PUC

Series: Monografias em Ciência da Computação,
No. 11/90

MODEL OF PTYXES FOR SUM AND EMPTY TYPES

Maria Cláudia Meré

Departamento de Informática

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO
RUA MARQUÊS DE SÃO VICENTE, 225 - CEP-22453
RIO DE JANEIRO - BRASIL

In charge of publications:

Rosane Teles Lins Castilho
Assessoria de Biblioteca, Documentação e Informação
PUC RIO, Departamento de Informática
Rua Marquês de São Vicente, 225 - Gávea
22453 - Rio de Janeiro, RJ
BRASIL

Tel.: (021)529-9386 TELEX:31078 FAX: (021)274-4546
BITNET: userrtlc@lncc.bitnet

PUC/RIO - DEPARTAMENTO DE INFORMÁTICA

Series: Monografias em Ciência da Computação, 11/90

Editor: Paulo Augusto Silva Veloso Sept., 1990

MODEL OF PTYXES FOR SUM AND EMPTY TYPES[1]

MARIA CLAUDIA MERÉ

[1]Presented by Professor Luiz Carlos P.D. Pereira-Departamento de Filosofia-PUC/RIO.

In charge of publications:

Rosane Teles Lins Castilho
Assessoria de Biblioteca, Documentação e Informação
PUC RIO, Departamento de Informática
Rua Marquês de São Vicente, 225 - Gávea
22453 - Rio de Janeiro, RJ
BRASIL

Tel.:(021)529-9386 TELEX:31078 FAX:(021)274-4546
BITNET:rtlc@lncc.bitnet

Model of Ptyxes for Sum and Empty Types

Maria Claudia Meré
Departamento de Informática
Pontifícia Universidade Católica do Rio de Janeiro-Brazil

Luiz Carlos P.D.Pereira
Departamento de Filosofia
Pontifícia Universidade Católica do Rio de Janeiro-Brazil

August of 1990

Abstract

In Chapter 12 of a forthcoming book, Girard [Gir] presents the model of Ptyxes of finite types as a model of a variant T 'of Gödel's T (a typed λ-calculus with primitive recursion). In Päppinghaus [Pap85] this model is extended to a typed lambda-calculus with ordinal structure (sup terms).

In the present paper we show how to extend the model of Ptyxes of finite types to a typed λ-calculus which also includes *sum* and *empty* types [GLP89].

KEY WORDS
typed λ-calculus, canonical model, Ptyxes, the category ON.

Resumo

No capitulo 12 de um livro ainda não publicado, Girard [Gir] apresenta o modelo dos Ptyxes de tipos finitos como um modelo de uma variante T ' da T de Gödel (um λ-calculus tipado com operador de recursão primitiva).

Em Päppinghaus [Pap85] este modelo é extendido a um lambda-calculus com estrutura ordinal (termos sup).

Nós mostramos como obter um modelo dos Ptyxes para tipos finitos o qual é um modelo para o λ-calculus tipado com o tipo *soma* e o *vazio* [GLP89].

PALAVRAS CHAVES
λ-calculus tipado, modelo canonico, Ptyxes, a categoria ON.

Contents

Acknowledgements

We wish to thank Nicolau Saldanha, Valeria Paiva and E.H. Haeusler for helpful discussions and suggestions during the elaboration of this work.

1 Introduction

The concept of Ptyxes of a certain finite types was introduced by Girard [Gir] as a generalization of the concept of Dilators [Gir81] to finite types. Girard also proved that the Ptyxes of finite types constituted a model of a variant T' of the well-known Gödel's T.

In 1985, Päppinghaus showed how to extend this model to an extended λ-calculus which included an ordinal structure (sup-terms) and showed how to obtain some results concerning ordinal measures for the evaluation of terms of type $0 \to 0$ on ω.

The aim of the present paper is to extend Girard's model of Ptyxes to an extended typed λ-calculus which includes besides sup-terms and ordinal structure, the types *sum* and *empty*.

The addition of these new types usually introduces difficulties which are very peculiar to them. It is well known that *sum* works with contexts. As Girard puts it in the context of Natural Deduction " What is catastrophic about them is the parasitic precense of a formula C which has no structural link with the formula which is eliminated. C plays the rôle of a context, and the writing of these rules is a concession to Sequent Calculus" [GLP89].

We will show that Ptyxes constitute a nice model of terms of these new types, providing an interesting treatement of contexts and of the equations related to them. In fact, we will present two different models of Ptyxes; one based on a partial evaluation function and another based on a total evaluation function. In spite of the technical differences between them, we think there are some interesting conceptual motivations (which we will not discuss here) for choosing one of them.

Most of the terminology and notation ussed in this paper is taken from [Gir] and [Pap85].

2 The Typed λ-calculus. λ-ND

Definition 2.1 *Types* are inductively defined by the following clauses :

1. EMP *is a finite type (the empty type).*
2. *0 is a finite type.*
3. *If σ and τ are finite types, then $(\sigma \to \tau)$, $(\sigma \times \tau)$, $(\sigma + \tau)$ are finite types.*
4. *The only finite types are those obtained by means of 1,2,3.*

As usual $\sigma_1 \to \sigma_2 \to \ldots \to \sigma_n$ is a shorthand for $(\sigma_1 \to (\sigma_2 \to (\ldots \to \sigma_n)))$.

Definition 2.2 The class $\lambda\text{-}ND_\sigma$ *of λ-terms of type σ* are inductively defined as :

variables : $a^\sigma, b^\sigma, c^\sigma, \ldots \in \lambda\text{-}ND_\sigma$.

constants : $0 \in \lambda\text{-}ND_0$, $S \in \lambda\text{-}ND_{0\to 0}$, $+ \in \lambda\text{-}ND_{0\to 0\to 0}$,
$R_\sigma \in \lambda\text{-}ND_{\sigma\to(\sigma\to 0\to\sigma)\to 0\to\sigma}$.

sup-terms : $t \in \lambda\text{-}ND_0$, *α limit ordinal, x^0 variable, then* $(\sup_{x^0<\alpha} t) \in \lambda\text{-}ND_0$.

$\otimes$-terms : $s \in \lambda\text{-}ND_\sigma$, $t \in \lambda\text{-}ND_\tau$, *then* $(s \otimes t) \in \lambda\text{-}ND_{\sigma\times\tau}$.

π-terms : $t \in \lambda\text{-}ND_{\sigma\times\tau}$, *then* $(\pi^1 t) \in \lambda\text{-}ND_\sigma$, $(\pi^2 t) \in \lambda\text{-}ND_\tau$.

app-terms : $t \in \lambda\text{-}ND_{\sigma\to\tau}$, $s \in \lambda\text{-}ND_\sigma$, *then* $(ts) \in \lambda\text{-}ND_\tau$.

λ-terms : $t \in \lambda\text{-}ND_\tau$, a^σ *variable, then* $(\lambda a^\sigma.t) \in \lambda\text{-}ND_{\sigma\to\tau}$.

$\oplus$-terms : $t \in \lambda\text{-}ND_\sigma$, *then* $(i^1 t) \in \lambda\text{-}ND_{\sigma+\tau}$, *and*
$t \in \lambda\text{-}ND_\tau$, *then* $(i^2 t) \in \lambda\text{-}ND_{\sigma+\tau}$.

case-terms : $t[a^\sigma] \in \lambda\text{-}ND_\delta$, $s[b^\tau] \in \lambda\text{-}ND_\delta$, $r \in \lambda\text{-}ND_{\sigma+\tau}$, *then*
$(\oplus(a,b)(r, t[a^\sigma], s[b^\tau])) \in \lambda\text{-}ND_\delta$.

empty-terms : $t \in \lambda\text{-}ND_{EMP}$, *then* $(\varepsilon^\sigma t) \in \lambda\text{-}ND_\sigma$.

Definition 2.3 1. $CT_\sigma :\equiv \{t \in \lambda\text{-}ND_\sigma \;/\; t \text{ closed}\}$

2. $+_0 :\equiv + \in CT_{0\to 0\to 0}$
$+_{\sigma\to\tau} :\equiv \lambda a.\lambda b.\lambda c.\, +_\tau (ac)(bc) \in CT_{(\sigma\to\tau)\to(\sigma\to\tau)\to(\sigma\to\tau)}$, $\tau \not\equiv EMP$

$+_{\sigma\times\tau} :\equiv \lambda a.\lambda b.\, +_\sigma (\pi^1 a)(\pi^1 b) \otimes +_\tau(\pi^2 a)(\pi^2 b) \in CT_{(\sigma\times\tau)\to(\sigma\times\tau)\to(\sigma\times\tau)}$, $\sigma, \tau \not\equiv EMP$

$$+_{\sigma+\tau} :\equiv \begin{cases} \lambda a.\lambda b.i^1(t_1 +_\sigma t_2) & \text{if } a^{\sigma+\tau} = i^1 t_1 \,,\ b^{\sigma+\tau} = i^1 t_2 \,,\ \sigma \not\equiv EMP \\ \lambda a.\lambda b.i^2(t_1 +_\tau t_2) & \text{if } a^{\sigma+\tau} = i^2 t_1 \,,\ b^{\sigma+\tau} = i^2 t_2 \,,\ \tau \not\equiv EMP \end{cases}$$

3. For $t \in \lambda\text{-}ND_{\sigma\to\tau}$: $(\sup_{x<\alpha} t) :\equiv \lambda a. \sup_{x<\alpha}(ta) \in \lambda\text{-}ND_{\sigma\to\tau}$, $\tau \neq EMP$
For $t \in \lambda\text{-}ND_{\sigma\times\tau}$: $(\sup_{x<\alpha} t) :\equiv \sup_{x<\alpha}(\pi^1 t) \otimes \sup_{x<\alpha}(\pi^2 t) \in \lambda\text{-}ND_{\sigma\times\tau}$
$\sigma, \tau \neq EMP$
If $t^{\sigma+\tau} = i^1 t_1^\sigma$ then : $(\sup_{x<\alpha} t) :\equiv i^1(\sup_{x<\alpha} t_1^\sigma) \in \lambda\text{-}ND_{\sigma+\tau}$, $\sigma \neq EMP$
If $t^{\sigma+\tau} = i^2 t_1^\sigma$ then : $(\sup_{x<\alpha} t) :\equiv i^2(\sup_{x<\alpha} t_1^\tau) \in \lambda\text{-}ND_{\sigma+\tau}$, $\tau \neq EMP$

As usual $t_1 t_2 \ldots t_n$ as an abbreviation of $(\ldots(t_1 t_2)\ldots t_n)$. Indexes will be drops whenever possible. The concepts of bound and free variables, closed terms, substitution, redex, contraction, normal terms, alphabetical variables are the usual ones. Substitution of a term s for a variable a in a term t (s and a of the same type) will be denoted by $t[a^\sigma :\equiv s]$, or if t is written as $t[a^\sigma]$ by $t[s/a^\sigma]$ or by $t[s]$ when is clear by the context.

Definition 2.4 The *reduction* $\vdash$ is defined by the following rules:
Operational reductions

- $+t0 \vdash t$
- $+t(Su) \vdash S(+tu)$
- $+t(\sup_{x<\alpha} u) \vdash \sup_{x<\alpha}(+tu)$
- $R_\sigma tu0 \vdash t$
- $R_\sigma tu(Sv) \vdash +_\sigma(R_\sigma tuv)(u(R_\sigma tuv)v)$
- $R_\sigma\ tu\ (\sup_{x<\alpha} v) \vdash \sup_{x<\alpha}(R_\sigma tuv)$
- $(\lambda a^\sigma.t[a])s \vdash t[s/a^\sigma]$ *for* $t[a] \in \lambda\text{-}ND_\tau, s \in \lambda\text{-}ND_\sigma$
- $\pi^1(s \otimes t) \vdash s$
  $\pi^2(s \otimes t) \vdash t$
- $(\oplus(a,b)(i^1 r^\sigma, t[a^\sigma], s[b^\tau])) \vdash t[r/a^\sigma]$
  $(\oplus(a,b)(i^2 r^\tau, t[a^\sigma], s[b^\tau])) \vdash s[r/b^\tau]$

Commuting reductions for $\oplus$

- $\pi^1(\oplus(a,b)(r,t,s)) \vdash (\oplus(a,b)(r,\pi^1 t,\pi^1 s))$
- $\pi^2(\oplus(a,b)(r,t,s)) \vdash (\oplus(a,b)(r,\pi^2 t,\pi^2 s))$
- $(\oplus(a,b)(r,t,s))w \vdash (\oplus(a,b)(r,tw,sw))$
- $\varepsilon^\alpha(\oplus(a,b)(r,t,s)) \vdash (\oplus(a,b)(r,\varepsilon^\alpha t,\varepsilon^\alpha s))$
- $(\oplus(a',b')((\oplus(a,b)(r,t[a^\sigma],s[b^\tau])),t'[a'^\sigma],s'[b'^\tau])) \vdash$
  $(\oplus(a,b)(r,(\oplus(a',b')(t[a^\sigma],t'[a'^\sigma],s'[b'^\tau])),(\oplus(a',b')(s[b^\tau],t'[a'^\sigma],s'[b'^\tau]))))$

Commuting reductions for *EMP*

- $\pi^1(\varepsilon^{\sigma\times\tau}t) \vdash \varepsilon^{\sigma}t$
- $\pi^2(\varepsilon^{\sigma\times\tau}t) \vdash \varepsilon^{\tau}t$
- $(\varepsilon^{\sigma\to\tau}t)u \vdash \varepsilon^{\tau}t$
- $\varepsilon^{\alpha}(\varepsilon^{EMP}t) \vdash \varepsilon^{\alpha}t$
- $(\oplus(a,b)(\varepsilon^{\sigma+\tau}r^{EMP}, t[a^{\sigma}]^{\alpha}, s[b^{\tau}]^{\alpha})) \vdash \varepsilon^{\alpha}r$

Since there is no introduction rules for $\perp$, there is no standard reduction for the ε symbol.

