THE JOURNAL OF SYMBOLIC LOGIC
Volume 52, Number 4, Dec. 1987

CATEGORICAL SEMANTICS FOR HIGHER ORDER POLYMORPHIC LAMBDA CALCULUS

R. A. G. SEELY

Abstract. A categorical structure suitable for interpreting polymorphic lambda calculus (PLC) is defined, providing an algebraic semantics for PLC which is sound and complete. In fact, there is an equivalence between the theories and the categories. Also presented is a definitional extension of PLC including "subtypes", for example, equality subtypes, together with a construction providing models of the extended language, and a context for Girard's extension of the Dialectica interpretation.

§0. Introduction. Since its development in the early 1970's by Girard [1972] and independently by Reynolds [1974], the polymorphic lambda calculus (PLC) has been the object of increasing study. In this paper I shall define a categorical structure suitable for interpreting PLC, thus providing a smooth, algebraic semantics for PLC. The usual soundness and completeness theorems are valid for this semantics; in fact, we shall show an equivalence between the theories and the categorical structures. Furthermore, I shall present a definitional extension of PLC to include "subtypes", the most important of which are equality subtypes, and a construction on the categorical structures which provides a suitable semantics for the extended language. A bonus of this construction is the creation of a type of natural numbers, thus providing the context needed for the Dialectica interpretation given by Girard [1972]. Finally, in order that the semantics presented here may be compared easily to that of Bruce and Meyer [1984], I shall describe the model of closure operators in $\mathscr{P}\omega$, due to McCracken [1979] and Scott [1976], in some detail.

Girard introduced PLC in [1971], [1972] to extend Gödel's Dialectica interpretation to analysis; this origin is illuminating because it suggests we regard PLC as a higher order version of Gödel's system of functionals. If for simplicity we ignore the natural numbers component of these systems, Gödel's system reduces to typed lambda calculus and PLC is just a higher order typed lambda calculus. (The term "polymorphic" has two connotations: one is just "multi-sorted" or "typed", and the second refers to the ability to pass types as parameters in term and type expressions, as we shall see.) This means that PLC permits "type abstraction" as well as the usual first order lambda abstraction. For example, for any type σ we have the identity function for σ, $\lambda x \in \sigma \cdot x$; abstracting with respect to σ gives a "universal identity" function $\Lambda \alpha \cdot \lambda x \in \alpha \cdot x$. To these two kinds of lambda abstraction correspond two

Received May 29, 1986; revised October 1, 1986.

0022-4812/87/5204-0008/$03.10

kinds of "function type": the identity for σ is of type $\sigma \supset \sigma$ (or σ^σ), while the universal identity is of type $\prod\alpha \cdot \alpha \supset \alpha$.

Independently, Reynolds [1974] introduced the second order fragment of PLC as an illustration of the type structure of programming languages which are typed (to allow syntactic checking) and yet which allow the formation of such "uniform" constructions as the universal identity considered above. In fact, PLC is very powerful: all provably total recursive functions can be represented in PLC, provable, that is, in higher order arithmetic (second order, for Reynold's fragment) (Girard [1972]). Nevertheless, it is simple enough to be a useful prototype for studying properties of programming languages embodying the principles Reynolds described, such as HOPE and ML (Burstall et al. [1980]; Milner [1984]).

Using the Howard formulae-as-types isomorphism, we can look at PLC in another way: the types of PLC are formulae of intuitionist higher order type theory (or propositional calculus), and the terms are derivations (of their types, with assumptions as the types of free variables). Under this interpretation $\prod\alpha \cdot \sigma[\alpha]$ is universal quantification: if one has a derivation $a[\alpha]$ of $\sigma[\alpha]$, with appropriate conditions on α, then $\Lambda\alpha \cdot a[\alpha]$ is the corresponding derivation of $\forall\alpha \cdot \sigma[\alpha]$. It was via this idea that Girard proved the syntactic form of Takeuti's conjecture from normalisation for PLC [1971].

Probably the most elegant formulation of the semantics of lambda calculus is via cartesian closed categories (Lambek and Scott [1986]). The purpose of this paper is to describe the corresponding categorical semantics for PLC, with the intent of achieving a similar conceptual simplification of other known semantics, such as that of Bruce and Meyer [1984]. Indeed, from the considerations above, a good idea of what to expect is evident. For we start from the well-known equivalence between typed lambda calculii and cartesian closed categories (Lambek and Scott [1986]). To permit the higher order function types $\prod\alpha \cdot \sigma[\alpha]$, the categorical structure must also be "complete" in some sense, i.e., must allow the formation of "arbitrary" products (or at least "uniformly defined" ones). Of course this is impossible (see Reynolds [1984], for example). The way around this dilemma is to interpret the categorical structure in an appropriate "universe": a "PL category" should be an *internal* complete cartesian closed category in some other category than the category of **Sets** (see §2.5).

Another approach to this notion is via the analogy with type theory: the semantics for type theory is given by toposes (Lambek and Scott [1986]), and so to get PLC we need only replace the notion of entailment (used to construct toposes) with the richer structure of deductions as given in PLC (see §2.4).

Some remarks concerning presentation: Girard's original system [1972] was a higher order type theory, unlike his system in [1971] or Reynolds [1974], which only contain a second order fragment of the full system. I have retained the full higher order feature, since the categorical semantics seem more natural in this setting: see, for example, the description of the closure operator model in §3. Whereever "functions types" occur, I have also included "product types"; in the case of "orders" this introduces new orders, but does not essentially alter the content of the theory. (Girard's system had product types, but not product orders.) Girard's system also had sum types and a type of natural numbers: I completely omit the former, and leave a discussion of the latter to §7.

A more important modification of the original Girard/Reynolds system has to do with the equality rules or axioms: in addition to the usual reduction (or β) rules, I also require expansion (or η) rules. So, for instance, every term of a function type is a function, and every term of a product type is an ordered pair. This is standard from the categorical viewpoint, where it amounts to admitting the two "triangle equalities" which define an adjunction. (See the book by Lambek and Scott [1986], for instance.)

An early version of the semantics defined in this paper was developed in Seely [1979], amounting to "internal PL categories", as in §2. The current presentation owes much to conversations with F. Lamarche, whom I thank for his helpful insights. In particular, the construction of §6 is based on a generalisation due to Lamarche [1985] of the "Freyd cover" construction. This work was done while I had the help of a grant from the Fonds F.C.A.R., Québec. A preliminary announcement of these results appeared in Seely [1986a], [1986b]. I would also like to acknowledge the suggestions of the referee, especially with regard to the amplification of my introductory remarks.

§1. Polymorphic lambda calculus. To begin with, I shall give an informal definition of the type theory or polymorphic lambda calculus used in the paper, leaving the most important technical points until later. (The lesser points will be left to the reader.) However, I should note that, again following categorical precedent, I shall define what is meant by "a" (not "the") polymorphic lambda calculus; what is usually meant by "the" polymorphic lambda calculus is the *free* one (on some collection of generators).

Roughly speaking, PLC consists of four sorts of objects: *orders*, *operators*, *types*, and *terms*. The orders describe the kinds of objects the theory is talking about: one kind of object is the types, and these make up the order Ω; another kind of object is functions which assign types to types, and these make up the order Ω^{Ω}. Operators are just functions from one kind of objects to another kind; in particular, operators from some kind of object to types (i.e., $A \to \Omega$) are themselves called types (with a free variable of order A). (In this paragraph, there is a definite blurring of the distinction between functions from A to B and objects in B^A, but I think it can be useful at times to think of an expression with a free variable as the corresponding closed functional expression, just as is done in the phrase "the function $x^2 + 1$".) Finally, the types carry a structure allowing for the creation of products and function types, as described in the Introduction; the terms are the objects of the individual types. Notice that a term may have free variables of two different sorts: term variables of specified types, and operator variables of specified orders. (For example the term $\Lambda\beta \cdot \lambda x \in \alpha \cdot \lambda y \in \beta \cdot x$ has no free term variables of any type, but it has a free type variable α (of order Ω). It is of type $\prod\beta \cdot \alpha \supset \beta \supset \alpha$.) I hope these remarks will make the intended interpretation of the syntax below clear; a more precise version of these remarks is the essential content of the semantics of §4, in terms of the categorical structures of §2. (See especially the remarks in 1.2, 2.4, and 2.5.)

DEFINITION 1.1. A *PL theory* $\mathfrak{T}$ consists of three collections of objects: orders, operators, and terms. Each operator and each term has an "arity"; that is to say, each operator is of a certain order, and may have free variables in it ranging over certain orders. Similarly, each term is of a certain type, and may have free variables in it ranging over certain types. (A type is a special kind of operator.) Each of these collections may have a given set of constant symbols, with appropriate arities for operators and terms. Furthermore, they must be closed under the following rules:

(1.1.1. *Orders*) 1 and Ω are orders; if A and B are orders, then $A \times B$ and Ω^A are also orders.

(1.1.2. *Operators*) In the following, "$\sigma \in A$" means σ is an operator of order A; the rest of the arity is left unspecified for simplicity.

For each order, there is a countable set of variable operators (called "indeterminates").

$* \in 1$. $\top \in \Omega$.

If $\sigma, \tau \in \Omega$, then $\sigma \wedge \tau$ and $\sigma \supset \tau \in \Omega$.

If $\sigma \in \Omega$ and α is an indeterminate of order A, then $\sum\alpha \in A \cdot \sigma$ and $\prod\alpha \in A \cdot \sigma \in \Omega$.

$(\times I)$ If $\sigma \in A$, $\tau \in B$, then $\langle\sigma, \tau\rangle \in A \times B$.

$(\times E)$ If $\sigma \in A \times B$, then $\pi_1\sigma \in A$, $\pi_2\sigma \in B$.

(PI) If α is an indeterminate of order A and $\sigma \in \Omega$, then $[\alpha \in A: \sigma] \in \Omega^A$.

(PE) If $\tau \in A$, $\sigma \in \Omega^A$, then $\sigma(\tau) \in \Omega$.

DEFINITION 1.1.3. A type is an operator of order Ω.

(1.1.4. *Terms*) In the following, "$a \in \tau$" means a is a term of type τ; the rest of the arity is left unspecified for simplicity.

For each type, there is a countable set of variable terms (called "variables").

$(\top I)$ $* \in \top$.

$(\supset I)$ If $a \in \tau$, x a variable of type σ, then $\lambda x \in \sigma \cdot a \in \sigma \supset \tau$.

$(\supset E)$ If $a \in \sigma \supset \tau$, $b \in \sigma$, then $a(b) \in \tau$.

$(\wedge I)$ If $a \in \sigma$, $b \in \tau$, then $\langle a, b\rangle \in \sigma \wedge \tau$.

$(\wedge E)$ If $a \in \sigma \wedge \tau$, then $\pi_1 a \in \sigma$, $\pi_2 a \in \tau$.

$(\sum I)$ If α is an indeterminate of order A, $\sigma \in \Omega$, $\tau \in A$, then $I_{\sum\alpha \cdot \sigma, \tau} \in \sigma[\tau/\alpha] \supset \sum\alpha \in A \cdot \sigma$. When clear from the context, we shall denote this term by I_τ, or even by I; in particular, if $b \in \sigma[\tau/\alpha]$, then $I(b) \in \sum\alpha \in A \cdot \sigma$.

