesis, I assumed that the reader knew what models of the lambda calculus were and that he was able to build them. In this appendix, I show how these models can be built and cite important properties about them. My introduction to those models is not definitive however, and the interested reader is referred to [BA1] and [BA2].

*I. Definitions:*

*Def1: semantic domains:* A domain  $D$  with a binary relation  $\subseteq$  on  $D$  is a semantic domain iff:

(i)  $D$  is a set which has a bottom element  $u$  satisfying:

$$(\forall x \in D) [u \subseteq x].$$

(ii) The binary relation  $\subseteq$  is a partial ordering on  $D$ . I.e:

(1)  $\subseteq$  is reflexive:  $(\forall x \in D) [x \subseteq x]$

(2)  $\subseteq$  is antisymmetric:

$$(\forall x \in D) (\forall x' \in D) [(x \subseteq x' \text{ and } x' \subseteq x) \implies x = x']$$

(3)  $\subseteq$  is transitive:

$$(\forall x, y, z \in D) [(x \subseteq y \text{ and } y \subseteq z) \implies x \subseteq z].$$

(iii) every  $\omega$ -sequence has a least upper bound in  $D$ . (see Def2,3,4)

We denote the least upper bound of  $(x_n)_{n \in \omega}$  by  $\bigcup_{n \in \omega} x_n$

and when no confusion occurs, we write  $\bigcup x_n$ .

We denote a semantic domain by  $(D, \subseteq)$ .

*Def2:  $\omega$ -sequences:* An  $\omega$ -sequence in a semantic domain  $(D, \subseteq)$  is a sequence  $(x_n)_{n \in \omega}$  of elements of  $D$  such that  $(\forall n \geq 0) [x_n \subseteq x_{n+1}]$ . When no confusion occurs, we write  $(x_n)_n$ .

*Def3: Upper bound:* An element  $d$  in  $D$  is an upper bound of a subset  $X$  of  $D$ , iff

$$(\forall d' \in X) [d' \subseteq d].$$

*Def4: Least upper bound:* An element  $d$  in  $D$  is the least upper bound of a subset  $X$  of  $D$  iff

- (1)  $d$  is an upper bound of  $X$
- (2)  $(\forall d' \in D) [(d' \text{ is an upper bound of } X) \implies d \subseteq d']$

*Def5: Continuous functions:* A function  $f$  from a semantic domain  $D$  into another semantic domain  $D'$  is continuous iff

$$(\text{for each } \omega\text{-sequence } (d_n)_n \in D) [f(\cup d_n) = \cup f(d_n)]$$

Hereafter,  $D^*$  will denote the domain  $D$  without its bottom element. I.e.  $D^* = D - \{u\}$ .

*II. Domains out of other domains:* Now that we have the notion of a domain, we need to do useful things with it and for that we should be able to build domains out of other domains. These constructions will be based upon three functors:

*II.1. Domains out of old ones using '+':*

*II.1.1. Sum of two domains:* Let  $(D_1, \subseteq_1)$  and  $(D_2, \subseteq_2)$  be two semantic domains. We define  $(D_1 + D_2, \subseteq)$  to be:

$$D_1 + D_2 = \{(d_i, i) \text{ such that } d_i \in D_i^*\} \cup \{u\}^{139}$$

$$\text{and } (\forall d = (d_i, i), d' = (d'_j, j) \in D_1 + D_2) [d \subseteq d' \iff (d=u \text{ or } (i=j \text{ and } d_i \subseteq_1 d'_j))]$$

*Lemma:*  $(D_1 + D_2, \subseteq)$  as constructed above is a semantic domain:

*Proof:*

(i)  $u$  is the bottom element because:

$$(\forall d \in D_1 + D_2) [u \subseteq d] \text{ following the definition of } \subseteq.$$

(ii)  $\subseteq$  is a partial order on  $D_1 + D_2$ :

(1)  $\subseteq$  is reflexive:

$$\text{Let } d \in D_1 + D_2.$$

$$\text{If } d = u \text{ then } d \subseteq d.$$

Otherwise  $d = (d_i, i)$  where  $d_i \in D_i^*$ . But  $\subseteq_1$  is reflexive,

$$\text{then } d_i \subseteq_1 d_i. \text{ Hence, } d \subseteq d.$$

---

<sup>139</sup>  $= (u \in D_1 \cup D_2)$

(2)  $\subseteq$  is antisymmetric:

Let  $d, d' \in D_1 + D_2$  such that  $d \subseteq d'$  and  $d' \subseteq d$ .

If  $d = u$  then also  $d' = u$  and so  $d = d'$ .

Otherwise  $d = (d_i, i)$  and  $d' = (d'_j, j)$ ,

$d \subseteq d' \implies i = j$  and  $d_i \subseteq_1 d'_j$

$d' \subseteq d \implies d'_j \subseteq_1 d_i$

but  $\subseteq_1$  is antisymmetric, then  $d_i = d'_j$  and therefore  $d = d'$ .

(3)  $\subseteq$  is transitive:

Let  $d, d', d'' \in D_1 + D_2$  such that  $d \subseteq d'$  and  $d' \subseteq d''$ .

If  $d = u$  then  $d \subseteq d''$  (definition of  $\subseteq$ )

Otherwise  $d = (d_i, i)$  where  $d_i \in D_1^*$

$d \subseteq d' \implies d' = (d'_j, j)$  where  $d'_j \in D_2^*$ ,  $i = j$  and  $d_i \subseteq_1 d'_j$

$d' \subseteq d'' \implies d'' = (d''_k, k)$  where  $d''_k \in D_2^*$ ,  $j = k$  and  $d'_j \subseteq_1 d''_k$ ,

therefore  $i = j = k$ ,  $d_i \subseteq_1 d'_j$  and  $d'_j \subseteq_1 d''_k$ .

But  $\subseteq_1$  is transitive hence  $i = k$  and  $d_i \subseteq_1 d''_k$ ,

by definition, this is:  $d \subseteq d''$ .

Hence  $\subseteq$  is transitive.

Combining (1), (2) and (3), we get that  $\subseteq$  is a partial order.

(iii) Every sequence of  $D_1 + D_2$  has a least upper bound in  $D_1 + D_2$ :

Let  $(d_n)_{n \in \omega}$  be an  $\omega$ -sequence in  $D_1 + D_2$ .

If the  $(d_n)_{n \in \omega}$  is of the form:  $u \subseteq u \dots \subseteq u \dots$ , then  $u$  is the limit.

Otherwise, there is a certain natural number  $k$  such that the elements of indices  $\geq k$  are  $\neq u$  and

$(\forall n \geq k) [d_n \in D_1^*]$  or  $(\forall n \geq k) [d_n \in D_2^*]$ . This is because:

1. If  $d_k \in D_1^*$  then so is  $d_{k+1}$  (as  $d_k \subseteq d_{k+1}$ )

by induction assume that  $d_n \in D_1^*$ . As  $d_n \subseteq d_{n+1}$  we get

$d_{n+1} \in D_1^*$ . So  $(\forall n \geq k) [d_n \in D_1^*]$

2. If  $d_k \in D_2^*$ , we prove as above that  $(\forall n \geq k) [d_n \in D_2^*]$ .

(a) If  $(\forall n \geq k) [d_n \in D_1^*]$  then:

$(d_n)_{n \geq k}$  is an  $\omega$ -sequence of  $D_1$  and so it has a limit in  $D_1$ .

Actually it does not matter if we start at  $k$  instead of  $0$ , because we can always stick bottom to the first  $k$  places of the sequence.

Let  $d$  be the limit of  $(d_n)_{n \geq k}$  in  $D_1$  then,

$d$  is the limit of  $(d_n)_{n \in \omega}$  in  $D_1 + D_2$ . To see this:

(1)  $d$  is an upper bound in  $D_1 + D_2$ :

$$\forall n, \text{ if } 0 \leq n < k \implies d_n = u$$

$$\implies d_n \subseteq d \text{ (by definition of } \subseteq \text{)}$$

$$\text{if } n \geq k \implies d_n \subseteq_1 d \implies d_n \subseteq d \text{ (definition of } \subseteq \text{)}$$

So  $(\forall n \in \omega) [d_n \subseteq d]$ .

(2)  $d$  is the least upper bound of  $(d_n)_n$  in  $D_1 + D_2$ .

If  $d'$  is an upper bound of  $(d_n)_n \in D_1 + D_2$  then:

$d'$  is an upper bound of  $(d_n)_{n \geq k}$  because

$$\forall n \geq k \ d_n \subseteq d' \text{ and } d' \in D_1^* \implies d_n \subseteq_1 d'$$

$d \subseteq d' \implies (\forall n \geq k) [d_n \subseteq_1 d']$ . But  $d$  is the least upper

bound in  $D_1 \implies d \subseteq_1 d'$ .

Therefore  $d$  is the least upper bound of  $(d_n)_n$  in  $D_1 + D_2$ .

(b) If  $(\forall n \geq k) [d_n \in D_2^*]$  the proof goes as in (a)

Hence (iii)

From (i), (ii) and (iii) we conclude that  $D_1 + D_2$  is a semantic domain.  $\square$

II.1.2. Sum of any number (possibly infinite) of domains: Let  $(D_i)_{i \in I}$  be a set of semantic domains with the ordering

$\subseteq_1$  on each  $D_i$ .

Let  $D = \{ \langle d, i \rangle : d \in D_i^* \} \cup \{u\}$ , we denote  $D$  by  $\cup D_i$ . For  $d, d'$  in  $D$  we define:

$$d \subseteq d' \iff (d = u \text{ or } (\exists i \in I) (\exists x, y \in D_i) [x \subseteq_1 y \text{ and } d = \langle x, i \rangle \text{ and } d' = \langle y, i \rangle])$$

*Lemma:*  $(D, \subseteq)$  as defined above is a semantic domain:

Proof:

(1)  $u$  is the least element (bottom) of  $D$ :

following the definition of  $\subseteq$ ,  $(\forall d \in D) [u \subseteq d]$ .

(2)  $\subseteq$  is a partial ordering:

(i)  $\subseteq$  is reflexive

Let  $d \in D$ ,

If  $d=u$ , nothing to prove.

If  $d \neq u$  then  $(\exists i \in I) (\exists d_i \in D_i^*) [\langle d_i, i \rangle = d]$

But  $\subseteq_1$  is reflexive, then  $d_i \subseteq_1 d_i$

Therefore  $d \subseteq d$ .

(ii)  $\subseteq$  is antisymmetric

Let  $d, d'$  be in  $D$  such that  $d \subseteq d'$  and  $d' \subseteq d$

If  $d=u$  or  $d'=u$  nothing to prove

Otherwise:

(1)  $d \subseteq d' \implies$

$(\exists i \in I) (\exists x_i, y_i \in D_i^*) [x_i \subseteq_1 y_i \text{ and}$

$d = \langle x_i, i \rangle \text{ and}$

$d' = \langle y_i, i \rangle]$

(11)  $d' \subseteq d \implies$

$(\exists j \in I) (\exists x'_j, y'_j \in D_j^*) [y'_j \subseteq_1 x'_j \text{ and}$

$d = \langle x'_j, j \rangle \text{ and}$

$d' = \langle y'_j, j \rangle]$ .

$d = \langle x_i, i \rangle = \langle x'_j, j \rangle \implies i=j \text{ and } x_i = x'_j$ .

$d' = \langle y_i, i \rangle = \langle y'_j, j \rangle \implies i=j \text{ and } y_j = y'_j$ .

The set of equations  $\{x_i \subseteq_1 y_i \text{ and } y'_j \subseteq_1 x'_j\}$  reduces to

$\{x_i \subseteq_1 y_i \text{ and } y_i \subseteq_1 x_i\}$ .

But  $\subseteq_1$  is antisymmetric, so we get  $x_i = y_i$ .

Therefore  $d = \langle x_i, i \rangle = \langle y_i, i \rangle$  and  $\subseteq$  is antisymmetric.

(iii)  $\subseteq$  is transitive:

Let  $d, d', d'' \in D$  such that  $d \subseteq d'$  and  $d' \subseteq d''$

If  $d = u$  then  $d \subseteq d''$

Otherwise,

$d \subseteq d' \implies (\exists i, x_i, x'_i) [d = \langle x_i, i \rangle, d' = \langle x'_i, i \rangle \text{ and } x_i \subseteq_1 x'_i]$ ,

$d' \subseteq d'' \implies (\exists j, x'_j, x''_j) [d' = \langle x'_j, j \rangle, d'' = \langle x''_j, j \rangle \text{ and } x'_j \subseteq_1 x''_j]$ .

$d' = \langle x'_i, i \rangle = \langle x'_j, j \rangle \implies x'_i = x'_j$  and  $i = j$ .

The set of equations:  $\{x_i \subseteq_1 x'_i \text{ and } x'_j \subseteq_1 x''_j\}$  reduces to:

$\{x_i \subseteq_1 x'_i \text{ and } x'_i \subseteq_1 x''_i\}$ .

But  $\subseteq_1$  is transitive  $\implies x_i \subseteq_1 x''_i$  and  $x''_i \in D_i$ .

$d = \langle x_i, i \rangle$  and  $d'' = \langle x''_i, i \rangle$  and  $x_i \subseteq_1 x''_i \implies d \subseteq_1 d''$ .

Therefore  $\subseteq$  is transitive.

(3) The least upper bound of sequences exist:

Let  $(x_n)_{n \in \omega}$  be an  $\omega$ -sequence. Each  $x_n$  is of the form  $\langle d_{i_n}, i \rangle$ , where  $d_{i_n}$  is in  $D_i^*$ . It should be clear now that if  $(d_i, i) \subseteq (d_j, j)$  then  $d_i$  and  $d_j$  belong to the same domain and  $i = j$ .

Therefore for each  $i$ ,  $(d_{i_n})_n$  is an  $\omega$ -sequence of  $D_i$ .

But  $D_i$  is a semantic domain, so let  $d_i = \bigcup_n d_{i_n}$  in  $D_i$ .

The task now is to prove that  $d = \langle d_i, i \rangle = \bigcup_n x_n$ .

(i)  $(\forall n \in \omega) [x_n = \langle d_{i_n}, i \rangle \subseteq \langle d_i, i \rangle]$ ?

$d_i = \bigcup_n d_{i_n} \implies d_{i_n} \subseteq_1 d_i$ .

So by definition of  $\subseteq$ ,  $x_n \subseteq \langle d_i, i \rangle = d$ .