We have the commuting reductions in order to resolve additional difficulties we encounter in the full calculus. It is no longer true that the conclusion of an elimination is a subformula of the major premise. This fact has as immediate consequence that whithout the commuting reductions we loose the important property that if $t_1 \vdash t_2$ then, in t_2 we do not have more "complex" redexes that those we have in t_1. In a way, the sum operator combined with other operators allows us to construct "bad forms" (in Natural Deduction terminology "bad eliminations") and the commuting reductions are introduced in order to transform that "bad forms" into "good forms" ("good eliminations") restoring thus the desired properties (complexity, sub-formula). On the other hand, the adoption of the new reduction rules still has undesirable consequences, as for example the a priori identification of different deductions. As Girard puts it:

"Moreover, the extensions are long and difficult, and for all that you will not learn anything new apart from technical variations on reducibility. So it will suffice to know that the Strong Normalisation Theorem also holds in this case" [GLP89, p.80].

The one step reduction $\longrightarrow$ is defined to be the closure of the reduction $\vdash$ under the term forming rules. The reduction $\leadsto$ is defined to be the reflexive, transitive closure of the one step reduction $\longrightarrow$.

Proposition 2.5 $\leadsto$ *is strongly normalizable and Church-Rosser.*

Proof : see [Pra70] and [Gir72]. □

3 Canonical Model

Definition 3.1 The notion of a *distinguished variable (d. var.)* of a term t is defined by the following inductive clauses :

1. *If* $t \equiv x \in \lambda\text{-}ND_0$*, x is the d. var. of t .*
2. *If x is the d. var. of* $u \in \lambda\text{-}ND_0$ *and x is not free in* $t \in \lambda\text{-}ND_0$*, then x is the d. var. of* $+tu \in \lambda\text{-}ND_0$.
3. *If x is the d. var. of* $v \in \lambda\text{-}ND_0$*, x is not free in* $t \in \lambda\text{-}ND_\sigma$ *and* $u \in \lambda\text{-}ND_{\sigma\to 0\to\sigma}$ *, then x is the d. var. of* $R_\sigma tuv \in \lambda\text{-}ND_\sigma$.
4. *If x is the d. var. of* $t \in \lambda\text{-}ND_{\sigma\to\tau}$ *and x is not free in* $s \in \lambda\text{-}ND_\sigma$*, then x is the d. var. of* $ts \in \lambda\text{-}ND_\tau$.
5. *If x is the d. var. of* $t \in \lambda\text{-}ND_{\sigma\times\tau}$*, then x is the d. var. of* $\pi^1 t \in \lambda\text{-}ND_\sigma$ *,* $\pi^2 t \in \lambda\text{-}ND_\tau$.
6. *If x is the d. var. of* $t \in \lambda\text{-}ND_{EMP}$*, then x is the d. var. of* $\varepsilon^\sigma t \in \lambda\text{-}ND_\sigma$.
7. *If x is the d. var. of* $t,s \in \lambda\text{-}ND_\delta$ *and x is not free in* $r \in \lambda\text{-}ND_{\sigma+\tau}$*, then x is the d. var. of* $(\oplus(a,b)(r,s,t))$.

Because of the fact that for Ptyxes the sup does not always exist, we restrict our typed λ-calculus to a sintactically defined subclass of it, which is big enough for our purpose. Let Λ be a fixed ordinal or Λ =On. Then, we change our definition for sup-terms in the following way:

$$\text{If} t \in \lambda\text{-ND}_0, x^0 \text{the d. var. of t}, \alpha < \Lambda \text{and} \alpha \text{ limit ordinal, then } \sup_{x<\alpha} t \in \lambda\text{-ND}_0.$$

The restricted λ-calculus thus obtained is called λ-ND$< \Lambda$.

Definition 3.2 A *type structure* is a sequence
$M =< \{M_\sigma \ / \ \sigma$ *finite type*$\}, \{\odot_{\sigma\tau} \ / \ \sigma,\tau$ *finite types*$\}, \{\otimes_{\sigma\tau} \ / \ \sigma,\tau$ *finite types*$\}$,
$\{\pi^j_{\sigma\tau} \ / \ \sigma,\tau$ *finite types, j= 1,2* $\}, \{i^j_{\sigma\tau} \ / \ \sigma,\tau$ *finite types, j= 1,2* $\} >$
such that

1. M_{EMP} *is empty.*
2. $\odot_{\sigma\tau} : M_{\sigma\to\tau} \times M_\sigma \longrightarrow M_\tau$
3. $\otimes_{\sigma\tau} : M_\sigma \times M_\tau \longrightarrow M_{\sigma\times\tau}$
4. $\pi^1_{\sigma\tau} : M_{\sigma\times\tau} \longrightarrow M_\sigma$
   $\pi^2_{\sigma\tau} : M_{\sigma\times\tau} \longrightarrow M_\tau$
5. $\forall a \in M_\sigma$, $\forall b \in M_\tau$ $\pi^1_{\sigma\tau}(a \otimes_{\sigma\tau} b) = a$, $\pi^2_{\sigma\tau}(a \otimes_{\sigma\tau} b) = b$

6. $i^1_{\sigma\tau} : M_\sigma \longrightarrow M_{\sigma+\tau}$
$i^2_{\sigma\tau} : M_\tau \longrightarrow M_{\sigma+\tau}$

Definition 3.3 We define an *ordinal operator structure over an ordinal* $\Lambda(\Lambda\text{-}OOS)$ to be a pair $< M, * >$ s.t.

1. The basic domain M_0 is a limit ordinal $\geq \Lambda$ or $\Lambda = \mathrm{On}$.
2. $*$ is a map that associates to each term $t \in \lambda\text{-ND}_\tau$ and to each sequence $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$ of distinct variables that contains the free variables of t, a map $t^* : M_{\sigma_1} \times \ldots \times M_{\sigma_n} \longrightarrow M_\tau$ that satisfies the following conditions :
   (a) *t^* is independent of the naming and ordering of the free variables of t.*
   (b) $(a^\sigma)^*\ (c) = c \in M_\sigma,\ \ \forall \sigma \neq EMP.$[1]
   (c) • *If $u^*(c_1, \ldots, c_n)$ and $v^*(c_1, \ldots, c_n)$ are defined and if u is not a case-term or an empty-term, then*
   $(uv)^*(c_1, \ldots, c_n) = u^*(c_1, \ldots, c_n) \odot_{\sigma\tau} v^*(c_1, \ldots, c_n)$
   • $((\varepsilon^{\sigma\to\tau}t)v)^*(c_1, \ldots, c_n) = (\varepsilon^\tau t)^*(c_1, \ldots, c_n)$
   • $((\oplus(a,b)(r,t,s))v)^*(c_1, \ldots, c_n) = (\oplus(a,b)(r,tv,sv))^*(c_1, \ldots, c_n)$
   (d) • *If $v^*(c_1, \ldots, c_n)$ is defined and if v is not a case-term or an empty-term , then*
   $(\pi^j v)^*(c_1, \ldots, c_n) = \pi^j_{\sigma\tau}(v^*(c_1, \ldots, c_n))$
   • $(\pi^1(\varepsilon^{\sigma\times\tau}t))^*(c_1, \ldots, c_n) = (\varepsilon^\sigma t)^*(c_1, \ldots, c_n)$
   $(\pi^2(\varepsilon^{\sigma\times\tau}t))^*(c_1, \ldots, c_n) = (\varepsilon^\tau t)^*(c_1, \ldots, c_n)$
   • $(\pi^j(\oplus(a,b)(r,t,s)))^*(c_1, \ldots, c_n) = (\oplus(a,b)(r,\pi^j t,\pi^j s))^*(c_1, \ldots, c_n)$
   (e) *If $s^*(c_1, \ldots, c_n)$ and $t^*(c_1, \ldots, c_n)$ are defined, then*
   $(s \otimes t)^*(c_1, \ldots, c_n) = s^*(c_1, \ldots, c_n) \otimes t^*(c_1, \ldots, c_n)$
   (f) • *If $t^*(c_1, \ldots, c_n, c_{n+1})$ is defined, then*
   $(\lambda a^\sigma.t)^*(c_1, \ldots, c_n) \odot_{\sigma\tau} d = t^*(c_1, \ldots, c_n, d)$ *for every* $d \in M_\sigma$.
   • *If* $\forall\, d \in M_\sigma\ \ t^*(c_1, \ldots, c_n, d) = s^*(c_1, \ldots, c_n, d)$ *then*
   $(\lambda x^\sigma t)^*(c_1, \ldots, c_n) = (\lambda x^\sigma s)^*(c_1, \ldots, c_n)$
   (g) $0^* = 0 \in M_0$.
   (h) $S^* \odot_{0\to 0\,0} \alpha = \alpha + 1$ *for every* $\alpha \in M_0$.
   (i) *If $t^*(c_1, \ldots, c_n, c_{n+1})$ is defined, then*
   $(\sup_{x<\alpha} t)^*(c_1, \ldots, c_n) = \sup_{\beta<\alpha} t^*(c_1, \ldots, c_n, \beta) \in M_0$.
   (j) *If $t^*(c_1, \ldots, c_n)$ is defined, then*
   $(i^j t)^*(c_1, \ldots, c_n) = i^j_{\sigma\tau}(t^*(c_1, \ldots, c_n))$

[1]Note that for M_{EMP}, we have $(x^{EMP})^*(c) =$ undefined , since $M_{EMP} = \emptyset$.

(k) •

$$(\oplus(a,b)(r^{\sigma+\tau}, t[a^\sigma], s[b^\tau]))^*(c_1,\ldots,c_n) = \begin{cases} t^*(c_1,\ldots,c_n,u^*(c_1,\ldots,c_n)) & \text{if } r^{\sigma+\tau} = i^1u \\ s^*(c_1,\ldots,c_n,u^*(c_1,\ldots,c_n)) & \text{if } r^{\sigma+\tau} = i^2u \end{cases}$$

In the first case we need that $t^*(c_1,\ldots,c_n,c_{n+1})$ and $u^*(c_1,\ldots,c_n)$ are defined, and in the second case we need that $s^*(c_1,\ldots,c_n,c_{n+1})$ and $u^*(c_1,\ldots,c_n)$ are defined.

- $(\oplus(a,b)(\varepsilon^{\sigma+\tau}r, t^\alpha, s^\alpha))^*(c_1,\ldots,c_n) = (\varepsilon^\alpha r)^*(c_1,\ldots,c_n)$
- $(\oplus(a',b')((\oplus(a,b)(r,t,s)),t',s'))^*(c_1,\ldots,c_n) =$
  $(\oplus(a,b)(r,(\oplus(a',b')(t,t',s')),(\oplus(a',b')(s,t',s'))))^*(c_1,\ldots,c_n)$
- *If* $(t[a])^*(c_1,\ldots,c_n) = (t'[a])^*(c_1,\ldots,c_n)$,
  $(s[b])^*(c_1,\ldots,c_n) = (s'[b])^*(c_1,\ldots,c_n)$,
  $r^*(c_1,\ldots,c_n) = r'^*(c_1,\ldots,c_n)$ *then*
  $(\oplus(a,b)(r,t[a],s[b]))^*(c_1,\ldots,c_n) = (\oplus(a,b)(r',t'[a],s'[b]))^*(c_1,\ldots,c_n)$

(l)
- $(\varepsilon^\sigma t)^*(c_1,\ldots,c_n) = t^*(c_1,\ldots,c_n)$ *if t is not a case-term.*[2]
- $(\varepsilon^\sigma(\oplus(a,b)(r,t,s)))^*(c_1,\ldots,c_n) = (\oplus(a,b)(r,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(c_1,\ldots,c_n)$

Definition 3.4 A *canonical model of λ-ND$<\Lambda$* is a Λ-*OOS* $<M,*>$ that satisfies the following additional conditions :

1. *If $t[a^\sigma] \in \lambda\text{-}ND_\tau$, $s \in \lambda\text{-}ND_\sigma$, a^σ not free in s, t^* and s^* defined, then* $(t[s])^*(c_1,\ldots,c_n) = t^*(c_1,\ldots,c_n,s^*(c_1,\ldots,c_n))$
2. *$\alpha < \Lambda$ limit ordinal, t^* defined, then* $(\sup_{x<\alpha} t)^*(c_1,\ldots,c_n) = t^*(c_1,\ldots,c_n,\alpha)$
3. $+^* \odot \alpha \odot \beta = \alpha + \beta \in M_0$ *for all* $\alpha,\beta \in M_0$.
4. $R^*_\sigma \odot c \odot d \odot 0 = c$,
   $R^*_\sigma \odot c \odot d \odot (\alpha+1) = +^*_\sigma \odot (R^*_\sigma \odot c \odot d \odot \alpha) \odot (d \odot (R^*_\sigma \odot c \odot d \odot \alpha) \odot \alpha)$
   for every $\alpha \in M_0$.

Lemma 3.5 *Let $<M,*>$ be a canonical model of λ-ND$<\Lambda$.*
Then, for any t, $s \in \lambda$-ND$<\Lambda$ s.t. t^* and s^* are defined we have: If $t \vdash s$ then $t^* = s^*$

Proof : By inspection of the reduction rules.

1. $(\lambda a.t[a])s \vdash t[s]$. *In [Pap85]*
2. $+t(\sup\ u) \vdash \sup(+tu)$. *In [Pap85]*

[2]It is interesting to notice that since every term t of type EMP is necessarily open (it's a consequence of normalization!) none of them has value in the model.