$(\sum E)$ If $a \in \sigma \supset \rho$, α an indeterminate of order A which is not free in ρ nor in the type of any free variable in a, then $\mathrm{V}\alpha \in A \cdot a \in (\sum\alpha \in A \cdot \sigma) \supset \rho$.

$(\prod I)$ If $a \in \sigma$, α an indeterminate of order A which is not free in the type of any free variable in a, then $\Lambda\alpha \in A \cdot a \in \prod\alpha \in A \cdot \sigma$.

$(\prod E)$ If $a \in \prod\alpha \in A \cdot \alpha$, $\tau \in A$, then $a\{\tau\} \in \sigma[\tau/\alpha]$, where $\sigma[\tau/\alpha]$ is σ with τ replacing α.

(1.1.5. *Equalities*) In addition to statements of the form "$\alpha \in A$", "$a \in \sigma$", (the latter only if "$\sigma \in \Omega$" has been derived), the PL theory $\mathfrak{T}$ will also contain equations, i.e., statements of the form "$\sigma = \tau$" and "$a = b$" (the former only if "$\sigma \in A$" and "$\tau \in A$", the latter only if "$\sigma \in \Omega$", "$a \in \sigma$", and "$b \in \sigma$", have been derived.) $\mathfrak{T}$ may contain certain equations as (nonlogical) axioms. In addition, $\mathfrak{T}$ must contain all the usual equality rules of reflexivity, symmetry, transitivity, substitution, and change of bound variables or indeterminates. (The variable binding symbols are $\sum$, $\prod$, [:], Λ and V for indeterminates, and λ for variables.) Furthermore $\mathfrak{T}$ has the following rules.

(1.1.6. *Equalities for operators*) Using the notation of §1.1.2:

(1 red) $\tau = *$ for $\tau \in 1$.

(P red) $[\alpha \in A: \sigma](\tau) = \sigma[\tau/\alpha]$.

(P exp) $\sigma = [\alpha \in A: \sigma(\alpha)]$.

($\times$ red) $\pi_1\langle\sigma,\tau\rangle = \sigma$, $\pi_2\langle\sigma,\tau\rangle = \tau$.

($\times$ exp) $\sigma = \langle\pi_1\sigma, \pi_2\sigma\rangle$.

(1.1.7. *Equalities for terms*) Using the notation of §1.1.4:

($\top$ red) $a = *$ for $a \in \top$.

($\supset$ red) $(\lambda x \in \sigma \cdot a)(b) = a[b/x]$.

($\supset$ exp) $a = \lambda x \in \sigma \cdot a(x)$.

($\wedge$ red) $\pi_1\langle a,b\rangle = a$, $\pi_2\langle a,b\rangle = b$.

($\wedge$ exp) $a = \langle\pi_1 a, \pi_2 a\rangle$.

($\sum$red) $(\mathrm{V}\alpha \in A.a)(I(b)) = a[\tau/\alpha](b)$.

($\sum$exp) If $f \in (\sum\alpha \in A \cdot \sigma) \supset \rho$ then $f = \mathrm{V}\alpha \in A.\lambda x \in \sigma \cdot f(I_{\Sigma\alpha\cdot\sigma,\alpha}(x))$.

(This should be thought of as $f = \mathrm{V}\alpha \in A(f \circ I)$.)

($\prod$ red) $(\Lambda\alpha \in A \cdot a)\{\tau\} = a[\tau/\alpha]$.

($\prod$exp) $a = \Lambda\,\alpha \in A \cdot a\{\alpha\}$.

1.2. Remarks. (1) When the order (respectively, type) of an indeterminate (variable) is clear from the context, we will frequently omit it when using the indeterminate (or variable) binding symbols, or alternately we might indicate it by the use of a subscript, as in $\prod\alpha \cdot (\alpha \supset \alpha)$ or $\lambda x_\sigma \cdot a$. (Note in the former, $\alpha \in \Omega$ is implied by forming $\alpha \supset \alpha$.)

(2) The restrictions in ($\sum E$) and ($\prod I$) are just the usual restrictions familiar from first order logic: for instance $\Lambda\,\alpha \cdot x_\alpha$ is not well-formed, which is just as well, for it would then give a term of type $\prod\alpha \in \Omega \cdot \alpha$, in general not a desirable thing. On the other hand, $\Lambda\,\alpha \cdot \lambda x_\alpha \cdot x$ is perfectly harmless, being the "universal identity" term of type $\prod\alpha \cdot (\alpha \supset \alpha)$.

(3) In §5 we shall construct PL theories from certain categories, which will have the property that all operators and terms will be "stratified", in the sense below. In order that the construction generate an equivalence, we shall impose that condition now on PL theories. (However, if the reader is unhappy with this restriction, it may be ignored—then our construction will just give an adjunction, which in practice is sufficient for our purposes.)

So, in forming operators $\sigma \wedge \tau$, $\sigma \supset \tau$, $\sigma(\tau)$, $\langle\sigma,\tau\rangle$, we shall require that the same free indeterminates appear in σ and in τ. (This may be accomplished by using "dummy" indeterminates, if necessary.) Similarly, in forming terms a, we assume that the types of a and of all free variables in a have the same free indeterminates; in forming $a(b)$ and $\langle a,b\rangle$ we suppose a and b have the same free variables. (Again, this may involve the use of "dummy" variables or indeterminates.) However, we shall not consider dummy indeterminates as possible violations of the restrictions in ($\sum E$) and ($\prod I$): for example, in ($\prod I$) α may occur as a dummy in a free variable of a, in which case it would then no longer occur free (even as a dummy) in $\Lambda\,\alpha.\,a$. (A similar approach to first order logic may be found in Seely [1983].)

(4) Although from a syntactic viewpoint stratification may seem unnatural, it does result in giving a PL theory the following structure: we have a "base-universe" of orders and operators, and over each order, we have a "fibre" filled with all the types whose free indeterminate ranges over the given order, and all the terms of those types (whose free variables also range over such types.) (Note that with products, we can assume without loss of generality that an expression has exactly one free

indeterminate or one free variable.) There are two ways of going from one fibre to another: substituting an operator for an indeterminate, and quantifying with $\sum$ or $\prod$, which replaces an indeterminate of order $A \times B$ by one of order B. Note that the fibre over 1 consists of the closed types. It is precisely this kind of structure we shall abstract in defining a PL category in §2.

(5) In §4, we shall interpret operators as maps in a certain category. It will be convenient for this purpose to suppose each set of indeterminates has a canonical well-ordering, and that in a well-formed expression, indeterminates are chosen from each appropriate order according to these well-orders. The intention is this: if $\alpha_1, \alpha_2 \in A$, we shall interpret the operator α_1 as the identity on A. If one were to write the expression α_2 in isolation, there is no reason to suppose it should be interpreted differently; it is only in contexts such as $\langle \alpha_1, \alpha_2 \rangle$ that α_1 and α_2 have different roles: contrast this with $\langle \alpha_1, \alpha_1 \rangle$, for instance. However, minor technical problems arise if we say α_1 and α_2 have the same interpretation when they stand alone, but different ones when together (but clearly $\langle \alpha_1, \alpha_2 \rangle$ should be the identity on $A \times A$, whereas $\langle \alpha_1, \alpha_1 \rangle$ should be the "diagonal" map $A \to A \times A$). So we adopt a "variable-labelling" convention to avoid this problem. Of course, ideally we should dispense with variables and indeterminates completely, and give a variable-free and indeterminate-free presentation of PLC. This is essentially what PL categories amount to: "the basic idea of categorical logic is that a logical theory has an intrinsic existence independent of its presentation, and that this existence is best represented by a category" (Seely [1982]).

§2. PL categories. We assume familiarity with the basic notions of category theory, e.g. Mac Lane [1971].

DEFINITION 2.1. A *PL category* $(\mathbf{G}, \mathbf{S})$ consists of:

(i) a category $\mathbf{S}$ with finite products, a distinguished object Ω, and exponentiation of the form Ω^A for A in $\mathbf{S}$ (precisely, for each object A of $\mathbf{S}$, there is an object Ω^A so that $\mathrm{Hom}_{\mathbf{S}}(B, \Omega^A) \cong \mathrm{Hom}_{\mathbf{S}}(A \times B, \Omega)$, naturally in B), and

(ii) an indexed category $\mathbf{G}$ over $\mathbf{S}$ satisfying:

(a) for each object A in $\mathbf{S}$, $\mathrm{Obj}(\mathbf{G}(A)) \cong \mathrm{Hom}_{\mathbf{S}}(A, \Omega)$, and for each morphism $f: A \to B$, $\mathbf{G}(f)$ acts as $\mathrm{Hom}_{\mathbf{S}}(f, \Omega)$ on objects (so $\mathbf{G}(f)$ is defined by composition);

(b) for each object A in $\mathbf{S}$, $\mathbf{G}(A)$ is a cartesian closed category, and for each $f: A \to B$ in $\mathbf{S}$, $\mathbf{G}(f)$ preserves the cartesian closed structure; and

(c) $\mathbf{G}$ is "weakly complete and cocomplete": for each object C in $\mathbf{S}$, the canonical indexed functor $\kappa_C: \mathbf{G} \to \mathbf{G}^C$ has left and right adjoints $\sum_C \dashv \kappa_C \dashv \prod_C$.

2.2. REMARKS. (1) For more on indexed categories, one may consult Paré and Schumacher [1978]. However, in view of condition (a), one may view $\mathbf{G}$ as a functor $\mathbf{S}^{\mathrm{op}} \to \mathbf{Cat}$, where $\mathbf{Cat}$ is the category of (small) categories, functors, and natural transformations. For C in $\mathbf{S}$, $\mathbf{G}^C$ is then the functor defined by $\mathbf{G}^C(A) = \mathbf{G}(A \times C)$, and $\kappa_C: \mathbf{G} \to \mathbf{G}^C$ is the natural transformation defined by $\kappa_C(A) = \mathbf{G}(\pi): \mathbf{G}(A) \to \mathbf{G}(A \times C)$, for $\pi: A \times C \to A$ the projection. For f a morphism of $\mathbf{S}$, it is usual to write f^* for $\mathbf{G}(f)$ when $\mathbf{G}$ is clear from the context.

(2) The effect of $\mathbf{G}$ is to give a categorical structure (externally) to the object Ω of $\mathbf{S}$. In fact, the Yoneda lemma implies that in a PL category, there will be morphisms $\top: 1 \to \Omega$, $\wedge$, $\supset: \Omega \times \Omega \to \Omega$, $\sum_C$, $\prod_C: \Omega^C \to \Omega$ which induce the

structure on **G** for objects. For example, if $\sigma, \tau\colon A \to \Omega$ are objects of $\mathbf{G}(A)$, then

$$\sigma \wedge \tau = A \xrightarrow{\langle \sigma,\tau\rangle} \Omega \times \Omega \xrightarrow{\wedge} \Omega$$

is their product. The morphism $\kappa_C\colon \Omega \to \Omega^C$ inducing the external κ_C always exists—it is the "constant" (or K-combinator) $\sigma \mapsto \lambda x \cdot \sigma$.

DEFINITION 2.3. A functor $F\colon(\mathbf{G},\mathbf{S}) \to (\mathbf{G}',\mathbf{S}')$ of PL categories consists of a functor $F_0\colon \mathbf{S} \to \mathbf{S}'$ preserving finite products and exponents of the form Ω^A, and an **S**-indexed functor $F_1\colon \mathbf{G} \to F_0^*\mathbf{G}'$ which preserves cartesian closedness and weak completeness and cocompleteness.