Therefore  $(\forall n \in \omega) [x_n \subseteq d]$

(2i) Let  $d'$  in  $D$  such that  $(\forall n \in \omega) [x_n \subseteq d']$ , prove  $d \subseteq d'$ ?

$d'$  is of the form  $\langle d'_j, j \rangle$  where  $j$  in  $I$  and  $d'_j$  in  $D_j$ .

$$(\forall n \in \omega) [x_n \subseteq d'] \implies$$

$$(\forall n \in \omega) [\langle d_{i_n}, i \rangle \subseteq \langle d'_j, j \rangle].$$

$$(\forall n \in \omega) [(i=j) \text{ and } (d_{i_n} \subseteq_1 d'_j)] \implies$$

$d'_j$  is an upper bound of  $(d_{i_n})_{n \in \omega}$  in  $D_i$ .

But  $d_i$  is the least upper bound so:

$$d_i \subseteq_1 d'_j \text{ and } \langle d_i, i \rangle \subseteq \langle d', i \rangle,$$

hence  $d \subseteq d'$ .

Therefore  $d$  is the least upper bound of  $(x_n)_{n \in \omega}$  in  $D$ .

Using (1), (2) and (3) we get:  $(D, \subseteq)$  is a semantic domain.  $\square$

II.2. Domains out of old domains using 'x': Let  $(D', \subseteq')$  and  $(D'', \subseteq'')$  be two semantic domains. We define  $(D' \times D'', \subseteq)$  as follows:

$$D = D' \times D'' = \{ \langle d', d'' \rangle \text{ where } d' \in D' \text{ and } d'' \in D'' \} \text{ and}$$

$$(\forall \langle d'_0, d''_0 \rangle, \langle d'_1, d''_1 \rangle \in D' \times D'')$$

$$[\langle d'_0, d''_0 \rangle \subseteq \langle d'_1, d''_1 \rangle \iff d'_0 \subseteq d'_1 \text{ and } d''_0 \subseteq d''_1]$$

*Lemma:*  $(D' \times D'', \subseteq)$  is a semantic domain.

Proof: (1) Let  $u = \langle u', u'' \rangle$  where  $u'$  is the bottom of  $D'$  and

$u''$  is the bottom of  $D''$ .

$u$  is the bottom of  $D' \times D''$  because:

$$\text{If } \langle d', d'' \rangle \in D, \text{ then } d' \in D' \text{ and } d'' \in D''$$

$$\implies u' \subseteq d' \text{ and } u'' \subseteq d'',$$

hence by definition of  $\subseteq$ ,  $\langle u', u'' \rangle \subseteq \langle d', d'' \rangle$ .

(2)  $\subseteq$  is a partial order:

(i)  $\subseteq$  is reflexive:

$$\text{Let } \langle d', d'' \rangle \text{ be in } D \implies (d' \in D') \text{ and } (d'' \in D'') \implies$$

$$d' \subseteq d' \text{ and } d'' \subseteq d'' \implies$$

$$\langle d', d' \rangle \subseteq \langle d', d'' \rangle.$$

(ii)  $\subseteq$  is transitive:



Let  $X, Y, Z$  be in  $D' \times D''$  such that  $X \subseteq Y$  and  $Y \subseteq Z$ .

$X = \langle x', x'' \rangle$ ,  $Y = \langle y', y'' \rangle$  and  $Z = \langle z', z'' \rangle$ .

$X \subseteq Y \implies x' \subseteq y'$  and  $x'' \subseteq y''$

$Y \subseteq Z \implies y' \subseteq z'$  and  $y'' \subseteq z''$

By transitivity of  $\subseteq$  and  $\subseteq''$  we get:

$x' \subseteq z'$  and  $x'' \subseteq z''$ .

Hence,  $X \subseteq Z$ .

(iii)  $\subseteq$  is antisymmetric:

Let  $X$  and  $Y$  be in  $D' \times D''$  such that  $X \subseteq Y$  and  $Y \subseteq X$ .

$X = \langle x', x'' \rangle$  and  $Y = \langle y', y'' \rangle$ .

$X \subseteq Y \implies x' \subseteq y'$  and  $x'' \subseteq y''$

$Y \subseteq X \implies y' \subseteq x'$  and  $y'' \subseteq x''$

$\implies$  (by transitivity of  $\subseteq$ ,  $\subseteq''$ ),

$x' = y'$  and  $x'' = y'' \implies X = Y$

Using (i), (ii) and (iii) we get that  $\subseteq$  is a partial order.

(3)  $\omega$ -sequences have limits in  $D' \times D''$ :

Let  $(X_i)_{i \in I}$  be an  $\omega$ -sequence in  $D' \times D''$ .

Each  $X_i$  is of the form  $\langle x_i', x_i'' \rangle$ .

As  $(X_i)_{i \in \omega}$  is an  $\omega$ -sequence in  $D$ , we can prove that:

$\{(x_i')_{i \in I}$  is an  $\omega$ -sequence in  $D'$  and

$(x_i'')_{i \in I}$  is an  $\omega$ -sequence in  $D''\}$

But  $D'$  and  $D''$  are semantic domains, So:

Let  $d'$  be the limit of  $(x_i')_{i \in \omega}$  in  $D'$

and  $d''$  be the limit of  $(x_i'')_{i \in \omega}$  in  $D''$ .

Our task now is to prove that:

$\langle d', d'' \rangle$  is the limit of  $(X_i)_{i \in \omega}$  in  $D$ .

(a)  $(\forall i \in \omega) [(x_i' \subseteq d') \text{ and } (x_i'' \subseteq d'')]$

because  $d'$  and  $d''$  are the limits.

Hence  $(\forall i \in \omega) [\langle x_i', x_i'' \rangle \subseteq \langle d', d'' \rangle]$

and so  $(\forall i \in \omega) [x_i \subseteq \langle d', d'' \rangle]$ .

Therefore  $\langle d', d'' \rangle$  is an upper bound of  $(X_i)_{i \in \omega}$  in  $D' \times D''$ .

(b) Let  $d_1$  be an upper bound of  $(X_i)_{i \in \omega}$  in  $D' \times D''$ .

$d_1 = \langle d_1', d_1'' \rangle$  where  $(d_1' \in D')$  and  $(d_1'' \in D'')$ .

$(\forall i \in \omega) [x_i \subseteq d_1]$  therefore

$(\forall i \in \omega) [(x_i' \subseteq d_1') \text{ and } x_i'' \subseteq d_1'']$ .

Because  $d_1'$  is an upper bound of  $(X_i')_{i \in \omega}$  in  $D'$  and

$d_1''$  is an upper bound of  $(X_i'')_{i \in \omega}$  in  $D''$ ,

$d' \subseteq d_1'$  and  $d'' \subseteq d_1''$  hence

$\langle d', d'' \rangle \subseteq \langle d_1', d_1'' \rangle$ .

Hence  $\langle d', d'' \rangle$  is the least upper bound of  $(X_i)_{i \in \omega}$  in  $D' \times D''$ .

Using (1), (2) and (3), we get that  $D' \times D''$  is a semantic domain.  $\square$

### II.3. New domains out of old ones using $\rightarrow$

Let  $[D_1 \rightarrow D_2]$  be the set of continuous functions from the domain  $(D_1, \subseteq_1)$  to the domain  $(D_2, \subseteq_2)$ .

We shall define a binary relation on  $[D_1 \rightarrow D_2]$

as follows:

$$(\forall f, g \text{ in } [D_1 \rightarrow D_2]) [f \subseteq g \iff (\forall d \in D) [f(d) \subseteq_2 g(d) ]].$$

*Lemma:*

$([D_1 \rightarrow D_2], \subseteq)$  as defined above is a semantic domain.

Proof:

(1) Let us take  $u = \lambda d. u_2$ ,  $u$  is the bottom of  $D$ .

(i)  $u$  is well defined:

Obvious, for  $u$  is the constant function.

(ii)  $u$  is continuous:

For each  $\omega$ -sequence  $(d_n)_{n \in \omega}$  in  $D_1$ ,

$$u(\cup d_n) = u_2 = \cup u_2 = \cup (u(d_n)).$$

(iii)  $u$  is the bottom of  $D$ :

Let  $f$  be in  $D$ .

$$(\forall d \in D_1) [u(d) = u_2]. \text{ But } u_2 \subseteq_2 f(d) \implies$$

$$(\forall d \in D_1) [u(d) \subseteq_2 f(d)] \implies$$

$$u \subseteq f.$$

Therefore  $u$  is the bottom of  $D$ .

(2)  $\subseteq$  is a partial order

(i)  $\subseteq$  is reflexive:

Let  $f$  be in  $[D_1 \dashrightarrow D_2]$ ,

$$(\forall d \in D) [f(d) \subseteq_2 f(d)].$$

Therefore,  $f \subseteq f$ .

(ii)  $\subseteq$  is antisymmetric:

Let  $f, g$  in  $D$  such that  $f \subseteq g$  and  $g \subseteq f$ .

Consider  $d$  in  $D_1$ .

$$(a) f \subseteq g \implies f(d) \subseteq_2 g(d).$$

$$(b) g \subseteq f \implies g(d) \subseteq_2 f(d).$$

But  $\subseteq_2$  is antisymmetric, hence (a)+(b)  $\implies f(d) = g(d)$ .

Therefore  $(\forall d \in D) [f(d) = g(d)]$

Using extensionality, we get  $f = g$ .

Hence  $\subseteq$  is antisymmetric.

(iii)  $\subseteq$  is transitive:

Let  $f, g, h$  be in  $D$  such that  $f \subseteq g$  and  $g \subseteq h$ .

Consider  $d$  in  $D$ ,

$$(a) f \subseteq g \implies f(d) \subseteq_2 g(d)$$

$$(b) g \subseteq h \implies g(d) \subseteq_2 h(d)$$

But  $\subseteq_2$  is transitive, hence (a)+(b)  $\implies f(d) \subseteq_2 h(d)$ .

Hence:  $(\forall d \in D) [f(d) \subseteq_2 h(d)]$  and so  $f \subseteq h$ .

By (i), (ii) and (iii)  $\subseteq$  is a partial order.

(3) Every  $\omega$ -sequence has a least upper bound.

Let  $(f_n)_{n \in \omega}$  be an  $\omega$ -sequence of  $[D_1 \rightarrow D_2]$  and consider  $f = \lambda d. \cup f_n(d)$ .

$f = \cup f_n$  because:

(i)  $f$  is well defined:

Let  $d$  in  $D_1 \in (f_n(d))_{n \in \omega}$  is an  $\omega$ -sequence of  $D_2$  because:

$$(\forall n \in \omega) [f_n \subseteq f_{n+1}] \implies$$

$$(\forall n \in \omega) [f_n(d) \subseteq_2 f_{n+1}(d)] \implies$$

$(f_n(d))_{n \in \omega}$  is an  $\omega$ -sequence of  $D_2$  and

its limit  $\cup f_n(d)$  is well defined.

So  $f = \lambda d. \cup f_n(d)$  is well-defined.

(ii)  $f$  is continuous:

Let  $(d_m)_{m \in \omega}$  be an  $\omega$ -sequence of  $D_1$ .

$$f(\cup d_m) = \cup_n f_n(\cup_m d_m) = \cup_n (\cup_m f_n(d_m)) \text{ (} f_n \text{ is continuous).}$$

$$= \cup_m (\cup_n f_n(d_m)) \text{ (using next lemma).}$$

$$= \cup_m f(d_m) \text{ (by definition of } f \text{).}$$

Therefore  $f(\cup d_m) = \cup f(d_m)$  and  $f$  is continuous.

Before continuing the proof that  $f = \cup f_n$ , we need the following lemma:

Lemma: Let  $(f_n)_n$  and  $(d_m)_m$  be  $\omega$ -sequences of  $[D_1 \rightarrow D_2]$  and  $D_1$  respectively.

$$\text{Then } \cup_n (\cup_m f_n(d_m)) = \cup_m (\cup_n f_n(d_m))$$

Proof: First we need to prove a little sublemma:

Sublemma: Let  $(a_n)_{n \in \omega}$  and  $(b_m)_{m \in \omega}$  be  $\omega$ -sequences of a

semantic domain  $D$  such that  $(\forall n \in \omega) [a_n \subseteq b_n]$

$$\text{Then } \cup a_n \subseteq \cup b_n.$$

Proof:  $\forall n \in \omega, a_n \subseteq b_n$  (hyp.)

and  $b_n \subseteq \cup b_n$  (def. of limit)

But  $\subseteq$  is transitive  $\implies$

$$(\forall n \in \omega) [a_n \subseteq \cup b_n].$$

Therefore  $\cup b_n$  is an upper bound of  $(a_n)_{n \in \omega} \implies$

$\cup a_n \subseteq \cup b_n$ . Hence the proof of the sublemma.

$(f_n(d_m))_{n \in \omega}$  is an  $\omega$ -sequence in  $D_2$  because  $(f_n)_n$  is an  $\omega$ -sequence in  $D$ .

Therefore  $\cup_n f_n(d_m)$  exists in  $D_2$  and  $(\forall n \in \omega) [f_n(d_m) \subseteq_2 \cup_n f_n(d_m)]$ .

But  $(f_n(d_m))_{m \in \omega}$  is an  $\omega$ -sequence because :

$f_n$  is continuous  $\implies$

$f_n$  is monotonic, i.e  $(\forall m \in \omega) [d_m \subseteq d_{m+1} \implies f_n(d_m) \subseteq f_n(d_{m+1})]$ ,

this implies that  $(\forall m \in \omega) [f_n(d_m) \subseteq f_n(d_{m+1})]$ .

$\implies (f_n(d_m))_{m \in \omega}$  is an  $\omega$ -sequence.

Also  $(\cup_n f_n(d_m))_{m \in \omega}$  is an  $\omega$ -sequence because:

$(\forall m \in \omega) [d_m \subseteq d_{m+1}] \implies$

$(\forall m \in \omega) [f_n(d_m) \subseteq_2 f_n(d_{m+1})]$  as  $f_n$  is monotonic  $\implies$

$\cup_n f_n(d_m) \subseteq_2 \cup_n f_n(d_{m+1})$  using above sublemma  $\implies$

$(\cup_n f_n(d_m))_{m \in \omega}$  is an  $\omega$ -sequence.

Applying the above sublemma on  $(\cup_n f_n(d_m))_{m \in \omega}$  and  $(f_n(d_m))_{m \in \omega}$ ,

we get  $\cup_m f_n(d_m) \subseteq_2 \cup_m (\cup_n f_n(d_m))$ .

