REINHARD MUSKENS

A RELATIONAL FORMULATION OF THE THEORY OF TYPES*

INTRODUCTION

Relational Type Theory

In Montague semantics it is common procedure to specify a translation function taking the expressions of some fragment of natural language to logical expressions. If all is done well, the translated phrases and their translations show the same logical behaviour. Their truth conditions should match, for example, and the relation of logical consequence on the translations should mirror the relation of entailment that is imposed on the natural language fragment by natural logic.

The logic that is usually taken as the range of values of this translation function is Montague's IL (Intensional Logic), defined in Montague [1973] and extensively described in Gallin [1975]. Being an intensional extension of Church's [1940] beautiful formulation of the simple theory of types, it can be embedded in a two-sorted version, TY_2, of this theory, as was shown by Gallin.

Historically, Church's formulation of type theory was much influenced by his formulations of the lambda calculus, which is a theory of functions. The 1940 article defining the logic is mainly of a syntactical character, but in the first section a brief suggestion is made concerning the intended interpretation of the system. This interpretation is to be *functional*. While in earlier and less precise formulations of type theory (see Russell [1908], Carnap [1929]) classes and relations played an important and more independent rôle, these now seem to have to be equated with their characteristic functions. Multi-argument relations are identified in this way with multi-argument functions, which in their turn, following Schönfinkel [1924], are equated with functions in one argument whose values are functions again.

Now these moves seem innocent enough. Technically it is clearly equivalent to consider relations directly or to explain them recursively with the help of Schönfinkel's Trick. But, although equivalent, the identification is—I claim—not very felicitous. Relations are 'moved up' recursively in the set-theoretical hierarchy and this complication makes it extremely difficult to formulate the usual model-theoretical notions for the logic. In fact, in almost all cases where an interesting

From: *Linguistics and Philosophy* **12**: 325-346, 1989.

notion *is* defined, this is done by the use of a recursion that reverses the effect of Schönfinkel's Trick.[1]

This kind of problem motivated S. Orey to define his higher-order predicate calculus in 1959 (see also Gallin [1975], Van Benthem & Doets [1983]). Avoiding the Trick, he formulated type theory in such a way that model-theoretic concepts as, for example, *substructure* or *end extension* (of a general model) have simple and natural formulations. Types, in his system, are of a *relational* character, not of a functional one as they are in Church's; and the objects in his domains are either individuals or relations. Here are the relevant definitions:

DEFINITION 1. The set of *types* is to be the smallest set such that:
i. e and s are types,
ii. if $\alpha_1, \ldots, \alpha_n$ are types ($n \geq 0$), then $\langle\alpha_1 \ldots \alpha_n\rangle$ is a type.
We shall equate $\langle\rangle$ with $\emptyset$ or, equivalently, with 0. The types e and s we call *basic*, all other types *relational*.

DEFINITION 2. A *standard Orey frame* is a family of sets $\{D_\alpha \mid \alpha \text{ is a type}\}$ such that $D_e \neq \emptyset$, $D_s \neq \emptyset$ and $D_{\langle\alpha_1 \ldots \alpha_n\rangle} = P(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$.
(The cartesian product of the empty sequence of sets is to be equated with $\{\langle\rangle\}$. So $D_{\langle\rangle} = P(\{\emptyset\}) = \{0,1\}$, the set of *truth values*.)

Orey's relational *models* can now be defined in the usual way by adding an interpretation function to the frames just given.The use of these relational models instead of the standard functional ones is not only advantageous from a model-theoretic point of view, but has also much to recommend it from the perspective of applications of type theory in Montague semantics. I shall give four arguments in support of this.

I. Although the standard logic has, in a sense to be explained below, 'more' types than relational type theory has, these extra types are in fact seldom used. Almost all proposed translations of natural language expressions have (functional) types that correspond closely to the relational types defined above. In order to put this more accurately, I shall first give the familiar definition of Church's types and then define the subclass of them that is in fact—I claim—most popular.

DEFINITION 3. The set of *Church types* is to be the smallest set such that:
i. e, s and t are Church types,

ii. if α and β are Church types, then $(\alpha\beta)$ is a Church type.

DEFINITION 4. Define the function Σ (Σ is for Schönfinkel) taking types to Church types by the following double recursion:

I $\Sigma(e) = e$, $\Sigma(s) = s$

II i. $\Sigma(\langle\rangle) = t$

ii. $\Sigma(\langle\alpha_1 \ldots \alpha_n\rangle) = (\Sigma(\alpha_1)\Sigma(\langle\alpha_2 \ldots \alpha_n\rangle))$ if n≥1.

So, for example, $\Sigma(\langle e\rangle) = (et)$, $\Sigma(\langle\langle e\rangle\rangle) = ((et)t)$, $\Sigma(\langle ee\rangle) = (e(et))$ and $\Sigma(\langle\langle se\rangle\langle se\rangle\rangle) = ((s(et))((s(et))t))$. If α is the type of some relation then $\Sigma(\alpha)$ is the Church type of the unary function that codes this relation in functional type theory. Note that arguments of Σ tend to have less length than the corresponding values. Let's call any Church type that is a value of Σ *quasi-relational*. It is not difficult to characterize the quasi-relational Church types: A Church type is a value of Σ if and only if no occurrence of *e* or *s* immediately precedes a right bracket in it.

Ever since Bennett [1974] removed individual concepts from the standard formulation of Montague Grammar, the vast majority of types that have been proposed as denotation sets for linguistic categories have been quasi-relational. If the semantics of a natural language is described with the help of a functional type theory, then linguistic expressions tend to get semantic values having values of Σ as their types. This seems to be an important fact about semantics, but it is a fact that is not reflected in the overall organization of current Montague Grammar. It would be so reflected if we would trade the usual type theory for a relational one and assign relational types α to linguistic categories instead of the more complex quasi-relational types $\Sigma(\alpha)$[2].

II. The complexity of the objects that are used in functional semantics exceeds the complexity of their relational counterparts. In the functional theory, elements from Orey frames are coded as elements from Church frames:

DEFINITION 5. A *Church frame* is a family of sets $\{D_\alpha \mid \alpha \text{ is a Church type}\}$ such that $D_e \neq \emptyset$, $D_s \neq \emptyset$, $D_t = \{0,1\}$ and $D_{(\alpha\beta)}$ is the set of functions from D_α to D_β.

The Schönfinkel identification that codes multi-argument relations as unary functions may look simple if only relations of individuals are considered. In the higher-order case however, where relations can take relations as arguments, which in their turn can

again take relations as arguments, etc., the identification is somewhat less than transparent. Let's fully write out the definition of the encoding function:

DEFINITION 6. Let $\{D_\alpha \mid \alpha \text{ is a type}\}$ be a standard Orey frame and $\{D'_\alpha \mid \alpha$ is a Church type$\}$ the Church frame such that $D_e = D'_e$ and $D_s = D'_s$. For each type α, define a bijection S_α: $D_\alpha \rightarrow D'_{\Sigma(\alpha)}$ by the following double recursion:

I $S_e(d) = d$, if $d \in D_e$;
$S_s(d) = d$, if $d \in D_s$;

II i. $S_{\langle\rangle}(d) = d$, if $d \in D_{\langle\rangle}$;

ii. If n > 0, $\alpha = \langle\alpha_1 \ldots \alpha_n\rangle$ and $R \in D_\alpha$, then $S_\alpha(R)$ is the function F of type $(\Sigma(\alpha_1)\Sigma(\langle\alpha_2 \ldots \alpha_n\rangle))$ such that, for each $f \in D'_{\Sigma(\alpha_1)}$, $F(f) = S_{\langle\alpha_2 \ldots \alpha_n\rangle}(\{\langle d_2, \ldots, d_n\rangle \mid \langle S_{\alpha_1}{}^{-1}(f), d_2, \ldots, d_n\rangle \in R\})$.