Here $F_0^*\mathbf{G}'$ is the indexed category over **S** defined by $F_0^*\mathbf{G}'(A) = \mathbf{G}'(F_0(A))$. F_1 preserving weak completeness means that for A, C in **S**, the following must commute:

$$\begin{tikzcd}
\mathbf{G}(A \times C) = \mathbf{G}^C(A) \arrow[r, "\Pi_C(A)"] \arrow[d, "F_1(A\times C)"'] & \mathbf{G}(A) \arrow[d, "F_1(A)"] \\
\mathbf{G}'(F_0A \times F_0C) = \mathbf{G}'^{F_0C}(F_0A) \arrow[r, "\Pi_{F_0C}(F_0A)"] & \mathbf{G}'(F_0A)
\end{tikzcd}$$

A similar condition holds for $\sum$.

The condition that F_1 be **S**-indexed is just to guarantee that F_1 commutes with f^*; viewing **G** and **G**′ as functors, this means F_1 is a natural transformation. **PL Cat** is the category of PL categories and functors.

2.4. An analogy. For those who are familiar with the relationship between λ-calculus and cartesian closed categories, and more particularly between type theory and toposes (see Lambek and Scott [1986]), the following analogy should make the relationship between PLC (i.e. PL theories) and PL categories clear.

Type theory	*PLC*	*Topos/tripos; PL category*
type	order	object of base category
term	operator	morphism of base category
formula	type	object of fibre
derivation	term	morphism of fibre

The idea is that PLC may be regarded as type-theory-as-a-deductive-system (in the Lambek and Scott [1986] sense), replacing entailment between formulae with equivalence classes of derivations. (The equivalence relation is the smallest making $\wedge$ a product, $\supset$ an exponent, $\top$ a terminal object, and $\sum$, $\prod$ adjoints to κ, i.e. infinite sums and products.) In fact, ignoring the natural numbers object N of type theory, the orders, operators, and types of PLC, as given §1, are exactly the types, terms, and formulae respectively of type theory given in Lambek and Scott [1986]. (Actually, Lambek and Scott include $\perp$ and $\vee$ in their formulae, which we omit, but that is inessential.) The terms of PLC are exactly the derivations of type theory, were it to be considered a deductive system.

However, when we get around to including a natural numbers object, there is a significant breakdown in the analogy: each system would add N as a type, but that has a different meaning in the two systems. Similarly, when we consider equality types, in PLC two terms will determine an equality (sub-)type, whereas in type theory, two terms determine an equality formula—"term" has a different significance in each system.

The analogy is clearest when considering the categorical structures, particularly if instead of toposes we use triposes to interpret type theory. A tripos (Hyland, Johnstone, and Pitts [1980]) is a modest generalization of the structure of the subobject functor $\mathbf{E}^{\mathrm{op}} \to \mathbf{Poset}$, assigning to an object of a topos $\mathbf{E}$, the poset of its subobjects. A PL category generalises this notion, replacing poset structure with categorical structure. There is an adjunction between the categories of type theories and triposes, which is analogous to the adjunction between PLC and PL categories (although I have presented these so that we actually get an equivalence).

From the categorical view point, the different ways type theory and PLC treat a natural numbers object is clear: type theory adds N to the base category, whereas PLC adds it to the fibres. Similarly, equality in type theory permits the formation of generalised $\sum_f$ and $\prod_f$, adjoints for any f^*, whereas equality in PLC amounts to having finite limits (in particular, equalisers) in the fibres.

There is a small point that ought to be mentioned in connection with this analogy. In type theory, it is well known that one can define $\perp$, $\vee$, $\exists$ (our $\sum$) in terms of $\wedge$, $\supset$, $\forall$ (our $\prod$); this is not the case once one wishes to consider it as a deductive system. For instance $\prod\alpha \in \Omega \cdot \alpha$ is not an initial object (though it is a weak initial object), and $\prod\omega \in \Omega \cdot \prod\alpha \in A \cdot (\sigma \supset \omega) \supset \omega$ does not satisfy the adjointness conditions we require of $\sum\alpha \in A \cdot \sigma$. (Again, there is a "map" from each to the other, so they are equivalent in the poset case, but not in general.) For example, the closure operator model of §3 does not have an initial object.

2.5. Internal PL categories. With the analogy of 2.4 in mind, one might ask "what then corresponds to toposes, which are after all more natural than triposes"? The answer is the following. (As this has no effect on the rest of the paper, I give only a sketch here.)

An *internal PL category* is a PL category $(\mathbf{G}, \mathbf{S})$ in which Ω is in fact an internal category object (see Paré and Schumacher [1978]) and $\mathbf{G} \cong \mathrm{Hom}_{\mathbf{S}}(-, \Omega)$ (on morphisms as well as on objects.)

This means $\mathbf{S}$ has certain pullbacks, in order to accomodate the structure on Ω, that $\mathbf{S}$ has an object Ω_1 which is the internal representation of morphisms in $\mathbf{G}$, and that each morphism $a\colon \sigma \to \tau$ in $\mathbf{G}(A)$ is given by a morphism $A \xrightarrow{a} \Omega_1$ in $\mathbf{S}$, satisfying certain conditions. From the point of view of PLC, this would mean we had an order Ω_1 of terms, operators dom, cod: $\Omega_1 \to \Omega$ (among others), allowing us to say that a term a is an operator of order Ω_1, that a is of type $\mathrm{cod}(a)$, and the free variable of a is of order $\mathrm{dom}(a)$. Of course, Ω_1 has much more structure, since it must account for all the structure on terms. The point here is that this is very unnatural from the PLC point of view: models of PLC just do not internalise the notion of "term". This is a fact, but not a logical necessity, however. (But see Remarks 3.9(2) and 7.6(2).)

Indeed, there is an adjunction between the category of internal PL categories and the category of PL categories, analogous to that between toposes and triposes. The reflection, creating an internal PL category from a PL category, is constructed analogously to that for toposes: enlarge the base category by adding a new object Ω_1 and all the other objects and morphisms required by **G**. (The objects required include such things as an object Ω_2 of "composable pairs"; the morphisms include those such as dom and cod, required for the categorical structure on Ω, as well as morphisms $A \to \Omega_1$ and $A \to \Omega_2$, induced by the morphisms and composable pairs of morphisms in $\mathbf{G}(A)$, for example.)

Returning to the remarks in §0, we can see that the intended idea behind the notion of a PL category is in fact the notion of an internal PL category. That is, one should think of a PL category as being just a (weakly) complete and cocomplete cartesian closed category Ω; but this does not take place in the category of **Sets**, but rather in some other category **S**. For technical reasons, I have taken the categorical structure for Ω out of **S** and placed it in the fibres given by **G**, but this is really not crucial, in view of the adjunction discussed above. (Of course, it does keep the structure of **S** as simple as possible.)

One might wonder why in the type theory case, the "internal" notion, topos, turns out to be so natural. I would argue that, as a semantics for type theory, toposes are intrinsically less natural than triposes (than PL categories whose fibres are posets, in fact); however, since the order relation on Ω may be defined equationally for type theory ($p \leq q$ if $p \supset q \cdot = \cdot \top$), one does end up with natural models as toposes.

2.6. Second order PLC. Since the higher order system is less familiar than the second order one, a few remarks about the latter might be appropriate. Restricting to a second order structure essentially removes the need for **S** to have any exponents: at the most, one would expect exponentiation of "depth 1", giving objects like $\Omega^{\Omega \times \Omega}$, but not Ω^{Ω^Ω}. However, since we are mainly interested in behaviour at the fibre level, this means we will be considering fibres $\mathbf{G}^C$, where C has no exponentiation. Hence we might as well dispense with Ω^A altogether. In the case of the free structure (no other constants), **S** may as well then have the natural numbers as its objects, where the number n represents Ω^n. There is no alteration to make in the structure on **G**, but note that in essence the only instances of $\sum$, $\prod$ are $\Omega^\Omega \to \Omega$ (or the marginally more general $\Omega^{\Omega^n} \to \Omega$ which could be viewed as repeated instances of $\Omega^\Omega \to \Omega$), plus, if there are any constant orders A other than Ω, $\Omega^A \to \Omega$.

From the "internal" point of view, what we are saying is that Ω should be a cartesian closed category in a suitable universe **S**, with a restricted (weak) completeness and cocompleteness condition giving only these instances of $\sum$ and $\prod$.

§3. Closure operators as a PL category.

3.1. To illustrate the semantics defined in §4, we present a well-known model of PLC as an example of a PL category, viz. the closure operators in $\mathscr{P}\omega$. For basic results about $\mathscr{P}\omega$ as a model of the lambda calculus, and for proofs not given here, see Scott [1976].

DEFINITION 3.2. $K = \{a \,\varepsilon\, \mathscr{P}\omega : I \subseteq a = a \circ a\}$ is the subspace of $\mathscr{P}\omega$ of closure operators. K is the set of objects of a category **K**, whose morphisms $f: a \to b$ are $f \,\varepsilon\, \mathscr{P}\omega$ satisfying $f = b \circ f \circ a$.

REMARK. We use "ε" to denote set membership, as opposed to "$\in$" which is part of the arity of an operator or term.

DEFINITION 3.3. For $a \varepsilon K$, $T(a) = \{x \varepsilon \mathscr{P}\omega : a(x) = x\}$ is the set of fixed points of a. Note that $T(a)$ is the image of a, and is a subspace of $\mathscr{P}\omega$.

PROPOSITION 3.4. (1) $\mathbf{K}$ *is cartesian closed.*

(2) *There is an object* Ω *of* $\mathbf{K}$ *satisfying* $T(\Omega) = K$.

"PROOF". (1) 1 is the constant map with value ω. b^a is the map sending x to $b \circ x \circ a$. (In Scott [1976], b^a is denoted $a \circ\!\!\rightarrow b$; we shall also write $a \supset b$ for b^a and $a \wedge b$ or $a \times b$ for Scott's $a \boxtimes b$.) The main fact about $a \supset b$ is this:

LEMMA 3.5. *For* $a, b \varepsilon K$, $T(a \supset b) = \mathrm{Hom}_{\mathbf{K}}(a,b) \cong [T(a), T(b)]$ *and* $T(a \times b) \cong T(a) \times T(b)$, *where for topological spaces* X *and* Y, $[X, Y]$ *is the set of continuous maps from* X *to* Y.

(2) Ω *is Scott's* $V = \lambda a\, \lambda x. Y(\lambda y. x \cup a(y))$.

DEFINITION 3.6. (1) For $d \varepsilon K$, $\mathbf{G}(d)$ is the category $[T(d), \mathbf{K}]$ of continuous maps $T(d) \rightarrow K$. This means an object of $\mathbf{G}(d)$ is a map $d \rightarrow \Omega$ in $\mathbf{K}$, and a morphism $f: a \rightarrow b$ of $\mathbf{G}(d)$ is a continuous map $f: T(d) \rightarrow \mathscr{P}\omega$ satisfying $f(t): a(t) \rightarrow b(t)$ is a morphism of $\mathbf{K}$ for all $t \varepsilon T(d)$.

(2) For $g: e \rightarrow d$ in $\mathbf{K}$, $\mathbf{G}(g): \mathbf{G}(d) \rightarrow \mathbf{G}(e)$ is defined by composition: $g^*(a) = a \circ g$ and $g^*(f) = f \circ g$ (using the isomorphism of 3.5).