3. $R_\sigma tu(\text{sup } v) \vdash \text{sup}(R_\sigma tuv)$. *In [Pap85]*

4. $(\pi^1(s \otimes t)) \vdash s$
$(\pi^1(s \otimes t))^*(c_1, \ldots, c_n) =$
$\pi^1_{\sigma\tau}((s \otimes t)^*(c_1, \ldots, c_n)) =$
$\pi^1_{\sigma\tau}(s^*(c_1, \ldots, c_n) \otimes_{\sigma\tau} t^*(c_1 \ldots, c_n)) =$
$s^*(c_1, \ldots, c_n)$

5. $(\pi^2(s \otimes t)) \vdash t$. *Idem 4*

6. $(\oplus(a, b)(i^1 r^\sigma, t[a^\sigma], s[b^\tau])) \vdash t[r]$
$(\oplus(a, b)(i^1 r^\sigma, t[a^\sigma], s[b^\tau]))^*(c_1, \ldots, c_n) =$
$t^*(c_1, \ldots, c_n, r^*(c_1, \ldots, c_n)) =$
$(t[r])^*(c_1, \ldots, c_n)$

7. $(\oplus(a, b)(i^2 r^\sigma, t[a^\sigma], s[b^\tau])) \vdash s[r]$. *Idem 6*

8. $+t0 \vdash t$
$(+t0)^*(c_1, \ldots, c_n) =$
$+^* \odot t^*(c_1, \ldots, c_n) \odot 0^* =$
$t^*(c_1, \ldots, c_n) + 0 =$
$t^*(c_1, \ldots, c_n)$

9. $+t(Su) \vdash S(+tu)$
$(+t(Su))^*(c_1, \ldots, c_n) =$
$+^* \odot t^*(c_1, \ldots, c_n) \odot (S^* \odot u^*(c_1, \ldots, c_n)) =$
$t^*(c_1, \ldots, c_n) + (u^*(c_1, \ldots, c_n) + 1) =$
$(t^*(c_1, \ldots, c_n) + u^*(c_1, \ldots, c_n)) + 1 =$
$(S(+tu))^*(c_1, \ldots, c_n)$

10. $R_\sigma tu0 \vdash t$
$(R_\sigma tu0)^*(c_1, \ldots, c_n) =$
$R^*_\sigma \odot t^*(c_1, \ldots, c_n) \odot u^*(c_1, \ldots, c_n) \odot 0 =$
$t^*(c_1, \ldots, c_n)$

11. $R_\sigma tu(Sv) \vdash +_\sigma(R_\sigma tuv)(u(R_\sigma tuv)v)$
$(R_\sigma tu(Sv))^*(c_1, \ldots, c_n) =$
$R^*_\sigma \odot t^*(c_1, \ldots, c_n) \odot u^*(c_1, \ldots, c_n) \odot (S^* \odot v^*(c_1, \ldots, c_n)) =$
$R_\sigma \odot t^*(c_1, \ldots, c_n) \odot u^*(c_1, \ldots, c_n) \odot (v^*(c_1, \ldots, c_n) + 1) =$
$+^*_\sigma \odot (R^*_\sigma \odot t^*(c_1, \ldots, c_n) \odot u^*(c_1, \ldots, c_n) \odot v^*(c_1, \ldots, c_n)) \odot$
$(u^*(c_1, \ldots, c_n) \odot (R^*_\sigma \odot t^*(c_1, \ldots, c_n) \odot u^*(c_1, \ldots, c_n) \odot v^*(c_1, \ldots, c_n)) \odot v^*(c_1, \ldots, c_n)) =$
$(+_\sigma(R_\sigma tuv)(u(R_\sigma tuv)v))^*(c_1, \ldots, c_n)$

12. $\pi^1(\oplus(a, b)(r, t, s)) \;\vdash\; (\oplus(a, b)(r, \pi^1 t, \pi^1 s))$. *By 3.2.4*

13. $\pi^2(\oplus(a, b)(r, t, s)) \;\vdash\; (\oplus(a, b)(r, \pi^2 t, \pi^2 s))$. *By 3.2.4*

14. $(\oplus(a, b)(r, t, s))w \;\vdash\; (\oplus(a, b)(r, tw, sw))$. *By 3.2.3*

15. $\varepsilon^\alpha(\oplus(a, b)(r, t, s)) \;\vdash\; (\oplus(a, b)(r, \varepsilon^\alpha t, \varepsilon^\alpha s))$. *By 3.2.12*

16. $(\oplus(a',b')((\oplus(a,b)(r,t[a^\sigma],s[b^\tau])),t'[a'^\sigma],s'[b'^\tau])) \vdash$
$(\oplus(a,b)(r,(\oplus(a',b')(t[a^\sigma],t'[a'^\sigma],s'[b'^\tau])),(\oplus(a',b')(s[b^\tau],t'[a'^\sigma],s'[b'^\tau]))))$. *By 3.2.11*

17. $\pi^1(\varepsilon^{\sigma\times\tau}t) \vdash \varepsilon^\sigma t$. *By 3.2.4*

18. $\pi^2(\varepsilon^{\sigma\times\tau}t) \vdash \varepsilon^\tau t$. *By 3.2.4*

19. $(\varepsilon^{\sigma\to\tau}t)u \vdash \varepsilon^\tau t$. *By 3.2.3*

20. $\varepsilon^\alpha(\varepsilon^{EMP}t) \vdash \varepsilon^\alpha t$
$(\varepsilon^\alpha(\varepsilon^{EMP}t))^*(c_1,\ldots,c_n) =$
$(\varepsilon^{EMP}t)^*(c_1,\ldots,c_n) =$
$t^*(c_1,\ldots,c_n) =$
$(\varepsilon^\alpha t)^*(c_1,\ldots,c_n)$

21. $(\oplus(a,b)(\varepsilon^{\sigma+\tau}r^{EMP},t[a^\sigma]^\alpha,s[b^\tau]^\alpha)) \vdash \varepsilon^\alpha r$. *By 3.2.11* □

It is interesting to notice that this definition of canonical model allows us to cope with the equations $(\pi^1 t\otimes\pi^2 t)\vdash t$, $(\lambda x.tx)\vdash t$, $\varepsilon^{EMP}t\vdash t$, $(\oplus(a,b)(t,i^1a,i^2b))\vdash t$, which usually provide additional problems for other types of models (e.g. the model based on coherence spaces).

Definition 3.6 The *relation* $\leadsto$ *of reduction* is defined from the reduction $\vdash$ by the following inductive clauses :

1. $t\leadsto t$
2. $t\vdash s$ *then* $t\leadsto s$
3. $t\leadsto s, s\leadsto u$ *then* $t\leadsto u$
4. $t\leadsto t', s\leadsto s'$ *then* $ts\leadsto t's'$
5. $t\leadsto t', s\leadsto s'$ *then* $t\otimes s\leadsto t'\otimes s'$
6. $t\leadsto t'$ *then* $\pi^j t\leadsto \pi^j t'$
7. $t\leadsto t'$ *then* $i^j t\leadsto i^j t'$
8. $t\leadsto t'$ *then* $\varepsilon^\alpha t\leadsto \varepsilon^\alpha t'$
9. $t\leadsto s$ *then* $\lambda x^\sigma.t\leadsto \lambda x^\sigma.s$
10. $t[a]\leadsto t'[a], s[b]\leadsto s'[b], r\leadsto r'$ *then*
$(\oplus(a,b)(r,t[a],s[b]))\leadsto(\oplus(a,b)(r',t'[a],s'[b]))$

Proposition 3.7 *Let* $<M,*>$ *be a canonical model of* λ-*ND*$<\Lambda$. *Then for* $t,s\in\lambda$-*ND*$<\Lambda$ *s.t.* t^* *and* s^* *are defined we have :*
If $t\leadsto s$ *then* $t^*=s^*$

Proof : By induction on $\leadsto$

1. $t \leadsto t$. *Obvious* $t^*(c_1, \ldots c_n) = t^*(c_1, \ldots, c_n)$
2. *By Lemma 3.4*
3. *By inductive hypothesis* $t^*(c_1, \ldots, c_n) = s^*(c_1, \ldots, c_n)$ *and* $s^*(c_1, \ldots, c_n) = u^*(c_1, \ldots, c_n)$ *then, for transitivity of* $=$
   $t^*(c_1, \ldots, c_n) = u^*(c_1, \ldots, c_n)$
4. *By inductive hypothesis* $t^*(c_1, \ldots, c_n) = t'^*(c_1, \ldots, c_n)$ *and* $s^*(c_1, \ldots, c_n) = s'^*(c_1, \ldots, c_n)$ *because of the operators preserve iquality in the model we have*
   $t^*(c_1, \ldots, c_n) \odot_{\sigma\tau} s^*(c_1, \ldots, c_n) = t'^*(c_1, \ldots, c_n) \odot_{\sigma\tau} s'^*(c_1, \ldots, c_n)$
   then $(ts)^*(c_1, \ldots, c_n) = (t's')^*(c_1, \ldots, c_n)$
5. *Idem 4*
6. *By inductive hypothesis* $t^*(c_1, \ldots, c_n) = t'^*(c_1, \ldots, c_n)$ *then*
   $\pi^j_{\sigma\tau} t^*(c_1, \ldots, c_n) = \pi^j_{\sigma\tau} t'^*(c_1, \ldots, c_n)$ *then*
   $(\pi^j t)^*(c_1, \ldots, c_n) = (\pi^j t')^*(c_1, \ldots, c_n)$
7. *Idem 6*
8. *By IH and 3.2.12*
9. *By IH and 3.2.6*
10. *By IH and 3.2.11* □

4 Ptyxes

As we said above, Ptyxes are generalizations of the concept of Dilators (introduced in Girard [Gir81]) to finite types. We associate with each type σ a category PT^σ whose objects form a class denoted Pt^σ and are called *Ptyxes of type* σ. The set of morphisms from A to B in Pt^σ is denoted by I^σ(A,B).

Definition 4.1 1. *PT^{EMP} is the empty category.*

2. *PT^0 is ON; i.e., Pt^0 is On, and $I^0(x, y)$ is $I(x, y)$ the set of all strictly increasing mappings from x into y.*

3. *$PT^{\sigma\to\tau}$ is such that $Pt^{\sigma\to\tau}$ is the class of all functors from PT^σ to PT^τ preserving direct limits and pull-backs. If A, B are in $Pt^{\sigma\to\tau}$, then $I^{\sigma\to\tau}(A,B)$ is the set of all natural transformations from A to B.*

4. *$PT^{\sigma\times\tau}$ is the product of the categories PT^σ and PT^τ : $Pt^{\sigma\times\tau}$ is the class of all pairs (A,B) such that A is in Pt^σ and B is in Pt^τ; $I^{\sigma\times\tau}((A,A'),(B,B'))$ is the set of all pairs (T,T') such that T is in $I^\sigma(A,B)$ and T' is in $I^\tau(A',B')$.*

5. *$PT^{\sigma+\tau}$ is the sum (or coproduct) of the categories PT^σ and PT^τ : $Pt^{\sigma+\tau}$ consist of pairs (i,A) with $i=1$ and $A\in Pt^\sigma$ or $i=2$ and $A\in Pt^\tau$; $I^{\sigma+\tau}((i,A),(j,B))$ is void when $i\neq j$, and consists of pairs $(1,T)$ such that T is in $I^\sigma(A,B)$ if $i=j=1$ or consists of pairs $(2,T)$ such that T is in $I^\tau(A,B)$ if $i=j=2$.*

Examples:

1. Ptyxes of type 0 are ordinals.
2. Ptyxes of type $0\to 0$ are dilators [Gir81].
3. The functor Λ [Gir81] is a nice example of a Ptyx of type $(0\to 0)\to(0\to 0)$

Definition 4.2 We shall now define for some Ptyxes A,B of type σ an *embedding morphism* $E^\sigma_{AB}\in I^\sigma(A,B)$: this morphism is a generalization of the morphism E_{xy} (where $x,y\in$ On) introduced in [GV84]. The definition of E_{AB} is taken from [Gir].

1. *If $A,B\in Pt^0$, then E^0_{AB} is defined when $A\leq B$. In this case $E^0_{AB}=E_{AB}$ is defined by $E_{AB}(z)=z\ \forall z<A$.*
2. *If A , $B\in Pt^{\sigma\to\tau}$, then $E^{\sigma\to\tau}_{AB}$ is defined iff the following conditions are satisfied :*
   - *$\forall a\in Pt^\sigma$, $E^\tau_{A(a)B(a)}$ is defined.*
   - *If $T_a=E^\tau_{A(a)B(a)}$ then, T defines a natural transformation from A to B.*

   If these conditions are satisfied then, $(E^{\sigma\to\tau}_{AB})(a)=E^\tau_{A(a)B(a)}$.
3. *If $A,B\in Pt^{\sigma\times\tau}$, then $E^{\sigma\times\tau}_{AB}$ is defined iff $E^\sigma_{\pi^1_{\sigma\tau}(A)\pi^1_{\sigma\tau}(B)}, E^\tau_{\pi^2_{\sigma\tau}(A)\pi^2_{\sigma\tau}(B)}$ are defined. In this case, $E^{\sigma\times\tau}_{AB}=(E^\sigma_{\pi^1_{\sigma\tau}(A)\pi^1_{\sigma\tau}(B)}, E^\tau_{\pi^2_{\sigma\tau}(A)\pi^2_{\sigma\tau}(B)})$.*
4. *If A, $B\in Pt^{\sigma+\tau}$, then $E^{\sigma+\tau}_{AB}$ is defined iff $A=i^jA'$, $B=i^jB'$ and, if $j=1$, $E^\sigma_{A'B'}$ is defined, or, if $j=2$, $E^\tau_{A'B'}$ is defined.*
   If $j=1$ $E^{\sigma+\tau}_{AB}=(1,E^\sigma_{A'B'})$.
   If $j=2$ $E^{\sigma+\tau}_{AB}=(2,E^\tau_{A'B'})$.