This means that  $\cup_m (\cup_n f_n(d_m))$  is an upper bound of  $(\cup_m f_n(d_m))_{n \in \omega}$

$\implies$  (a) :  $\cup_n (\cup_m f_n(d_m)) \subseteq_2 \cup_m (\cup_n f_n(d_m))$ .

Now let us prove that  $\cup_m (\cup_n f_n(d_m)) \subseteq_2 \cup_n (\cup_m f_n(d_m))$ :

$$d_m \subseteq_1 \cup_m d_m$$

$\implies (\forall n \in \omega) [f_n(d_m) \subseteq_2 f_n(\cup_m d_m)]$  as  $f$  is monotonic.

$\implies (\forall n \in \omega) [f_n(d_m) \subseteq_2 \cup_m f_n(d_m)]$  as  $f$  is continuous.

However  $(f_n(d_m))_{n \in \omega}$  and  $(\cup_m f_n(d_m))_{n \in \omega}$  are  $\omega$ -sequences,

Therefore by the above sublemma:  $\bigcup_n f_n(d_m) \subseteq_2 \bigcup_n (\bigcup_{m \in \omega} f_n(d_m))$ .

Therefore  $\bigcup_n (\bigcup_m f_n(d_m))$  is an upper bound of  $(\bigcup_n f_n(d_m))_{m \in \omega}$ .

$\implies$  (b) :  $\bigcup_m (\bigcup_n f_n(d_m)) \subseteq_2 \bigcup_n (\bigcup_m f_n(d_m))$ .

But  $\subseteq_2$  is antisymmetric, hence (a)+(b)  $\implies$

$\bigcup_n (\bigcup_m f_n(d_m)) = \bigcup_m (\bigcup_n f_n(d_m))$ . End of proof of lemma.

Now back to the proof that every  $\omega$ -sequence of  $[D_1 \dashrightarrow D_2]$

has a least upper bound. We continue as follows:

(iii)  $f$  is the limit of  $(f_n)_{n \in \omega}$ :

$\forall n \in \omega, f_n \subseteq f$  because:

If  $d \in D_1 \implies f(d) = \bigcup f_n(d) \implies f_n(d) \subseteq_2 f(d)$ .

Therefore  $f$  is an upper bound of  $(f_n)_{n \in \omega}$ .

Suppose  $g$  is an upper bound of  $(f_n)_{n \in \omega}$  and consider  $d$  in  $D$ .

$f(d) = \bigcup f_n(d)$ .

$(\forall n \in \omega) [f_n \subseteq g] \implies (\forall n \in \omega) [f_n(d) \subseteq_2 g(d)] \implies$

$\bigcup f_n(d) \subseteq_2 g(d) \implies f(d) \subseteq_2 g(d)$ .

Therefore  $(\forall d \in D) [f(d) \subseteq_2 g(d)] \implies f \subseteq g$ .

Hence each  $\omega$ -sequence in  $[D_1 \dashrightarrow D_2]$  has a limit.

using (1),(2) and (3) we get that  $([D_1 \dashrightarrow D_2], \subseteq)$  is a semantic domain.  $\square$

So far we have seen a way of building a domain out of two (or more) old domains.

Later, we shall see that we are really interested in domains  $E$  which satisfy an

equation of the form:  $E \cong [E \dashrightarrow E]$ . We define  $B$  the set of truth values, i.e.  $B =$

$\{0, 1, u_0\}$  where  $u_0 \subseteq 1, u_0 \subseteq 0$  ( $B$  is a semantic domain). We build our domain  $E$  by

building a sequence of domains (by induction). We start with  $E_0 = B$  and build  $E_{n+1} =$

$B + [E_n \dashrightarrow E_n]$  for  $n \geq 0$ . For all  $n, E_n$  is a semantic domain. We would like, however,

to relate all those domains with an ordering relation and find the limit of such a

sequence. This limit is going to be the required  $E$ . We start with some definitions:

*Definition:* A projection pair of  $D_1$  on  $D_2$  is a pair  $\langle \Phi, \Psi \rangle$  such that:

$\Phi : D_1 \dashrightarrow D_2, \Psi : D_2 \dashrightarrow D_1$  and

(a)  $\Phi, \Psi$  are both continuous,

(b)  $(\forall x \in D_1) [\Psi(\Phi(x)) = x]$

(c)  $(\forall x \in D_2) [\Phi(\Psi(x)) \subseteq_2 x]$

For each  $n \geq 0$ , we define a projection pair  $\langle \Psi_n, \Phi_n \rangle$ . The aim of each  $\Phi_n$  is to embed  $E_n$  into  $E_{n+1}$ , whereas  $\Psi_n$  is a surjection from  $E_{n+1}$  to  $E_n$ . Our construction of  $(\Phi_n)_{n \in \omega}$  is done by induction as follows:<sup>140</sup>

$\Phi_0 : E_0 \dashrightarrow E_1 \quad \Psi_0 : E_1 \dashrightarrow E_0$

$\Phi_0(x) = x \in B^* \dashrightarrow x, u_1 \quad \Psi_0(x) = x \in B^* \dashrightarrow x, u_0$

(1i<sub>0</sub>)  $\Phi_0$  is well defined and is an injection: obvious.

(2i<sub>0</sub>)  $\Psi_0$  is well defined and is a surjection: obvious.

(3i<sub>0</sub>)  $\Psi_0(\Phi_0(x)) = x$  for all  $x$  in  $E_0$ :

$$\begin{aligned} \Psi_0(\Phi_0(x)) &= x \in B^* \dashrightarrow \Psi_0(x), \Psi_0(u_1) \\ &= x \in B^* \dashrightarrow x, u_0 \\ &= x. \end{aligned}$$

(4i<sub>0</sub>)  $\Phi_0(\Psi_0(x)) \subseteq_1 x$  for all  $x$  in  $E_1$ :

$$\begin{aligned} \Phi_0(\Psi_0(x)) &= x \in B^* \dashrightarrow \Phi_0(x), \Phi_0(u_0) \\ &= x \in B^* \dashrightarrow x, u_1 \end{aligned}$$

If  $\Phi_0(\Psi_0(x)) = x \implies \Phi_0(\Psi_0(x)) \subseteq_1 x$  as  $\subseteq_1$  is reflexive.

If  $\Phi_0(\Psi_0(x)) = u_1 \implies \Phi_0(\Psi_0(x)) \subseteq_1 x$  (bottom element).

Therefore  $(\forall x \in E_1) [\Phi_0(\Psi_0(x)) \subseteq_1 x]$ .

(5i<sub>0</sub>)  $\Phi_0$  is continuous:

Let  $(x_n)_{n \in \omega}$  be an  $\omega$ -sequence of  $E_0$ . Two cases arise:

(a)  $\cup x_n = u_0 \implies$

$$\Phi_0(\cup x_n) = u_1 \text{ and } (\forall n \in \omega) [x_n = u_0] \implies$$

$$\Phi_0(\cup x_n) = u_1 \text{ and } (\forall n \in \omega) [\Phi_0(x_n) = u_1] \implies$$

<sup>140</sup> (the notation " $f(x) = p(x) \dashrightarrow a, b$ " is to be understood as: if  $p(x)$  is true then  $f(x) = a$  otherwise  $f(x) = b$ )

$$\Phi_0(\cup x_n) = u_1 \text{ and } \cup \Phi_0(x_n) = u_1 \implies$$

$$\Phi_0(\cup x_n) = \cup \Phi_0(x_n)$$

(b)  $\cup x_n$  is in  $B^*$   $\implies$

$$\Phi_0(\cup x_n) = \cup x_n \text{ and } (\exists k \in \omega) (\forall n \geq k) [x_k = x_n = \cup x_n \in B^*] \implies$$

$$\Phi_0(\cup x_n) = \cup x_n = x_n = x_k \in B^* \text{ for all } n \geq k \implies$$

$$(\Phi_0(\cup x_n) = \cup x_n) \text{ and } (\Phi_0(x_n) = x_n = x_k = \cup x_n \text{ for all } n \geq k) \implies$$

$$(\Phi_0(\cup x_n) = \cup x_n) \text{ and } (\cup \Phi_0(x_n) = x_k = \cup x_n) \implies$$

$$\Phi_0(\cup x_n) = \cup \Phi_0(x_n).$$

Therefore we always have  $\Phi_0(\cup x_n) = \cup \Phi_0(x_n)$ .

(6i<sub>0</sub>)  $\Psi_0$  is continuous:

Let  $(x_n)_{n \in \omega}$  be an  $\omega$ -sequence of  $E_1$ . Two cases arise:

(a)  $\cup x_n \in B^*$ : the proof as in (b) above.

(b)  $\cup x_n$  is in  $[E_0 \dashrightarrow E_0] \cup \{u_1\}$

(i) If  $\cup x_n = u_1$ :

the proof is as in (5i<sub>0</sub>,a) with interchanging  $u_0, u_1$ .

(ii) If  $\cup x_n \in [E_0 \dashrightarrow E_0]$  then:

$$(\exists k \in \omega) (\forall n \geq k) [x_n \in [E_0 \dashrightarrow E_0]] \text{ (definition of '')} \implies$$

$$\Psi_0(\cup x_n) = u_0 \text{ and } (\forall n \geq k) [\Psi_0(x_n) = u_0] \implies$$

$$\Psi_0(\cup x_n) = u_0 \text{ and } \cup \Psi_0(x_n) = u_0 \implies$$

$$\Psi_0(\cup x_n) = \cup \Psi_0(x_n).$$

Therefore, we always have  $\Psi_0(\cup x_n) = \cup \Psi_0(x_n)$ .

By induction, we build  $\Phi_{n+1}$  and  $\Psi_{n+1}$  assuming that  $\Phi_n$  and  $\Psi_n$  have been defined satisfying (i<sub>n</sub>), (2i<sub>n</sub>), (3i<sub>n</sub>), (4i<sub>n</sub>), (5i<sub>n</sub>) and (6i<sub>n</sub>).

$$\Phi_{n+1}: E_{n+1} \dashrightarrow E_{n+2}$$

$$\Phi_{n+1}(x) = x \in B^* \dashrightarrow x, (x = u_{n+1} \dashrightarrow u_{n+2}, \Phi_n \circ x \circ \Psi_n)$$

$$\Psi_{n+1}: E_{n+2} \dashrightarrow E_{n+1}$$

$$\Psi_{n+1}(x) = x \in B^* \dashrightarrow x, (x = u_{n+2} \dashrightarrow u_{n+1}, \Psi_n \circ x \circ \Phi_n)$$



(i<sub>n+1</sub>)  $\Phi_{n+1}$  is well defined and injective:

Well defined: obvious.

Injective: Let  $x, x'$  be in  $E_{n+1}$  such that  $\Phi_{n+1}(x) = \Phi_{n+1}(x')$ .

$$\text{If } x \in B^* \implies \Phi_{n+1}(x) = x \in B^* \implies$$

$$\Phi_{n+1}(x') = \Phi_{n+1}(x) \in B^* \implies$$

$$x' \text{ is in } B \text{ and } \Phi_{n+1}(x') = x'$$

$$\implies x = x'.$$

$$\text{If } x = u_{n+1} \implies \Phi_{n+1}(x) = u_{n+2} = \Phi_{n+1}(x')$$

$$\implies x' = u_{n+1} \implies x = x'.$$

$$\text{If } x \in [E_n \dashrightarrow E_n] \implies$$

$$x' \in [E_n \dashrightarrow E_n] \text{ and } \Phi_{n+1}(x) = \Phi_{n+1}(x') \implies$$

$$\Phi_n \circ x \circ \Psi_n = \Phi_n \circ x' \circ \Psi_n \implies$$

$$\Psi_n \circ \Phi_n \circ x \circ \Psi_n = \Psi_n \circ \Phi_n \circ x' \circ \Psi_n \implies$$

$$x \circ \Psi_n = x' \circ \Psi_n \text{ (by induction } \Psi_n \circ \Phi_n(x) = x) \implies$$

$$x \circ \Psi_n \circ \Phi_n = x' \circ \Psi_n \circ \Phi_n \implies x = x'.$$

Therefore  $\Phi_{n+1}(x) = \Phi_{n+1}(x') \implies x = x'$ . Hence  $\Phi_{n+1}$  is injective.

(2i<sub>n+1</sub>)  $\Psi_n$  is surjective and well-defined:

Well defined: obvious.

Surjective: Let  $f$  in  $E_{n+1}$ .

$$\text{If } f = u_1 \text{ then } u_2 \in E_{n+2} \text{ and } \Psi_{n+1}(u_2) = u_1.$$

$$\text{If } f \in B^* \text{ then } f \in E_{n+2} \text{ and } \Psi_{n+1}(f) = f.$$

$$\text{Otherwise } f \in [E_n \dashrightarrow E_n] \implies \Phi_{n+1}(f) \in E_{n+2}$$

$$\Psi_{n+1}(\Phi_{n+1}(f)) = f \text{ As we shall prove in (3i}_{n+1}\text{)}.$$

Therefore  $\Psi_{n+1}$  is surjective.