It is routine to prove that this is well-defined. Define S to be $\bigcup_\alpha S_\alpha$.

Obviously, the function S tends to rather dramatically increase complexity. For example, an object of type ‹‹*se*›‹*se*›› (arguably the kind of object that can be taken to be the extension of a natural language determiner), which is a two-place relation taking relations between indices and entities in both its argument places, is coded as a function taking functions from indices to functions from entities to truth values to functions taking functions from indices to functions from entities to truth values to truth values.

Now, if there would be any need to do so, we could gladly accept these intricacies, since the functions S_α are *isomorphisms*: for all relations R (of any type) $\langle d_1, \ldots, d_n\rangle \in R$ iff $S(R)(S(d_1)) \ldots (S(d_n)) = 1$, as can easily be verified. But I think that this doubly recursive encoding is just a needless complication. If we want Montague Grammar to look a little less like a Rube Goldberg machine (the comparison is taken from Barwise & Cooper [1981]), we may as well skip it.

III. In view of the fact that natural language **and**, **or** and **not** can be used with expressions of almost all linguistic categories, type domains should have a Boolean structure. This has been argued for by a variety of authors, beginning with Von Stechow [1974] (see also Keenan & Faltz [1978]). Obviously, Orey's relational models have a Boolean structure on all their (non-basic) domains, since these are power sets. So we can give a very simple rule for the interpretation of natural language conjunction, disjunction and negation: they are to be treated as $\cap$, $\cup$ and $-$ (complementation within a typed domain) respectively. Entailment between expressions of the same category is to be treated as inclusion.

This does not differ much, of course, from the usual treatment of both entailment and the expressions just mentioned. The point is rather that the relevant Boolean operations are not as easily available in a functional type theory as they are here. They can only be obtained by Gazdar's [1980] pointwise recursive definitions. Let's have a look at one of them. Before we can give it, we must characterize a certain subclass of the Church types, the so-called *conjoinable* ones:

DEFINITION 7.
i. t is conjoinable;
ii. If β is conjoinable, then $(\alpha\beta)$ is conjoinable.

Note that, while not all conjoinable Church types are quasi-relational, there is a close kinship between the two classes of types: A Church type is quasi-relational if and only if all its subtypes are either basic or conjoinable.

Having defined the conjoinable types we can define generalized conjunction in functional type theory thus:

DEFINITION 8.
i. $a \;\; b := \mathrm{a} \cap \mathrm{b}$, if $a,b \in \{0, 1\}$;
ii. If F_1 and F_2 are functions of some conjoinable type $(\alpha\beta)$, then the function $F_1 \;\; F_2$ is defined by $(F_1 \;\; F_2)(z) = F_1(z) \;\; F_2(z)$, for all z of Church type α.

Similar definitions can be given for generalized disjunction, complementation and inclusion (see Groenendijk & Stokhof [1984] for the last operation).

These definitions are an artefact of the functional formulation of type theory. They enable us to treat generalized co-ordination by reversing the effect of Schönfinkels Trick: It is not difficult to prove that, for any R_1, R_2 of relational type, $S(R_1 \cap R_2) = S(R_1) \;\; S(R_2)$. But as soon as we get rid of the Trick, the need for its reversals, these pointwise definitions, vanishes too. So let's omit them and, since having no definitions is simpler than having some, get a less complicated theory.

IV. Simplification should lead to generalization; that is the reason why we strive for it. Thus far, my arguments for adopting a relational version of type theory were mainly concerned with the simplification and—I hope—esthetic improvement of existing semantic theories. But my fourth and last reason for going relational is that it allows a generalization of Montague Semantics which I think is urgently needed

and which seems hard to get in the standard approach: Doing things the relational way makes it possible to *partialize* the existing theory.

The view that partial structures (structures that may leave the truth values of some sentences undefined) are crucial for an adequate description of many semantic facts is now widely accepted. Attractive analyses of various linguistic phenomena have been carried out within frameworks that stress the partial, incomplete, character of semantic objects. Examples are the treatments of propositional attitudes and neutral perception verbs in Barwise & Perry's Situation Semantics, forcing accounts of conditionals and modals (see Kratzer [1977], Veltman [1985] and Landman [1986]), and the treatment of anaphora in Kamps' Discourse Representation Theory (Kamp [1981]). The reader will have no difficulty in supplying more examples of appealing semantical theories in which non-complete objects play an essential rôle. In the last ten years or so there has been a widespread tendency towards going partial.

But while Montague Semantics aspires to be a very general vehicle for the description of linguistic meaning it doesn't seem possible to carry out similar analyses within this framework; it simply lacks the partiality that is needed. The existing type hierarchies seem to be inherently total in character and thus far the logic has resisted all attempts at generalization in the desired direction.

One problem one has to deal with when trying to partialize standard type theory is that the one-to-one correspondence between multi-argument functions and unary functions of certain type breaks down: If, for example, D is some domain then the partial functions from $D \times D$ into D cannot in general be isomorphic to the partial functions from D into the partial functions from D into D. If D has two elements then the first of these sets has $3^{2\times 2} = 81$ elements, while the cardinality of the second one is $(3^2+1)^2 = 100$. So the Schönfinkel identification is no longer possible.

No such problems arise in the relational theory. Let's define a *partial relation* R on domains $D_1, \ldots, D_n$ as a tuple $\langle R^+, R^-\rangle$ of ordinary relations on these domains. The relation R^+ is called R's *denotation*, its companion R^- we call R's *antidenotation*. The n-tuples that are neither in R^+ nor in R^-, the set $D_1 \times \cdots \times D_n \setminus R^+ \cup R^-$, form R's *gap*; those that are in both form its *glut*.[3]

We may now consider hierarchies of partial relations:

Definition 9. A *standard partial frame* is a family of sets $\{D_\alpha \mid \alpha \text{ is a type}\}$ such that $D_e \neq \emptyset$, $D_s \neq \emptyset$ and $D_{\langle \alpha_1 \ldots \alpha_n \rangle} = P(D_{\alpha_1} \times \cdots \times D_{\alpha_n}) \times P(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$.

The above definition is of course a straightforward generalization of Definition 2.

I shall not describe the logic of these partial frames here. I have done so elsewhere (see Muskens [forthcoming]), where it turns out—as could be expected—that this logic can be obtained by simply 'doubling' some of the crucial concepts that are discussed in this paper. It also turns out that the partial logic enables us to give Montague analyses of propositional attitude and naked perception verbs very much along the lines of Barwise & Perry [1983] and Barwise [1981].

But for the moment let's return to the total relational frames. It is from these, conceptually, that the partial ones are derived and they should be studied in their own right.

Although there are, as I have just argued, good reasons to prefer Orey's relational *models* over Church's functional ones, there are equally good reasons to prefer Church's *syntax* over Orey's when it comes to choosing a logic for our purposes. In fact the latter logic, as it was defined in Orey [1959], lacks the operations of *application* and *abstraction*, which are absolutely crucial for the Montague semanticist. So at this point it may seem that we can either have an applicable logic with a complex model theory or a logic with a simpler model theory which is inapplicable.