We write E^σ_A for E^σ_{AA}.

Because of the presence of sup-terms we need a notion of partial order (for every finite type σ) $\leq^\sigma$ on Pt^σ and on morphisms in order to introduce sup's in the model. This *order on Ptyxes* is characterized by:

1. $A \leq^\sigma B$ iff an "embedding" morphism $E^\sigma_{AB} \in I^\sigma(A, B)$ is well defined.
2. $\leq^0$ is the usual order on ordinals, and for higher types we have:

   If $A \leq^{\sigma\to\tau} B$ then $\forall C \in Pt^\sigma : AC \leq^\tau BC$.
   (The converse does not hold in general).

   $$(A, A') \leq^{\sigma\times\tau} (B, B') \quad \text{iff} \quad A \leq^\sigma B \text{ and } A' \leq^\tau B'$$

   $$(i, A) \leq^{\sigma+\tau} (j, B) \quad \text{iff} \quad i = j = 1 \text{ and } A \leq^\sigma B \quad \text{or} \quad i = j = 2 \text{ and } A \leq^\tau B$$

We have also defined an *ordering on morphisms*. We order the sets $I^\sigma(A, B)$ as follows :

1. If $f, g \in I^0(A, B)$, then $f \leq g$ iff $\forall z \in A \; f(z) \leq g(z)$.
2. If $T, U \in I^{\sigma\to\tau}(A, B)$, then $T \leq U$ iff $T(a) \leq U(a) \; \forall a \in Pt^\sigma$.
3. If $T, U \in I^{\sigma\times\tau}(A, B)$ and $T = (T', T'')$, $U = (U', U'')$ then $T \leq U$ iff $T' \leq U', T'' \leq U''$.
4. If $T, U \in I^{\sigma+\tau}(A, B)$ and $T = (j, T')$, $U = (j, U')$, $j = 1, 2$ then $T \leq U$ iff $T' \leq U'$.

Thus, we can define $\sup_{i\in I} A_i$ (*the sup on Ptyxes*)in a natural way as the direct limit of the directed system $(A_i, E^\sigma_{A_iA_j})_{i,j\in I}$ provided that for all $i, j \in I : i \leq j$ then $A_i \leq^\sigma A_j$. This notion of sup satisfies:

$$\forall\tau \neq EMP, \; if \; A = \sup_{i\in I} A_i \quad \text{in} \quad PT^{\sigma\to\tau} \quad \text{then} \quad \forall A' \in Pt^\sigma : AA' = \sup_{i\in I} A_i A'$$

$$\forall\sigma, \tau \neq EMP, \quad (A, B) = \sup_{i\in I}(A_i, B_i) \quad \text{in} \quad PT^{\sigma\times\tau} \quad \text{iff}$$
$$A = \sup_{i\in I} A_i \quad \text{in} \quad PT^\sigma \quad \text{and } B = \sup_{i\in I} B_i \quad \text{in} \quad PT^\tau$$

$$\forall\sigma \neq EMP, \; (1, A) = \sup_{l\in L}(1, A_l) \quad \text{in} \quad PT^{\sigma+\tau} \quad \text{iff} \quad A = \sup_{l\in L} A_l \quad \text{in} \quad PT^\sigma$$

$$\forall\tau \neq EMP, \; (2, A) = \sup_{l\in L}(2, A_l) \quad \text{in} \quad PT^{\sigma+\tau} \quad \text{iff} \quad A = \sup_{l\in L} A_l \quad \text{in} \quad PT^\tau$$

And we can define (*the sup on morphism*) $\sup_{i\in I} F_i$ as the direct limit of the system of morphisms (F_i) (with associated function φ) when $F_i \in I^\sigma(A_i, B_{\varphi(i)})$ is such that $F_i \circ E^\sigma_{A_iA_j} = E^\sigma_{B_iB_j} \; \forall i, j \in I$ and $i \leq j$.

We are interested in seeing that the Ptyxes form a canonical model of our typed λ-calculus.

Definition 4.3 Let be $MPT =< \{PT^\sigma\}, \odot_{\sigma\tau}, \otimes_{\sigma\tau}, \pi^j_{\sigma\tau}, i^j_{\sigma\tau} >$ *for* σ, τ *finite types and* $j = 1, 2$ where

$\odot_{\sigma\tau}$ *is the application of Ptyxes and of morphisms of corresponding types (we write A(B) for A* $\odot_{\sigma\tau}$ *B for A,B Ptyxes or morphisms).*

$\otimes_{\sigma\tau}$ *is the pairing of Ptyxes and of morphism of corresponding types (we write (A,B) for A* $\otimes_{\sigma\tau}$ *B for A,B Ptyxes or morphisms).*

$\pi^j_{\sigma\tau}$ *is the unpairing of Ptyxes and of morphisms of corresponding types* ($j = 1, 2$).

$i^j_{\sigma\tau}$ *are the injections of Ptyxes and of morphisms of corresponding types (we write (1,A) for* $i^1_{\sigma\tau}$ *and (2,A) for* $i^2_{\sigma\tau}$, *for A a Ptyx or a morphism).*

Thus, MPT is a type structure.

Definition 4.4 The Λ-OOS is the pair $< MPT, * >$ where M_0 *is ON and* $*$ is a map associating to each term t $\in$ λ-ND_τ and every sequence $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$ of different variables containing the free variables of t a multifunctor which preserves pull-back and direct limit $t^* : PT^{\sigma_1} \times \ldots \times PT^{\sigma_n} \longrightarrow PT^\tau$,more precisely, assume that $t(a_1^{\sigma_1}, \ldots, a_n^{\sigma_n})$ is a λ-term of type τ whose only free variables are contained in the sequence $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$ then we define for all Ptyxes $A_1, \ldots, A_n$ of respective types $\sigma_1, \ldots, \sigma_n$ a Ptyx $t^*(A_1, \ldots, A_n)$ *of type* τ*;* and for every sequence of morphisms $F_1 \in I^{\sigma_1}(A_1, A_1'), \ldots, F_n \in I^{\sigma_n}(A_n, A_n')$ we define a morphism $t^*(F_1, \ldots, F_n) \in I^\sigma(t^*(A_1, \ldots, A_n), t^*(A_1', \ldots, A_n'))$.

Then, beyond the requeriments of definition 3.3 we have to check that :

1. $t^*(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n}) = E^\tau_{t^*(A_1,\ldots,A_n)}$
2. $t^*(F_1, \ldots, F_n) \circ t^*(F_1', \ldots, F_n') = t^*(F_1 \circ F_1', \ldots, F_n \circ F_n')$ (eq)
3. $t^*(F_1, \ldots, F_n)\ \&\ t^*(F_1', \ldots, F_n') = t^*(F_1 \& F_1', \ldots, F_n \& F_n')$
4. If $(A_1, F_{1_i}) = \lim_{i\in I}(A_{1_i}, F_{1_{ij}}), \ldots, (A_n, F_{n_i}) = \lim_{i\in I}(A_{n_i}, F_{n_{ij}})$ *then* $(t^*(A_1, \ldots, A_n), t^*(F_{1_i}, \ldots, F_{n_i})) = \lim_{i\in I}(t^*(A_{1_i}, \ldots, A_{n_i}), t^*(F_{1_{ij}}, \ldots, F_{n_{ij}}))$

Where $\circ$ is the composition of morphisms, $F_1 \& F_2$ is a pull-back of F_1 and F_2;and $(x, F_i) = \lim_{i\in I}(x_i, F_{i_j})$ is the direct limit (or colimit) of the direct system (or diagram) of (x_i, F_{i_j}).

1 and 2 guarantee that t^* is a multifunctor, 3 guarantees that the functor preserves pull-backs and 4 that it preserves direct limit. The requirement of preserving direct limit, as the condition the continuity of Dana Scott's model and the condition about limit in coherence spaces of Girard , corresponds to the existence of one denotation (each Ptyx can be represented as a direct limit of finite dimensional Ptyxes : those determined by finitary data). And preservation of pull-back, as the stability condition of Girard's model of Coherence Spaces, corresponds to the unicity of this denotation.

Now, we defined these multifunctors t^* by induction on the complexity of the term t as in [Gir].

Definition 4.5 1.variables $(a^\sigma)^*(A) = A$ *and* $(a^\sigma)^*(F) = F \quad \forall\sigma \neq EMP$

Thus, the condition 3.3.2.b is satisfied and for conditions (eq) we have:

1. $(a^\sigma)^*(E^\sigma_A) \stackrel{\text{def}}{=} E^\sigma_A \stackrel{\text{def}}{=} E^\sigma_{(a^\sigma)^*(A)}$
2. $(a^\sigma)^*(F) \circ (a^\sigma)^*(F') \stackrel{\text{def}}{=} F \circ F' \stackrel{\text{def}}{=} (a^\sigma)^*(F \circ F')$
3. *As above.*
4. $((a^\sigma)^*(A_1), (a^\sigma)^*(F_{1_i})) \stackrel{\text{def}}{=}$
   $(A_1, F_{1_i}) \stackrel{\text{hip}}{=}$
   $\lim_{i\in I}(A_{1_i}, F_{1_{ij}}) \stackrel{\text{def}}{=}$
   $\lim_{i\in I}((a^\sigma)^*(A_{1_i}), (a^\sigma)^*(F_{1_{ij}}))$

2.constants $(0)^*$ *is the Ptyx 0 of type 0 (the ordinal 0).*

$(S)^*$ *is the Ptyx Id + 1 of type* $0 \to 0$ *(dilator) [Gir81]*

$(+)^*$ *is the functor sum (bilator) of type* $0 \to 0 \to 0$ *[Gir81]*

$(R_\sigma)^*$ *is define in [Gir] and [Pap85].*

3.app-terms *We want to defined* $(tu)^*$, *if* $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$ *are the free variables of tu.*

1. *Assume* t^* *and* u^* *are defined ,enjoy (eq) and t is not a case or an empty-term. Then, we define :*
   $(tu)^*(A_1, \ldots, A_n) = t^*(A_1, \ldots, A_n)(u^*(A_1, \ldots, A_n))$
   $(tu)^*(F_1, \ldots, F_n) = t^*(F_1, \ldots, F_n)(u^*(F_1, \ldots, F_n))$
   The conditions 3.3.2.c are satisfied because $\odot_{\sigma\tau}$ *is the application of Ptyxes and the application of morphisms*[3] *. And for (eq) we have :*
   (a) $(tu)^{\tau *}(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n}) \stackrel{\text{def}}{=}$
   $t^{\sigma\to\tau *}(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n})(u^{\sigma *}(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n})) \stackrel{\text{IH}}{=}$
   $E^{\sigma\to\tau}_{t^*(A_1,\ldots,A_n)}(E^\sigma_{u^*(A_1,\ldots,A_n)}) =^4$
   $E^{\sigma\to\tau}_{t^*(A_1,\ldots,A_n)}(u^*(A_1, \ldots, A_n)) \stackrel{\text{def4.2}}{=}$
   $E^\tau_{t^*(A_1,\ldots,A_n)(u^*(A_1,\ldots,A_n))} \stackrel{\text{def}}{=}$
   $E^\tau_{(tu)^*(A_1,\ldots,A_n)}$
   (b) $(tu)^*(F_1, \ldots, F_n) \circ (tu)^*(F'_1, \ldots, F'_n) \stackrel{\text{def}}{=}$
   $t^*(F_1, \ldots, F_n)(u^*(F_1, \ldots, F_n)) \circ t^*(F'_1, \ldots, F'_n)(u^*(F'_1, \ldots, F'_n)) =^5$
   $t^*(F_1, \ldots, F_n) \circ t^*(F'_1, \ldots, F'_n)(u^*(F_1, \ldots, F_n) \circ u^*(F'_1, \ldots, F'_n)) \stackrel{\text{IH}}{=}$
   $t^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)(u^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)) \stackrel{\text{def}}{=}$
   $(tu)^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)$
   (c) $(tu)^*(F_1, \ldots, F_n)$ & $(tu)^*(F'_1, \ldots, F'_n) \stackrel{\text{def}}{=}$
   $t^*(F_1, \ldots, F_n)(u^*(F_1, \ldots, F_n))$ & $t^*(F'_1, \ldots, F'_n)(u^*(F'_1, \ldots, F'_n)) \stackrel{\text{Gir12.2.18.IV}}{=}$

[3]Let $F_1, F_2 : A \longrightarrow B$ be functors and let $T : F_1 \Longrightarrow F_2$ be a natural transformation. Then given $g : a \longrightarrow b$ a morphism in A, we write T(g) for the morphism in B such that $T(g) = F_2(g) \circ T_a = T_b \circ F_1(g)$.

[4]$\forall T, T' : F_1 \Longrightarrow F_2$ Natural transformations, $F_1, F_2 : A \longrightarrow B$ functors, $a \in Obj(A)$, g,g',E_a morphisms in A, we have $T(E_a) = E_{F_1(a)} \circ T(a) = T(a) \circ E_{F_2(a)} = T(a)$, and also $T(g) \circ T'(g') = T \circ T'(g \circ g')$.