(3i<sub>n+1</sub>)  $\Psi_{n+1}(\Phi_{n+1}(f)) = f$ : (remember our notation  $f(x) = p(x) \dashrightarrow a, b$ )

$$\Psi_{n+1}(\Phi_{n+1}(f)) = f \in B^* \dashrightarrow \Psi_{n+1}(f),$$

$$(f = u_{n+1} \dashrightarrow \Psi_{n+1}(u_{n+2}), \Psi_{n+1}(\Phi_n \circ f \circ \Psi_n)),$$

$$= f \in B^* \text{----} > f,$$

$$(f = u_{n+1} \text{----} > u_{n+1}, \Psi_n \circ (\Phi_n \circ \Psi_n) \circ \Phi_n)$$

$$\text{But } \Psi_n \circ (\Phi_n \circ \Psi_n) \circ \Phi_n = ((\Psi_n \circ \Phi_n) \circ \Psi_n) \circ \Phi_n$$

$$= f \text{ because } \Psi_n \circ \Phi_n = \text{the identity function.}$$

$$\text{Therefore } \Psi_{n+1}(\Phi_{n+1}(f)) = f.$$

$$(4i_{n+1}) \Phi_{n+1}(\Psi_{n+1}(f)) \subseteq_{n+2} f, \text{ for all } f \text{ in } E_{n+2}:$$

Let  $f$  in  $E_{n+2}$ ,

$$\Phi_{n+1}(\Psi_{n+1}(f)) = f \in B^* \text{----} > \Phi_{n+1}(f),$$

$$(f = u_{n+2} \text{----} > \Phi_{n+1}(u_{n+1}), \Phi_{n+1}(\Psi_n \circ \Phi_n))$$

$$= f \in B^* \text{----} > f,$$

$$(f = u_{n+2} \text{----} > u_{n+2}, \Phi_n \circ (\Psi_n \circ \Phi_n) \circ \Psi_n).$$

Let  $x$  in  $E_{n+1}$ ,

$$(\Phi_n \circ (\Psi_n \circ \Phi_n) \circ \Psi_n)(x) = \Phi_n(\Psi_n(f(\Phi_n(\Psi_n(x)))))$$

But  $\Phi_n(\Psi_n(f)) \subseteq_{n+1} f$  and  $\Phi_n \circ \Psi_n$  is monotonic  $\implies$

$$\Phi_n \circ (\Psi_n \circ \Phi_n) \circ \Psi_n(x) \subseteq_{n+1} f(\Phi_n(\Psi_n(x))).$$

But again  $\Phi_n(\Psi_n(x)) \subseteq_{n+1} x$  and  $f$  is continuous (monotonic)

$$\implies f(\Phi_n(\Psi_n(x))) \subseteq_{n+1} f(x).$$

Using transitivity of  $\subseteq_{n+1}$  we get:

$$(\Phi_n \circ (\Psi_n \circ \Phi_n) \circ \Psi_n)(x) \subseteq_{n+1} f(x) \text{ for all } x \text{ in } E_{n+1} \implies$$

$$\Phi_n \circ (\Psi_n \circ \Phi_n) \circ \Psi_n \subseteq_{n+2} f$$

$$f \subseteq_{n+2} f \text{ (reflexivity)}$$

$$u_{n+2} \subseteq_{n+2} f \text{ (bottom element)}$$

$$\text{Therefore always } \Phi_{n+1}(\Psi_{n+1}(f)) \subseteq_{n+2} f.$$

(5i<sub>n+1</sub>)  $\Phi_{n+1}$  is continuous:

Let  $(x_n)_{n \in \omega}$  be an  $\omega$ -sequence of  $E_{n+1}$ .

$$\Phi_{n+1}(\bigcup x_n) = \bigcup x_n \in B^* \text{----} > \bigcup x_n,$$

$$(\bigcup x_n = u_{n+1} \text{----} > u_{n+2}, \Phi_n \circ (\bigcup x_n) \circ \Psi_n)$$

But  $\Phi_{n+1}(x_n) = x_n \in B^* \text{----} > x_n$ ,

$$(x_n = u_{n+1} \text{----} > u_{n+2}, \Phi_n \circ x_n \circ \Psi_n)$$

(a) If  $\cup x_n$  is in  $B^*$ , the proof goes as in (5i<sub>0</sub>,b)

$$\text{and we get } \Phi_{n+1}(\cup x_n) = \cup \Phi_{n+1}(x_n).$$

(b) If  $\cup x_n = u_{n+1}$ , the proof goes as in (5i<sub>0</sub>,a)

$$\text{and we get } \Phi_{n+1}(\cup x_n) = \cup \Phi_{n+1}(x_n).$$

(c) Otherwise,  $\cup x_n$  is in  $[E_n \text{--} > E_n]$  and we can use the

continuity together with the definition of  $\cup f_n$ , where

$$f_n \text{ is a function, to prove that } \Phi_n \circ (\cup x_n) \circ \Psi_n = \cup (\Phi_n \circ x_n \circ \Psi_n).$$

$$\text{And we get } \Phi_{n+1}(\cup x_n) = \cup \Phi_{n+1}(x_n).$$

(a), (b) and (c)  $\implies \Phi_{n+1}(\cup x_n) = \cup \Phi_{n+1}(x_n)$

$$\implies \Phi_{n+1} \text{ is continuous.}$$

(6i<sub>n+1</sub>)  $\Psi_{n+1}$  is continuous:

Let  $(x_n)_n \in \omega$  be an  $\omega$ -sequence of  $E_{n+2}$ .

$$\Psi_{n+1}(\cup x_n) = \cup x_n \in B^* \text{----} > \cup x_n,$$

$$(\cup x_n = u_{n+2} \text{----} > u_{n+1}, \Psi_n \circ (\cup x_n) \circ \Phi_n)$$

$$\Psi_n(x_n) = x_n \in B^* \text{----} > x_n,$$

$$(x_n = u_{n+2} \text{----} > u_{n+1}, \Psi_n \circ x_n \circ \Phi_n)$$

(a) If  $\cup x_n \in B^*$ , the proof goes as in (a) above

$$\text{and we have } \Psi_{n+1}(\cup x_n) = \cup \Psi_{n+1}(x_n)$$

(b) If  $\cup x_n = u_{n+1}$ , then the proof goes as in (b) above

$$\text{and we have } \Psi_{n+1}(\cup x_n) = \cup \Psi_{n+1}(x_n)$$

(c) If  $\cup x_n$  is in  $[E_{n+1} \text{----} > E_{n+1}]$  then we can use the

continuity of  $\Psi_n$  together with the definition of

$\cup x_n$  where  $x$  is a function, to prove that:

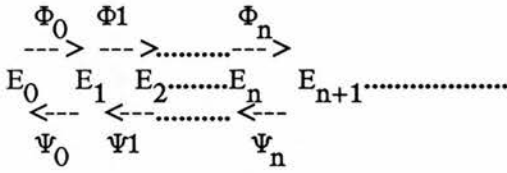
$$\Psi_n \circ (\cup x_n) \circ \Phi_n = \cup (\Psi_n \circ x_n \circ \Phi_n).$$

$$\text{And so we get } \Psi_{n+1}(\cup x_n) = \cup \Psi_{n+1}(x_n).$$

$$(a), (b) \text{ and } (c) \implies \Psi_{n+1}(\cup x_n) = \cup \Psi_{n+1}(x_n)$$

$$\implies \Psi_{n+1} \text{ is continuous.}$$

To conclude the construction, we draw the picture which shows the relations clearly:



Where: For all  $n \geq 0$ ,

- $\Phi_n$  is injective continuous,
- $\Psi_n$  is surjective continuous,
- $\Phi_n(\Psi_n(f)) \subseteq_{n+1} f$ ,
- $\Psi_n(\Phi_n(f)) = f$

$\Phi_n$  and  $\Psi_n$  can be so extended so that instead of running through two consecutive domains, they run through any 2 domains. This is done as follows:

$$\Phi_{nm} : E_n \dashrightarrow E_m \text{ such that:}$$

for all  $n, m$  in  $N$ ,

$$n = m \implies \Phi_{nm} = Id_n = \lambda x \in E_n. x$$

$$n < m \implies \Phi_{nm} = \Phi_{m-1} \circ \Phi_{nm-1}$$

$$n > m \implies \Phi_{nm} = \Phi_{n-1m} \circ \Psi_{n-1}$$

*Lemma:*  $(\forall n, m \geq 0) [\Phi_{nm} \text{ is continuous}]$

*Proof:*

*Sublemma1:* Let  $n \geq 0, (\forall m \geq n) [\Phi_{nm} \text{ is continuous}]$

*Proof:* By induction,

- case  $n=m$  true because  $\Phi_{nm} = Id_n$  continuous.
- Assume that  $(\forall m \geq n) [\Phi_{nm} \text{ is continuous}]$ ,
- and prove that  $\Phi_{nm+1}$  is continuous.

$$\Phi_{nm+1} = \Phi_m \circ \Phi_{nm}$$

$\Phi_m$  and  $\Phi_{nm}$  are continuous (induction hypotheses),

Then  $\Phi_m \circ \Phi_{nm}$  is continuous.

Therefore  $(\forall m \geq n) [\Phi_{nm} \text{ is continuous}]$ .

Sublemma2:  $(\forall n \geq m) [\Phi_{nm} \text{ is continuous}]$ .

Proof: similar to above.

Using sublemma1,2 we get  $\Phi_{nm}$  continuous, for all  $n, m \geq 0$ .  $\square$

Lemma:  $(\forall n, m: 0 \leq n \leq m) [\Phi_{mn}(\Phi_{nm}(f)) = f]$ .

Proof: If  $m=n \implies \Phi_{mn}(\Phi_{nm}(f)) = \Phi_{mn}(f) = f$ .

Assume that the property is true for all  $m \geq n \geq 0$  such that  $m-n \leq k$ .

Let us prove the property holds for  $m \geq n \geq 0$  such that  $m=n+k+1$ .

$$\begin{aligned} \Phi_{mn}(\Phi_{nm}(f)) &= \Phi_{mn}((\Phi_{m-1} \circ \Phi_{nm-1})(f)) \text{ as } n < m \\ &= (\Phi_{m-1} \circ \Psi_{m-1})(\Phi_{nm-1}(f)) \text{ as } m > n \\ &= \Phi_{m-1}((\Psi_{m-1} \circ \Phi_{nm-1})(\Phi_{nm-1}(f))) \text{ as } \circ \text{ is associative.} \\ &= \Phi_{m-1}(\Phi_{nm-1}(f)) \text{ as } \Psi_{m-1} \circ \Phi_{m-1}(x) = x. \\ &= f \text{ by induction, because } m-n-1=k \leq k. \square \end{aligned}$$

Note that here a stronger lemma could be proved. That is:  $\Phi_{m1} \circ \Phi_{nm} = \Phi_{n1}$ . Our above lemma will be a special case of this one by taking  $l=n$ .

Lemma:  $\Phi_{nm}(\Phi_{mn}(g)) \subseteq_m g$  for  $0 \leq n \leq m$ .

Proof: By induction as above.

1.  $n=m \implies \Phi_{nm}(\Phi_{mn}(g)) = g \subseteq_m g$  (reflexivity)
2. Suppose that for all  $0 \leq n \leq m$  such that  $m-n \leq k$  and  $k \geq 0$ , the property that  $\Phi_{nm}(\Phi_{mn}(g)) \subseteq_m g$  holds.
3. Let us prove that it also holds for  $m \geq n \geq 0$  such that  $m=n+k+1$ :

$$\begin{aligned} \Phi_{nm}(\Phi_{mn}(g)) &= \Phi_{nm}((\Phi_{m-1} \circ \Psi_{m-1})(g)) \text{ as } m > n \\ &= (\Phi_{m-1} \circ \Phi_{nm-1})(\Phi_{m-1} \circ \Psi_{m-1}(g)) \text{ as } n < m \\ &= \Phi_{m-1}(\Phi_{nm-1}(\Phi_{m-1}(\Psi_{m-1}(g)))) \text{ as } \circ \text{ is associative.} \end{aligned}$$

However,  $m-1-n \leq k$ , so following the hypotheses of induction we get:

$$(a) \Phi_{nm-1}(\Phi_{m-1}(x)) \subseteq_{m-1} x.$$

But  $(\forall n \in \omega) [\Phi_n \text{ is continuous}] \implies$

$(\forall n \in \omega) [\Phi_n \text{ is monotonic}] \implies$

$\Phi_{m-1}$  is monotonic.

Therefore using (a) and the monotonicity of  $\Phi_{m-1}$ , we get:

$$\begin{aligned} \Phi_{m-1}(\Phi_{nm-1}(\Phi_{m-1n}(x))) &\subseteq_m \Phi_{m-1}(x) \\ \implies (b) : \Phi_{nm}(\Phi_{mn}(g)) &\subseteq_m \Phi_{m-1}(\Psi_{m-1}(g)) \end{aligned}$$

But (c):  $\Phi_{m-1}(\Psi_{m-1}(g)) \subseteq_m g$ .

As  $\subseteq$  is transitive, we get from (b)+(c):

$$\Phi_{nm}(\Phi_{mn}(g)) \subseteq_m g.$$

And so for all  $0 \leq n \leq m$ ,  $\Phi_{nm}(\Phi_{mn}(g)) \subseteq_m g$ .  $\square$

We still have not found a domain  $E \simeq [E \rightarrow E]$ . Here we see how to do it.

Having constructed all the  $(E_n)_{n \in \omega}$ , we can construct a domain  $E_\infty$  which will contain all the  $E_n$  for  $n \in \omega$ .

$$E_\infty = \{ \langle f_n \rangle : f_n \in E_n \text{ and } \Psi_n(f_{n+1}) = f_n \}.$$

The ordering relation on  $E_\infty$  will be:

$$(\forall \langle f_n \rangle, \langle g_n \rangle \in E_\infty) [ \langle f_n \rangle_n \in \omega \subseteq \langle g_n \rangle_n \in \omega \iff (\forall n \in \omega) [f_n \subseteq_n g_n] ]$$

we shall prove next that  $(E_\infty, \subseteq)$  is a semantic domain.

*Lemma:*  $(E_\infty, \subseteq)$  is a semantic domain.

*Proof:*

(1) Bottom element:

Let  $\langle u_n \rangle_n$  where  $u_n$  is the bottom element of  $E_n$ .

$\langle u_n \rangle_n$  is the bottom element of  $E_\infty$ :

(i)  $\langle u_n \rangle_n$  is in  $E_\infty$ :

$u_n$  is in  $E_n$  for all  $n$ ,

$\Psi_n(u_{n+1}) = u_n$  by definition of  $\Psi_n$ ,

Therefore  $\langle u_n \rangle_n$  is in  $E_\infty$

(ii)  $\langle u_n \rangle_n$  is the bottom element of  $E_\infty$ :

Let  $\langle f_n \rangle_n \in E_\infty$

$(\forall n \in \omega) [u_n \text{ is the bottom element of } E_n] \implies$

$(\forall n \in \omega) [u_n \subseteq_n f_n] \implies$

$\langle u_n \rangle_n \subseteq \langle f_n \rangle_n \implies$

$\langle u_n \rangle_n$  is the bottom element of  $E_\infty$ ;

(2)  $\subseteq$  is a partial order:

(i)  $\subseteq$  is reflexive:

Let  $\langle f_n \rangle_n$  be in  $E_\infty$ ;

As  $f_n$  is in  $E_n$  which is a semantic domain,  $f_n \subseteq_n f_n$  (reflexivity)

$\implies (\forall n \in \omega) [f_n \subseteq_n f_n] \implies \langle f_n \rangle_n \subseteq \langle f_n \rangle_n$ .

Hence  $\subseteq$  is reflexive.