PROPOSITION 3.7. $(\mathbf{G}, \mathbf{K})$ *is a PL category.*

"PROOF". (1) For $d \varepsilon K$, $\mathbf{G}(d)$ is cartesian closed; the structure is given "pointwise" by the structure on $\mathbf{K}$, and so is preserved by any g^*.

(2) For $c \varepsilon K$, there are morphisms $\sum_c$, $\prod_c$: $\Omega^c \rightarrow \Omega$ of $\mathbf{K}$ defined as follows:

$$\sum_c = \lambda x \cdot \lambda\langle t \in c, y\rangle \cdot \langle t, x(t)(y)\rangle$$

and

$$\prod_c = \lambda x \cdot \lambda y \cdot \lambda t \in c. x(t)(y(t)).$$

For clarity, we have used some natural conventions here: $\lambda x \in a \cdot e[x]$ means $\lambda x \cdot e[a(x)]$, for an expression $e[x]$ in which x appears, $e[a(x)]$ being e with $a(x)$ replacing x. $\langle , \rangle$ are the pairing brackets for K, and $\lambda\langle x, y\rangle \cdot e[x, y]$ means $\lambda z \cdot e[\pi_1 z, \pi_2 z]$, where π_1, π_2 are the projections in K. Note that K has surjective pairing, so there is no confusion here. (Remark in passing that $\prod_c$ is almost the S-combinator; $\prod$ is in fact the combinator G of Barendregt and Rezus [1983].)

There are several routine things to check: that $\sum_c$ and $\prod_c$ are continuous maps $[T(c), K] \rightarrow K$; that they extend to functors $\mathbf{G}^c \rightarrow \mathbf{G}$ by composition (to define $\sum_c$ and $\prod_c$ for morphisms of $\mathbf{G}^c(d)$, use the same definitions as above, interpreting the "x" as a morphism of $\mathbf{G}^c(d)$); and that $\sum_c \dashv \kappa_c \dashv \prod_c$. These are all straightforward; the main point is this: We are thinking of the closure operators as "types", and of their fixed points as their "terms". Given $f: T(c) \rightarrow K$, we think of $\sum_c(f)$ as the "sum" of all $f(t)$, $t \in c$ (i.e. $t \varepsilon T(c)$), and of $\prod_c(f)$ as the "product" of all $f(t)$, $t \in c$: $\sum_c(f) =$ "$\sum t \in c \cdot f(t)$", $\prod_c(f) =$ "$\prod t \in c \cdot f(t)$". So a "term" (fixed point) of $\sum_c(f)$ ought to be a pair $\langle t, y\rangle$, where $t \in c$ and y is a "term" (fixed point) of $f(t)$: if $z = \sum_c(f)(z)$ then $z = \langle t, y\rangle$ where $y = f(t)(y)$ and $t \in c$, so that $\sum_c(f)(\langle t \in c, y\rangle) = \langle t, f(t)(y)\rangle$. Similarly, if $z = \prod_c(f)(z)$, then z must be a function on $T(c)$, so that $z(t) = f(t)(z(t))$, giving $\prod_c(f)(z)(t) = f(t)(z(t))$. Hence:

LEMMA 3.8. *Given* $c \in K$ *and* $f \in [T(c), K]$:

$$T\Big(\sum_c(f)\Big) \cong \sum t \in T(c) \cdot T(f(t)), \tag{1}$$

$$T\Big(\prod_c(f)\Big) \cong \prod t \in T(c) \cdot T(f(t)). \tag{2}$$

The adjunctions $\sum_c \dashv \kappa_c \dashv \prod_c$ follow immediately.

3.9. REMARKS. (1) The intention in forming the PL category $(\mathbf{G}, \mathbf{K})$ is to interpret types of PLC as closure operators and terms as fixed points. Indeed, $\mathbf{G}(1) = \mathbf{K}$, so closed types are exactly the closure operators, and since $\mathrm{Hom}_{\mathbf{K}}(1, a) \cong T(a)$, closed terms are fixed points. However, in this structure the closure operators are also playing the role of the orders. For a "leaner" order structure, we could have taken the full subcategory of $\mathbf{K}$ generated from 1 and Ω by the operations of $\times$ and $\Omega^{(\cdot)}$ as the base category instead of $\mathbf{K}$.

(2) In fact $\mathbf{K}$ is an internal PL category, via the following trick (which is possible since in this model, orders = closed types). Define $\Omega_1 = \sum\langle\alpha, \beta\rangle \in \Omega \times \Omega \cdot \alpha \supset \beta$. By 3.5 and 3.8,

$$T(\Omega_1) \cong \sum\langle a, b\rangle \in K \times K.[T(a), T(b)],$$

so a typical term of type Ω_1 is just a triple $\langle a, b, f\colon a \to b\rangle$. It is now easy to define id, dom, cod, to show that Ω_1 gives an internal category structure to Ω, and to show that $\mathbf{G} \cong \mathrm{Hom}_{\mathbf{K}}(-, \Omega)$ (on morphisms as well as on objects).

§4. Categorical semantics.

4.1. We now define what is meant by an interpretation of a PL theory $\mathfrak{T}$ in a PL category $(\mathbf{G}, \mathbf{S})$. For simplicity we assume exactly one free indeterminate or variable in any operator or term; since we have products, this causes no loss of generality. We shall say an operator σ has arity $A \to B$ (denoted $\sigma\colon A \to B$) if σ is of order B and its free indeterminate has order A. (Of course, then $\sigma\colon 1 \to B$ indicates σ is a closed operator.) Similarly $a\colon \sigma \to \tau$ denotes the arity of a term of type τ with free variable of type σ. By our stratification condition (1.2(3)), a, σ, and τ all have the same free indeterminate, of order A, say; in this case, we say $a\colon \sigma \to \tau$ is over A. (Note that a closed term of a closed type is $a\colon \top \to \sigma$ over 1.)

DEFINITION. An *interpretation* $I\colon \mathfrak{T} \to (\mathbf{G}, \mathbf{S})$ of a PL theory $\mathfrak{T}$ in a PL category $(\mathbf{G}, \mathbf{S})$ is a function which assigns to each order A of $\mathfrak{T}$ an object $I(A)$ of $\mathbf{S}$, to each operator $\sigma\colon A \to B$ of $\mathfrak{T}$ a morphism $I(\sigma)\colon I(A) \to I(B)$ of $\mathbf{S}$, and to each term $a\colon \sigma \to \tau$ over A of $\mathfrak{T}$ a morphism $I(a)\colon I(\sigma) \to I(\tau)$ of $\mathbf{G}(A)$. Note that $I(\sigma)$ and $I(\tau)$ are objects of $\mathbf{G}(A)$, provided we require $I(\Omega) = \Omega$ (which we do).

DEFINITION 4.2. An interpretation $I\colon \mathfrak{T} \to (\mathbf{G}, \mathbf{S})$ is a *model* of $\mathfrak{T}$ if I preserves all the (nonlogical) axioms of $\mathfrak{T}$, and satisfies the following conditions. (We follow the notation of Definition 1.1.)

(4.2.1. *Orders*) $I(1) = 1$; $I(\Omega) = \Omega$; $I(A \times B) = I(A) \times I(B)$; $I(\Omega^A) = \Omega^{I(A)}$.

(4.2.2. *Operators*) If α is an indeterminate of order A, $I(\alpha) = \mathrm{id}\colon I(A) \to I(A)$; $I(*) = \mathrm{id}\colon 1 \to 1$; $I(\top) = \top\colon 1 \to \Omega$ (as in 2.2(2)). For $\sigma, \tau\colon A \to \Omega$, $I(\sigma \wedge \tau) = I(\sigma) \wedge I(\tau)$ (as in 2.2(2)), and similarly $I(\sigma \supset \tau) = I(\sigma) \supset I(\tau)$. (Note these are product and exponentiation in $\mathbf{G}(I(A))$.) For $\sigma\colon B \times A \to \Omega$, $\alpha \in A$, note that $\sum\alpha \in A \cdot \sigma$, $\prod\alpha \in A \cdot \sigma\colon B \to \Omega$, and that $I(\sigma)\colon I(B) \times I(A) \to \Omega$. Let $\tilde{\sigma}$ be the corresponding morphism $I(B) \to \Omega^{I(A)}$. Then $I(\sum\alpha \in A \cdot \sigma) = \sum_{I(A)} \circ \tilde{\sigma}$, $I(\prod\alpha \in A \cdot \sigma) = \prod_{I(A)} \circ \tilde{\sigma}\colon I(B) \to \Omega$. (Of course, these are just $\sum_{I(A)}(I(\sigma))$ and $\prod_{I(A)}(I(\sigma))$ respectively.)

($\times I$) $I(\langle\sigma,\tau\rangle) = \langle I(\sigma), I(\tau)\rangle$.

($\times E$) $I(\pi_1\sigma) = \pi_1 I(\sigma)$, $I(\pi_2\sigma) = \pi_2 I(\sigma)$.

(PI) $I([\alpha \in A\colon \sigma]) = \tilde{\sigma}\colon I(B) \to \Omega^{I(A)}$, as above.

(PE) $I(\sigma(\tau)) = \mathrm{ev} \circ \langle I(\sigma), I(\tau)\rangle$, where $\mathrm{ev}\colon \Omega^{I(A)} \times I(A) \to \Omega$ is the standard "evaluation" map.

Substitution is given by the functor $\mathbf{G}$: for example, if $\tau\colon C \to A$ and $\sigma\colon B \times A \to \Omega$, let $\bar{\tau} = I(B) \times I(\tau)$. Then if $\alpha \in A$, $I(\sigma[\tau/\alpha]) = \bar{\tau}^*(I(\sigma))\colon I(B) \times I(C) \to \Omega$. (Of course this is just given by composition.) Similarly, if $a\colon \nu \to \sigma$ over $B \times A$, then $I(a[\tau/\alpha]) = \bar{\tau}^*(I(a))\colon I(\nu[\tau]) \to I(\sigma[\tau])$ in $\mathbf{G}(I(B) \times I(C))$.

(4.2.3. *Terms*) If x is a variable of type σ over A, $I(x) = \mathrm{id}\colon I(\sigma) \to I(\sigma)$ in $\mathbf{G}(I(A))$; $I(*) = \mathrm{id}\colon \top \to \top$ in $\mathbf{G}(1)$.

($\supset I$) For $a\colon \nu \wedge \sigma \to \tau$ over A, $I(a)\colon I(\nu) \wedge I(\sigma) \to I(\tau)$ in $\mathbf{G}(A)$, and $I(\lambda x \in \sigma \cdot a)\colon I(\nu) \to I(\sigma) \supset I(\tau) = I(\sigma \supset \tau)$ is the corresponding morphism of $\mathbf{G}(A)$, given by cartesian closedness.

($\supset E$) For $a\colon \nu \to \sigma \supset \tau$, $b\colon \nu \to \sigma$ over A, $I(a(b)) = \mathrm{ev} \circ \langle I(a), I(b)\rangle$ in $\mathbf{G}(A)$.

($\wedge I$) $I(\langle a,b\rangle) = \langle I(a), I(b)\rangle$.

($\wedge E$) $I(\pi_1 a) = \pi_1 I(a)$, $I(\pi_2 a) = \pi_2 I(a)$.