$t^*(F_1,\ldots,F_n)\ \&\ t^*(F'_1,\ldots,F'_n)(u^*(F_1,\ldots,F_n)\ \&\ u^*(F'_1,\ldots,F'_n)) \stackrel{IH}{=}$
$t^*(F_1\ \&\ F'_1,\ldots,F_n\ \&\ F'_n)(u^*(F_1\ \&\ F'_1,\ldots,F_n\ \&\ F'_n)) \stackrel{def}{=}$
$(tu)^*(F_1\ \&\ F'_1,\ldots,F_n\ \&\ F'_n)$

(d) $((tu)^*(A_1,\ldots A_n),(tu)^*(F_{1_i},\ldots,F_{n_i})) \stackrel{def}{=}$
$(t^*(A_1,\ldots,A_n)(u^*(A_1,\ldots,A_n)), t^*(F_{1_i},\ldots,F_{n_i})(u^*(F_{1_i},\ldots,F_{n_i}))) \stackrel{Gir12.2.17.IV}{=}$
$\lim_{i\in I}(t^*(A_{1_i},\ldots,A_{n_i})(u^*(A_{1_i},\ldots,A_{n_i})), t^*(F_{1_{ij}},\ldots,F_{n_{ij}})(u^*(F_{1_{ij}},\ldots,F_{1_{ij}}))) \stackrel{def}{=}$

$\lim_{i\in I}((tu)^*(A_{1_i},\ldots,A_{n_i}),(tu)^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

2. *If t is an empty-term, then we have:*
$((\varepsilon^{\sigma\to\tau}t^{EMP})v)^*(A_1,\ldots,A_n) = (\varepsilon^{\tau}t^{EMP})^*(A_1,\ldots,A_n)$
$((\varepsilon^{\sigma\to\tau}t^{EMP})v)^*(F_1,\ldots,F_n) = (\varepsilon^{\tau}t^{EMP})^*(F_1,\ldots,F_n)$
This satisfies 3.3.2.c and for the properties (eq) we don't have to check anything because t^* for t of type EMP is undefined.

3. *If t is a case-term, then we have:*
$((\oplus(a,b)(r^{\sigma+\tau},t^{\alpha\to\delta},s^{\alpha\to\delta}))v^{\alpha})^*(A_1,\ldots,A_n) = (\oplus(a,b)(r,tv,sv))^*(A_1,\ldots,A_n)$
$((\oplus(a,b)(r^{\sigma+\tau},t^{\alpha\to\delta},s^{\alpha\to\delta}))v^{\alpha})^*(F_1,\ldots,F_n) = (\oplus(a,b)(r,tv,sv))^*(F_1,\ldots,F_n)$
This satisfies 3.3.2.c and for the properties (eq) we have:

(a) $((\oplus(a,b)(r,s,t))v)^*(E_{A_1},\ldots,E_{A_n}) \stackrel{def}{=} (\oplus(a,b)(r,tv,sv))^*(E_{A_1},\ldots,E_{A_n})$ (1)
Here we have 4 cases to consider
Case 1 *If $r = i^1u$ idem item for case-terms replacing tv by t and sv by s.*
Case 2 *If $r = i^2u$ idem case 1.*
Case 3 *If $r = \varepsilon^{\sigma+\tau}u^{EMP}$. The equation 1 is equal to:*
$(\varepsilon^{\delta}u)^*(E_{A_1},\ldots,E_{A_n}) \stackrel{def}{=} u^*(E_{A_1},\ldots,E_{A_n})$*, and is undefined.*
Case 4 *If $r = (\oplus(a',b')(r',t',s'))^{\sigma+\tau}$ then, we just have to iterate the rules above.*

(b) $((\oplus(a,b)(r,t,s))v)^*(F_1,\ldots,F_n) \circ ((\oplus(a,b)(r,t,s))v)^*(F'_1,\ldots,F'_n) \stackrel{def}{=}$
$(\oplus(a,b)(r,tv,sv))^*(F_1,\ldots,F_n) \circ (\oplus(a,b)(r,tv,sv))^*(F'_1,\ldots,F'_n)$ (2)
We have 4 cases to consider:
Case 1 *If $r = i^1u$ then, it is treated as case-terms replacing tv by t and sv by s.*
Case 2 *If $r = i^2u$ then, it is treated as in case 1.*
Case 3 *If $r = \varepsilon^{\delta}u^{EMP}$. The equation 2 is equal to:*
$(\varepsilon^{\delta}u^{EMP})^*(F_1,\ldots,F_n) \circ (\varepsilon^{\delta}u^{EMP})^*(F'_1,\ldots,F'_n) \stackrel{def}{=}$
$u^*(F_1,\ldots,F_n) \circ u^*(F'_1,\ldots,F'_n)$ *,and is undefined.*
Case 4 *If $r = (\oplus(a',b')(r',t',s'))$ then, we just have to iterate the rules above*

(c) As for property 2.

(d) $(((\oplus(a,b)(r,t,s))v)^*(A_1,\ldots,A_n),((\oplus(a,b)(r,t,s))v)^*(F_{1_i},\ldots,F_{n_i})) \stackrel{def}{=}$
$((\oplus(a,b)(r,tv,sv))^*(A_1,\ldots,A_n),(\oplus(a,b)(r,tv,sv))^*(F_{1_i},\ldots,F_{n_i}))$ (3)
We have 4 cases to consider:

Case 1 *If* $r = i^1 u$ *then, it is treated as case-terms replacing tv by t and sv by s.*

Case 2 *If* $r = i^2 u$ *is analogous to case 1.*

Case 3 *If* $r = \varepsilon^\delta u^{EMP}$. *The equation 3 is equal to:*
$((\varepsilon^\delta u)^*(A_1,\dots,A_n),(\varepsilon^\delta u)^*(F_{1_i},\dots,F_{n_i})) \stackrel{\text{def}}{=}$
$(u^*(A_1,\dots,A_n),u^*(F_{1_i},\dots,F_{n_i}))$, *and is undefined*

Case 4 *If* $r = (\oplus(a',b')(r',t's'))$*then, we just have to iterate the rules above.*

4.λ-terms *If* t^* *has been defined and it enjoys the properties (eq), then we define*
$(\lambda a^\sigma t)^*(A_1,\dots,A_n)(A_{n+1}) = t^*(A_1,\dots,A_n,A_{n+1})$
$(\lambda a^\sigma t)^*(A_1,\dots,A_n)(F_{n+1}) = t^*(E_{A_1},\dots,E_{A_n},F_{n+1})$
$(\lambda a^\sigma t)^*(F_1,\dots,F_n)(A_{n+1}) = t^*(F_1,\dots,F_n,E_{A_{n+1}})$[5]
This satisfies the first condition of 3.3.2.f and we have to check the second condition:
If $\forall A \in Pt^\sigma$, $\forall F \in I^\sigma(A,A')$ *we have*
$t^*(A_1,\dots,A_n,A) = s^*(A_1,\dots,A_n,A)$ *and*
$t^*(A_1,\dots,A_n,F) = s^*(A_1,\dots,A_n,F)$, *then by definition,*
$\forall A\ Pt^\sigma\ (\lambda a.t)^*(A_1,\dots,A_n)(A) = (\lambda a.s)^*(A_1,\dots,A_n)(A)$*and*
$\forall F \in I^\sigma(A,A')\ (\lambda a.t)^*(A_1,\dots,A_n)(F) = (\lambda a.s)^*(A_1,\dots,A_n)(F)$ *.Thus*
$(\lambda a.t)^*(A_1,\dots,A_n) = (\lambda a.s)^*(A_1,\dots,A_n)$
And we also have:
If $\forall A \in Pt^\sigma$ *we have* $t^*(F_1,\dots,F_n,A) = s^*(F_1,\dots,F_n,A)$
then, by definition $\forall A \in Pt^\sigma\ (\lambda a.t)^*(F_1,\dots,F_n)(A) = (\lambda a.s)^*(F_1,\dots,F_n)(A)$
Thus $(\lambda a.t)^*(F_1,\dots,F_n) = (\lambda a.s)^*(F_1,\dots,F_n)$
And for the properties (eq) we have:

1. $(\lambda a^\sigma t^\tau)^*(E_{A_1},\dots,E_{A_n})(A_{n+1}) \stackrel{\text{def}}{=}$
$t^*(E_{A_1},\dots,E_{A_n},E_{A_{n+1}}) \stackrel{\text{IH}}{=}$
$E^\tau_{t^*(A_1,\dots,A_n,A_{n+1})} \stackrel{\text{def}}{=}$
$E^\tau_{(\lambda at)(A_1,\dots,A_n)(A_{n+1})} \stackrel{\text{def4.2}}{=}$
$E^{\sigma\to\tau}_{(\lambda at)^*(A_1,\dots,A_n)}(A_{n+1})\quad \forall A_{n+1} \in Pt^\sigma$. *Thus, we have*
$(\lambda at)^*(E_{a_1},\dots,E_{A_n}) = E^{\sigma\to\tau}_{(\lambda at)^*(A_1,\dots,A_n)}$

2. $((\lambda at)^*(F_1,\dots,F_n)\circ(\lambda at)^*(F_1,\dots,F_n))(A_{n+1}) \stackrel{\text{compNT}}{=}$
$(\lambda a^\sigma t)^*(F_1,\dots,F_n)(A_{n+1})\circ(\lambda a^\sigma t)^*(F'_1,\dots,F'_n)(A_{n+1}) \stackrel{\text{def}}{=}$
$t^*(F_1,\dots,F_n,E_{A_{n+1}})\circ t^*(F'_1,\dots,F'_n,E_{A_{n+1}}) \stackrel{\text{IH}}{=}$
$t^*(F_1\circ F'_1,\dots,F_n\circ F'_n,E_{A_{n+1}}\circ E_{A_{n+1}}) =$
$t^*(F_1\circ F'_1,\dots,F_n\circ F'_n,E_{A_{n+1}}) \stackrel{\text{def}}{=}$
$(\lambda a^\sigma t)^\sigma(F_1\circ F'_1,\dots,F_n\circ F'_n)(A_{n+1})\ \forall A_{n+1}\in Pt^\sigma$. *Thus, we have*
$(\lambda at)^*(F_1,\dots,F_n)\circ(\lambda at)^*(F'_1,\dots,F'_n) = (\lambda at)^*(F_1\circ F'_1,\dots,F_n\circ F'_n)$

[5] We don't need to define $(\lambda at)^*(F_1,\dots,F_n)(F_{n+1}) = t^*(F_1,\dots,F_n,F_{n+1})$ because this is a consequence of these three equalities.

3. $((\lambda at)^*(F_1,\ldots,F_n)\ \&\ (\lambda at)^*(F'_1,\ldots,F'_n))(A_{n+1}) \overset{\text{Gir12.2.18}}{=}$
$(\lambda a^\sigma t)^*(F_1,\ldots,F_n)(A_{n+1})\ \&\ (\lambda a^\sigma t)^*(F'_1,\ldots,F'_n)(A_{n+1}) \overset{\text{def}}{=}$
$t^*(F_1,\ldots,F_n,E_{A_{n+1}})\ \&\ t^*(F'_1,\ldots,F'_n,E_{A_{n+1}}) \overset{\text{IH}}{=}$
$t^*(F_1\&F'_1,\ldots,F_n\&F'_n,E_{A_{n+1}}\&E_{A_{n+1}}) =$
$t^*(F_1\&F'_1,\ldots,F_n\&F'_n,E_{A_{n+1}}) \overset{\text{def}}{=}$
$(\lambda at)^*(F_1\&F'_1,\ldots,F_n\&F'_n)(A_{n+1}) \quad \forall A_{n+1}\in Pt^\sigma$

4. $((\lambda a^\sigma t)^*(A_1,\ldots,A_n)(A_{n+1}),(\lambda a^\sigma t)^*(F_{1_i},\ldots,F_{n_i})(A_{n+1})) \overset{\text{def}}{=}$
$(t^*(A_1,\ldots,A_n,A_{n+1}),t^*(F_{1_i},\ldots,F_{n_i},E_{A_{n+1}})) \overset{\text{IH}}{=}$
$\lim_{i\in I}(t^*(A_{1_i},\ldots,A_{n_i},A_{n+1}),t^*(F_{1_{ij}},\ldots,F_{n_{ij}},E_{A_{n+1}})) \overset{\text{def}}{=}$
$\lim_{i\in I}((\lambda at)^*(A_{1_i},\ldots,A_{n_i})(A_{n+1}),(\lambda at)^*(F_{1_{ij}},\ldots,F_{n_{ij}})(A_{n+1})) \quad \forall A_{n+1}\in Pt^\sigma.$
By [Gir] 12.2.17 this is equivalent to
$((\lambda at)^*(A_1,\ldots,A_n),(\lambda at)^*(F_{1_i},\ldots,F_{n_i})) =$
$\lim((\lambda at)^*(A_{1_i},\ldots,A_{n_i}),(\lambda at)^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

5.$\otimes$-terms *Assume that t^* and u^* have been defined and enjoy the properties (eq). Then, by 3.3.2.1*
$(t\otimes u)^*(A_1,\ldots,A_n) = (t^*(A_1,\ldots,A_n),u^*(A_1,\ldots,A_n))$
$(t\otimes u)^*(F_1,\ldots,F_n) = (t^*(F_1,\ldots,F_n),u^*(F_1,\ldots,F_n))$
We now check the properties (eq) for $(t\otimes u)^*$

1. $(t\otimes u)^*(E_{A_1},\ldots,E_{A_n}) \overset{\text{def}}{=}$
$(t^*(E_{A_1},\ldots,E_{A_n}),u^*(E_{A_1},\ldots,E_{A_n})) \overset{\text{IH}}{=}$
$(E^\sigma_{t^*(A_1,\ldots,A_n)},E^\tau_{u^*(A_1,\ldots,A_n)}) \overset{\text{def4.2}}{=}$
$E^{\sigma\times\tau}_{(t^*(A_1,\ldots,A_n),u^*(A_1,\ldots,A_n))} \overset{\text{def}}{=}$
$E^{\sigma\times\tau}_{(t\otimes u)^*(A_1,\ldots,A_n)}$