(ii)  $\subseteq$  is transitive:

Let  $\langle f_n \rangle_n$ ,  $\langle g_n \rangle_n$  and  $\langle h_n \rangle_n$  be in  $E_\infty$  such that:

$\langle f_n \rangle_n \subseteq \langle g_n \rangle_n$  and  $\langle g_n \rangle_n \subseteq \langle h_n \rangle_n \implies$

$(\forall n \in \omega) [f_n \subseteq_n g_n]$  and  $(\forall n \in \omega) [g_n \subseteq_n h_n] \implies$

$(\forall n \in \omega) [f_n \subseteq_n g_n \text{ and } g_n \subseteq_n h_n]$ , but  $\subseteq_n$  is transitive  $\implies$

$(\forall n \in \omega) [f_n \subseteq_n h_n] \implies \langle f_n \rangle_n \subseteq \langle h_n \rangle_n \implies$

$\subseteq$  is transitive.

(iii)  $\subseteq$  is antisymmetric:

Let  $\langle f_n \rangle_n$  and  $\langle g_n \rangle_n$  be in  $E_\infty$  such that:

$\langle f_n \rangle_n \subseteq \langle g_n \rangle_n$  and  $\langle g_n \rangle_n \subseteq \langle f_n \rangle_n$ . Then,

$(\forall n \in \omega) [f_n \subseteq_n g_n]$  and  $(\forall n \in \omega) [g_n \subseteq_n f_n] \implies$

$(\forall n \in \omega) [f_n \subseteq_n g_n \text{ and } g_n \subseteq_n f_n]$ , but  $\subseteq_n$  is antisymmetric.

$\implies (\forall n \in \omega) [f_n = g_n] \implies \langle f_n \rangle_n = \langle g_n \rangle_n$

Hence,  $\subseteq$  is antisymmetric.

Using (i), (ii) and (iii) we get that  $\subseteq$  is a partial order on  $E_\infty$ ;

(3) Every  $\omega$ -sequence has a limit in  $E_\infty$ ;

Let  $(X_m)_m$  be an  $\omega$ -sequence in  $E_\infty$

Every  $X_m$  is of the form  $\langle f_{nm} \rangle_n$  where:

$$f_{nm} \text{ is in } E_n \text{ and } \Psi_n(f_{n+1m}) = f_{nm}.$$

But  $(\forall m \in \omega) [X_m \subseteq X_{m+1}] \implies$

$$(\forall m \in \omega) [(\forall n \in \omega) [f_{nm} \subseteq_n f_{n+1m}]] \implies$$

$$(\forall n \in \omega) [(\forall m \in \omega) [f_{nm} \subseteq_n f_{n+1m}]] \implies$$

$$(\forall n \in \omega) [(f_{nm})_m \text{ is an } \omega\text{-sequence of } E_n].$$

$E_n$  is a semantic domain  $\implies (f_{nm})_m$  has a limit  $g_n = \bigcup_m f_{nm}$  in  $E_n$ .

Let  $X = \langle g_n \rangle_n$ , then  $X$  is the limit of  $(X_m)_m$  in  $E_\infty$ :

Proof:

(a)  $X$  is in  $E_\infty$ :

$$(\forall n \in \omega) [g_n \in E_n],$$

$$(\forall n \in \omega) [\Psi_n(g_{n+1}) = \Psi_n(\bigcup_m f_{n+1m})]$$

$$= \bigcup_m \Psi_n(f_{n+1m}) \text{ (as } \Psi_n \text{ is continuous).}$$

$$= \bigcup_m f_{nm} \text{ (as } X_m = \langle f_{nm} \rangle_n \implies \Psi_n(f_{n+1m}) = f_{nm}).$$

$$= g_n.$$

So  $X$  is in  $E_\infty$

(b)  $X$  is the limit of  $(X_m)_{m \in \omega}$  in  $E_\infty$ :

(i)  $X$  is an upper bound:

$$\text{Let } m \in \omega, \text{ then } X_m = \langle f_{nm} \rangle_n,$$

$$\text{Let us prove that } X_m \subseteq X.$$

$$\text{For } n \text{ in } \omega, X_{nm} = f_{nm} \subseteq_n g_n = \bigcup_m f_{nm} \implies$$

$$(\forall n \in \omega) [f_{nm} \subseteq_n g_n] \implies X_m \subseteq X \implies$$

$$(\forall m \in \omega) [X_m \subseteq X] \implies X \text{ is an upper bound of } (X_m)_m.$$

(ii)  $X$  is the least upper bound:

$$\text{Let } Y \text{ be an upper bound of } (X_m)_{m \in \omega} \text{ in } E_\infty \implies Y = \langle y_n \rangle_n.$$

For  $n \in \omega$ ,



$$(\forall m \in \omega) [X_m \subseteq Y] \implies$$

$$(\forall m \in \omega) [f_{nm} \subseteq_n y_n] \implies$$

$$y_n \text{ is an upper bound for } (f_{nm})_{m \in \omega} \implies$$

$$\bigcup_m f_{nm} \subseteq_n y_n \implies$$

$$(\forall n \in \omega) [g_n \subseteq_n y_n] \implies \langle g_n \rangle_n \subseteq \langle y_n \rangle_n \implies X \subseteq Y.$$

Therefore  $X$  is the least upper bound of  $(X_m)_m$  in  $E_\infty$ .

Using (1), (2) and (3) we get that  $E_\infty$  is a semantic domain.  $\square$

We define  $\Phi_{n\infty}: E_n \dashrightarrow E_\infty$  and  $\Phi_{\infty n}: E_\infty \dashrightarrow E_n$  such that:

$$\Phi_{n\infty}(f) = \langle \Phi_{nk}(f) \rangle_k \text{ and } \Phi_{\infty n}(f) = f_n.$$

*Lemma:*  $\Phi_{\infty n}$  is well defined and is continuous for all  $n \in \omega$ .

Proof:

(1) Well defined: Obvious.

(2) Continuous: Obvious, as being the  $n$ th projection.

*Lemma:*  $\Phi_{n\infty}$  is well defined and is continuous, for all  $n \in \omega$ .

Proof:

(1) Well defined:

$$\Phi_{n\infty}(f) = \langle \Phi_{nk}(f) \rangle_k,$$

$$(\forall k \in \omega) [\Phi_{nk}(f) \in E_k \text{ and } \Psi_k(\Phi_{nk+1}(f)) = \Phi_{nk}(f)]$$

$$\text{Therefore } \Phi_{n\infty}(f) \in E_\infty$$

(2) Continuous:

Let  $(f_m)_m$  be an  $\omega$ -sequence is in  $E_n$ .

$$\Phi_{n\infty}(\bigcup f_m) = \langle \Phi_{nk}(\bigcup f_m) \rangle_k$$

$$= \langle \bigcup_m \Phi_{nk}(f_m) \rangle_k \text{ (as } \Phi_{nk} \text{ is continuous)}$$

$$= \bigcup_m \langle \Phi_{nk}(f_m) \rangle_k \text{ (can be proved)}$$

$$= \bigcup_m \Phi_{n\infty}(f_m)$$

Hence  $\Phi_{n\infty}$  is continuous.  $\square$

*Lemma:*  $\Phi_{\infty n}(\Phi_{n\infty}(f)) = f$  for all  $n, f$ .

Proof:

$$\begin{aligned}\Phi_{\infty n}(\Phi_{n\infty}(f)) &= \Phi_{\infty n}(\langle \Phi_{nk}(f) \rangle_k) \\ &= \Phi_{nn}(f) = f. \quad \square\end{aligned}$$

Lemma:  $\Phi_{n\infty}(\Phi_{\infty n}(g)) \subseteq g$  for all  $n, g$ .

Proof:

$$\begin{aligned}\Phi_{n\infty}(\Phi_{\infty n}(g)) &= \Phi_{n\infty}(g_n) = \langle \Phi_{nk}(g) \rangle_k \\ &= \langle \Phi_{n0}(g_n), \Phi_{n1}(g_n), \dots, \Phi_{nn}(g_n), \Phi_{nn+1}(g_n), \dots \rangle \\ &= \langle \Psi_0 \dots \Psi_{n-1}(g), \Psi_1 \dots \Psi_{n-1}(g), \dots, g, \Phi_n(g), \dots \rangle \\ &= \langle g_0, g_1, \dots, g_n, \Phi_n(g_n), \dots \rangle \text{ (as } \Psi_n(g_{n+1}) = g_n) \\ (a) \quad (\forall m < n) \quad &[(\Phi_{n\infty}(\Phi_{\infty n}(g)))_{mth} = g_m \subseteq_m g_m] \text{ (refl.)}^{141}\end{aligned}$$

Actually this is not necessary.

(b) We shall prove that:

$$(\forall m \geq n) [(\Phi_{n\infty}(\Phi_{\infty n}(g)))_{mth} = \Phi_{nm}(g_n) \subseteq_m g_m]$$

$$(i) \quad m=n \quad \Phi_{nm}(g_n) = g_n \subseteq_n g_n = g_m.$$

(ii) Suppose that the property holds for  $m > n$ .

(iii) Prove it for  $m+1$ :

$$\Phi_{nm+1}(g_n) = \Phi_m \circ \Phi_{nm}(g_n).$$

$$\text{But } g_m = \Psi_m(f_{m+1}) \implies$$

$$\Phi_m(g_m) = \Phi_m(\Psi_m(g_{m+1})) \subseteq_{m+1} g_{m+1} \text{ but}$$

$$\Phi_{nm+1}(g_n) = \Phi_m(\Phi_{nm}(g_n)) \text{ hence}$$

$$\Phi_{nm+1}(g_n) \subseteq_{m+1} \Phi_m(g_m)$$

(by induction and monotonicity of  $\Phi_m$ ).

$$\text{Therefore } (\forall m \geq n) [(\Phi_{n\infty}(\Phi_{\infty n}(g)))_{mth} \subseteq_m g_m]$$

$$(a)+(b) \text{ implies } (\forall m \in \omega) [(\Phi_{n\infty}(\Phi_{\infty n}(g)))_{mth} \subseteq_m g_m \implies$$

$$\Phi_{n\infty}(\Phi_{\infty n}(g)) \subseteq g. \quad \square$$

<sup>141</sup> We use  $mth$  to say that it is element  $m$  in the sequence. If there was no parenthesis, then we can say  $f_n$ , meaning the  $n$ th element in the sequence  $(f_n)$ . However, for the above case, it will be confusing to say  $(\Phi_{n\infty}(\Phi_{\infty n}(g)))_m$ , as we will not understand whether we mean the sequence or the element.

*Lemma:*  $(\forall f \in E_\infty) [(f \in E_n) \implies (f = f_n)]$ .

Proof:  $f$  is in  $E_\infty \implies$

$$f = \langle \dots \Psi(f), f, \Phi(f), \Phi(\Phi(f)), \dots \rangle$$

And so  $f_n = f$ .  $\square$

*Lemma:*  $(\forall f \in E_\infty) [(f \in E_n) \implies (\Phi_n(f) = f)]$ .

Proof:  $(\Phi_n(f))$  is in  $E_\infty \implies \Phi_n(f)$  in  $E_\infty$  is written as:

$$\langle \dots, \Psi_n(\Phi_n(f)), \Phi_n(f), \Phi_{n+1}(\Phi_n(f)), \dots \rangle,$$

I.e.:  $(\Phi_n(f))_{k\text{th}} = \Phi_{n+1k}(\Phi_n(f))$ , but  $\circ$  is associative:

$$= (\Phi_{k-1} \circ \dots \circ \Phi_n)(f) \text{ for } k > n+1.$$

$$(\Phi_n(f))_{n+1\text{th}} = \Phi_{n+1n+1}(\Phi_n(f)) = \Phi_n(f).$$

$(\Phi_n(f))_{k\text{th}} = \Phi_{n+1k}(\Phi_n(f))$ , but  $\circ$  is associative:

$$= (\Psi_k \circ \dots \circ \Psi_n \circ \Phi_n)(f), \text{ but } \Psi_n \circ \Phi_n = \text{Id}_n:$$

$$= (\Psi_k \circ \dots \circ \Psi_{n-1})(f) \text{ for } k \leq n+1.$$

$f$  is in  $E_n \implies$

$$f_k = (f)_{k\text{th}} = \Phi_{nk}(f) = (\Phi_{k-1} \circ \dots \circ \Phi_n)(f) \text{ for } k > n+1.$$

$$f_{n+1} = (f)_{n+1\text{th}} = \Phi_{nn+1}(f) = \Phi_n(f)$$

$$f_k = (f)_{k\text{th}} = \Phi_{nk}(f) = (\Psi_k \circ \dots \circ \Psi_{n-1})(f) \text{ } k \leq n < n+1.$$

Therefore,  $f = \Phi_n(f)$ .  $\square$

*Lemma:*  $(\forall f \in E_\infty) [(f \in E_{n+1}) \implies (\Psi_n(f) \subseteq f)]$ .

Proof:  $(\Psi_n(f))_k = \Phi_{nk}(\Psi_n(f))$  for all  $k \in \omega$ , as  $\Psi_n(f)$  is in  $E_n$ .

$k > n+1 > n \implies$

$(\Psi_n(f))_k = (\Phi_{k-1} \circ \dots \circ \Phi_n)(\Psi_n(f))$ , but  $\circ$  is associative.  $\implies$

$$= (\Phi_{k-1} \circ \dots \circ \Phi_{n+1})(\Phi_n(\Psi_n(f)))$$

and  $f_k = (\Phi_{k-1} \circ \dots \circ \Phi_{n+1})(f)$

But  $\Phi_n(\Psi_n(f)) \subseteq_{n+1} f$  and  $\Phi_{k-1} \circ \dots \circ \Phi_{n+1}$  is monotonic  $\implies$

$$\Psi_n(f)_{k\text{th}} = (\Phi_{k-1} \circ \dots \circ \Phi_{n+1})(\Phi_n(\Psi_n(f)))$$

$$\subseteq_k (\Phi_{k-1} \circ \dots \circ \Phi_{n+1})(f) = f_k$$

Hence,  $(\forall k > n+1) [(\Psi_n(f))_{k\text{th}} \subseteq_k f_k]$ .

We conclude from here that:  $\Psi_n(f) \subseteq f$ .

Note: Here, we do not have to go through the cases

$k \leq n+1$ , But I shall do it for the sake of completeness:

$(\Psi_n(f))_n = \Psi_n(f)$  because  $\Psi_n(f)$  is in  $E_n$ .

$f_n = \Psi_n(f)$  because  $f$  is in  $E_{n+1}$ .

$(\Psi_n(f))_{n+1} = \Phi_n(\Psi_n(f)) \subseteq_{n+1} f$  and  $f_{n+1} = f$  ( $f$  in  $E_{n+1}$ )

$\implies (\Psi_n(f))_{n+1} \subseteq_{n+1} f_{n+1}$ .