But the dilemma is only apparent. We can have our cake and eat it by taking the syntax of standard type theory and evaluating it on relational models. Let's consider application. Suppose that A and B are terms of types $\langle\beta\alpha_1 \ldots \alpha_n\rangle$ and β respectively. Then the value of the term AB (of type $\langle\alpha_1 \ldots \alpha_n\rangle$) in some model M (under an assignment a) is given by the following rule:

$$(1) \qquad \|AB\|^{M,a} = \{\langle d_1, \ldots, d_n\rangle \mid \langle \|B\|^{M,a}, d_1, \ldots, d_n\rangle \in \|A\|^{M,a}\}$$

Let, for example, the domain of M be some set of people and let *love* be a constant of type $\langle e\, e\rangle$ that is to be interpreted as the love relation among them ($\{\langle d_1,d_2\rangle \mid d_2$ loves $d_1\}$). Let j and m be constants of type e, interpreted on M by John and Mary respectively. Then $\|love\ j\|^M$ will be equal to $\{d_2 \mid d_2$ loves John$\}$, the set of persons loving John, while $\|love\ j\ m\|^M$ equals $\{\langle\rangle \mid$ Mary loves John$\}$, which is equal to the value 1 just in case Mary indeed loves John.

Suppose now that A is a term of some type $\langle\alpha_1 \ldots \alpha_n\rangle$ and that x is a variable of type β. Then we can define the value of the term λxA (of type $\langle\beta\alpha_1 \ldots \alpha_n\rangle$) in M under a as follows:

(2) $\|\lambda xA\|^{M,a} = \{\langle d,d_1,...,d_n\rangle \mid d \in D_\beta \text{ and } \langle d_1,...,d_n\rangle \in \|A\|^{M,a[d/x]}\}$

For example the term $\lambda x_e \lambda y_e \exists z_e(x \prec z \wedge y \prec z)$ will receive the relation 'having a common successor' on the D_e domain as its interpretation in any model, as the reader can easily verify.

Definite and indefinite description operators

In the preceding pages I have sketched how type theory can be interpreted in a relational way. It would be easy now to fill in the details of this sketch and obtain a completely defined relational semantics. The crucial clauses in the Tarski truth definition would be (1) and (2) of course, and the resulting system would look a lot like Gallin's TY_2, although its model theory would be much simpler.

Note, however, the following little asymmetry: while in the standard type theory it is possible to obtain terms of a basic (*e* or *s*) type by application, this is not so in relational type theory; the results of clauses (1) and (2) are always relations. In the functional theory the result of applying a (say) (*ee*)-type function to an *e*-type argument gives a value of type *e*, but in the relational formulation the same function, seen as an ‹*ee*›-type relation now, applied to the same argument, gives an ‹*e*›-type singleton as a result. To get the original value we need a *description operator.*

Since a description operator is generally useful, we may add it to the logic[4] and define $\iota x_\alpha(\varphi)$ to be a term of type α if φ is a formula (a term of type ‹›) and x_α is a variable of type α and demand that at least:

(3a) $\|\iota x_\alpha(\varphi)\|^{M,a}$ = the unique $d \in D_\alpha$ such that $\|\varphi\|^{M,a[d/x]} = 1$, if there is such an object $d \in D_\alpha$

What to do if there is no such unique *d*? This is a classical problem of course and it has been discussed extensively in the literature from Frege onwards (see Scott [1967], Renardel [1984] for short expositions of the main points of view). If α is a relational type, a type of the form $\langle\alpha_1...\alpha_n\rangle$ that is, then there is an obvious candidate for the value of $\|\iota x_\alpha(\varphi)\|^{M,a}$ in case there is no unique *d* such that *d* satisfies φ: we can let it be the empty set[5]. If, on the other hand, α is basic, that is if $\alpha = e$ or $\alpha = s$, we must proceed in some other way.

To this end we shall follow Scott [1967] in distinguishing between the proper objects of some basic type and an *improper* one, designed to be the 'non-referent' of non-referring expressions. The proper objects are just the elements of D_e or D_s. To

these we now add an improper one. Since we can—up to isomorphism—take any set to play the part of this object, we might as well choose $\varnothing$ again for uniformity's sake. We shall stipulate that $\varnothing \notin D_e$ and $\varnothing \notin D_s$ (since we want to restrict quantification and abstraction to the proper objects) and demand that:

(3b) $||\iota x_\alpha(\varphi)||^{M,a} = \varnothing$, if there is no unique $d \in D_\alpha$ such that $||\varphi||^{M,a[d/x]} = 1$

Scott's treatment of the iota-operator makes it possible to give *e*-type translations to natural language descriptions[6] and have them behave in a Russellian way. Consider the famous sentence:

(4) **The present king of France is bald**,

which may be formalized by:

(5) *bald* ιx(*king x*) (where both *bald* and *king* are type ‹*e*› constants).

In a model M where there is no unique king of France, such that $||king||^M$ is not a singleton, the interpretation of **the present king of France**, $||\iota x(king\ x)||^M$, will be equal to $\varnothing$. Since $||bald||^M \subseteq D_e$, but $\varnothing \notin D_e$, rule (1) will ensure that $||bald\ \iota x(king\ x)||^M = 0$, so the sentence is false in M. Of course this implies that its direct negation

(6) **It is not the case that the present king of France is bald**

is true in M. On the other hand the sentence

(7) **The present king of France is not bald**,

containing a verb phrase negation, will come out false in M. Again, since the interpretation of the verb phrase **is not bald** is a subset of D_e (the complement of $||bald||^M$ in D_e) and $||\iota x(king\ x)||^M = \varnothing \notin D_e$, rule (1) ensures (7)'s falsity in M.

Now that we have seen that abstraction and the definite description operator can be given a precise semantics on our models, we turn our attention to the *in*definite description operator. In this paper, we take an indefinite description operator as a primitive logical symbol, defining all other variable binding operators from it.

To have indefinite descriptions in the theory of types is no innovation. In Church [1940] the author takes 'selection operators' (somewhat misleadingly denoted by iotas) as some of his logical primitives. These operators are constants of types $((\alpha t)\alpha)$ (in our notation), so intuitively they take sets of objects of type α to objects of type α. Church then proposes two alternative axiom schemes that should govern the behaviour of these iotas. The first gives a set of *axioms of descriptions*: the iotas should assign to singletons their unique elements. This is of course still in line with the usual interpretation of the iota symbol. But the latter remark doesn't hold true for the second, stronger, axiom scheme that Church proposes. This scheme gives *axioms of choice*: the iotas should pick out some element from *every* non-empty set, which makes the symbol into an indefinite description operator. Henkin, in his famous article in which the generalized completeness of Church's system is proved (Henkin [1950]), gives a (very sketchy) semantics for the selection operator that seems to be close in spirit to the semantics that we shall give to our indefinite description operator in section 2 below.

Probably the first treatment of an indefinite description operator was given by Hilbert & Bernays in their classical *Grundlagen der Mathematik* (Hilbert & Bernays [1939]), to which Church acknowledges a debt. It often happens in mathematical texts that when a statement of the form

(10) There are x such that φ . . .

is derived, the author continues with a statement like

(11) Now let a be an arbitrary x such that φ . . .

It is easy to reason away such talk about arbitrary objects by translating the whole mathematical argument in question into standard predicate logic. But Hilbert & Bernays do not take such a course. Instead, they take the arbitrary φ seriously, give it a name, $\varepsilon x(\varphi)$, treat this as a term, and give axioms ruling its proof theory (first-order equivalent to Church's axiom of choice for $\alpha = e$). The ordinary quantifiers can then be defined using ε-terms and ordinary quantification theory can be derived from their ε-calculus.

What is the appropriate semantics for Hilbert's ε-symbol? Hilbert & Bernays themselves give none, since they are only interested in proof-theoretical investigations, but a semantics is given in Asser [1957] (see also Leisenring [1969]). Asser uses *choice functions*, choosing an element from every non-empty subset of the

domain. The value of the term $\varepsilon x(\varphi)$ in some model M is then a choice from the set of objects that satisfy φ in M. This seems a good way to interpret the indefinite description operator.