2. $(t\otimes u)^*(F_1,\ldots,F_n)\circ(t\otimes u)^*(F'_1,\ldots,F'_n) \overset{\text{def}}{=}$
$(t^*(F_1,\ldots,F_n),u^*(F_1,\ldots,F_n))\circ(t^*(F'_1,\ldots,F'_n),u^*(F'_1,\ldots,F'_n)) \overset{\text{prodcat}}{=}$
$(t^*(F_1,\ldots,F_n)\circ t^*(F'_1,\ldots,F'_n),u^*(F_1,\ldots,F_n)\circ u^*(F'_1,\ldots,F'_n)) \overset{\text{IH}}{=}$
$(t^*(F_1\circ F'_1,\ldots,F_n\circ F'_n),u^*(F_1\circ F'_1,\ldots,F_n\circ F'_n)) \overset{\text{def}}{=}$
$(tu)^*(F_1\circ F'_1,\ldots,F_n\circ F'_n)$

3. *As above.*

4. $((t\otimes u)^*(A_1,\ldots,A_n),(t\otimes u)^*(F_{1_i},\ldots,F_{n_i})) \overset{\text{def}}{=}$
$((t^*(A_1,\ldots,A_n),u^*(A_1,\ldots,A_n)),(t^*(F_{1_i},\ldots,F_{n_i}),u^*(F_{1_i},\ldots,F_{n_i}))) \overset{\text{catprod-IH}}{=}$
$\lim_{i\in I}((t^*(A_{1_i},\ldots,A_{n_i}),u^*(A_{1_i},\ldots,A_{n_i})),(t^*(F_{1_{ij}},\ldots,F_{n_{ij}}))) \overset{\text{def}}{=}$
$\lim_{i\in I}((tu)^*(A_{1_i},\ldots,A_{n_i}),(tu)^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

6.π-terms *1. Assume that t^* has been defined, and it is not a case or an empty-term. Then we define:*
$(\pi^1 t)^*(A_1,\ldots,A_n) = \pi^1_{\sigma\tau}(t^*(A_1,\ldots,A_n))$
$(\pi^1 t)^*(F_1,\ldots,F_n) = \pi^1_{\sigma\tau}(t^*(F_1,\ldots,F_n))$

$(\pi^2 t)^*(A_1,\ldots,A_n) = \pi^2_{\sigma\tau}(t^*(A_1,\ldots,A_n))$
$(\pi^2 t)^*(F_1,\ldots,F_n) = \pi^2_{\sigma\tau}(t^*(F_1,\ldots,F_n))$
3.3.2.d is satisfied, and for (eq), we have:

(a) $(\pi^1 t)^{*\sigma}(E^{\sigma_1}_{A_1},\ldots,E^{\sigma_n}_{A_n}) \stackrel{\text{def}}{=}$
$\pi^1_{\sigma\tau}(t^*(E^{\sigma_1}_{A_1},\ldots,E^{\sigma_n}_{A_n})) \stackrel{\text{III}}{=}$
$\pi^1_{\sigma\tau}(E^{\sigma\times\tau}_{t^*(A_1,\ldots,A_n)}) \stackrel{\text{def4.2}}{=}$
$E^{\sigma}_{\pi^1_{\sigma\tau}(t^*(A_1,\ldots,A_n))} \stackrel{\text{def}}{=}$
$E^{\sigma}_{(\pi^1 t)^*(A_1,\ldots,A_n)}$

(b) $(\pi^1 t)^*(F_1,\ldots,F_n) \circ (\pi^1 t)^*(F'_1,\ldots,F'_n) \stackrel{\text{def}}{=}$
$\pi^1_{\sigma\tau}(t^*(F_1,\ldots,F_1)) \circ \pi^1_{\sigma\tau}(t^*(F'_1,\ldots,F'_n)) =$
$\pi^1_{\sigma\tau}(t^*(F_1,\ldots,F_n) \circ t^*(F'_1,\ldots,F'_n)) \stackrel{\text{III}}{=}$
$\pi^1_{\sigma\tau}(t^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n)) \stackrel{\text{def}}{=}$
$(\pi^1 t)^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n)$

(c) As above.

(d) $((\pi^1 t)^*(A_1,\ldots,A_n),(\pi^1 t)^*(F_{1_i},\ldots,F_{n_i})) \stackrel{\text{def}}{=}$
$(\pi^1_{\sigma\tau}(t^*(A_1,\ldots,A_n)),\pi^1_{\sigma\tau}(t^*(F_{1_i},\ldots,F_{n_i}))) \stackrel{\text{III}}{=}$
$\lim_{i\in I}(\pi^1_{\sigma\tau}(t^*(A_{1_i},\ldots,A_{n_i})),\pi^1_{\sigma\tau}(t^*(F_{1_{ij}},\ldots,F_{n_{ij}}))) \stackrel{\text{def}}{=}$
$\lim_{i\in I}((\pi^1 t)^*(A_{1_i},\ldots,A_{n_i}),(\pi^1 t)^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

Idem for $\pi^2 t$.

2. *If t is an empty-term then, we define:*
$(\pi^1(\varepsilon^{\sigma\times\tau}t))^*(A_1,\ldots,A_n) = (\varepsilon^{\sigma}t)^*(A_1,\ldots,A_n)$
$(\pi^1(\varepsilon^{\sigma\times\tau}t))^*(F_1,\ldots,F_n) = (\varepsilon^{\sigma}t)^*(F_1,\ldots,F_n)$
$(\pi^2(\varepsilon^{\sigma\times\tau}t))^*(A_1,\ldots,A_n) = (\varepsilon^{\tau}t)^*(A_1,\ldots,A_n)$
$(\pi^2(\varepsilon^{\sigma\times\tau}t))^*(F_1,\ldots,F_n) = (\varepsilon^{\tau}t)^*(F_1,\ldots,F_n)$
This satisfies 3.3.2.d and we don't have to check (eq) because t^* is undefined for t of type EMP.

3. *If t is a case-term, then we define:*
$(\pi^i(\oplus(a,b)(r,t,s)))^*(A_1,\ldots,A_n) = (\oplus(a,b)(r,\pi^i t,\pi^i s))^*(A_1,\ldots,A_n)$
$(\pi^i(\oplus(a,b)(r,t,s)))^*(F_1,\ldots,F_n) = (\oplus(a,b)(r,\pi^i t,\pi^i s))^*(F_1,\ldots,F_n)$
This satisfies 3.3.4 and for the properties (eq) we have:

(a) $(\pi^i(\oplus(a,b)(r,t,s)))^*(E_{A_1},\ldots,E_{A_n}) \stackrel{\text{def}}{=}$
$(\oplus(a,b)(r,\pi^i t,\pi^i s))^*(E_{A_1},\ldots,E_{A_n})$. *We have several cases to examine :*

Case 1 *If $r = i^1 u$ then, it is treated as case-terms replacing $\pi^i t$ by t and $\pi^i s$ by s.*

Case 2 *If $r = i^2 u$ then, it is treated as in case 1.*

Case 3 *If $r = \varepsilon^{\sigma+\tau}u$ then, it is undefined.*

Case 4 *If $r = (\oplus(a',b')(r',t's'))$, then, we just have to iterate the rules above.*

(b) *Analogous to property 1.*

(c) *Analogous to property 1.*

(d) *Analogous to property 1.*

7.sup-terms *Assume t^* has been defined and enjoys the properties (eq). Then, we define*

$(\sup_{x<\alpha} t)^*(A_1, \ldots A_n) = t^*(A_1, \ldots, A_n, \alpha)$

$(\sup_{x<\alpha} t)^*(F_1, \ldots, F_n) = t^*(F_1, \ldots, F_n, E^0_\alpha)$

This satisfies 3.3.2.i because for α limit ordinal we have

$\sup_{\beta<\alpha} t^*(A_1, \ldots, A_n, \beta) = t^*(A_1, \ldots, A_n, \alpha)$

As to the properties (eq), we have:

1. $(\sup_{x<\alpha} t)^*(E_{A_1}, \ldots, E_{A_n}) \stackrel{\text{def}}{=}$
$t^*(E_{A_1}, \ldots, E_{A_n}, E^0_\alpha) \stackrel{\text{IH}}{=}$
$E^0_{t^*(A_1,\ldots,A_n,\alpha)} \stackrel{\text{def}}{=}$
$E^0_{(\sup_{x<\alpha} t)^*(A_1,\ldots,A_n)}$

2. $(\sup_{x<\alpha} t)^*(F_1, \ldots, F_n) \circ (\sup_{x<\alpha} t)^*(F'_1, \ldots, F'_n) \stackrel{\text{def}}{=}$
$t^*(F_1, \ldots, F_n, E^0_\alpha) \circ t^*(F'_1, \ldots, F'_n, E^0_\alpha) \stackrel{\text{IH}}{=}$
$t^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n, E^0_\alpha \circ E^0_\alpha) =$
$t^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n, E^0_\alpha) \stackrel{\text{def}}{=}$
$(\sup_{x<\alpha})^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)$

3. *As above*

4. $((\sup_{x<\alpha} t)^*(A_1, \ldots, A_n), (\sup_{x<\alpha} t)^*(F_{1_i}, \ldots, F_{n_i})) \stackrel{\text{def}}{=}$
$(t^*(A_1, \ldots, A_n, \alpha), t^*(F_{1_i}, \ldots, F_{n_i}, E^0_\alpha)) \stackrel{\text{IH}}{=}$
$\lim_{i\in I}(t^*(A_{1_i}, \ldots, A_{n_i}, \alpha), t^*(F_{1_{ij}}, \ldots, F_{n_{ij}}, E^0_\alpha)) \stackrel{\text{def}}{=}$
$\lim_{i\in I}((\sup_{x<\alpha} t)^*(A_{1_i}, A_{n_i}), (\sup_{i\in J} t)^*(F_{1_{ij}}, \ldots, F_{n_{ij}}))$

8.⊕-terms *Assume that t^* has been defined and enjoys the properties (eq). Then we define*

$(i^1 t^\sigma)^*(A_1, \ldots, A_n) = (1, t^*(A_1, \ldots, A_n))$

$(i^1 t^\sigma)^*(F_1, \ldots, F_n) = (1, t^*(F_1, \ldots, F_n))$

$(i^2 t^\tau)^*(A_1, \ldots, A_n) = (2, t^*(A_1, \ldots, A_n))$

$(i^2 t^\tau)^*(F_1, \ldots, F_n) = (2, t^*(F_1, \ldots, F_n))$

3.3.2.j is satisfied because $i^j_{\sigma\tau}$ is the injection in the sum category. We have:

1. $(i^1 t^\sigma)^*(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n}) \stackrel{\text{def}}{=}$
$(1, t^*(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n})) \stackrel{\text{IH}}{=}$
$(1, E^\sigma_{t^*(A_1,\ldots,A_n)}) \stackrel{\text{def4.2}}{=}$
$E^{\sigma+\tau}_{(i^1 t)^*(A_1,\ldots,A_n)}$

2. $(i^1 t^\sigma)^*(F_1, \ldots, F_n) \circ (i^1 t^\sigma)^*(F'_1, \ldots, F'_n) \stackrel{\text{def}}{=}$
$(1, t^*(F_1, \ldots, F_n)) \circ (1, t^*(F'_1, \ldots, F'_n)) \stackrel{\text{sumcat}}{=}$
$(1, t^*(F_1, \ldots, F_n) \circ t^*(F'_1, \ldots, F'_n)) \stackrel{\text{IH}}{=}$

$(1, t^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)) \stackrel{\text{def}}{=}$
$(i^1 t)^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)$

3. As above.

4. $((i^1 t)^*(A_1, \ldots, A_n), (i^1 t)^*(F_{1_i}, \ldots, F_{n_i})) \stackrel{\text{def}}{=}$
$((1, t^*(A_1, \ldots, A_n)), (1, t^*(F_{1_i}, \ldots, F_{n_i}))) \stackrel{\text{catsum-III}}{=}$
$\lim_{i \in I}((1, t^*(A_{1_i}, \ldots, A_{n_i})), (1, t^*(F_{1_{ij}}, \ldots, F_{n_{ij}}))) \stackrel{\text{def}}{=}$
$\lim_{i \in I}((i^1 t)^*(A_{1_i}, \ldots, A_{n_i}), (i^1 t)^*(F_{1_{ij}}, \ldots, F_{n_{ij}})))$

Idem for i^2.

9.case-terms *1. Assume that r is not an empty or case-term. Then, we define*

$$(\oplus(a,b)(r^{\sigma+\tau}, t^\alpha, s^\alpha))^{\alpha *}(A_1, \ldots, A_n) = \begin{cases} t^*(A_1, \ldots, A_n, u^*(A_1, \ldots, A_n)) & \text{if } r = i^1 u^\sigma \\ s^*(A_1, \ldots, A_n, u^*(A_1, \ldots, A_n)) & \text{if } r = i^2 u^\tau \end{cases}$$

$$(\oplus(a,b)(r^{\sigma+\tau}, t^\alpha, s^\alpha))^{\alpha *}(F_1, \ldots, F_n) = \begin{cases} t^*(F_1, \ldots, F_n, u^*(F_1, \ldots, F_n)) & \text{if } r = i^1 u^\sigma \\ s^*(F_1, \ldots, F_n, u^*(F_1, \ldots, F_n)) & \text{if } r = i^2 u^\tau \end{cases}$$

In the first case we need that t^* *and* u^* *be defined and enjoy (eq). And in the second case, we need that* s^* *and* u^* *be defined and enjoy (eq).*
We have to check that the conditions 3.3.2.k are satisfied. The first condition is satisfied by definition. The second and the third conditions are vacuous. As to the fourth condition, we can see that:
Let

- $(t[a])^*(A_1, \ldots, A_n) = (t'[a])^*(A_1, \ldots, A_n)$
- $(s[b])^*(A_1, \ldots, A_n) = (s'[b])^*(A_1, \ldots, A_n)$
- $r^*(A_1, \ldots, A_n) = r'^*(A_1, \ldots, A_n)$

Given that, $r^*(A_1, \ldots, A_n) = (j, u^*(A_1, \ldots, A_n))$ *and*
$r'^*(A_1, \ldots, A_n) = (j, u'^*(A_1, \ldots, A_n))$
then we have $j = j'$ *and* $u^*(A_1, \ldots, A_n) = u'^*(A_1, \ldots, A_n)$
Suppose $j = j' = 1$ *and suppose, for the sake of contradiction, that*
$(\oplus(a,b)(i^1 u, t, s))^*(A_1, \ldots, A_n) \neq (\oplus(a,b)(i^1 u', t', s'))^*(A_1, \ldots, A_n)$ *.Then*
$t^*(A_1, \ldots, A_n, u^*(A_1, \ldots, A_n)) \neq t'^*(A_1, \ldots, A_n, u'^*(A_1, \ldots, A_n))$
what implies that $t^*(A_1, \ldots, A_n, u^*(A_1, \ldots, A_n)) \neq t'^*(A_1, \ldots, A_n, u^*(A_1, \ldots, A_n))$
Which is a contradiction. Idem for j= j'=2 and for $F_1, \ldots, F_n$.