($\sum I$) For $\sigma\colon B \times A \to \Omega$, note that $\kappa_{I(A)}\sum_{I(A)}(I(\sigma))$ is the interpretation of $\sum\alpha \in A \cdot \sigma$ with a dummy indeterminate $\alpha \in A$. $I(I_{\Sigma\alpha\cdot\sigma,\alpha})\colon I(\sigma) \to \kappa_{I(A)}\sum_{I(A)}(I(\sigma))$ is the unit of the adjunction $\sum \dashv \kappa$. If $\tau\colon C \to A$, then

$$I(I_{\Sigma\alpha\cdot\sigma,\tau}) = \bar{\tau}^*(I(I_{\Sigma\alpha\cdot\sigma,\alpha}))\colon I(\sigma[\tau]) \to \pi_1^* I\left(\sum\alpha \in A \cdot \sigma\right),$$

where $\pi_1\colon I(B) \times I(C) \to I(B)$, so π_1^* adds a dummy indeterminate of order C, in effect.

($\sum E$) For $\nu, \rho\colon B \to \Omega$, $\sigma\colon B \times A \to \Omega$, and $a\colon \nu \to \sigma \supset \rho$ over $B \times A$ (so that now ν and ρ have a dummy indeterminate $\alpha \in A$), note that

$$I(a)\colon \kappa_{I(A)}(I(\nu)) \to I(\sigma) \supset \kappa_{I(A)}(I(\rho))$$

in $\mathbf{G}(I(B) \times I(A))$. Then

$$I(V\alpha \in A \cdot a)\colon I(\nu) \to I\left(\sum\alpha \in A \cdot \sigma\right) \supset I(\rho)$$

in $\mathbf{G}(I(B))$ must be given by these correspondences:

$$\begin{array}{ll}
\kappa_{I(A)}(I(\nu)) \to I(\sigma) \supset \kappa_{I(A)}(I(\rho)) & \text{in } \mathbf{G}(I(B \times A)), \\
\hline
\kappa_{I(A)}(I(\nu)) \wedge I(\sigma) \to \kappa_{I(A)}(I(\rho)) & \text{in } \mathbf{G}(I(B \times A)), \\
\hline
\sum_{I(A)} (\kappa_{I(A)}(I(\nu)) \wedge I(\sigma)) \to I(\rho) & \text{in } \mathbf{G}(I(B)), \\
\hline
I(\nu) \wedge \sum_{I(A)} (I(\sigma)) \to I(\rho) & \text{in } \mathbf{G}(I(B)), \\
\hline
I(\nu) \to \sum_{I(A)} (I(\sigma)) \supset I(\rho) & \text{in } \mathbf{G}(I(B)).
\end{array}$$

(We have used here the result, Frobenius reciprocity,

$$\sum_A(\kappa_A(v) \wedge \sigma) \xrightarrow{\simeq} v \wedge \sum_A(\sigma),$$

which is true for any PL category. This is equivalent to κ_A preserving exponentiation, and since $\kappa_A = \pi_1^*$, this was guaranteed by Definition 2.1(ii)(b).)

($\prod I$) For $v: B \to \Omega$, $\sigma: B \times A \to \Omega$, $a: v \to \sigma$ over $B \times A$, then v has a dummy $\alpha \in A$ and $I(a): \kappa_{I(A)}(I(v)) \to I(\sigma)$. Then $I(\Lambda\alpha \in A \cdot a): I(v) \to \prod_{I(A)}(I(\sigma))$ must be the corresponding morphism under $\kappa \dashv \prod$.

($\prod E$) For $v: B \to \Omega$, $\sigma: B \times A \to \Omega$, $a: v \to \prod\alpha \in A \cdot \sigma$ over B, $I(a): I(v) \to \prod_{I(A)}(I(\sigma))$; $I(a\{\alpha\}): \kappa_{I(A)}(I(v)) \to I(\sigma)$ in $\mathbf{G}(I(B \times A))$ is the corresponding morphism under $\kappa \dashv \prod$. For $\tau: C \to A$, $I(a\{\tau\}) = \bar{\tau}^*(I(a\{\alpha\})): \pi_1^* I(v) \to I(\sigma[\tau/\alpha])$ in $\mathbf{G}(I(B \times C))$, where $\pi_1: I(B) \times I(C) \to I(B)$.

Substitution is given by composition: if $a: v \to \sigma$, $b: \rho \wedge \sigma \to \tau$ over A, then

$$I(b[a/x_\sigma]) = I(b) \circ (I(\rho) \wedge I(a)): I(\rho) \wedge I(v) \to I(\rho) \wedge I(\sigma) \to I(\tau)$$

in $\mathbf{G}(A)$.

4.3. REMARK. The essence of 4.1 and 4.2 is that an interpretation I is given by what it does to the constants of $\mathfrak{T}$. For the rest, I is extended canonically to all orders, operators, and terms. I is a model if I respects the axioms of $\mathfrak{T}$. For the rest, the canonical definition of I automatically respects the logical equalities of §1.1.5. We summarise this in the next proposition.

PROPOSITION 4.4 (Soundness). *If $I: \mathfrak{T} \to (\mathbf{G}, \mathbf{S})$ is a model, then all the equality rules of* §1.1.5 *are valid under I.*

The proof of soundness is by now a standard procedure in categorical logic. For example, similar results may be found in Seely [1983] for first order logic, Seely [1984] for Martin-Löf type theory, and Lambek and Scott [1986] for λ-calculus and type theory.

4.5. In fact, the notions of 4.2 can be used to define a PL category $(\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T}))$ from a PL theory $\mathfrak{T}$: the objects of $\mathbf{S}(\mathfrak{T})$ are orders of $\mathfrak{T}$; morphisms $\sigma: A \to B$ of $\mathbf{S}(\mathfrak{T})$ are operators $\sigma: A \to B$ of $\mathfrak{T}$, modulo the equivalence relation given by the equations of $\mathfrak{T}$; morphisms of $\mathbf{G}(\mathfrak{T})(A)$ are terms $a: \sigma \to \tau$ over A of $\mathfrak{T}$, modulo the equivalence relation given by equations of $\mathfrak{T}$; σ^* is defined by substitution.

PROPOSITION 4.6. $(\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T}))$ *is a PL category.*

PROOF. Again, the details are standard and straightforward. For example, we show that for any A in $\mathbf{S}(\mathfrak{T})$, $\kappa_A: \mathbf{G}(\mathfrak{T}) \to \mathbf{G}(\mathfrak{T})^A$ has a left adjoint $\prod_A$. Given B in $\mathbf{S}(\mathfrak{T})$, σ in $\mathbf{G}(\mathfrak{T})^A(B) = \mathbf{G}(\mathfrak{T})(B \times A)$, we know that $\sigma: B \times A \to \Omega$ in $\mathfrak{T}$. Also, $\prod_A(\sigma) = \prod\alpha \in A \cdot \sigma: B \to \Omega$, where α is a new indeterminate of order A. Given $a: \sigma \to \tau$ in $\mathbf{G}(\mathfrak{T})^A(B)$, i.e. $a: \sigma \to \tau$ over $B \times A$, $\prod_A(a) = \Lambda\alpha \in A \cdot a[x\{\alpha\}]: \prod\alpha \in A \cdot \sigma \to \prod\alpha \in A \cdot \tau$ over B, where x is a free variable of type $\prod\alpha \in A \cdot \sigma$ and $a[x\{\alpha\}]$ is a with $x\{\alpha\}$ replacing the free variable of type σ in a. The bijection

$$\frac{\kappa_A(\rho) \to \sigma \quad \text{in } \mathbf{G}(\mathfrak{T})^A(B),}{\rho \to \prod_A(\sigma) \quad \text{in } \mathbf{G}(\mathfrak{T})(B)}$$

is given thus: For $a: \kappa_A(\rho) \to \sigma$ over $B \times A$, i.e. $a: \rho \to \sigma$ where $\alpha \in A$ is only a dummy in ρ, let $\hat{a}: \rho \to \prod_A(\sigma)$ be $\Lambda\, \alpha \in A \cdot a: \rho \to \prod \alpha \in A \cdot \sigma$ over B. For $b: \rho \to \prod_A(\sigma)$ over B, let $\check{b}: \kappa_A(\rho) \to \sigma$ be $b\{\alpha\}$. That these are inverse is immediate from ($\prod$ red) and ($\prod$ exp).

PROPOSITION 4.7. *There is a model* $I_0: \mathfrak{T} \to (\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T}))$. *Moreover, given any model* $I: \mathfrak{T} \to (\mathbf{G}, \mathbf{S})$, *I factors through* I_0: *there is a unique functor* $F_I: (\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T})) \to (\mathbf{G}, \mathbf{S})$ *of PL categories so that* $F_I \circ I_0 = I$ (*with the evident definition of composition*).

PROOF. Briefly, I_0 sends each order, operator, and term to itself (or rather the equivalence class containing it, in the case of operators and terms). F_I is essentially defined as I; this is well defined on equivalence classes since I is a model.

DEFINITION 4.8. (1) $\mathrm{Mod}(\mathfrak{T}; \mathbf{G}, \mathbf{S})$ is the set of all models $\mathfrak{T} \to (\mathbf{G}, \mathbf{S})$, for any PL theory $\mathfrak{T}$ and PL category $(\mathbf{G}, \mathbf{S})$.

(2) For PL theories $\mathfrak{T}'$, $\mathfrak{T}$, an interpretation $\mathfrak{T}' \to \mathfrak{T}$ is a model $\mathfrak{T}' \to (\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T}))$.

(3) **PLC** is the category of PL theories and interpretations.

REMARKS. (i) It is easy to check that (2) is equivalent to the usual syntactic notion of interpretation of one theory in another.

(ii) From 4.7 we have, for any PL theories $\mathfrak{T}$, $\mathfrak{T}'$ and PL category $(\mathbf{G}, \mathbf{S})$,

$$\mathrm{Mod}(\mathfrak{T}; \mathbf{G}, \mathbf{S}) \cong \mathbf{PL\ Cat}((\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T})), (\mathbf{G}, \mathbf{S})).$$
$$\mathbf{PLC}(\mathfrak{T}', \mathfrak{T}) \cong \mathbf{PL\ Cat}((\mathbf{G}(\mathfrak{T}'), \mathbf{S}(\mathfrak{T}')), (\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T})).$$

Moreover $\mathrm{Mod}(-;-)$ is functorial, contravariant in the first position, covariant in the second, and these isomorphisms are natural in each variable.

DEFINITION 4.9. Two PL theories $\mathfrak{T}'$ and $\mathfrak{T}$ are *equivalent* if the functors $\mathrm{Mod}(\mathfrak{T}', -)$ and $\mathrm{Mod}(\mathfrak{T}, -)$ are naturally isomorphic. Equivalently, $(\mathbf{G}(\mathfrak{T}'), \mathbf{S}(\mathfrak{T}')) \cong (\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T}))$ in **PL Cat**.

§5. Equivalences.

5.1. Given a PL category $(\mathbf{G}, \mathbf{S})$, we define a PL theory $\mathfrak{T}(\mathbf{G}, \mathbf{S})$ as follows: The orders of $\mathfrak{T}(\mathbf{G}, \mathbf{S})$ are objects of $\mathbf{S}$, and the operators $\sigma: A \to B$ of $\mathfrak{T}(\mathbf{G}, \mathbf{S})$ are morphisms $\sigma: A \to B$ of $\mathbf{S}$. Ω is Ω, so types are morphisms $\sigma: A \to \Omega$. Terms $a: \sigma \to \tau$ over A are morphisms in $\mathbf{G}(A)$. Equations are given by equality in $(\mathbf{G}, \mathbf{S})$.