Again the classical problem arises: what if the set of φ's is empty? Asser considers two possibilities to solve this problem. Either one can let choice functions assign some arbitrary element of the domain to the empty set or one can leave them undefined for that set. As Leisenring correctly remarks, the first option gives a nice semantics for Hilbert's ε-symbol, but the second one is better suited to the interpretation of the η-symbol, another indefinite description operator that Hilbert & Bernays consider briefly, the one that is discussed in Reichenbach [1947][7].

In this paper we shall interpret the η-symbol in a manner that resembles Asser's second way. Thus, the value of a term $\eta x(\varphi)$ in a model M will be an arbitrary x such that φ (given by the choice function on M) if there are φ's in M and it will be $\varnothing$ if there are none. The usual variable-binding operators (to wit the lambda-operator, the iota-operator and the quantifiers) as well as the epsilon-operator can then be defined using η and the propositional connectives.

THE SYSTEM $TT^{\eta,2}$

In this section I shall present a formal development of the logical system $TT^{\eta,2}$, a two-sorted relational type theory with an indefinite description operator.

Syntax and semantics

Symbols come in four kinds. First, for each type α, we shall assume the existence of a set of *constants* of type α. There are two special constants, denoted by $\bot$ and $\rightarrow$, of types $\langle\rangle$ and $\langle\langle\rangle\langle\rangle\rangle$ respectively, called *logical* constants. They will get a fixed interpretation. All other constants are called *non-logical*. Second, for each type α, there is a denumerably infinite set of *free variables* of type α and, third, there is a countable infinity of *bound variables* of type α.[8] I shall sometimes, but not always, indicate the type of a constant or a free or bound variable by a subscript. Fourth, there are four improper symbols, denoted by), (, η and =. It is clear that the four sets of symbols should be disjoint. If σ and σ' are strings of symbols and s is a symbol, then $[\sigma'/s]\sigma$ denotes the string of symbols obtained from σ by replacing every occurrence of s in σ by σ'.[9] Sometimes, if no confusion is likely to result, I shall streamline notation a bit by writing $\sigma(\sigma')$ for $[\sigma'/s]\sigma$.

DEFINITION 10. We define, for each α, the set of *terms* of that type by the following inductive definition:

i. Every constant or free variable of some type α is a term of that type.

ii. If A is a term of type $\langle\beta\alpha_1 \ldots \alpha_n\rangle$ and B is a term of type β, then (AB) is a term of type $\langle\alpha_1 \ldots \alpha_n\rangle$.

iii. If A and B are terms of the same type, then $A = B$ is a term of type $\langle\rangle$ (a *formula*).

iv. If φ is a formula, u_α a free variable of type α and x_α a bound variable of that type and u does not occur in any substring of φ of the form $\eta x(\sigma)$, where an equal number of left and right brackets occur in σ, then $\eta x([x/u]\varphi)$ is a term of type α.

A term A of type α may be denoted by A_α. I shall suppress parentheses wherever this does not lead to confusion (under the understanding that association is to the left). Terms of the form $\eta x([x/u]\varphi)$ will be called η-*terms*. Using the convention given above, we shall often write $\eta x(\varphi(x))$, or even $\eta x\varphi(x)$, for $\eta x([x/u]\varphi)$. A term is *closed* if it contains no free variables. A closed formula is called a *sentence*; a set of sentences is a *theory*.

Now, let us turn to the semantics of the logic. We shall give a standard interpretation as well as a generalized one (see Henkin [1950]).

DEFINITION 11. A *frame* is a family of sets $\{D_\alpha | \ \alpha \text{ is a type}\}$ such that $D_\alpha \subseteq P(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$ for each type $\alpha = \langle\alpha_1 \ldots \alpha_n\rangle$, $\emptyset \in D_\alpha$ for each relational α, but $\emptyset \notin D_\alpha$ if α is basic. A frame is *standard* if $D_\alpha = P(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$ for each $\alpha = \langle\alpha_1 \ldots \alpha_n\rangle$.

Note that basic domains may be empty. Our logic makes no existence assumptions and consequently the sentences $\exists x_e\, x = x$ and $\exists x_s\, x = x$ will not be logically valid.

DEFINITION 12. Let $F = \{D_\alpha\}_\alpha$ be a frame. An *interpretation* for F is a function I having the set of constants as its domain, such that $I(c) \in D_\alpha \cup \{\emptyset\}$ for each non-logical constant c of type α, and such that $I(\bot) = 0$ and $I(\rightarrow) = \{\langle 0,0\rangle, \langle 1,1\rangle, \langle 0,1\rangle\}$. (Note that $I(\rightarrow)$ is not necessarily an element of $D_{\langle\langle\rangle\langle\rangle\rangle}$, but see below.) An *assignment* for F is a function a, taking free variables as its arguments, such that $a(u) \in D_\alpha \cup \{\emptyset\}$ if u is a free variable of type α. If a is an assignment, then $a[d/u]$ is to be the assignment a' such that $a'(v) = a(v)$ if $v \neq u$ and $a'(u) = d$.

In order to be able to interpret η-terms we need choice functions:

DEFINITION 13. A *choice function* for a set D is a function G: $P(D) \to D \cup \{\emptyset\}$ such that:
i. $G(X) \in X$, if $X \subseteq D$ and $X \neq \emptyset$,
ii. $G(\emptyset) = \emptyset$.
Let $F = \{D_\alpha\}_\alpha$ be a frame. A *set of choice functions* for F is a set $\{H_\alpha\}_\alpha$ such that each H_α is a choice function for D_α.

DEFINITION 14. A *weak general model* is a triple $\langle F,I,H\rangle$ such that F is a frame, I is an interpretation for F, and H is a set of choice functions for F. A weak general model is a (*standard*) *model* if its frame is standard.

A note on notation: I shall follow the convention that a weak general model M, its frame F, its interpretation I, its set of choice functions H, and all the elements of both F and H will be denoted by metalinguistic variables that carry the same superscripts.

Even though the domains of weak general models may be very sparsely inhabited (note for example that all relational domains may be equal to $\{\emptyset\}$), we are able to give a Tarski truth definition (or, more adequately expressed, a Tarski value definition) at this point:

DEFINITION 15. The *value* $\|A\|^{M,a}$ of a term A on a weak general model M under an assignment a is defined by induction on the complexity of terms in the following way:
i. $\|c\|^{M,a} = I(c)$ if c is a constant
$\|u\|^{M,a} = a(u)$ if u is a free variable
ii. $\|AB\|^{M,a} = \{\langle d_1, \ldots, d_n\rangle \mid \langle \|B\|^{M,a}, d_1, \ldots, d_n\rangle \in \|A\|^{M,a}\}$
iii. $\|A = B\|^{M,a} = 1$ iff $\|A\|^{M,a} = \|B\|^{M,a}$
iv. $\|\eta x_\alpha([x/u]\varphi)\|^{M,a} = H_\alpha(\{d \in D_\alpha \mid \|\varphi\|^{M,a[d/u]} = 1\})$

It would have been misleading to speak of 'the value of a term *in* a weak general model' since, in general, there is of course no guarantee that the value of a term A_α on M will be an element of D_α or even of $D_\alpha \cup \{\emptyset\}$. This does not effect the correctness of the definition, however. In standard models, as well as in the general models to be defined below, the value of a term A_α will be in D_α if α is relational and it will be an element of $D_\alpha \cup \{\emptyset\}$ if α is basic.

We say that a formula φ is *true* on a weak general model M under an assignment a, or, alternatively, that M *satisfies* φ under a, or, to use still another phrase, that M is a weak general model *of* φ under a, if $\|\varphi\|^{M,a} = 1$. As is usual, $\|A\|^{M,a}$ depends only on the values that a assigns to the free variables actually occurring in A. So if A is a closed term, we may write $\|A\|^{M}$ instead of $\|A\|^{M,a}$.