$k \leq n \implies (\Psi_n(f))_k = (\Psi_k \circ \dots \circ \Psi_{n-1})(\Psi_n(f))$ ,  $\circ$  is associative:

$$= (\Psi_k \circ \dots \circ \Psi_n)(f)$$

$$= f_k$$

Therefore,  $(\forall k \geq 0) [(\Psi_n(f))_k \subseteq_k f_k] \implies \Psi_n(f) \subseteq f$ .  $\square$

*Lemma:* In  $E_\infty$   $f_{nm} = f_{\min(n,m)}$ .

Proof:  $f_{nm} = \Phi_{nm}(f)$

$n = m \implies f_{nm} = \Phi_{nn}(f_n) = f_n = f_{\min(n,m)}$ .

$n > m \implies f_{nm} = (\Psi_m \circ \dots \circ \Psi_{n-1})(f_n)$ , as  $\Psi_{n-1}(f_n) = f_{n-1}$

:

: By induction

:

$$= \Psi_m(f_{m+1}) = f_m = f_{\min(n,m)}$$

$n < m \implies f_{nm} = (\Phi_{m-1} \circ \dots \circ \Phi_n)(f_n)$ , as  $\Phi_n(f_n) = f_n$  for  $f_n$  in  $E_n$ .

But  $f_n = \Phi_n(f_n)$  is in  $E_{n+1} \implies \Phi_{n+1}(f_n) = f_n$ .

By induction we prove that  $\Phi_k(f_n) = f_n$  for all  $k \geq n$ .

Therefore,  $f_{nm} = f_n = f_{\min(n,m)}$ .  $\square$

*Lemma:*  $n \leq m$  then  $f_n \subseteq f_m \subseteq f$ .

Proof:

(a)  $f_n \subseteq f_{n+1}$  for all  $n \geq 0$ :

We proved that if  $f$  is in  $E_{n+1}$  then  $\Psi_n(f) \subseteq f$ ,

So as  $f_{n+1}$  is in  $E_{n+1}$ ,  $\Psi_n(f_{n+1}) \subseteq f_{n+1}$ .

But  $\Psi_n(f_{n+1}) = f_n$ . So  $f_n \subseteq f_{n+1}$ .

(b) We could therefore prove by induction that:

$$(\forall m \geq n \geq 0) [f_n \subseteq f_m].$$

This is by using (a)+transitivity+reflexivity of  $\subseteq$ .

(c)  $f = \langle f_n \rangle_n$ . So to prove  $f_n \subseteq f$ , we have to prove that:

$$(\forall i \in \omega) [(f_n)_i \subseteq_{1i} f_i].$$

Let  $i \in \omega$ ,

$$\begin{aligned} f_{ni} &= f_{\min(n,i)} \\ &= (n < i) \text{----} \times f_n, f_i. \end{aligned}$$

But  $f_n \subseteq_{1i} f_i$  for  $n < i$ , and  $f_i \subseteq_{1i} f_i \implies$

$$(\forall i \in \omega) [f_{ni} \subseteq_{1i} f_i] \implies f_n \subseteq f.$$

Therefore,  $(\forall m \geq n \geq 0) [f_n \subseteq f_m \subseteq f]$ .  $\square$

*Lemma:*  $f = \cup f_n$

Proof: We proved above that:  $(\forall n \leq m) [f_n \subseteq f_m]$ .

Therefore  $(f_n)_n$  is an  $\omega$ -sequence in  $E_\infty$

and so its limit  $\cup f_n$  exists in  $E_\infty$ .

We also proved that:  $(\forall n \in \omega) [f_n \subseteq f]$ .

Therefore  $f$  is an upper bound of  $(f_n)_n$  in  $E_\infty$ .

Let us prove that  $f = \cup f_n$ :

Each  $f_n$  is of the form  $\langle (f_n)_i \rangle_i = \langle f_{ni} \rangle_i$ ,

$$\text{So } \cup f_n = \cup_n \langle f_{ni} \rangle_i$$

$$= \langle \cup_n f_{ni} \rangle_i$$

$$= \langle \cup_n f_{n0}, \cup_n f_{n1}, \dots \rangle$$

But  $(\forall n \geq 0) [f_{n0} = f_0] \implies$

$\cup_n f_{n0} = f_0$ . The same for  $\cup_n f_{nk} = f_k$  for all  $k \geq 0$ .

and we get:  $\cup f_n = \langle f_0, f_1, \dots \rangle = \langle f_1 \rangle_1 = f$ .  $\square$

Now we define application in  $E_\infty$

Let  $f, e$  be in  $E_\infty$  and define  $f \bullet e = \cup f_{n+1}(e_n)$

*Lemma:* Application is well defined.

Proof:

(i) This is because if  $f_{n+1}$  is in  $E_{n+1}$  and  $e_n$  in  $E_n$  then:

$$f_{n+1} \bullet e_n = f_{n+1}(e_n).$$

Because:

$$\begin{aligned} f_{n+1} \bullet e_n &= \cup_m (f_{n+1})_{m+1}((e_n)_m) \\ &= \cup_{m > n} (f_{n+1})_{m+1}((e_n)_m) \\ &= \cup f_{n+1}(e_n) \text{ as } f_{nm} = f_{\min(n,m)} \\ &= f_{n+1}(e_n) \end{aligned}$$

Therefore it makes sense to talk of  $f_{n+1} \bullet e_n$  as application

from  $E_{n+1} \times E_n \rightarrow E_n$ .

(ii) We know that we are talking about limits here for  $\omega$ -sequences,

therefore we have to show that  $(f_{n+1}(e_n))_n$  is an  $\omega$ -sequence.

Proof:

$$\begin{aligned} e_0 \subseteq e_1 \text{ and } f_1 \subseteq f_2 &\implies \\ f_1(e_0) \subseteq f_2(e_0). \text{ But } f_2 \text{ is continuous } &\implies \text{monotonic} \implies \\ f_2(e_0) \subseteq f_2(e_1). \text{ But } \subseteq \text{ is transitive } &\implies \\ f_1(e_0) \subseteq f_2(e_1). & \end{aligned}$$

By induction we can assume that  $f_n(e_{n-1}) \subseteq f_{n+1}(e_n)$ .

We prove that  $f_{n+1}(e_n) \subseteq f_{n+2}(e_{n+1})$ :

$$\begin{aligned} e_n \subseteq e_{n+1} \text{ and } f_{n+1} \text{ monotonic } &\implies \\ f_{n+1}(e_n) \subseteq f_{n+1}(e_{n+1}). \text{ But } f_{n+1} \subseteq f_{n+2} &\implies \\ f_{n+1}(e_{n+1}) \subseteq f_{n+2}(e_{n+1}). \text{ But } \subseteq \text{ is transitive } &\implies \end{aligned}$$

$$f_{n+1}(e_n) \subseteq f_{n+2}(e_{n+1}).$$

Therefore  $(f_{n+1}(e_n))_n$  is an  $\omega$ -sequence in  $E_\infty$  and

$$\cup f_{n+1}(e_n) \text{ exists in } E_\infty.$$

Therefore the application from  $E_\infty \times E_\infty$  to  $E_\infty$  as above is well defined.

*Lemma:* Let  $f: D_1 \times D_2 \rightarrow D_3$  (3 semantic domains),

Then  $f$  is continuous iff

$f$  is continuous in terms of its arguments taken separately.

Proof:

(a) Assume  $f$  is continuous and let  $a$  in  $D_1$ .

We will show that  $f_a: D_2 \rightarrow D_3$  such that  $f_a(x) = f(a, x)$  is continuous.

$$\begin{aligned} f_a(\cup x_n) &= f(a, \cup x_n) = f(\cup a, \cup x_n) = f(\cup (a, x_n)). \text{ But } f \text{ is continuous } \implies \\ &= \cup f(a, x_n) \\ &= \cup f_a(x_n). \end{aligned}$$

We can do the same to prove that  $g_a(x) = f(x, a)$  is continuous.

(b) Assume  $f$  is continuous in terms of each of its arguments separately.

$$\begin{aligned} f(\cup a_n) &= f(\cup (x_n, y_n)) = f(\cup x_n, \cup y_n). \text{ But } f \text{ is continuous in terms of its 1st argument:} \\ &= \cup_n f(x_n, \cup y_n). \text{ But } f \text{ is continuous in terms of its 2nd argument:} \\ &= \cup_n \cup_m f(x_n, y_m). \\ &= \cup f(x_n, y_n). \quad \square \end{aligned}$$

*Lemma:* Let  $D_1, D_2$  be two semantic domains and  $f: D_1 \rightarrow D_2$ .

$f$  is continuous  $\implies f$  is monotonic.

Proof: Take  $x, y$  in  $D_1$  such that  $x \subseteq_1 y$ .

we construct the  $\omega$ -sequence  $(x_n)_n$  where  $x_0 = x$  and  $x_n = y \forall n > 0$ .

This is obviously an  $\omega$ -sequence and its limit is  $y = \cup x_n$ .

But  $f$  is continuous  $\implies f(y) = f(\cup x_n) = \cup f(x_n)$ .

$(f(x_n))_n$  is an  $\omega$ -sequence which has  $\cup f(x_n)$  as a limit  $\implies$

$f(x_0) = f(x) \subseteq_2 \cup f(x_n)$ . Therefore  $f(x) \subseteq_2 f(y)$  and so

every continuous function is monotonic.  $\square$

*Lemma:* Application  $\text{App}: E_\infty \times E_\infty \rightarrow E_\infty$  is continuous.

Proof: we have that  $g: E_{n+1} \times E_{n+1} \rightarrow E_{n+1}$  such that

$$g(f_{n+1}, e_n) = f_{n+1}(e_n) \text{ is continuous.}$$

$$\text{We also have that } \text{App}(f, e) = \bigcup_{n+1} f_{n+1}(e_n).$$

So  $\text{App}$  is the limit of continuous functions.

Therefore  $\text{App}$  is continuous.  $\square$

*Lemma:*  $(\forall m \geq n) [(f_{n+1 m+1}(e_m)) = f_{n+1}(e_n)]$

Proof: By induction.

(1)  $m=n$ : obvious.

(2) assume the property holds for  $m$  and prove it for  $m+1$ .

$$\begin{aligned} (3) ((f_{n+1})_{m+2}(e_{m+1})) &= (\Phi_{m+1}((f_{n+1})_{m+1}))(e_{m+1}) \\ &= (\Phi_m \circ (f_{n+1})_{m+1} \circ \Psi_m)(e_{m+1}). \text{ But } \circ \text{ is associative:} \\ &= \Phi_m((f_{n+1})_{m+1}(\Psi_m(e_{m+1}))). \text{ But } \Psi_m(e_{m+1}) = e_m: \\ &= \Phi_m((f_{n+1})_{m+1}(e_m)). \end{aligned}$$

But the argument of  $\Phi_m$  is in  $E_m$ :

$$\begin{aligned} &= (f_{n+1})_{m+1}(e_m). \text{ By induction hypotheses we get:} \\ &= f_{n+1}(e_n). \end{aligned}$$

Therefore,  $(\forall m \geq n) [(f_{n+1})_{m+1}(e_m) = f_{n+1}(e_n)]. \square$

*Lemma:*  $f_{n+1} \bullet e = f_{n+1}(e_n)$

$$\begin{aligned} \text{Proof: } f_{n+1} \bullet e &= \bigcup_m ((f_{n+1})_{m+1}(e_m)) \\ &= \bigcup_{m \geq n} ((f_{n+1})_{m+1}(e_m)) \\ &= \bigcup_{m \geq n} f_{n+1}(e_n), \text{ by previous lemma.} \\ &= f_{n+1}(e_n), \text{ by independence of } m. \square \end{aligned}$$

*Lemma:*  $(\forall m \geq n) [(f_{m+1}(e_{nm}))_n = f_{n+1}(e_n)]$

Proof:



(1)  $m=n \implies (f_{m+1}(e_{nm}))_n = f_{n+1}(e_n)$ .

(2) Assume the property holds for  $m$ .

(3)  $(f_{m+2}(e_{nm+1}))_n = \Phi_{m+1n}(f_{m+2}(e_{nm+1}))$ . But  $m > n$ :

$$\begin{aligned} &= \Phi_{mn}(\Psi_m(f_{m+2}(e_{nm+1}))) \\ &= \Phi_{mn}((\Psi_m \circ f_{m+2} \circ \Phi_m)(e_{nm})) \\ &= \Phi_{mn}((\Psi_{m+1}(f_{m+2}))(e_{nm})) \\ &= \Phi_{mn}(f_{m+1}(e_{nm})) \\ &= (f_{m+1}(e_{nm}))_n. \text{ But by induction:} \\ &= f_{n+1}(e_n). \end{aligned}$$

Therefore  $f_{n+1}(e) = f_{n+1}(e_n) = (f(e_n))_n$ .  $\square$

*Lemma:*  $(f(e))_n = f_{n+1}(e_n)$

*Proof:*  $(f(e_n))_n = (\cup_m f_{m+1}(e_{nm}))_n$   
 $= \cup_m (f_{m+1}(e_{nm}))_{nth}$  But by above lemma:  
 $= \cup_m f_{n+1}(e_n)$   
 $= f_{n+1}(e_n)$ .

Therefore,  $f_{n+1} \bullet e = f_{n+1}(e_n) = (f \bullet e_n)_{nth}$ , for all  $n \in \omega$ .  $\square$

We have still not proved that  $E_\infty \cong [E_\infty \dashrightarrow E_\infty]$ .

Let us see how to do it.

First we shall need the following theorem:

*Theorem:*  $(\forall f \in [E_\infty \dashrightarrow E_\infty]) (\exists Xf \in E_\infty) ((\forall e \in E_\infty) [f(e) = Xf \bullet e])$ .

*Proof:* Let  $f$  be in  $E_\infty$  and take  $Xf = \cup(\lambda y \in E_n (f(y))_n)$ .

$Xf$  is what we are looking for because:

(i)  $Xf$  is well defined and is in  $E_\infty$  (Obvious).