In §1 we defined a PL theory as given by sets of orders, operators, and terms, sets which had to be closed under certain operations and which had to satisfy certain equations. It is of course more usual to treat these operations as rules for freely generating the sets of orders, operators, and terms, and then impose the equations on the freely generated sets. I have not done that, so that now it is not necessary to worry about the "duplication" that would arise if we were to treat the orders, operators, and terms of $\mathfrak{T}(\mathbf{G}, \mathbf{S})$ above as constant symbols, and then used the formation rules to generate sets of orders, operators, and terms freely from them. With our approach it is only necessary to add indeterminates and variables (which are essentially unnecessary anyway!); the rest of the orders, operators and terms already are present as appropriate objects and morphisms of $(\mathbf{G}, \mathbf{S})$, as indicated in §4.2.

PROPOSITION 5.2. (i) $\mathfrak{T}(\mathbf{G}, \mathbf{S})$ *is a PL theory.*

(ii) *Any functor* $F: (\mathbf{G}, \mathbf{S}) \to (\mathbf{G}', \mathbf{S}')$ *of PL categories induces a model* $\mathfrak{T}(\mathbf{G}, \mathbf{S}) \to (\mathbf{G}', \mathbf{S}')$; *in fact* $\mathrm{Mod}(\mathfrak{T}(\mathbf{G}, \mathbf{S}); \mathbf{G}', \mathbf{S}') \cong \mathbf{PL\ Cat}((\mathbf{G}, \mathbf{S}), (\mathbf{G}', \mathbf{S}'))$ (*naturally in each position*).

PROPOSITION 5.3. (i) *For any PL category* $(\mathbf{G}, \mathbf{S})$, *there is an equivalence*

$$\varepsilon: (\mathbf{G}(\mathfrak{T}(\mathbf{G},\mathbf{S})), \mathbf{S}(\mathfrak{T}(\mathbf{G},\mathbf{S}))) \Rightarrow (\mathbf{G},\mathbf{S}) \quad \textit{in } \mathbf{PL\ Cat}.$$

(2) *For any PL theory* $\mathfrak{T}$, *there is an equivalence*

$$\eta: \mathfrak{T} \to \mathfrak{T}(\mathbf{G}(\mathfrak{T}), \mathbf{S}(\mathfrak{T})) \quad \textit{in } \mathbf{PLC}.$$

(3) *There is an equivalence of categories* **PL Cat** $\simeq$ **PLC**.

REMARK We have defined the structure in PLC so that η is an equivalence of theories, but a more usual approach would give only a conservative extension. Then in (3) we would only have an adjunction between **PL Cat** and **PLC**. This would be sufficient, however; see Lambek and Scott [1986], where such an approach is used for type theory.

§6. Subtypes.

6.1. In view of the equivalence 5.3, it is sufficient to work in terms of PL categories. However, we shall continue to refer to PL theories as well, in the interests of clarity; our definitions and constructions will be informal in **PLC**, and more formal in **PL Cat**.

DEFINITION 6.2. A *PLS theory* $\mathfrak{T}$ consists of four collections of objects: orders, operators (which include types), subtypes, and terms. The orders, operators (and types), and terms are essentially those of a PL theory, except that terms now may have subtypes in their arity. A subtype X has an arity which, in addition to the orders of free indeterminates appearing in X, includes a type σ (over the same orders). Furthermore, X may have terms occurring in it, as we shall see below. We say such an X is a subtype of σ (over the appropriate orders), or σ is the supertype of X, and write $X: \subset\sigma$ or $\sigma: \supset X$. X may contain a free variable of type σ; if we wish to indicate the occurrences of a term a in X, we shall use the notation $X[a]$, as before. If a term a is of subtype $X: \subset\sigma$ and has a free variable of subtype $Y: \subset\tau$, we write $a: Y \subset \tau \to X \subset \sigma$. Every type σ is a subtype $\sigma: \subset\sigma$; we write this as just σ. The key new formation rule for subtypes is:

$(=)$ If a, b: $Y \subset \tau \to X \subset \sigma$ over A are terms with the same free variable, then $E(a,b)$ is a subtype of τ over A.

We have built the stratification into $(=)$: note then, for example, if x_1, x_2 are two different free variables of type σ, then $E(x_1,x_2)$ must be understood to be an abbreviation for $E(\pi_1\langle x_1,x_2\rangle, \pi_2\langle x_1,x_2\rangle)$, and is then a subtype of $\sigma \wedge \sigma$.

Subtypes are closed under the same rules as types (1.1.2), the supertypes being given by the corresponding rules. For example, $\prod\alpha \in A \cdot X: \subset \prod\alpha \in A \cdot \sigma$. Of course, $\mathfrak{T}$ may also contain constant subtype symbols as well. Notice there are two kinds of substitution for subtypes: of operators for indeterminates, and of terms for variables.

Terms in $\mathfrak{T}$ have arities including subtypes. (We use products to account for terms containing several free variables.) There are two new term formation rules, corresponding to the creation of the $E(a,b)$ subtypes:

$(=I)$ If a: $Y \subset \tau \to X \subset \sigma$ over A, then $r(a) \in E(a,a)$: $\subset\tau$ over A.

$(=E)$ If a, b: $Y \subset \tau \to X \subset \sigma$ over A, $c \in E(a,b)$, $Z[x_X, y_X]$: $\subset\sigma \wedge \sigma$, and $d \in Z[a,a]$, then $\mathbf{s}(a,b,c;d) \in Z[a,b]$ We write $\mathbf{s}(d)$ if a, b, c are clear from the context.

Furthermore, there may be constants given by $\mathfrak{T}$, and terms for subtypes must be closed under the usual term formation rules, as in 1.1.4.

Finally there are three new equality rules for the terms introduced above, as well as the rules in 1.1.7 for subtypes, and axioms of $\mathfrak{T}$:

(= red) $\mathbf{s}(a, a, r(a); d) = d$.

(= exp) If $f[a, b, c] \in Z[a, b]$, then $f = \mathbf{s}(a, b, c; f[a, a, r(a)])$.

(= rule) If $c \in E(a, b)$, then $a = b$ and $c = r(a)$.

6.3. In fact, one may construct a PLS theory $\tilde{\mathfrak{T}}$ as a definitional extension of a PL theory $\mathfrak{T}$ simply: the orders, operators, and terms are those of $\mathfrak{T}$, and a subtype is a pair (X, σ), where σ is a type, over A say, and X is a function which to each closed operator ρ of order A assigns a set $X(\rho)$ of closed terms of type $\sigma[\rho/\alpha_A]$. A term of subtype (X, σ) with free variable of subtype (Y, τ) is an equivalence class of terms $a: \tau \to \sigma$ such that, for each closed operator $\rho \in A$, for each closed term $b \in \tau[\rho/\alpha_A]$ in $Y(\rho)$, $a[\rho/\alpha_A; b/x_\tau]$ is in $X(\rho)$. (Note that variables are always such terms.) Two such terms a_1 and a_2 are equivalent if $a_1[\rho; b] = a_2[\rho; b]$ for all closed $\rho \in A$ and $b \in Y(\rho)$. A type σ is made a subtype (X, σ) by taking $X(\rho)$ to be all closed terms of type $\sigma[\rho]$. The equivalence relation above imposes equations on $\tilde{\mathfrak{T}}$, so that although $\mathfrak{T}$ and $\tilde{\mathfrak{T}}$ have the same types, $\tilde{\mathfrak{T}}$ has more equations between types, in general. Finally, notice that for terms a_1, a_2 as above, $E(a_1, a_2)$ is the subtype consisting of all $b \in Y(\rho)$ so that $a_1[\rho; b] = a_2[\rho; b]$.

DEFINITION 6.4. A *PLS category* $(\mathbf{G}, \mathbf{S})$ consists of

(i) a category $\mathbf{S}$ with finite products, a distinguished object Ω, and exponentiation of the form Ω^A for A in $\mathbf{S}$, and

(ii) an indexed category $\mathbf{G}$ over $\mathbf{S}$ satisfying:

(a) for each object A in $\mathbf{S}$, $\mathbf{G}(A)$ is "subrepresentable on objects by Ω": this means there is a full subcategory $\mathbf{Rep}_\mathbf{G}(A) \subseteq \mathbf{G}(A)$ so that $\mathrm{Obj}(\mathbf{Rep}_\mathbf{G}(A)) \cong \mathrm{Hom}_\mathbf{S}(A, \Omega)$, every object of $\mathbf{G}(A)$ is a subobject of a representable object (i.e. one in $\mathbf{Rep}_\mathbf{G}(A)$), and every morphism of $\mathbf{G}(A)$ lifts to a morphism in $\mathbf{Rep}_\mathbf{G}(A)$ between the corresponding representable objects; furthermore, for each $f: A \to B$ of $\mathbf{S}$, f^* acts as $\mathrm{Hom}(f, \Omega)$ on $\mathrm{Obj}(\mathbf{Rep}_\mathbf{G}(A))$;

(b) for each object A in $\mathbf{S}$, $\mathbf{G}(A)$ is cartesian closed, has finite limits, and for each $f: A \to B$ in $\mathbf{S}$, $\mathbf{f}^*$ preserves this structure;

(c) $\mathbf{G}$ is "weakly complete and cocomplete", as in 2.1(c);

(d) as a subindexed category, $(\mathbf{Rep}_\mathbf{G}, \mathbf{S})$ is a PL category.

6.5. REMARK. In 6.4(ii)(a) we do not require morphisms in $\mathbf{G}(A)$ to lift uniquely to $\mathbf{Rep}_\mathbf{G}(A)$; that would amount to requiring $\mathbf{Rep}_\mathbf{G}$ to be a reflective sub(indexed) category of $\mathbf{G}$. Thinking of types as parametrised sets and subtypes as uniformly defined subsets, the idea is that maps of subtypes should be restrictions of maps of types, but several such maps could restrict to the same map of subtypes.

6.6. The definition of a functor of PLS categories is clear; **PLS Cat** denotes the category of such categories and functors. As mentioned in 2.4, adding equality to PLC has added equalisers (and so all finite limits) to the fibres of the PL categories. The price of adding these subobjects is that we must replace "representability" with "subrepresentability". It would be nice to have a PLS category which in fact was a PL category: $\mathbf{Rep}_\mathbf{G} = \mathbf{G}$.

In fact, many such PL categories (with equalisers) can be constructed via the "partial equivalence relation" construction of models of PLC from models of untyped lambda calculus. The most interesting variant of this is to do the construction *internally*, creating an object in a topos which is an (internal) locally cartesian closed weakly complete category; this was originally observed by E. Moggi and J. M. E. Hyland.