Unsurprisingly, the ordinary kind of substitution theorem holds for this logic.

PROPOSITION 1. (Substitution Theorem). Let M be a weak general model, a an assignment for M, A a term and B a term of the same type as the free variable u, then: $\|[B/u]A\|^{M,a} = \|A\|^{M,a[d/u]}$, where $d = \|B\|^{M,a}$.
PROOF. This is proved by an induction on the complexity of the term A.

The usual logical operators may be obtained by means of definition now. The following definition needs no comment:

DEFINITION 16. Let φ and ψ be formulae.

$\neg\ \varphi$	abbreviates	$\varphi \rightarrow \bot$
$\varphi \vee \psi$	abbreviates	$\neg\ \varphi \rightarrow \psi$
$\varphi \wedge \psi$	abbreviates	$\neg\ (\varphi \rightarrow \neg\ \psi)$
$\varphi \leftrightarrow \psi$	abbreviates	$(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$

We can define the quantifiers with the help of the η-operator. Quantification over relational domains is defined in essentially the same way as Hilbert & Bernays defined quantification in their ε-calculus with the help of the ε-symbol. Quantification over basic domains is defined in a different way since we do not wish to quantify over the improper object:

DEFINITION 17. Let φ be a formula

$\exists x_\alpha([x/u]\varphi)$	abbreviates	$[\eta x_\alpha([x/u]\varphi)/u]\varphi$,	if α is relational.
$\exists x_\alpha([x/u]\varphi)$	abbreviates	$\neg\ (\eta x_\alpha([x/u]\varphi) = \eta x(\bot))$,	if α is basic.
$\forall x_\alpha([x/u]\varphi)$	abbreviates	$\neg\ \exists x_\alpha([x/u]\neg\ \varphi)$	

At first sight these stipulations may perhaps look somewhat unintuitive, but the quantifiers get the interpretations we want them to have:

PROPOSITION 2. For any weak general model M:
$\|\exists x_\alpha([x/u]\varphi)\|^{M,a} = 1$ iff there is a $d \in D_\alpha$ such that $\|\varphi\|^{M,a[d/u]} = 1$

$\|\forall x_\alpha([x/u]\varphi)\|^{M,a} = 1$ iff for all $d \in D_\alpha$ $\|\varphi\|^{M,a[d/u]} = 1$

PROOF. We prove the first statement, from which the second one readily follows. Let $d' = \|\eta x_\alpha([x/u]\varphi)\|^{M,a} = H_\alpha(\{d \in D_\alpha \mid \|\varphi\|^{M,a[d/u]} = 1\})$. If α is relational then $d' \in D_\alpha$ and hence by the definition of choice functions: $\|\varphi\|^{M,a[d/u]} = 1$ for any $d \in D_\alpha$ iff $\|\varphi\|^{M,a[d'/u]} = 1$. By the Substitution Theorem above this last statement is equivalent to $\|[\eta x_\alpha([x/u]\varphi)/u]\varphi\|^{M,a} = 1$ and hence to $\|\exists x_\alpha \varphi(x)\|^{M,a} = 1$. If α is basic, we see that if $d' \in D_\alpha$ then $d' \neq \emptyset$ since $\emptyset \notin D_\alpha$. From this and the definition of choice functions it follows that $\|\varphi\|^{M,a[d/u]} = 1$ for some $d \in D_\alpha$ iff $d' \neq \emptyset$ iff $\|\neg(\eta x_\alpha \varphi(x) = \eta x(\bot))\|^{M,a} = 1$ iff $\|\exists x_\alpha \varphi(x)\|^{M,a} = 1$.

Turning to the definite description operator now, we define:

DEFINITION 18. $\iota x_\alpha([x/u]\varphi)$ abbreviates $\eta x_\alpha \forall y_\alpha(x = y \leftrightarrow [y/u]\varphi)$, where y is the first bound variable of type α different from x in some fixed ordering.

Again, it is easy to see that the abbreviation thus defined has its intended interpretation:

PROPOSITION 3. For any weak general model M:

$\|\iota x_\alpha([x/u]\varphi)\|^{M,a} = d$, if d is the unique element of D_α such that $\|\varphi\|^{M,a[d/u]} = 1$,

$= \emptyset$, if there is no such unique object.

Abstraction is defined in the following manner:

DEFINITION 19. Let A be a term of type $\langle\alpha_1 \ldots \alpha_n\rangle$, x a bound variable of type β and u a free variable of that type, then $\lambda x([x/u]A)$ abbreviates $\eta R \forall x(Rx = [x/u]A)$, where R is the first variable of type $\langle\beta\alpha_1 \ldots \alpha_n\rangle$ that does not occur in A.

This time there is no guarantee that an expression $\lambda x A(x)$ will get its intended interpretation on a weak general model. The reason for this is that the required relation may simply not be present in the relevant domain (in that case $\lambda x A(x)$ will get the value $\emptyset$). We may want to restrict our attention to those weak general models wherein all lambda-terms *do* get their intended interpretations:

DEFINITION 20. Any sentence of the form $\forall y_1 \ldots y_n \exists R \forall x(Rx = [x/u]A)$, where A, x, u and R are as in definition 19, is called a *comprehension axiom*. A *general model* is a weak general model that satisfies all comprehension axioms.

All standard models are of course general models.

Note that by definition any general model satisfies all sentences of the form $\forall y_1 \ldots y_n \forall z(\lambda x([x/u]A)z = [z/u]A)$. So, in a sense, *lambda-conversion* is just another form of the comprehension axioms under our abbreviatory definitions. But note also that on basic domains abstraction, like quantification, is restricted to the proper objects: the more usual (and useable) form of lambda-conversion, $\lambda x_\alpha([x/u]A)B_\alpha = [B/u]A$ for all B, holds only for relational α; for basic α we have the restricted schema $\neg(B_\alpha = \eta x_\alpha \bot) \rightarrow \lambda x([x/u]A)B = [B/u]A$. While, for example, $\eta x_e \bot = \eta x_e \bot$ is true on all models, $\lambda x_e(x = x)\eta x_e \bot$ is true on none.

It is not difficult to see that general models conform to requirement (2) of the introduction:

PROPOSITION 4. Let M be a general model, a an assignment to M, A a term of type $\langle \alpha_1 \ldots \alpha_n \rangle$ and x a variable of type β, then:
$\|\lambda x([x/u]A)\|^{M,a} = \{\langle d, d_1, \ldots, d_n \rangle \mid d \in D_\beta \text{ and } \langle d_1, \ldots, d_n \rangle \in \|A\|^{M,a[d/u]}\}$.

The last operator that we consider is Hilbert's ε-symbol. The following definition is suggested in Hilbert & Bernays [1939]:

DEFINITION 21. $\varepsilon x_\alpha([x/u]\varphi)$ abbreviates $\eta x_\alpha(\exists y_\alpha[y/u]\varphi \rightarrow [x/u]\varphi)$, where y is the first bound variable of type α different from x in some fixed ordering.

This gives a semantics for the ε-symbol that is closely analogous to the one given in Hermes [1965]:

PROPOSITION 5. For any weak general model M:

$$\|\varepsilon x_\alpha([x/u]\varphi)\|^{M,a} = \begin{cases} H_\alpha(\{d \in D_\alpha \mid \|\varphi\|^{M,a[d/u]} = 1\}), & \text{if there is a } d \in D_\alpha \\ & \text{such that } \|\varphi\|^{M,a[d/u]} = 1. \\ H_\alpha(D_\alpha), & \text{otherwise.} \end{cases}$$

Relational type theory enables us to generalize the notion of entailment somewhat. Not only formulae can entail another formula, but any set of terms of a relational type can entail some term of that type:

DEFINITION 22. Let $\Gamma \cup \{A\}$ be a set of terms of some type $\alpha = \langle \alpha_1 ... \alpha_n \rangle$. Γ *entails* A (Γ *g-entails* A, Γ *wg-entails* A), $\Gamma \models A$ ($\Gamma \models_g A$, $\Gamma \models_{wg} A$), if $\bigcap_{B \in \Gamma} \|B\|^{M,a} \subseteq \|A\|^{M,a}$ for all models (general models, weak general models) M and assignments a to M.