(ii) Let  $e$  be in  $E_\infty$

$$\begin{aligned} Xf \bullet e &= \cup_m Xf_{m+1}(e_m), \text{ by definition of application.} \\ &= \cup_m (Xf \bullet e_m)_m, \text{ by above lemma.} \\ &= \cup_m ((\cup_n (\lambda y \in E_n (f(y))_n)) \bullet e_m)_m, \text{ by definition of } Xf. \end{aligned}$$

$$\begin{aligned}
 &= \bigcup_m (\bigcup_n ((\lambda y \in E_n(f(y)))_n) \bullet e_m)_m, \text{ by continuity in terms of } \bigcup_n. \\
 &= \bigcup_m \bigcup_n ((\lambda y \in E_n(f(y)))_n) \bullet e_m)_m, \text{ by continuity in terms of } \bigcup_m. \\
 &= \bigcup_k ((\lambda y \in E_k(f(y)))_k) \bullet e_k)_k, \text{ by a lemma.} \\
 &= \bigcup_k (f(e_k))_k, \text{ by } \lambda\text{-conversion.} \\
 \\
 &= \bigcup_{n,m} (f(e_m))_n \text{ by a lemma.} \\
 &= \bigcup_n (\bigcup_m f(e_m))_n \text{ by continuity.} \\
 &= \bigcup_n (f(\bigcup_m e_m))_n, \text{ by continuity.} \\
 &= \bigcup_n (f(e))_n. \\
 &= f(e), \text{ by continuity of } f. \square
 \end{aligned}$$

*Theorem:*  $E_\infty \cong [E_\infty \dashrightarrow E_\infty]$ .

*Proof:* Let  $R: [E_\infty \dashrightarrow E_\infty] \dashrightarrow E_\infty$  such that  $R(f) = Xf$  (as above).

(i)  $R$  is well defined: obvious.

(ii)  $R$  is injective:

$$R(f) = R(g) \implies Xf = Xg \implies f = g?$$

$$f(e) = Xf \bullet e = Xg \bullet e = g(e), \text{ Assuming extensionality we get } f = g.$$

(iii)  $R$  is surjective: obvious.

(iv)  $R$  is continuous:

$$\begin{aligned}
 R(\bigcup_n f_n) &= X \bigcup_n f_n \\
 &= \bigcup_m (\lambda y \in E_m (\bigcup_n f_n(y)))_m, \text{ using continuity } \implies \\
 &= \bigcup_m (\bigcup_n (\lambda y \in E_m (f_n(y)))_m), \text{ using limit } \implies \\
 &= \bigcup_n (\bigcup_m (\lambda y \in E_m (f_n(y)))_m), \text{ using definition of } Xf_n \implies \\
 &= \bigcup_n Xf_n. \\
 &= \bigcup_n R(f_n).
 \end{aligned}$$

Therefore  $R$  is an isomorphism and so we have:

$$E_\infty \cong [E_\infty \dashrightarrow E_\infty]. \square$$

Actually,  $E_\infty$  is the least upper bound of the sequence of domains  $(E_n)_n$ , but as it

takes a lot of work to prove it, I shall ignore the proof.

That was a short introduction to how one can define the Scott domain  $E_\infty$ . However, what I have not given is a definition of the model of a lambda calculus in general. For such a definition, the reader is referred to [ME1].<sup>142</sup>

---

<sup>142</sup> For those who are familiar with category theory, it is worth reminding them of the categorical side of this construction. They would have guessed already that  $\Phi$  is a contraction if the category is the category of domains and continuous functions and that  $E_\infty = (E_1, \Psi_1)$ .

## BIBLIOGRAPHY

- [AB1] Abbott J.C (1969), *Sets, Lattices and Boolean Algebras*; Allyn and Bacon, Inc., Boston.
- [AC1] Aczel P. (1974), The strength of Martin-Löf's intuitionistic type theory with one universe; *Proceeding of the Symposium on Mathematical Logic*, in Oulu 1974, Report No 2 of Department of Philosophy, University of Helsinki (1977), pp 1-32.
- [AC2] Aczel P. (1977), The type theoretic interpretation of constructive set theory; *Logic Colloquium '77*, edited by A. MacIntyre, L. Pacholski and J. Paris, North-Holland, Amsterdam, pp 55-66.
- [AC3] Aczel P. (1980), Frege structures and the notions of proposition, truth and set; *The Kleene Symposium*, Edited by Barwise et.al, Studies in Logic 101, North-Holland, NewYork, pp 31-60.
- [AC4] Aczel P. (1981), A formal language; privately circulated note, Manchester University.
- [AC5] Aczel P. (1984), Lecture notes on non-well founded sets; Lecture notes at Stanford University.
- [AC6] Aczel P. (1985), Properties and propositional functions; privately circulated note, Manchester University.
- [AL1] Allen J.F and Perrault C.R (1980), Analysing intensions in utterances; *Artificial Intelligence 15*, pp 143-178.
- [AL2] Allen J.F (1981), A general model of action and time; privately circulated note, University of Rochester.
- [AL3] Allen J.F (1981), An interval-based representation of temporal knowledge; *International Journal of Computing and Artificial Intelligence '81*, pp 221-226.
- [AL4] Allen J.F and Koomen J.A (1983), Planning using a temporal world model; *International Journal of Computing and Artificial Intelligence '83*, pp 741-747.
- [AL5] Allen J.F (1984), Towards a general theory of Action and Time; *Artificial Intelligence 23*, pp 123-154.
- [AL6] Allison L. (1984), Denotational semantics; Computer Science 4, University of Edinburgh.
- [AV1] Avron A. (1987), Simple Consequence relations; Talk given at the conference on General Logic in Computer Science, March '87, Edinburgh University.
- [BA1] Barendregt H.P (1981), The type free lambda calculus; *Handbook of Mathematical Logic* edited by J. Barwise. North-Holland, New York, pp 1091-1132.
- [BA2] Barendregt H.P (1981), *The lambda calculus: its syntax and semantics*. North-Holland, Amsterdam.
- [BA3] Barwise J. and Cooper R. (1981), Generalised quantifiers and natural language.

*Linguistics & Philosophy* 4, pp 159-219.

[BA4] Barwise J. and Perry J. (1983), *Situations and Attitudes*. MIT Press.

[BA5] Backhouse R. (1986), Notes on Martin-Löf's theory of types; Sections 1 and 2. University of Essex. CSM 81-82.

[BE1] Van Benthem J. (1984), Questions about quantifiers; *Journal of Symbolic Logic* 49, pp 443-469.

[BE2] Bethke I. (1986), How to construct extensional Combinatory Algebras; *Proceedings of the Koninklijke Nederlandse Akademie Van Wetenschappen* 89, pp 243-257.

[BE3] Beeson M.J (1978), Continuity in intuitionistic set theories; *Logic Colloquium '78*, edited by M. Boffa et al, North-Holland, Amsterdam, pp 1-52.

[BE4] Beeson M.J (1987), *Foundations of constructive Mathematics*; Springer-Verlag, Berlin.

[BE5] Ben-Chorin S. (1982), Sentence meanings as a formal basis for intensional Logic; *Proceedings of the first West Coast Conference on Formal Linguistics*, Stanford University.

[BE6] Bealer G. (1982), *Quality and concept*; Clarendon press, Oxford.

[BE7] Van Benthem J. (1983), Determiners and logic; *Linguistics & Philosophy* 6, pp 447-478.

[BI1] Bishop E. (1967), *Foundations of constructive analysis*; McGraw-Hill, New York.

[BO1] Boolos G. (1971), The iterative conception of set; *Journal of Philosophy* LXVIII, pp 215-231.

[BO2] Boolos G. and Jeffrey R. (1980). *Computability and Logic*; 2nd edition, Cambridge University Press.

[BO3] Boolos G. (1982) Extremely undecidable sentences; *Journal of Symbolic Logic* 47, pp 191-197.

[BU1] Burgess J.P (1986), The Truth is never simple; *Journal of Symbolic Logic* 51, pp 663-681.

[CA1] Carlson G. (1987), *Reference to kinds in English*; Ph.D. Dissertation, University of Massachusetts.

[CA2] Carnap (1947), *meaning and necessity*; Chicago press.

[CH1] Chierchia G. (1982), Nominalisation and Montague grammar: A semantics without types for natural languages; *Linguistics and Philosophy* 5, pp 303-354.

[CH2] Chierchia G. (1982), Bare plurals, Mass nouns and Nominalisation; In D.Flickinger, M.Macken and N.Wiegand (eds.), *Proceedings of West Coast Conference on Formal Linguistics I*, Stanford, California.

- [CH3] Chierchia G. (1984), *Topics in the Syntax and Semantics of infinitives and Gerunds*; PhD dissertation, University of Massachusetts.
- [CH4] Chierchia G. and Turner R. (1985), *Semantics and property theory*; Essex/Massachusetts.
- [CH5] Chierchia G. (1985), *Formal semantics and the grammar of predication*; *Linguistic Inquiry* 16, pp 417-443.
- [CH6] Church A. (1940), *A formulation of the simple theory of types*; *Journal of Symbolic Logic* 5, pp 56-68.
- [CH7] Church A. (1941), *Calculi of lambda-conversion*; *Annals of Mathematical Studies* 6, Princeton University Press, Princeton.
- [CH8] Church A. (1956), *Introduction to Mathematical Logic I*; Princeton University Press, Princeton.
- [CO1] Cocchiarella N.B (1984), *Frege's Double Correlation Thesis and Quine's set theories NF and ML*; *Journal of Philosophical Logic* 13.
- [CO2] Cocchiarella N.B (1986), *Conceptualism, Ramified Logic and Nominalised predicates*; privately circulated note, Indiana University.
- [CO3] Cocchiarella N.B (1986), *Philosophical Perspectives on Formal theories of Predication*; *Handbook of Philosophical Logic* 4, D. Reidel, Dordrecht.
- [CO4] Coquand Th. (1985), *An analysis of Girard's paradox*; CMU and INRIA, privately circulated note.
- [CO5] Constable R. and Howe D. (1987), *NUPRL as a framework for defining logics*; technical report, Computer Science, Cornell University.
- [CU1] Curry H.B and Feys R. (1968), *Combinatory Logic I*; North-Holland, Amsterdam.
- [DA1] Van Dalen D. (1985), *Intuitionistic Logic*; *Handbook of Philosophical Logic* 3, D.Reidel, Dordrecht.
- [DI1] Diller J. (1980), *To H.B.Curry: Essays on Combinatory Logic, lambda calculus and formalism*; edited by J.P. Seldin and J. R. Hindley, Academic press, New York.
- [DO1] Dowty D., Wall R. and Peters S. (1981), *Introduction to Montague semantics*; D.Reidel, Dordrecht.
- [DU1] Dummett M. (1977), *Elements of intuitionism*; Clarendon press, Oxford.
- [DU2] Dummett M. (1981), *Frege philosophy of language*; Duckworth.
- [DU3] Dunn J.M. and Belnap N.D (1968), *The substitution interpretation of the quantifiers*; *Noûs* 2, pp 177-185.
- [EN1] Enderton H.B (1972), *A Mathematical introduction to Logic*; Academic press, New York, London.

- [FE1] Feferman S. (1975), A language and axioms for explicit Mathematics; *Algebra & Logic*, lecture notes in mathematics 450, Springer, Berlin, pp 87-139.
- [FE2] Feferman S (1979), Constructive theories of functions and classes; In M.Boffa, D.Van Dalen, K. McAloon (eds), *Logic Colloquium '78*, edited by M. Boffa et al, North-Holland, Amsterdam, pp 159-224.
- [FE3] Feferman S. (1981), Working foundations; A revised version of a paper presented to the workshop *The present state of the problem of foundations of Mathematics*, Florence.
- [FE4] Feferman S. (1981), A theory of variable types; privately circulated note, Stanford University.
- [FE5] Feferman S. (1982), Inductively presented systems and the formalisation of Meta-Mathematics; *Logic Colloquium '80*, edited by D. Lascar and J. Smiley, North-Holland, Amsterdam, pp 95-128.
- [FE6] Feferman S (1983), Intensional Mathematics; *Logic Colloquium '83*.
- [FE7] Feferman S. (1984), Between constructive and classical Mathematics; in *Logic Colloquium '83*.
- [FE8] Feferman S (1984), Intensionality in Mathematics; For the symposium on intensions and set theory; Meeting of the Pacific Division of the American Philosophical Association.
- [FE9] Feferman S (1984), Towards useful type-free theories I; *Journal of Symbolic Logic* 49, pp 75-111.
- [FI1] Fine K. (1974), Models for entailments; *Journal of Philosophical Logic* 3, pp 347-372.
- [FI2] Fine K. (1984), Critical review of Parsons' non-existent objects; *Philosophical Studies* 45, pp 95-142.
- [FL1] Flagg R. and Myhill J. (1984), Notes on a type-free system extending ZFC; privately circulated note, Ohio State University and University of New York at Buffalo.
- [FL2] Flagg R. and Myhill J. (1984), An extension of Frege structures; privately circulated note, Ohio State University and University of New York at Buffalo.
- [FL3] Flagg R. and Myhill J. (1984), Implication and Analysis in classical Frege Structures; privately circulated note, Ohio State University and University of New York at Buffalo.
- [FL4] Flagg R. (1984), K-continuous lattices and Comprehension principles for Frege Structures; privately circulated note, Ohio State University.
- [FL5] Flagg R. (1985), Church's thesis is consistent with epistemic arithmetic; *Intensional Mathematics*; edited by S. Shapiro, Elsevier Science publishers B.V., North-Holland.

- [FO1] Font J.M (1985), *Modality and Possibility in some intuitionistic modal logics*; Faculty of Mathematics, University of Barcelona.
- [FR1] Fraenkel A. (1966), *Set theory and Logic*; Addison-Wesley, U.S.A.
- [FR2] Fraenkel A., Bar-Hillel Y. and Levy A. (1973), *Foundations of set theory*; North-Holland, Amsterdam.
- [FR3] Frege G. (1970), *Translations from the philosophical writings of Frege*; Edited by Peter Geach and Max Black, Basil Blackwell, Oxford.
- [GA1] Gallin D. (1975), *Intensional and higher-order modal Logic*; North-Holland, Mathematics Studies 19.
- [GE1] Geminiani M., *Elementary topology*; Addison-Wesley, U.S.A.
- [GO1] Gödel K. (1962), *On formally undecidable propositions of Principia Mathematica and related systems*; Translated by Meltzer, 1st English edition, Oliver and Boyd.
- [GO2] Gordon M. (1979), *The denotational description of programming languages, An introduction*; Springer-Verlag, New York.
- [GO3] Goodman N. (1985), *A Genuinely Intensional set theory*; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V., North-Holland.
- [GU1] Gupta A. (1982), *Truth and Paradox*; *Journal of Philosophical Logic II*, pp 1-60.
- [HA1] Hanna and Saber (1971), *Sets and Logic*; Irwin, Inc.
- [HA2] Harper R. (1986), *Introduction to standard ML*; technical report, Computer Science, University of Edinburgh.
- [HE1] Henderson P. (1980), *Functional programming, Application and Implementation*; Prentice Hall-International Series in Computer Science, London.
- [HE2] Henkin L. (1950), *Completeness in the theory of types*; *Journal of Symbolic Logic 15*, pp 81-91.
- [HE3] Heyting A. (1966), *Intuitionism, An introduction*; North-Holland, Amsterdam.
- [HE4] Herzberger, H. (1982), *Notes on Naive semantics*; *Journal of Philosophical Logic II*, pp 62-102.
- [HI1] Hindley J.R and Seldin J.P (1986), *Introduction to Combinators and Lambda-Calculus*; Cambridge University Press, Cambridge.
- [HI2] Hintikka K.J (1956), *Identity, variables, and impredicative definitions*; *Journal of Symbolic Logic 21*, pp 225-245.
- [HI3] Hintikka K.J (1975), *Impossible possible worlds vindicated*; *Journal of Philosophical Logic 4*, pp 475-484.
- [HO1] Hodges W. (1975), *Logic*; Penguin books, London.