6.7. As with PLS theories, we can construct a PLS category $(\tilde{\mathbf{G}}, \mathbf{S})$ from a PL category $(\mathbf{G}, \mathbf{S})$. The base category $\mathbf{S}$ remains the same. For an object A of $\mathbf{S}$, $\tilde{\mathbf{G}}(A)$ is the following category: An object of $\tilde{\mathbf{G}}(A)$ is a pair (X, σ) where σ is an object of $\mathbf{G}(A)$ and X is a function which, for each morphism $\rho: 1 \to A$ in $\mathbf{S}$, assigns a set $X(\rho)$ of morphisms $\top \to \rho^*\sigma$ in $\mathbf{G}(1)$. ($\top$ is the terminal object of $\mathbf{G}(1)$, and $\rho^*\sigma = \mathbf{G}(\rho)(\sigma)$, which in fact is $1 \xrightarrow{\rho} A \xrightarrow{\sigma} \Omega$.) If we use Γ for the "global sections functor" Hom$(1, -)$ in the appropriate categories, and $\mathscr{P}$ for "powerset", then $X \, \varepsilon \prod \rho \, \varepsilon \, \Gamma(A). \mathscr{P}(\Gamma(\rho^*\sigma))$. A morphism $(Y, \tau) \to (X, \sigma)$ in $\tilde{\mathbf{G}}(A)$ is an equivalence class of morphisms $a: \tau \to \sigma$ of $\mathbf{G}(A)$ such that, for each $\rho \in \Gamma(A)$, $\Gamma(\rho^*a): \Gamma(\rho^*\tau) \to \Gamma(\rho^*\sigma)$ restricts to a map $Y(\rho) \to X(\rho)$. Two such morphisms a_1, a_2 are equivalent if $\Gamma(\rho^*a_1) \restriction Y(\rho) = \Gamma(\rho^*a_2) \restriction Y(\rho)$ for all $\rho \, \varepsilon \, \Gamma(A)$. Note that identity morphisms are morphisms of $\tilde{\mathbf{G}}(A)$, that any object σ of $\mathbf{G}(A)$ induces an object (Γ_σ, σ) of $\tilde{\mathbf{G}}(A)$ where $\Gamma_\sigma(\rho) = \Gamma(\rho^*\sigma)$, and that any morphism $a: \tau \to \sigma$ of $\mathbf{G}(A)$ induces a morphism (represented by itself) $a: (\Gamma_\tau, \tau) \to (\Gamma_\sigma, \sigma)$. $\mathbf{REP}_{\tilde{\mathbf{G}}}(A)$ is the full subcategory whose objects are all (Γ_σ, σ), σ in $\mathbf{G}(A)$; it is a quotient of $\mathbf{G}(A)$ with the same objects as $\mathbf{G}(A)$ (morphisms may be identified).

If $f: A \to B$ in $\mathbf{S}$, $\tilde{\mathbf{G}}(f)$ is defined by composition: $\tilde{\mathbf{G}}(f)(X, \sigma) = (f^*X, f^*\sigma)$, where $f^*X(\rho) = X(f \circ \rho)$, and $f^*\sigma = \mathbf{G}(f)(\sigma) = \sigma \circ f$; $\tilde{\mathbf{G}}(f)(a) = f^*a$ (this is to be interpreted modulo the equivalence relation—it is well defined because of the functorality of $\mathbf{G}$).

PROPOSITION 6.8. *If* $(\mathbf{G}, \mathbf{S})$ *is a PL category, then* $(\tilde{\mathbf{G}}, \mathbf{S})$ *is a PLS category.*

PROOF. For A in $\mathbf{S}$, $\tilde{\mathbf{G}}(A)$ is cartesian closed:

The terminal object is $(\Gamma_\top, \top)$.

$(X, \sigma) \times (Y, \tau) = (X \times Y, \sigma \wedge \tau)$, where $X \times Y(\rho) = X(\rho) \times Y(\rho)$.

$(Y, \tau)^{(X, \sigma)} = (Y^X, \sigma \supset \tau)$, where $Y^X(\rho) = \{f: X(\rho) \to Y(\rho): f$ is the restriction of a map $\Gamma(\rho^*\sigma) \to \Gamma(\rho^*\tau)$ induced by a morphism $\rho^*\sigma \to \rho^*\tau$ of $\mathbf{G}(1)\}$.

$\tilde{\mathbf{G}}(A)$ has finite limits: Given morphisms a_1, $a_2: (Y, \tau) \to (X, \sigma)$, the equaliser $E(a_1, a_2)$ is $(E(a_1, a_2), \tau)$, where $E(a_1, a_2)(\rho) = \{x \, \varepsilon \, Y(\rho): \Gamma(\rho^*a_1)(x) = \Gamma(\rho^*a_2)(x)\}$.

$\tilde{\mathbf{G}}$ is weakly complete: if (X, σ) is an object of $\tilde{\mathbf{G}}^C(A) = \tilde{\mathbf{G}}(A \times C)$, then $\sigma: A \times C \to \Omega$, and for $\langle \rho_1, \rho_2 \rangle \, \varepsilon \, \Gamma(A \times C)$, $X(\rho_1, \rho_2) \subseteq \Gamma(\langle \rho_1, \rho_2 \rangle^*\sigma)$. Then

$$\prod_C (X, \sigma) = \left(\prod \rho_2 \, \varepsilon \, \Gamma(C) \cdot X(\cdot, \rho_2), \prod_C (\sigma) \right).$$

This means the following: If $\rho_1 \, \varepsilon \, \Gamma(A)$, then $\rho_1^* \prod_C(\sigma) = \prod_C(\rho_1^*\sigma)$, so its global sections (in $\mathbf{G}(1)$)

$$\frac{\top \to \rho_1^* \prod_C (\sigma) = \prod_C (\rho_1^* \sigma)}{\kappa_C(\top) = \top \to \rho_1^* \sigma}$$

correspond to the global sections of $\rho_1^*\sigma$ in $\mathbf{G}^C(1) = \mathbf{G}(C)$. (Here ρ_1^* means $\mathbf{G}^C(\rho_1)$, so $\rho_1^*\sigma = \mathbf{G}^C(\rho_1)(\sigma) = C \xrightarrow{\langle\rho_1, C\rangle} A \times C \xrightarrow{\sigma} \Omega$.) Then $\prod\rho_2 \varepsilon \Gamma(C) \cdot X(\rho_1, \rho_2)$ is the set of such global sections z of $\rho_1^*\sigma$ with the property that for any $\rho_2 \varepsilon \Gamma(C)$, $\rho_2^* z \varepsilon X(\rho_1, \rho_2)$. (Here ρ_2^* means $\mathbf{G}(\rho_2)$; note that $\rho_2^* z$ is a global section in $\mathbf{G}(\rho_2)\mathbf{G}(\rho_1 \times C)(\sigma) = \mathbf{G}(\langle\rho_1, \rho_2\rangle)(\sigma)$.)

The weak cocompleteness of $\tilde{\mathbf{G}}$ is somewhat more involved. The cartesian closedness and weak completeness of $\tilde{\mathbf{G}}$ used essentially the ideas of §3, where Γ generalises the notion of fixed point, T (see 3.5 and 3.8). But there is a problem with $\sum$, in that the "easy" description of $\Gamma(\sum_C(\sigma))$ in terms of $\Gamma(C)$ and $\Gamma(\langle\rho_1, \rho_2\rangle^*\sigma)$ fails to be true in general. More precisely, for $\rho_1 \varepsilon \Gamma(A)$, we do *not* have a bijection

$$\frac{T^z \to \rho_1^* \sum_C(\sigma) \qquad \text{in } \mathbf{G}(1)}{1 \xrightarrow[\rho_2]{} C \text{ in } \mathbf{S};\ \top \xrightarrow[x]{} \langle\rho_1, \rho_2\rangle^*\sigma \qquad \text{in } \mathbf{G}(1)}. \tag{1}$$

However, we do have a construction $\langle\rho_2, x\rangle \mapsto z$, described below. Then if (X, σ) is in $\tilde{\mathbf{G}}^C(A)$,

$$\sum_C(X, \sigma) = \left(\sum\rho_2 \varepsilon \Gamma(C) \cdot X(\cdot, \rho_2), \sum_C(\sigma)\right),$$

where $\sum\rho_2 \varepsilon \Gamma(C) \cdot X(\rho_1, \rho_2)$ is the image under this construction of all $\langle\rho_2, x\rangle$, for $\rho_2 \varepsilon \Gamma(C)$, $x \varepsilon X(\rho_1, \rho_2)$.

Syntactically, this amounts to the following: if $\sum_C(X, \sigma) = (\sum_C X, \sum_C \sigma)$, then $\sum_C X$ is a function so that if ρ_1 is a closed operator of order A, $(\sum_C X)(\rho_1)$ is the set of all closed terms of type $\sum\gamma \in C \cdot \sigma[\rho_1, \gamma]$ of the form $I_{\rho_2}(x)$, where $x \varepsilon X(\rho_1, \rho_2)$ and ρ_2 is a closed operator of order C. The lack of a bijection above reflects the fact that theories of the generality we allow may have closed terms of type $\sum\gamma \in C \cdot \sigma[\rho_1, \gamma]$ other than those of the form $I_{\rho_2}(x)$.

We sketch the details: given $\langle\rho_2, x\rangle$ as above, z is

$$\langle\rho_1, \rho_2\rangle^*\eta_\sigma \circ x\colon \top \to \langle\rho_1, \rho_2\rangle^*\sigma \to \langle\rho_1, \rho_2\rangle^*\kappa_C \sum_C(\sigma) = \rho_1^* \sum_C(\sigma),$$

since $\langle\rho_1, \rho_2\rangle^*\kappa_C = \rho_1^*$. Given $a\colon(\sum_C X, \sum_C\sigma) \to (Y, \tau)$ in $\tilde{\mathbf{G}}(A)$, where $\sum_C X = \sum\rho_2 \varepsilon \Gamma(C).X(\cdot, \rho_2)$, let $b = \kappa_C(a) \cdot \eta_\sigma$ be the usual morphism $\sigma \to \kappa_C\tau$ in $\mathbf{G}^C(A)$ induced by a. To see b is a morphism $(X, \sigma) \to (\kappa_C Y, \kappa_C\tau)$ of $\tilde{\mathbf{G}}^C(A)$, one must check that for any $\rho_1 \varepsilon \Gamma(A)$, $\rho_2 \varepsilon \Gamma(C)$, $x \varepsilon X(\rho_1, \rho_2)$, we have $(\langle\rho_1, \rho_2\rangle^*b) \circ x$ is in $Y(\rho_1)$. But

$$\langle\rho_1, \rho_2\rangle^*b = \langle\rho_1, \rho_2\rangle^*\kappa_C(a) \circ \langle\rho_1, \rho_2\rangle^*\eta_\sigma = \rho_1^*a \circ \langle\rho_1, \rho_2\rangle^*\eta_\sigma,$$

so this is just the condition that a is a morphism of $\tilde{\mathbf{G}}(A)$, viz. that for any $z \varepsilon \sum_C X(\rho_1)$, $\rho_1^*a \circ z$ is in $Y(\rho_1)$, since $z = \langle\rho_1, \rho_2\rangle^*\eta_\sigma \circ x$. Conversely, given $b\colon(X, \sigma) \to (\kappa_C Y, \kappa_C\tau)$ in $\tilde{\mathbf{G}}^C(A)$, the induced morphism $a = \varepsilon_\tau \circ \sum_C(b)\colon \sum_C\sigma \to \tau$ of $\mathbf{G}(A)$ is a morphism of $\tilde{\mathbf{G}}(A)$, $(\sum_C X, \sum_C\sigma) \to (Y, \tau)$. Now we must verify that for any $z = \langle\rho_1, \rho_2\rangle^*\eta_\sigma \circ x$ in $\sum X(\rho_1)$, $\rho_1^*a \circ z$ is in $Y(\rho_1)$. But by the triangle equality $b = \kappa_C(\varepsilon_\tau) \circ \kappa_C\sum_C(b) \circ \eta_\sigma$, so

$$\langle\rho_1, \rho_2\rangle^*b = \rho_1^*\left(\varepsilon_\tau \circ \sum_C(b)\right) \circ \langle\rho_1, \rho_2\rangle^*\eta_\sigma = \rho_1^*a \circ \langle\rho_1, \rho_2\rangle^*\eta_\sigma.$$

So $\rho_1^* a \circ z = (\langle \rho_1, \rho_2 \rangle^* b) \circ x$, and this is in $Y(\rho_1)$ since b is a morphism of $\tilde{\mathbf{G}}^C(A)$. (It is obvious these correspondences respect equivalence of morphisms in $\tilde{\mathbf{G}}$.)