In natural language too, expressions of many categories may entail one another (see Keenan & Faltz [1978], Groenendijk & Stokhof [1984]). It is obvious that definition 22 is indeed a generalization of the usual notion of entailment:

PROPOSITION 6. Let $\Gamma \cup \{\varphi\}$ be a set of formulae. $\Gamma \models \varphi$ ($\Gamma \models_g \varphi$, $\Gamma \models_{wg} \varphi$) iff for each model (general model, weak general model) M and assignment a to M it holds that if M satisfies all $\psi \in \Gamma$ under a, then M satisfies φ under a.

DEFINITION 23. We say that two terms A and B are (g- , wg-)*equivalent* if both A (g-, wg-) entails B and B (g-, wg-) entails A.

Proof theory and completeness

I shall finish by giving a standard Henkin proof to the effect that the notions $\models_g$ and $\models_{wg}$, defined in the preceding section, are recursively axiomatizable. Of course, by Gödel's Theorem and the fact that the natural number system is categorically definable in $TT^{\eta,2}$ with the standard semantics, $\models$ cannot be axiomatized.

DEFINITION 24. All formulae of one of the following forms are *axioms*:
Propositional axioms:
AS1 $\varphi \rightarrow (\psi \rightarrow \varphi)$
AS2 $(\varphi \rightarrow (\psi \rightarrow \chi)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \chi))$
AS3 $((\varphi \rightarrow \bot) \rightarrow \bot) \rightarrow \varphi$
Eta and quantification axioms:
AS4 $\neg A_\alpha B_1 ... B_n$, where $\alpha = \langle \alpha_1 ... \alpha_n \rangle$, each B_i is of type α_i and either A is $\eta x_\alpha(\bot)$ or for some i, α_i is basic and B_i is $\eta x_{\alpha_i}(\bot)$.
AS5^a $\neg(A_\alpha = \eta x_\alpha(\bot)) \rightarrow (\varphi(A_\alpha) \rightarrow \exists x \varphi(x))$, if α is basic.
AS5^b $\varphi(A_\alpha) \rightarrow \exists x \varphi(x)$, if α is relational.
AS6 $\exists x \varphi(x) \rightarrow \varphi(\eta x_\alpha \varphi(x))$,
AS7 $\forall x_\alpha(\varphi(x) \leftrightarrow \psi(x)) \rightarrow \eta y_\alpha \varphi(y) = \eta z_\alpha \psi(z)$
Extensionality:

AS8 $\forall R_\alpha \forall R'_\alpha(\forall x_{\alpha_1}\ldots x_{\alpha_n}(Rx_{\alpha_1}\ldots x_{\alpha_n} \leftrightarrow R'x_{\alpha_1}\ldots x_{\alpha_n}) \rightarrow R = R')$, where $\alpha = \langle\alpha_1\ldots\alpha_n\rangle$

Identity axioms:

AS9 $A = A$

AS10 $A = B \rightarrow (\varphi(A) \rightarrow \varphi(B))$ (Leibniz's Law).

DEFINITION 25. A *proof* for a formula φ is a sequence $\varphi_0, \ldots, \varphi_n$ of formulae such that $\varphi = \varphi_n$ and each φ_k ($k \leq n$) is either an axiom or follows from two formulae earlier in the sequence by the rule of *modus ponens* ($\varphi, \varphi \rightarrow \psi / \psi$). A formula φ is *provable*, $\vdash \varphi$, if there is a proof for it. The formula φ is said to be *derivable* from a set of formulae Γ, $\Gamma \vdash \varphi$, if there are $\varphi_0, \ldots, \varphi_n \in \Gamma$ such that $\vdash (\varphi_0 \wedge \ldots \wedge \varphi_n) \rightarrow \varphi$. A set of formulae is *consistent* if $\bot$ is not derivable from it.

Since AS1 – AS3 are axiom schemes and modus ponens is a rule we immediately see that all substitution-instances of tautologies in propositional logic are provable.

PROPOSITION 7. If $\vdash \varphi \rightarrow \psi(u)$ and the free variable u does not occur in φ, then $\vdash \varphi \rightarrow \forall x\psi(x)$.

PROOF. Suppose that $\chi_0, \ldots, \chi_n$ is a proof for $\varphi \rightarrow \psi(u)$; it is not difficult to verify that $[\eta x\neg\psi(x)/u]\chi_0, \ldots, [\eta x\neg\psi(x)/u]\chi_n$ is a proof for $\varphi \rightarrow \psi(\eta x\neg\psi(x))$. To obtain a proof for $\varphi \rightarrow \forall x\psi(x)$, use propositional logic and the dual of AS6, $\psi(\eta x\neg\psi(x)) \rightarrow \forall x\psi(x)$.

From the above proposition it follows that we can reason classically with the quantifiers $\forall x_\alpha$ and $\exists x_\alpha$, provided α is relational. If α is basic, then $\forall x_\alpha$ and $\exists x_\alpha$ behave like restricted quantifiers in standard logic.

THEOREM 1 (Soundness Theorem). Let T be a theory and φ a formula, then:

$$T \vdash \varphi \;\Rightarrow\; T \models_{wg} \varphi.$$

PROOF. By a straightforward induction on the length of proofs.

THEOREM 2. Let T be a theory and φ a formula, then:

$$T \models_{wg} \varphi \;\Rightarrow\; T \vdash \varphi.$$

PROOF. This is proved in the usual way, with the help of the Consistency Theorem below.

COROLLARY (Generalized Completeness Theorem). Let T be a theory and φ a formula, then $T \models_g \varphi \Leftrightarrow T \cup \text{COMP} \vdash \varphi$, where COMP is the set of all comprehension axioms.

THEOREM 3 (Consistency Theorem). If a theory is consistent, then it has a weak general model.

PROOF. Let T be a consistent theory. We construct a maximal consistent set of sentences $\Gamma \supseteq T$ having the so-called Henkin property. To this end let $A_0, \ldots, A_n, \ldots$ be some enumeration of all terms (of all types). For each $n \in \omega$ define a set of formulae Γ_n by: $\Gamma_0 = T$ and $\Gamma_{n+1} = \Gamma_n \cup \{A_n = u\}$, where u is the first free variable in our enumeration which has the same type as A_n has and which does neither occur in A_n nor in any of the formulae in Γ_n. Clearly, all Γ_n are consistent and hence $\bigcup_n \Gamma_n$ is consistent.

Next, we expand $\bigcup_n \Gamma_n$ to a maximal consistent set by the Lindenbaum construction. Let $\varphi_0, \ldots, \varphi_n, \ldots$ be an enumeration of all formulae. Let $\Gamma_0' = \bigcup_n \Gamma_n$ and let $\Gamma_{n+1}' = \Gamma_n' \cup \{\varphi_n\}$ if $\Gamma_n' \cup \{\varphi_n\}$ is consistent, otherwise let $\Gamma_{n+1}' = \Gamma_n'$. The union Γ of all Γ_n' is consistent; moreover, it is *maximal* (for each φ either $\varphi \in \Gamma$ or $\Gamma \cup \{\varphi\} \vdash \bot$) and—by the construction in the previous paragraph, the properties of maximal consistent sets of formulae, Leibniz's Law and (the dual of) AS6—it has the *Henkin property*: If $\varphi(u) \in \Gamma$ for all free variables u of some type α, then, since $\eta x \neg([x/u]\varphi) = u' \in \Gamma$ for some u', $\varphi(\eta x \neg([x/u]\varphi)) \in \Gamma$ and hence $\forall x \varphi(x) \in \Gamma$.