- [HU1] Hughes G.E and Cresswell M.J (1968), *An introduction to modal Logic*; Methuen.
- [KA1] Kamp, H. (1979), Events, Instants and temporal reference; edited by Bauerle, Egli and von Stechow. *semantics from different points of view*, De Gruyter: Berlin pp 376-471
- [KA2] Kamp H. (1983), A Scenic tour through the land of naked infinitives; unpublished.
- [KA3] Kamp H. (1981), A theory of Truth and semantic representation; In Groenendijk, J.T. Janssen and M. Stokhof eds, *Formal methods in the study of language*, Amsterdam: Mathematics centrum.
- [KA4] Kamareddine F. (1987), Second order quantifiers in Frege structures; in preparation.
- [KA5] Kamareddine F. (1987), Theories of truths in Frege structures; in preparation.
- [KA6] Kamareddine F. (1988), Interpreting IL in a two-levelled theory.
- [KE1] Kelley J.L (1955), *General Topology*; D.Van Nostrand co., Princeton.
- [KE2] Keenan E.L (1982), Eliminating the universe, A study in ontological perfection; *Proceedings of the First West Coast Conference on Formal Linguistics*; Stanford University.
- [KL1] Klein E. (1977), On sentences which report beliefs, desires and other mental attitudes; PhD dissertation, University of Cambridge.
- [KL2] Kleene, S.C (1952), *Introduction to Metamathematics*; D.Van Nostrand co, Inc., Princeton
- [KL3] Kleene, S.C (1968), *Mathematical Logic*; Wiley&Sons, Inc.
- [KO1] Kowalski R. and Sergot M (1984), A Logic-based calculus of events; technical report, Imperial college, London University.
- [KO2] Kock A. and Wraith G.C (1971), *Elementary toposes*; Lecture notes series 30, Aarhus University.
- [KR1] Kripke S.A (1963), Semantical considerations on Modal Logic; In *Reference and Modality*, L.Linsky.
- [KR2] Kripke S.A (1975), Outline of a theory of Truth; *The Journal of Philosophy LXXII*, pp 690-716.
- [KR3] Kripke S.A, A puzzle about belief; in *Margalit*, pp 239-283.
- [LA1] Lambek J & Scott P.J (1986), *Introduction to higher order categorial Logic*; Cambridge University Press, Cambridge.
- [LE1] Lemmon E.J (1965), *Beginning Logic*; Nelson's University Paperbacks series, London.

- [LI1] Link G., Plural; to appear in the *Handbook of Semantics*, edited by D. Wunderlich and A. V. Stechow.
- [LI2] Link G. (1983), The logical analysis of plural and mass terms: A lattice theoretic approach; In *Meaning, Use and Interpretation of Language*, De Gruyter, pp 302-323.
- [LI3] Link G. (1986), Generalised quantifiers and plurals; CSLI.
- [LI4] Lifschitz V. (1985), Calculable natural numbers; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V. North-Holland.
- [LL1] Lloyd J.W (1984), *Foundations of Logic programming*; Springer-Verlag, Berlin Heidelberg New-York Tokyo.
- [LO1] Longo G. (1983), Set theoretical models of Lambda Calculus: Theories, Expansions, Isomorphisms; *Annals of Pure and Applied Logic 24*, pp 153-188.
- [MA1] Martin-Löf P. (1973), An intuitionistic theory of Types: Predicative part; *Logic Colloquium '73*, edited by H.E. Rose and J.C. Shepherdson, North-Holland, Amsterdam, pp 73-118.
- [MA2] Martin-Löf P. (1980), Intuitionistic Type Theory; Notes by Giovanni Sambin of a series of lectures given in Padova.
- [MC1] McCarty D.C (1983), Intuitionism: an introduction to a seminar; *Journal of Philosophical Logic 12*, pp 105-149.
- [MC2] McCarty D.C (1984), Subcountability and Realisability; technical report at Monash University.
- [MC3] McDermott D. (1982), A temporal logic for reasoning about processes and plans; *Cognitive Science 6*, pp 101-155.
- [ME1] Meyer A. (1981), What is a model of the Lambda Calculus? Unpublished ms., M.I.T. Lab., Computer Science.
- [MO1] Mönnich U. (1983), Toward a calculus of concepts as a semantical metalanguage; In *Meaning, Use and Interpretation of Language*, De Gruyter, pp 342-360.
- [MY1] Myhill J. (1975), Constructive set theory, *Journal of Symbolic Logic 40*, pp 347-382.
- [MY2] Myhill J. (1985), Intensional set theory; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V. North-Holland.
- [NA1] Nagel E. and Newman J. (1959), Gödel's Proof; Routledge & Kegan Paul.
- [PA1] Parsons C. (1971), A plea for substitutional quantification; *The Journal of Philosophy LXVIII*, pp 231-237.
- [PA2] Partee B. (1986), Noun phrase Interpretation and type shifting principles; *Studies in discourse representation and the theory of generalised quantifiers*, edited by J. Groenendijk and M. Stokhof, Grass 8, Foris, Dordrecht.

- [PA4] Partee B. (1986), Ambiguous pseudoclefts with unambiguous Be; To appear in *Proceedings of North Eastern Linguistic Society 16*, edited by Steve Berman, Joe Choe and Joyce McDonough, GLSA.
- [PA5] Parsons T. (1979), The theory of types and ordinary language; in Davies S. and Mithun M. (eds), *Linguistics, Philosophy and Montague Grammar*, University of Texas Press, Austin
- [PE1] Peirce (1955), *Philosophical Writings*; selected and edited by Buchler, Dover publications, New York.
- [PE2] Perlis D. (1987), *Languages with Self-Reference II: Knowledge, Belief and Modality*; University of Maryland.
- [PO1] Poincaré H. (1902), Du rôle de l'intuition et de la logique en mathématiques; *C.R. du 11<sup>me</sup> Congr. Intern. des Math.*, Paris 1900, pp 200-202.
- [QU1] Quine W.V (1952), *Methods of Logic*; Routledge & Kegan Paul ltd.
- [QU2] Quine W.V (1964), New foundations for Mathematical Logic; In *From a logical point of view, Nine Logico-philosophical essays*, Harvard University Press, Harvard.
- [QU3] Quine W.V (1969), *Set theory and its Logic*; Belknap Press, Harvard.
- [RO1] Partee B. and Rooth M. (1983), Generalised conjunction and type ambiguity; *Meaning, Use and Interpretation of Language*, De Gruyter, pp 361-381.
- [RU1] Russell B. (1908), Mathematical logic as based on the theory of types; *American Journal of of Math.* 30, pp 222-262.
- [RU2] Russell B. (1905), On denoting; in *Logic and Knowledge, Essays 1901-1950*, edited by R. Marsh, Capricorn books, New York.
- [SC1] Scott D.S (1968), Constructive validity; *Symposium on automatic demonstration, Lecture notes in Mathematics 25*, Springer-Verlag, Berlin.
- [SC2] Scott D.S (1975), Combinators and classes; In *Lambda Calculus and Computer Science*, edited by C. Böhm, Lecture notes in computer Science 37, Springer, Berlin. pp.1-26.
- [SC3] Scott D.S (1976), Data types as Lattices; Technical Monograph PRG-5, *Siam Journal on Computing* 5, pp 522-587.
- [SC4] Scott D.S (1980), Lambda calculus: Some models, some philosophy; edited by J. Barwise, H.J. Keisler and K. Kunen, *The Kleene Symposium*, North-Holland, Amsterdam, pp 223-265.
- [SC5] Scott D.S (1981), *Notes on the Formalization of Logic*, Parts I&II, III&IV.
- [SC6] Scott D.S (1981), Lectures on a Mathematical theory of computation; Technical Monograph PRG-19, Oxford University Computing Laboratory.
- [SC7] Scedrov A. (1985), Extending Gödel's modal interpretation to type theory and set theory; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers

B.V. North-Holland.

[SH1] Shapiro S. (1985), Introduction: Intensional Mathematics and Constructive Mathematics; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V. North-Holland.

[SH2] Shapiro S. (1985), Epistemic and Intuitionistic Arithmetic; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V., North-Holland.

[SH3] Shoenfield J. (1976), *Mathematical Logic*; Addison Wesley.

[SM1] Smith J. (1984), An interpretation of Martin-Löf's type theory in a type-free theory of propositions; *Journal of Symbolic Logic* 49, pp 730-753.

[SM2] Smith J. (1985), On a nonconstructive type theory and program derivation; technical report, Computer Science Department, University of Göteborg.

[SM3] Smorynski C. (1977), The incompleteness theorems; *Handbook of Mathematical Logic*, pp 821-865, Edited by Barwise, North-Holland.

[SM4] Smullyan R. (1985), Modality and Self-Reference; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V. North-Holland.

[SM5] Smullyan R. (1985). Some principles related to Löb's theorem; *Intensional Mathematics*, edited by S. Shapiro, Elsevier Science publishers B.V., North-Holland.

[ST1] Stenlund S. (1972), *Combinators, lambda-terms and proof-theory*; D.Reidel, Dordrecht, Holland.

[ST2] Stoy J. (1977), *Denotational semantics: The Scott-Strachey approach to programming language theory*; The MIT Press, Cambridge, Massachusetts.

[ST3] Stalnaker R.C (1984), *Inquiry*; The MIT Press, Cambridge, Massachusetts.

[ST4] Stalnaker R.C. and Thomason R.H. (1968), Abstraction in First-Order Modal Logic, *Theoria* 3, pp 203-207.

[SU1] Suppes P. and Hill S. (1964), *First course in Mathematical Logic*; Blaisdell.

[TE1] Tennent R.D (1976), The denotational semantics of programming languages; *Communications of the ACM* 19, pp 437-453.

[TE2] Tennant N. (1978), *Natural Logic*; Edinburgh University Press.

[TE3] Tennant N. (1982), Proof and Paradox; *Dialectica*.

[TH1] Thomason R.H (1970), *Symbolic Logic, An introduction*; Collier-Macmillan.

[TH2] Thomason R.H (1974), *Formal Philosophy*; Selected papers by Richmond Montague, Yale University.

[TH3] Thomason R.H (1976), Some extensions of Montague grammar; *Montague Grammar*, Edited by Barbara Partee, Academic Press.

- [TH4] Thomason R.H (1980), A model theory for propositional attitudes; *Linguistics and Philosophy* 4, pp 47-70.
- [TH5] Thomason R.H and Stalnaker R.C. (1973), A semantic theory of adverbs; *Linguistic Inquiry* 4, pp 195-220.
- [TH6] Thomason R.H (1968), Modal Logic and Metaphysics, *The logical way of doing things*, edited by K. Lambert, Yale University Press, New Haven.
- [TR1] Troelstra A.S (1969), *Principles of Intuitionism*; Lecture notes in Mathematics, Springer-Verlag, Berlin-Heidelberg, New York.
- [TU1] Turner R. (1982), Montague grammar and Scott-strachey semantics; In D.Flickinger, M.Macken and N.Wiegand (eds.), *Proceedings of the first West Coast Conference on Formal Linguistics*, Stanford University.
- [TU2] Turner R. (1983), Montague Semantics, Nominalization and Scott's Domains; *Linguistics & Philosophy* 6, pp 259-288.
- [TU3] Turner R. (1984), Formal semantics & type free theories; Proceedings of the 5th Amsterdam Symposium on Formal Semantics.
- [TU4] Turner R. (1984), *Logics for AI*; Ellis Horwood series in Artificial Intelligence.
- [TU5] Turner R. (1984), Three Theories of Nominalized Predicates; *Studia Logica XLIV*2, pp 165-186.
- [TU6] Turner R. (1984), Nominalization and Scott's Domains II; *Nôtre Dame Journal of Formal Logic* 26.
- [TU7] Turner R. (1984), The Semantics of Nominalized Predicates; privately circulated note.
- [TU8] Turner R. (1986), Towards a new foundation for semantic theory; Conference on property theory at Amherst,
- [TU9] Turner R. (1987), A theory of properties; *Journal of Symbolic Logic* 52.
- [TU10] Turner R. (1987), Intentional Semantics; privately circulated note.
- [TU11] Turner R. (1987), Procedural semantics; privately circulated note.
- [TU12] Turner R. (1987), Generalised Frege structures; Invited talk, American Philosophical Association.
- [TU13] Turner R. (1987), Semantics in the lambda calculus; privately circulated.
- [VE1] Veltman F. (1984), Data Semantics; MC Tract 136, *Formal methods in the study of language*, Edited by Groenendijk, Janssen and Stochof.
- [VE2] Veltman F. (1985), *Logics for conditionals*; PhD thesis, University of Amsterdam.
- [WA1] Wallace J. (1971), Convention T and Substitutional quantification; *Noûs* 5, pp

199-211.

[WA2] Wang H. (1950), A formal system of Logic; *Journal of Symbolic Logic* 15, pp 25-32.

[WA3] Wang H. (1955), On formalization; *Mind* 64, pp 226-238.

[WI1] Willard S. (1941), *General topology*; Addison-Wesley Series in Mathematics.

[WR1] Wright C. (1983), *Frege's conception of numbers as objects*; Scott Philosophical Monographs 2, Aberdeen University Press.

[YA1] Yablo S. (1985), Truth and Reflection; *Journal of Philosophical Logic* 14, pp 297-349.

[ZE1] Zeevat H. (1985), A treatment of belief sentences in discourse representation theory; privately circulated note.