Let us now check the properties (eq). Let $r = i^1 u^\sigma$ *and suppose* u^* *enjoys the properties (eq).*

(a) $(\oplus(a,b)(r,t,s))^*(E^{\sigma_1}_{A_1}, \ldots, E^{\sigma_n}_{A_n}) \stackrel{\text{def}}{=}$
$t^*(E_{A_1}, \ldots, E_{A_n}, u^*(E_{A_1}, \ldots, E_{A_n})) \stackrel{\text{IH}}{=}$
$t^*(E_{A_1}, \ldots, E_{A_n}, E^\sigma_{u^*(A_1, \ldots, A_n)}) \stackrel{\text{IH}}{=}$

$E^{\alpha}_{t^*(A_1,\ldots,A_n,u^*(A_1,\ldots,A_n))} \stackrel{\text{def}}{=}$
$E^{\alpha}_{(\oplus(a,b)(r,t,s))^*(A_1,\ldots,A_n)}$

(b) $(\oplus(a,b)(r,t,s))^*(F_1,\ldots,F_n) \circ (\oplus(a,b)(r,t,s))^*(F'_1,\ldots,F'_n) \stackrel{\text{def}}{=}$
$t^*(F_1,\ldots,F_n,u^*(F_1,\ldots,F_n)) \circ t^*(F'_1,\ldots,F'_n,u^*(F'_1,\ldots,F'_n)) \stackrel{\text{IH}}{=}$
$t^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n, u^*(F_1,\ldots,F_n) \circ u^*(F_1,\ldots,F_n)) \stackrel{\text{IH}}{=}$
$t^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n, u^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n)) \stackrel{\text{def}}{=}$
$(\oplus(a,b)(r,t,s))^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n)$

(c) As above.

(d) $((\oplus(a,b)(r,t,s))^*(A_1,\ldots,A_n),(\oplus(a,b)(r,t,s))^*(F_{1_i},\ldots,F_{n_i})) \stackrel{\text{def}}{=}$
$(t^*(A_1,\ldots,A_n,u^*(A_1,\ldots,A_n)),t^*(F_{1_i},\ldots,F_{n_i},u^*(F_{1_i},\ldots,F_{n_i}))) \stackrel{\text{IH}}{=}$
$\lim_{i<I}(t^*(A_{1_i},\ldots,A_{n_i},u^*(A_{1_i},\ldots,A_{n_i})),t^*(F_{1_{ij}},\ldots,F_{n_{ij}},u^*(F_{1_{ij}},\ldots,F_{n_{ij}})))$[6] $\stackrel{\text{def}}{=}$

$\lim_{i<I}((\oplus(a,b)(r,t,s))^*(A_{1_i},\ldots,A_{n_i}),(\oplus(a,b)(r,t,s))(F_{1_{ij}},\ldots,F_{n_{ij}}))$

Idem for $r = i^2u$

2. *If r is an empty-term then, we define*
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u^{EMP},t^{\alpha},s^{\alpha}))^*(A_1,\ldots,A_n) = (\varepsilon^{\alpha}u)^*(A_1,\ldots,A_n)$
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u^{EMP},t^{\alpha},s^{\alpha}))^*(F_1,\ldots,F_n) = (\varepsilon^{\alpha}u)^*(F_1,\ldots,F_n)$
This satisfies the first condition of 3.3.2.k and for the fourth one we have: Assume that

- $(t[a])^*(A_1,\ldots,A_n) = (t'[a])^*(A_1,\ldots,A_n)$
- $(s[b])^*(A_1,\ldots,A_n) = (s'[b])^*(A_1,\ldots,A_n)$
- $(\varepsilon^{\sigma+\tau}u)^*(A_1,\ldots,A_n) = (\varepsilon^{\sigma+\tau}u')^*(A_1,\ldots,A_n)$

Then we have $u^*(A_1,\ldots,A_n) = u'^*(A_1,\ldots,A_n)$
Suppose for the sake of contradiction
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(A_1,\ldots,A_n) \neq (\oplus(a,b)(\varepsilon^{\sigma+\tau}u',t's'))^*(A_1,\ldots,A_n)$
Then $(\varepsilon^{\sigma+\tau}u)^*(A_1,\ldots,A_n) \neq (\varepsilon^{\sigma+\tau}u')^*(A_1,\ldots,A_n)$
and by definition $u^*(A_1,\ldots,A_n) \neq u'^*(A_1,\ldots,A_n)$ *This is a contradiction.*
Idem for $F_1,\ldots,F_n$.
As to the properties (eq) we don't have to check anything because u^* *is undefined.*

3. *If r is a case-term then, we just have to iterate the rules above.*

10.empty-terms 1. $(\varepsilon^{\sigma}t)^*(A_1,\ldots,A_n) = t^*(A_1,\ldots,A_n)$
$(\varepsilon^{\sigma}t)^*(F_1,\ldots,F_n) = t^*(F_1,\ldots,F_n)$
This satisfies 3.3.2.l and as to (eq) we don't have to check anything because t^* *is undefined.*

2. $(\varepsilon^{\sigma}(\oplus(a,b)(r,s,t)))^*(A_1,\ldots,A_n) = (\oplus(a,b)(r,\varepsilon^{\sigma}t,\varepsilon^{\sigma}s))^*(A_1,\ldots,A_n)$
$(\varepsilon^{\sigma}(\oplus(a,b)(r,s,t)))^*(F_1,\ldots,F_n) = (\oplus(a,b)(r,\varepsilon^{\sigma}t,\varepsilon^{\sigma}s))^*(F_1,\ldots,F_n)$

[6]This is because for IH $(u^*(A_1,\ldots,A_n),u^*(F_{1_i},\ldots,F_{n_i})) = \lim_{i<I}(u^*(A_{1_i},\ldots,A_{n_i}),u^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$.

This satisfies 3.3.12 and as to (εq) we don't have to check anything because if $r = i^j u$ then t^* and s^* are undefined, if $r = \varepsilon^{\delta+\tau} r'$ then r'^* is undefined, and if $r = (\oplus(a', b')(r', t', s'))$ then, we just have to iterate these cases. □

Thus, showing how t^* acts on objects and on morphisms the values in the model are uniquely determined [Gir]. By iteration of these rules, it is possible to define $+_\sigma^*$ in $PT^{\sigma\to\sigma\to\sigma}$ $\forall\sigma$ which corresponds exactly to the following notion of sum for the Ptyxes and morphisms of type σ [Gir]:

If $\sigma = \delta \to \tau,\ \ \tau \neq EMP$ *then,*
$(A +_{\delta\to\tau} B)(A') = A(A') +_\tau B(A')$
$(A +_{\delta\to\tau} B)(F) = A(F) +_\tau A(F)$
$(T +_{\delta\to\tau} U)(A') = T(A') +_\tau U(A')$
For $A, B \in Pt^{\delta\to\tau}$, $A' \in Pt^\delta$, $T \in I^{\delta\to\tau}(A, C),\ U \in I^{\delta\to\tau}(B, D),\ F \in I^\delta(A', A'')$
If $\sigma = \delta \times \tau,\ \ \delta, \tau \neq EMP$ *then,*
$(A, B) +_{\delta\times\tau} (A', B') = (A +_\delta A', B +_\tau B')$
$(T, U) +_{\delta\times\tau} (T', U') = (T +_\delta T', U +_\tau U')$
For $(A, B), (A', B') \in Pt^{\delta\times\tau}$ and $(T, U) \in I^{\delta\times\tau}((A, B)(C, D)),\ (T', U') \in I^{\delta\times\tau}((A', B')(C', D')$
If $\sigma = \delta + \tau$ *then*
$(1, A) +_{\delta+\tau} (1, B) = (1, A +_\delta B)$ if $\delta \neq EMP.$
$(2, A) +_{\delta+\tau} (2, B) = (2, A +_\tau B)$ if $\tau \neq EMP.$
$(1, T) +_{\delta+\tau} (1, U) = (1, T +_\delta B)$ if $\delta \neq EMP.$
$(2, T) +_{\delta+\tau} (2, U) = (2, T +_\tau U)$ if $\tau \neq EMP.$
For $(j, A), (j, B) \in Pt^{\delta+\tau}$ and $(j, T)\ \in I^{\delta+\tau}((j, A)(j, A'))$, and$(j, U)\ \in I^{\delta+\tau}((j, B)(j, B'))$
j = 1,2

Also, by iteration, we can define $\forall\sigma\ \ \sup_{x<\alpha} t^\sigma$ and this results equal to the notion of sup for Ptyxes of type σ defined in page 12.

Lemma 4.6 *If the free variables of $t^\tau \in \lambda\text{-}ND_\tau$ are among $a_1^{\sigma_1}, \dots, a_n^{\sigma_n}, x^0$; x^0 is d. var. of t, and t^* is defined then,*
$\forall\alpha \leq \beta \in On\ t^*(E_{A_1}^{\sigma_1}, \dots, E_{A_n}^{\sigma_n}, E_{\alpha\beta}^0) = E^\tau_{t^*(A_1,\dots,A_n,\alpha)t^*(A_1,\dots,A_n,\beta)}$

Proof : by induction on the term t.

1. $a^{0*}(E^0_{\alpha\beta}) = E^0_{\alpha\beta} = E^0_{a^{0*}(\alpha)a^{0*}(\beta)}$
2. *Case 1. t is not a case or an elim-term.*
   $(tu)^*(E_{A_1}, \dots, E_{A_n}, E^0_{\alpha\beta}) =$
   $t^*(E_{A_1}, \dots, E_{A_n}, E_{\alpha\beta})(u^*(E_{A_1}, \dots, E_{A_n}, E_{\alpha\beta})) =$
   $t^*(E_{A_1}, \dots, E_{A_n}, E_{\alpha\beta})(u^*(E_{A_1}, \dots, E_{A_n})) =$
   $E^{\sigma\to\tau}_{t^*(A_1,\dots,A_n,\alpha)t^*(A_1,\dots,A_n,\beta)}(E^\sigma_{u^*(A_1,\dots,A_n)u^*(A_1,\dots,A_n)}) =$

$E^{\sigma\to\tau}_{t^*(A_1,\ldots,A_n,\alpha)t^*(A_1,\ldots,A_n,\beta)}(u^*(A_1,\ldots,A_n)) =$
$E^{\tau}_{t^*(A_1,\ldots,A_n,\alpha)(u^*(A_1,\ldots,A_n))t^*(A_1,\ldots,A_n,\beta)(u^*(A_1,\ldots,A_n))} =$
$E^{\tau}_{t^*(A_1,\ldots,A_n,\alpha)(u^*(A_1,\ldots,A_n,\alpha))t^*(A_1,\ldots,A_n,\beta)(u^*(A_1,\ldots,A_n,\beta))} =$
$E^{\tau}_{(tu)^*(A_1,\ldots,A_n,\alpha)(tu)^*(A_1,\ldots,A_n,\beta)}$
Case 2. If t is an elim-term then, it is undefined.
Case 3. If $t = (\oplus(a,b)(r,v,w))$ *then, it is treated as case-terms replacing vu for t and wu for s.*

3. $(t' \otimes t'')^{\sigma\times\tau *}(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta}) =$
$(t'^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta}), t''^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta})) =$
$(E^{\sigma}_{t'^*(A_1,\ldots,A_n,\alpha)t'^*(A_1,\ldots,A_n,\beta)}, E^{\tau}_{t''^*(A_1,\ldots,A_n,\alpha)t''^*(A_1,\ldots,A_n,\beta)}) =$
$E^{\sigma\times\tau}_{((t'^*(A_1,\ldots,A_n,\alpha),t''^*(A_1,\ldots,A_n,\alpha)),(t'^*(A_1,\ldots,A_n,\beta),t''^*(A_1,\ldots,A_n,\beta)))} =$
$E^{\sigma\times\tau}_{(t'\otimes t'')^*(A_1,\ldots,A_n,\alpha)(t'\otimes t'')^*(A_1,\ldots,A_n,\beta)}$

4. *Case 1. If t is not a case or elim-term, then*
$(\pi^1 t)^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta}) =$
$\pi^1_{\sigma\tau}(t^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta})) =$
$\pi^1_{\sigma\tau}(E^{\sigma\times\tau}_{t^*(A_1,\ldots,A_n,\alpha)t^*(A_1,\ldots,A_n,\beta)}) =$
$E^{\sigma}_{\pi^1_{\sigma\tau}(t^*(A_1,\ldots,A_n,\alpha))\pi^1_{\sigma\tau}(t^*(A_1,\ldots,A_n,\beta))} =$
$E^{\sigma}_{(\pi^1 t)^*(A_1,\ldots,A_n,\alpha)(\pi^1 t)^*(A_1,\ldots,A_n,\beta)}$
Idem for π^2
Case 2. If t is an elim-term then, this case is undefined.
Case 3. If $t = (\oplus(a,b)(r,t',t''))$ *then, it is treated as case-terms replacing* $\pi^i t'$ *for t and* $\pi^i t''$ *for s.*