Now define an equivalence relation $\sim$ between terms: $A \sim B$ iff $A = B \in \Gamma$. For each term A, let $[A]$ be the equivalence class $\{B \mid A \sim B\}$. For each type α we define a function Φ_α having the set $\{[A] \mid A \text{ is a term of type } \alpha\}$ as its domain. If $\alpha = e$ or $\alpha = s$, let $\Phi_\alpha([\eta x_\alpha \bot]) = \varnothing$ and let $\Phi_\alpha([A_\alpha]) = [A]$ if $[A] \neq [\eta x_\alpha \bot]$. If $\alpha = \langle \alpha_1 \ldots \alpha_n \rangle$, let $\Phi_\alpha([A_\alpha])$ be the relation $\{\langle \Phi_{\alpha_1}([B_{\alpha_1}]), \ldots, \Phi_{\alpha_n}([B_{\alpha_n}]) \rangle \mid AB_{\alpha_1} \ldots B_{\alpha_n} \in \Gamma\}$. This is well-defined by Leibniz's Law and the maximal consistency of Γ.

The functions Φ_α are injections. This is obvious in case $\alpha = e$ or $\alpha = s$, so let $\alpha = \langle \alpha_1 \ldots \alpha_n \rangle$. Suppose that $\Phi_\alpha([A]) = \Phi_\alpha([A'])$. Let $u_1, \ldots, u_n$ be free variables of types $\alpha_1, \ldots, \alpha_n$ respectively, then $Au_1 \ldots u_n \in \Gamma$ iff $A'u_1 \ldots u_n \in \Gamma$. From this it follows that $Au_1 \ldots u_n \leftrightarrow A'u_1 \ldots u_n \in \Gamma$. By the Henkin property: $\forall x_1 \ldots x_n (Ax_1 \ldots x_n \leftrightarrow A'x_1 \ldots x_n) \in \Gamma$, so, using Extensionality and the maximal consistency of Γ, we see that $A = A' \in \Gamma$ and $[A] = [A']$.

Define: $D_\alpha = \{\Phi_\alpha([A]) \mid A \text{ is a term of type } \alpha\}$ if α is relational and $D_\alpha = \{\Phi_\alpha([A]) \mid A \text{ is a term of type } \alpha\} \setminus \{\varnothing\}$ if α is basic. From the definition of the

functions Φ_α and the fact that AS4 is an axiom scheme it follows that $F = \{D_\alpha\}_\alpha$ is a frame. Define $I(c_\alpha) = \Phi_\alpha([c_\alpha])$ and define for each α and each D such that $D \subseteq D_\alpha$:

$$\begin{array}{lcl} H_\alpha(D) & = & \Phi_\alpha([\eta x_\alpha \varphi(x)]), \text{ if } D = \{\Phi_\alpha([u]) \mid \varphi(u) \in \Gamma\}; \\ & = & \Phi_\alpha([u]), \text{ where } u \text{ is the first free variable such that } \\ & & \Phi_\alpha([u]) \in D, \text{ otherwise.} \end{array}$$

The functions H_α are well-defined. First, note that $\emptyset = \{\Phi_\alpha([u]) \mid \bot \in \Gamma\}$, so the second clause is all right. Next, suppose that for all free variables u_α: $\varphi(u) \in \Gamma$ iff $\psi(u) \in \Gamma$. Then $\forall x(\varphi(x) \leftrightarrow \psi(x)) \in \Gamma$ and by AS7: $\eta x\varphi(x) = \eta y\psi(y) \in \Gamma$, from which it follows that $[\eta x\varphi(x)] = [\eta y\psi(y)]$.

The functions H_α are choice functions for the sets D_α. Clearly $H_\alpha(\emptyset) = \Phi_\alpha([\eta x_\alpha \bot]) = \emptyset$. Suppose that $D \subseteq D_\alpha$ is not empty. If the second clause of H_α's definition obtains, then obviously $H_\alpha(D) \in D$. So suppose that $D = \{\Phi_\alpha([u]) \mid \varphi(u) \in \Gamma\}$ for some φ. Since $D \neq \emptyset$ and $D \subseteq D_\alpha$ (and hence $\emptyset \notin D_\alpha$ if α is basic) there is a variable u such that $\varphi(u) \in \Gamma$ and such that moreover $\neg(u = \eta x \bot) \in \Gamma$ if α is basic. By AS5 we see that $\exists x\varphi(x) \in \Gamma$ and therefore by AS6 $\varphi(\eta x\varphi(x)) \in \Gamma$. By the construction of Γ there is a free variable u' such that $u' = \eta x\varphi(x) \in \Gamma$ and so for this variable both $\varphi(u') \in \Gamma$ and $[u'] = [\eta x\varphi(x)]$ hold, whence $\Phi_\alpha([\eta x\varphi(x)]) \in D$.

Now, let M be the weak general model $\langle F,I,H\rangle$ and let the assignment a be defined by $a(u_\alpha) = \Phi_\alpha([u])$. We prove by induction on term complexity that $\|A\|^{M,a} = \Phi_\alpha([A])$ for all terms A of type α, hence that $\|\varphi\|^{M,a} = 1$ iff $\varphi \in \Gamma$, for all formulae φ and hence that M is a weak general model of T:

i. $\|c\|^{M,a} = I(c) = \Phi([c])$ if c is a constant;
$\|u\|^{M,a} = a(u) = \Phi([u])$ if u is a free variable;

ii. $\|AB\|^{M,a} = \{\langle d_1, \ldots, d_n\rangle \mid \langle \|B\|^{M,a}, d_1, \ldots, d_n\rangle \in \|A\|^{M,a}\} = \{\langle d_1, \ldots, d_n\rangle \mid \langle \Phi([B]), d_1, \ldots, d_n\rangle \in \Phi([A])\} = \Phi([AB])$;

iii. $\|A = B\|^{M,a} = 1 \Leftrightarrow \|A\|^{M,a} = \|B\|^{M,a} \Leftrightarrow \Phi([A]) = \Phi([B]) \Leftrightarrow [A] = [B] \Leftrightarrow A = B \in \Gamma \Leftrightarrow \Phi([A = B]) = 1$;

iv. $\|\eta x_\alpha([x/u]\varphi)\|^{M,a} = H_\alpha(\{d \in D_\alpha \mid \|\varphi\|^{M,a[d/u]} = 1\}) = H_\alpha(\{\Phi_\alpha([u]) \mid \|\varphi\|^{M,a} = 1\}) = H_\alpha(\{\Phi_\alpha([u]) \mid \varphi \in \Gamma\}) = \Phi_\alpha([(\eta x[x/u]\varphi)])$.

NOTES

* I would like to thank Johan van Benthem and Dick de Jongh for their moral support and their critical comments on earlier versions of this paper. Martin Stokhof, Elias Thijsse and the anonymous referees made very valuable remarks.
The research for this paper was supported by the Netherlands Organization for the Advancement of Pure Research (ZWO).