§7. Natural numbers objects.

7.1. In a category, a natural numbers object N is an initial diagram of the form $1 \xrightarrow{0} N \xrightarrow{s} N$; i.e. for any diagram $1 \xrightarrow{a} A \xrightarrow{t} A$, there is a unique $N \xrightarrow{f} A$ making the following diagram commute:

$$\begin{tikzcd} & N \arrow[r, "s"] \arrow[d, "f"] & N \arrow[d, "f"] \\ 1 \arrow[ur, "0"] \arrow[dr, "a"'] & & \\ & A \arrow[r, "t"'] & A \end{tikzcd}$$

If f is not necessarily unique, then N is a weak natural numbers object.

For an indexed category $(\mathbf{G}, \mathbf{S})$ to have a natural numbers object N, each fibre $\mathbf{G}(A)$ should have a natural numbers object N_A, and for any $f: B \to A$ of $\mathbf{S}$, $f^*(N_A) = N_B$ (up to unique isomorphism). It follows from Freyd's characterisation of natural numbers objects that any (left and right) exact functor preserves them (Freyd [1972]), and so if $(\mathbf{G}, \mathbf{S})$ is a PL (or PLS) category, $\kappa_C(A)$ preserves N for any A, C in $\mathbf{S}$, since κ has left and right adjoints. In particular, if $!_A: A \to 1$, then $!_A^*$ preserves natural numbers objects, so $N_A = !_A^*(N_1)$, and so also for $f: B \to A$, $f^*(N_A) = f^* \, !_A^*(N_1) = !_B^*(N_1) = N_B$. Hence:

DEFINITION/COROLLARY 7.2. *A PL (or PLS) category* $(\mathbf{G}, \mathbf{S})$ *has a natural numbers object if* $\mathbf{G}(1)$ *has.*

7.3. Any PL category $(\mathbf{G}, \mathbf{S})$ has a natural candidate for a natural numbers object, namely the interpretation of the type $N = \prod \alpha \in \Omega \cdot \alpha \times (\alpha \supset \alpha) \supset \alpha$ in $\mathbf{G}(1)$. Then $0: \top \to N$ would be the interpretation of the (closed) term $\Lambda \; \alpha \in \Omega \cdot \lambda \langle x, y \rangle \in \alpha \times (\alpha \supset \alpha) \cdot x$ (with the evident abuse of notation to avoid projection terms), and s: $N \to N$ would be the interpretation of the term

$$\lambda n \in N \cdot \Lambda \, \alpha \in \Omega \cdot \lambda \langle x, y \rangle \in \alpha \times (\alpha \supset \alpha) \cdot y(n\{\alpha\}(\langle x, y \rangle)).$$

For any closed type σ (i.e. object of $\mathbf{G}(1)$), there is an "iterator"

$$I_\sigma: \sigma \times (\sigma \supset \sigma) \times N \to \sigma, \qquad I_\sigma = \lambda \langle x, y, n \rangle \in \sigma \times (\sigma \supset \sigma) \times N \cdot n\{\sigma\}(\langle x, y \rangle),$$

satisfying the equations

$$I_\sigma(\langle x, y, 0 \rangle) = x, \qquad I_\sigma(\langle x, y, s(n) \rangle) = y(I_\sigma(\langle x, y, n \rangle)).$$

So N is a weak natural numbers object (Lambek and Scott [1986]). Primitive recursion can then be defined: $R_\sigma: \sigma \times (N \times \sigma \supset \sigma) \times N \to \sigma$ is given by

$$R_\sigma(x, y, n) = \pi_2(n\{N \times \sigma\}(\langle 0, x \rangle, \langle s\pi_1, y \rangle)),$$

for projections $N \xleftarrow{\pi_1} N \times \sigma \xrightarrow{\pi_2} \sigma$, and

$$R_\sigma(x, y, 0) = x, \qquad R_\sigma(x, y, sn) = y(n\{N \times \sigma\}(\langle 0, x \rangle, \langle s\pi_1, y \rangle)).$$

The usual recursion equation is $R_\sigma(x, y, sn) = y(\langle n, R_\sigma(x, y, n) \rangle)$; this would follow if $\pi_1(n\{N \times \sigma\}(\langle 0,x\rangle, \langle s\pi_1, y\rangle)) = n$, which is true for standard numerals $s \circ s \circ \cdots \circ s(0)$, but not in general. However, we could make N a (strong) natural numbers object by "throwing away" nonstandard numerals:

PROPOSITION 7.4. *Any PL category* $(\mathbf{G}, \mathbf{S})$ *has a weak natural numbers object, the type N of "Church numerals". The corresponding PLS category* $(\tilde{\mathbf{G}}, \mathbf{S})$ *has a natural numbers object* (X, N), *where X is the set of standard numerals.*

7.5. EXAMPLE. In the closure operator model of §3, $N = \lambda x \cdot \lambda a \in \Omega \cdot a \circ x(a) \circ a \wedge (a \supset a)$; a "term" (= fixed point) of "type" N is a function $x: K \to \mathscr{P}\omega$ so that for any $a \in K$, $x(a): a \wedge (a \supset a) \to a$. Here the standard numerals are iterations of the "evaluation" function ($1(a) = \mathrm{ev}_a: a \wedge (a \supset a) \to a$), but N has other terms. (For instance, every closure operator has the fixed point $\top = \omega$, so the constant maps to $\top$ give a fixed point of N, which is not a standard numeral.)

7.6. REMARKS. (1) With 7.4 we have all the structure needed for a Dialectica interpretation, as carried out in Girard [1971]. This would follow a pattern similar to that of Scott [1978]; I hope to present the details in a sequel.

(2) There are other models of PLC that give rise to PL categories with very interesting structure. One example of an internal PL category with an internal natural numbers object, due to J. M. E. Hyland, is based on the partial (or restricted) equivalence relations of Scott [1976]. The idea is to construct the category **Per** of partial equivalence relations inside the effective topos **Eff**. **Per** then turns out to be a locally cartesian closed (internal) category with natural numbers object, and is weakly complete and cocomplete. I hope the details of this model will be available soon. It is related to the structure HEO of Girard [1972].

REFERENCES

H. BARENDREGT and A. REZUS [1983], *Semantics for classical AUTOMATH and related systems*, ***Information and Control***, vol. 59, pp. 127–147.

K. B. BRUCE and A. R. MEYER [1984], *The semantics of second order polymorphic lambda calculus*, ***Semantics of data types*** (G. Kahn et al., editors), Lecture Notes in Computer Science, vol. 173, Springer-Verlag, Berlin, pp. 131–144.

R. BURSTALL, D. MACQUEEN and D. SANNELLA [1980], ***HOPE: an experimental applicative language***, Report CSR-62-80, Computer Science Department, Edinburgh University, Edinburgh.

P. FREYD [1972], *Aspects of topoi*, ***Bulletin of the Australian Mathematical Society***, vol. 7, pp. 1–76.

J.-Y. GIRARD [1971], *Une extension de l'interpretation de Gödel à l'analyse, et son application à l'élimination des coupures dans l'analyse et la théorie des types*, ***Proceedings of the second Scandinavian logic symposium*** (J. E. Fenstad, editor), North-Holland, Amsterdam, pp. 63–92.

——— [1972], ***Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur***, Thèse de Doctorat d'État, Université Paris-VII, Paris. (Much of this is summarised in Girard [1971], [1973].)

——— [1973], *Quelques résultats sur les interprétations fonctionnelles*, ***Cambridge summer school in mathematical logic*** (A. R. D. Mathias and H. Rogers, editors), Lecture Notes in Mathematics, vol. 337, Springer-Verlag, Berlin, pp. 232–252.

J. M. E. HYLAND, P. T. JOHNSTONE and A. M. PITTS [1980], *Tripos theory*, ***Mathematical Proceedings of the Cambridge Philosophical Society***, vol. 88, pp. 205–232.

F. LAMARCHE [1985], Unpublished lecture notes, McGill University, Montréal.

J. LAMBEK and P. J. SCOTT [1986], ***Introduction to higher order categorical logic***, Cambridge Studies in Advanced Mathematics, vol. 7, Cambridge University Press, Cambridge.

S. MAC LANE [1971], ***Categories for the working mathematician***, Springer-Verlag, Berlin.

N. J. MCCRACKEN [1979], ***An investigation of a programming language with a polymorphic type structure***, Ph.D. Thesis, Syracuse University, Syracuse, New York.

R. MILNER [1984], ***The standard ML core language***, Report CSR-168-84, Computer Science Department, Edinburgh University, Edinburgh.

R. PARÉ and D. SCHUMACHER [1978], *Abstract families and the adjoint functor theorems*, ***Indexed categories and their applications*** (P. T. Johnstone and R. Paré, editors), Lecture Notes in Mathematics, vol. 661, Springer-Verlag, Berlin, pp. 1–125.

J. C. REYNOLDS [1974], *Towards a theory of type structure*, ***Colloque sur la programmation*** (B. Robinet editor), Lecture Notes in Computer Science, vol. 19, Springer-Verlag, Berlin, pp. 408–425.

——— [1984], *Polymorphism is not set-theoretic*, ***Semantics of data types*** (G. Kahn et al., editors), Lecture Notes in Computer Science, vol. 173, Springer-Verlag, Berlin, pp. 145–156.

D. S. SCOTT [1976], *Data types as lattices*, ***SIAM Journal on Computing***, vol. 5, pp. 522–587.

P. J. SCOTT [1978], *The "Dialectica" interpretation and categories*, ***Zeitschrift für Mathematische Logik und Grundlagen der Mathematik***, vol. 24, pp. 553–575.

R. A. G. SEELY [1979], ***Girard's type theory and categories***, Unpublished lecture notes, McGill University, Montréal.

——— [1982], Review of P. T. Johnstone, ***Topos theory***, this JOURNAL, vol. 47, pp. 448–450.

——— [1983], *Hyperdoctrines, natural deduction and the Beck condition*, ***Zeitschrift für Mathematische Logik und Grundlagen der Mathematik***, vol. 29, pp. 505–542.

——— [1984], *Locally cartesian closed categories and type theory*, ***Mathematical Proceedings of the Cambridge Philosophical Society***, vol. 95, pp. 33–48.

——— [1986a], *Higher order polymorphic lambda calculus and categories*. I, ***Mathematical Reports of the Academy of Science (Canada)***, vol. 8, pp. 135–139.

——— [1986b], *Higher order polymorphic lambda calculus and categories*. II, ***Mathematical Reports of the Academy of Science (Canada)***, vol. 8, pp. 197–201.

DEPARTMENT OF MATHEMATICS
JOHN ABBOTT COLLEGE
STE. ANNE DE BELLEVUE, QUÉBEC H9X 3L9, CANADA