1 See e.g. the definition of *persistence* in Gallin [1975, §4] (and compare it with the one in his §9).

2 The choice of a particular logic should of course not preclude certain analyses of natural language. It should, for example, not be made *impossible* by our logic to use individual concepts (type (se) objects, functions from worlds to entities). However, since all functions are relations, there is no problem. Those who think that individual concepts are useful (see Janssen [1984]) may keep them as ‹se›-type relations (relations between worlds and entities). Expressions like **the pope** can then be treated as individual concepts. Note that **the pope** can't be a function since there have been times that there was no pope and once, during the Avignon period, there were three.
Another example of a semantical theory that forms an exception to the rule that in current Montague Grammar only quasi-relational types are used and that should not be ruled out on a priori grounds is the analysis of questions with the help of Skolem functions (see Engdahl [1980], Groenendijk & Stokhof [1985]). On this account, one reading of the question ‘Whom does every man love’ should be rendered as $\lambda f_{(ee)} \forall x(man\ x \rightarrow love\ x\ fx)$ (I suppress index variables here). It is easily seen, however, that the theory can be reformulated in relational terms. For example, in this case, the relational term $\lambda R_{\langle ee \rangle} \forall x(man\ x \rightarrow love\ x\ \iota y(Rxy))$ (for notation see below) would do the same work and could be obtained compositionally along Groenendijk & Stokhof's lines.

3 Some may want to demand that all partial relations have empty glut. It is possible to add this condition without any technical difficulties. See Muskens [forthcoming] for a discussion.

4 It should be emphasized however, that the addition of description operators is not essential. Relational type theories can be formulated without them. See also Muskens [forthcoming]

5 As opposed to functional frames, our frames have the empty set in all their relational domains. This provides a particularly natural denotation for $\iota x \bot$ if x is of relational type. There are ways to repair the omission in functional type theory, but none of them is as straightforward as the present solution. This point is of course related to argument III above.

6 Of course, the ordinary generalized quantifier approach to noun phrases, including definites, is available as well in our logic. **The king** can be treated as $\lambda P \exists x(\forall y(king\ y \leftrightarrow x = y) \wedge Px)$, which by the way is equivalent to $\lambda P(P \iota x(king\ x))$.

7 The η-symbol has entered the linguistic literature on a modest scale through this book. However, despite its name, I do not think that it would be wise to apply the indefinite description operator in any straightforward way to the treatment of indefinite descriptions in natural language. While it is defensible (but not necessary) to treat **the man** as $\iota x(man\ x)$, treating **a man** as $\eta x(man\ x)$ leads to wrong results (under normal assumptions), as the reader can easily verify once he has seen the semantics of the η-symbol given below. Clearly nothing in the present proposals is in conflict with treating **a man** as $\lambda P \exists x(man\ x \wedge Px)$, like Montague did (compare note 6 above). I am indebted to two anonymous referees for pointing out to me that my presentation in an earlier version of this paper was misleading in this respect.

8 This distinction between free and bound variables makes it easy to avoid variable clashes in cases of substitution, but it is not an essential feature of the theory.

9 To avoid any confusion: the $[\sigma'/s]\sigma$ notation is a way to *refer* to strings. The square brackets are not part of any string.

REFERENCES

Asser, G.: 1957, Theorie der logischen Auswahlfunktionen, *Zeitschrift für mathematische Logik und Grundlagen der Mathematik* **3**, 30-68.

Barwise, J.: 1981, Scenes and Other Situations, *The Journal of Philosophy*, **78**, 369-397.

Barwise, J. and Cooper R.: 1981, Generalized Quantifiers and Natural Language, *Linguistics and Philosophy* **4**, 159-219.

Barwise, J. and Perry J.: 1983, *Situations and Attitudes*, MIT Press, Cambridge, Mass.

Bennett, M.: 1974, *Some Extensions of a Montague Fragment of English*, UCLA, Dissertation.

Van Benthem, J.F.A.K. and Doets, K.: 1983, Higher-Order Logic, in Gabbay & Guenthner [1983], pp. 275-329.

Carnap, R.: 1929, *Abriß der Logistik*, Verlag von Julius Springer, Vienna.

Church, A.: 1940, A Formulation of the Simple Theory of Types, *The Journal of Symbolic Logic* **5**, 56-68.

Engdahl, E.: 1980, *The Syntax and Semantics of Questions in Swedish,* University of Massachusetts at Amherst, Dissertation, reprinted as *Constituent Questions: The Syntax and Semantics of Questions with Special Reference to Swedish,* Reidel, Dordrecht 1985.

Gabbay, D. and Guenthner, F. (eds.): 1983, *Handbook of Philosophical Logic*, Vol. 1, Reidel, Dordrecht.

Gallin, D.: 1975, *Intensional and Higher-Order Modal Logic*, North-Holland, Amsterdam.

Gazdar, G.: 1980, A Cross-Categorial Semantics for Coordination, *Linguistics and Philosophy* **3**, 407-409.

Groenendijk, J. and Stokhof, M.: 1984, *Studies on the Semantics of Questions and the Pragmatics of Answers*, Dissertation, University of Amsterdam.

Henkin, L.: 1950, Completeness in the Theory of Types, *The Journal of Symbolic Logic* **15**, 81-91.

Hermes, H.: 1965, *Eine Termlogik mit Auswahloperator*, Springer, Berlin-Heidelberg-New York.

Hilbert, D. and Bernays, P.: 1939, *Grundlagen der Mathematik*, Vol. 2, Springer, Berlin.

Janssen, T.: 1984, Individual Concepts are Useful, in A. Landman and F. Veltman (eds.), *Varieties of Formal Semantics*, Foris, Dordrecht, 171-192.

Kamp, H.: 1981, A Theory of Truth and Semantic Representation, reprinted in J. Groenendijk, T. Janssen and M. Stokhof (eds.), *Truth, Interpretation and Information,* Foris, Dordrecht, 1984, 277-322.

Keenan, E. and Faltz, L.:1978, *Logical Types for Natural Language*, UCLA Occasional Papers in Linguistics, 3.

Kratzer, A.: 1977, What *Must* and *Can* Can and Must Mean, *Linguistics and Philosophy* **1**.

Landman, A.: 1986, *Towards a Theory of Information,* Dissertation, University of Amsterdam.

Leisenring, A.C.: 1969, *Mathematical Logic and Hilbert's ε-Symbol*, Macdonald, London.

Montague, R.: 1973, The Proper Treatment of Quantification in Ordinary English, reprinted in Montague [1974], pp. 247-270.

Montague, R.: 1974, *Formal Philosophy*, Yale University Press, New Haven.

Muskens, R.A.: forthcoming, Going Partial in Montague Grammar, to appear in R. Bartsch, J.F.A.K. van Benthem and P. van Emde Boas (eds.), *Proceedings of the Sixth Amsterdam Colloquium*, Foris, Dordrecht.

Orey, S.: 1959, Model Theory for the Higher Order Predicate Calculus, *Transactions of the American Mathematical Society* **92**, 72-84.

Reichenbach, H.: 1947, *Elements of Symbolic Logic, Macmillan,* New York.

Renardel de Lavalette, G.: 1984, Descriptions in Mathematical Logic, *Studia Logica* **43**, 281-294.

Russell, B.: 1908, Mathematical Logic as Based on the Theory of Types, *American Journal of Mathematics* **30**, 222-262.
Schönfinkel, M.: 1924, Über die Bausteine der mathematischen Logik, *Mathematische Annalen* **92**, 305-316.
Scott, D.: 1967, Existence and Description in Formal Logic, in R. Schoenman (ed.), *Bertrand Russell, Philosopher of the Century*, Allen & Unwin, London, 660-696.
Von Stechow, A.: 1974, ε-λ kontextfreie Sprachen: Ein Beitrag zu einer natürlichen formalen Semantik, *Linguistische Berichte* **34**, 1-33.
Veltman, F.: 1985, *Logics for Conditionals*, Dissertation, University of Amsterdam.

Department of Linguistics
Tilburg University
P. O. Box 90153
5000 LE Tilburg