5. $(i^1 t^{\sigma})^{\sigma+\tau *}(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta}) =$
$(1, t^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta})) =$
$(1, E^{\sigma}_{t^*(A_1,\ldots,A_n,\alpha)t^*(A_1,\ldots,A_n,\beta)}) =$
$E^{\sigma+\tau}_{(1,t)^*(A_1,\ldots,A_n,\alpha)(1,t)^*(A_1,\ldots,A_n,\beta)}$

6. $\varepsilon^{\sigma} t$. *This case is undefined.*

7. $(\sup_{x<\delta} t)^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta}) =$
$t^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta},E_{\delta}) =$
$t^*(E_{A_1},\ldots,E_{A_n},E_{\delta},E_{\alpha\beta}) =$
$E^{0}_{t^*(A_1,\ldots,A_n,\delta,\alpha)t^*(A_1,\ldots,A_n,\delta,\beta)} =$
$E^{0}_{t^*(A_1,\ldots,A_n,\alpha,\delta)t^*(A_1,\ldots,A_n,\beta,\delta)} =$
$E^{0}_{(\sup_{x<\delta} t)(A_1,\ldots,A_n,\alpha)(\sup_{x<\delta} t)(A_1,\ldots,A_n,\beta)}$

8. $(\lambda a.t)^{\sigma\to\tau *}(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta})(d) =$
$t^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta},E^{\sigma}_{d}) =$
$t^*(E_{A_1},\ldots,E_{A_n},E_d,E_{\alpha\beta}) =$
$E^{\tau}_{t^*(A_1,\ldots,A_n,d,\alpha)t^*(A_1,\ldots,A_n,d,\beta)} =$
$E^{\tau}_{t^*(A_1,\ldots,A_n,\alpha,d)t^*(A_1,\ldots,A_n,\beta,d)} =$
$E^{\tau}_{(\lambda a.t)^*(A_1,\ldots,A_n,\alpha)(d)(\lambda a.t)^*(A_1,\ldots,A_n,\beta)(d)} =$
$E^{\sigma\to\tau}_{(\lambda a.t)^*(A_1,\ldots,A_n,\alpha)(\lambda a.t)^*(A_1,\ldots,A_n,\beta)}(d) \quad \forall d \in Pt^{\sigma}$

9. *Case 1.* $r = i^1 u$. *Then*
$(\oplus(a,b)(i^1 u, t, s))^{\delta *}(E_{A_1}, \ldots, E_{A_n}, E_{\alpha\beta}) =$
$t^*(E_{A_1}, \ldots, E_{A_n}, E_{\alpha\beta}, u^*(E_{A_1}, \ldots, E_{A_n}, E_{\alpha\beta})) =$
$t^*(E_{A_1}, \ldots, E_{A_n}, E_{\alpha\beta}, u^*(E_{A_1}, \ldots, E_{A_n})) =$
$t^*(E_{A_1}, \ldots, E_{A_n}, E_{u^*(A_1,\ldots,A_n)}, E_{\alpha\beta}) =$
$E^{\delta}_{t^*(A_1,\ldots,A_n,u^*(A_1,\ldots,A_n),\alpha)t^*(A_1,\ldots,A_n,u^*(A_1,\ldots,A_n),\beta)} =$
$E^{\delta}_{t^*(A_1,\ldots,A_n,\alpha,u^*(A_1,\ldots,A_n))t^*(A_1,\ldots,A_n,\beta,u^*(A_1,\ldots,A_n))} =$
$E^{\delta}_{t^*(A_1,\ldots,A_n,\alpha,u^*(A_1,\ldots,A_n,\alpha))t^*(A_1,\ldots,A_n,\beta,u^*(A_1,\ldots,A_n,\beta))} =$
$E^{\delta}_{(\oplus(a,b)(i^1 u,t,s))^*(A_1,\ldots,A_n,\alpha)(\oplus(a,b)(i^1 u,t,s))^*(A_1,\ldots,A_n,\beta)}$
Case 2. $r = i^2 u$. *As in case 1.*
Case 3. $r = \varepsilon^{\sigma+\tau} u$. *This case is undefined.*
Case 4. $r = (\oplus(a',b')(r',t',s'))$. *Application iterative of the above cases.* □

Theorem 4.7 $< MPT, * >$ *is a canonical model.*
Proof : In order to show that $< MPT, * >$ *is a canonical model, we just have to prove that it satisfies 3.4.*

1. • *t is a variable. It is obvious.*
   • $t[a] = (uv)[a]$

   Case 1 *u is not a case or empty term*
   $(u[s]v[s])^*(A_1, \ldots, A_n) \overset{4.4.3}{=}$
   $(u[s])^*(A_1, \ldots, A_n)((v[s])^*(A_1, \ldots, A_n)) \overset{\text{IH}}{=}$
   $u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))(v^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))) \overset{4.4.3}{=}$
   $(uv)^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))$
   Idem for $F_1, \ldots, F_n$

   Case 2 $u[a] = \varepsilon^{\delta \to \tau} w[a]$. *It is trected as empty-terms replacing for* $\varepsilon^{\tau} w[s]$.

   Case 3 $u[a] = (\oplus(c,b)(r, w[c], l[b]))[a]$. *It is treated as in case-terms replacing u for wv and v for lv.*

   • $t[a] = (\pi^1 u[a])$

   Case 1 *u is not a case or empty term.*
   $(\pi^1 u[s])^*(A_1, \ldots, A_n) =$
   $\pi^1_{\tau\delta}(u[s]^*(A_1, \ldots, A_n) =$
   $\pi^1_{\tau\delta}(u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))) =$
   $(\pi^1 u)^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))$
   Idem for $F_1, \ldots, F_n$ *and for* π^2 .

   Case 2 $u[a] = \varepsilon^{\tau \times \delta} v[a]$. *As for empty-terms.*

   Case 3 $u[a] = (\oplus(c,b)(r, w, v))[a]$. *As in case-terms replacing u for* $\pi^1 w$ *and v for* $\pi^1 v$.

   Idem for π^2

- $t[a] = (t' \otimes t'')[a]$

  $(t'[s] \otimes t''[s])^*(A_1, \ldots, A_n) \overset{4.4.5}{=}$

  $((t'[s])^*(A_1, \ldots, A_n), (t''[s])^*(A_1, \ldots, A_n)) \overset{\text{III}}{=}$

  $(t'^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n)), t''^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))) \overset{4.4.5}{=}$

  $(t' \otimes t'')^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))$

  Idem for $F_1, \ldots, F_n$.

- $t[a] = i^1 u[a]$

  $(i^1 u[s])^*(A_1, \ldots, A_n) \overset{4.4.8}{=}$

  $(1, (u[s])^*(A_1, \ldots, A_n)) \overset{\text{III}}{=}$

  $(1, u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))) \overset{4.4.8}{=}$

  $(i^1 u)^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))$

  Idem for $F_1, \ldots, F_n$ *and for* i^2.

- $t[a] = sup_{x<\alpha} u[a]$

  $(sup_{x<\alpha} u[s])^*(A_1, \ldots, A_n) \overset{4.4.7}{=}$

  $(u[s])^*(A_1, \ldots, A_n, \alpha) \overset{\text{III}}{=}$

  $u^*(A_1, \ldots, A_n, \alpha, s^*(A_1, \ldots, A_n, \alpha)) =$

  $u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n, \alpha), \alpha) =$

  $u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n), \alpha) \overset{4.4.7}{=}$

  $(\sup_{x<\alpha} u)^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))$

  Idem for $F_1, \ldots, F_n$

- $t[a^\sigma] = \lambda b^\alpha . u[a^\sigma]$

  $(\lambda b.u[s])^*(A_1, \ldots, A_n)(A_{n+1}) \overset{4.4.4}{=}$

  $(u[s])^*(A_1, \ldots, A_n, A_{n+1}) \overset{\text{III}}{=}$

  $u^*(A_1, \ldots, A_n, A_{n+1}, s^*(A_1, \ldots, A_n, A_{n+1})) =$

  $u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n, A_{n+1}), A_{n+1}) =$

  $u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n), A_{n+1}) \overset{4.4.4}{=}$

  $(\lambda b.u)^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))(A_{n+1}) \quad \forall A_{n+1} \in Pt^\alpha$.

  Idem for $F_1, \ldots, F_n$.

- $t[a^\sigma] = (\oplus(c^\delta, b^\alpha)(r, u[c], v[b]))[a^\sigma] = (\oplus(c, b)(r, u[c, a], v[b, a]))$

Case 1 $r = i^1 w$

$((\oplus(c, b)(i^1 w, u[c, a], v[b, a]))[s])^*(A_1, \ldots, A_n) =$

$(\oplus(c, b)(r[s], u[c, s/a], v[b, s/a]))^*(A_1, \ldots, A_n) =$

$(u[s])^*(A_1, \ldots, A_n, w^*(A_1, \ldots, A_n)) =$

$u^*(A_1, \ldots, A_n, w^*(A_1, \ldots, A_n), s^*(A_1, \ldots, A_n, w^*(A_1, \ldots, A_n))) =$

$u^*(A_1, \ldots, A_n, w^*(A_1, \ldots, A_n), s^*(A_1, \ldots, A_n, w^*(A_1, \ldots, A_n))) =$

$u^*(A_1, \ldots, A_n, w^*(A_1, \ldots, A_n), s^*(A_1, \ldots, A_n)) =$

$u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n), w^*(A_1, \ldots, A_n)) =$

$u^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n), w^*(A_1, \ldots, A_n, s^*(A_1, \ldots, A_n))) =$

$(\oplus(c,b)(i^1 w,u,v))^*(A_1,\ldots,A_n,s^*(A_1,\ldots,A_n))$
Idem for $F_1,\ldots,F_n$.

Case 2 $r[a]=i^2w[a]$. *As in case 1.*

Case 3 $r[a]=\varepsilon^{\delta+\alpha}w[a]$. *Idem for empty-terms.*

Case 4 $r[a]=(\oplus(c',b')(r',u',v'))[a]$ *.Iterative application of the above cases.*

- $t[a]=\varepsilon^{\sigma}u[a]$. *Undefined .*

2. *By definition of sup.*
3. *By definition of the functor sum.*
4. *By definition [Gir].* □

5 Conclusion

Following Girard's concepts and Päppinghaus we extended the model of Ptyxes for finite types for a typed λ-calculus which includes *sum* and *empty* types.

The main work was to build up,step by step, the model for λ-terms and verify that the sum category actually models terms of type *sum*.

As girard puts it, the model of Ptyxes provides interesting refinements in the denotation of different closed normal terms, which may denote the same function on ordinals, "the interpretation by means of Ptyxes helps us to understand why these two normal forms are differents : they denote distinct dilators" [Gir].

References

[Gir] J. Y. Girard. *Proof Theory and Logic Complexity.* Volume 2, Chapter 12 (to appear).

[Gir72] J. Y. Girard. *Interprétation fonctionnelle et élimination des coupures dans l'arithmétique d'ordre supérieur.* PhD thesis, Université Paris VII, 1972.

[Gir81] J. Y. Girard. π^1_2-Logic, Part 1 : Dilators. *Annals of Mathematical Logic*, (21):75–219, 1981.

[GLP89] J.Y. Girard, Y. Lafont, and Taylor P. *Proofs and Types.* Volume 7 of *Cambridge Tracts in Theoretical Computer Science*, Cambridge, 1989.

[GV84] J. Y. Girard and J. Vauzeilles. Functors and Ordinal Notations I : a functorial construction of the Veblen Hierarchy. *The Journal of Symbolic Logic*, 49(3):713–729, September 1984.

[HS73] H. Herrlich and G. E. Strecker. *Category Theory.* Ally and Bacon Inc., Boston, 1973.

[HS86] J.R. Hindley and J.P. Seldin. *Introduction to Combinators and λ-calculus.* Cambridge University Press, London, 1986.

[Pap85] P. Pappinghaus. *A typed Lambda-calculus and Girard's model of Ptykes*, pages 245–279. Plenum Press, New York, 1985.

[Pra65] D. Prawitz. *Natural Deduction.* Almqvist and Wiksell, Stockholm, 1965.

[Pra70] D. Prawitz. Ideas and results in proof-theory. In *Proceedings of the second Scandinavian Logic Symposium*, pages 235–304, North-Holland, 1970.

[Sch77] K. Schutte. *Proof Theory.* Springer Verlag, 1977.

[Tro73] A. S. Troelstra. Metamathematical investigation of intuitionistic arithmetic and analysis. In *LNM*, Springer-Verlag, Berlin, 1973.

A Appendix

A.1 Canonical Total Model

In this section we will show another version of how to model the same λ -calculus using Ptyxes. Here, we introduce in each domain of type σ a constant symbol $\bot_\sigma$ which is the least symbol of the domain σ. This botton element "contains no information"; it serves to model the values of computations that are erroneous or never produce any information. Thus, any term of type EMP, or those which include terms of type EMP, will be modeled by botton elements. In the model defined in section 3, these terms had an undefined value in the model.

We only redo the new concepts and definitions we need, leting anythingelse unchanged.

Definition A.1 An operation $f : A_1 \times \ldots \times A_n \longrightarrow A_{n+1}$ is called *strict* if $f(a_1, \ldots, a_n) = \bot_{A_{n+1}}$ whenever $a_i = \bot_{A_i}$ for some $i = 1, \ldots, n$.

Definition A.2 A *total type structure* is a sequence $M =< \{M_\sigma / \sigma \textit{ finite type}\}, \{\odot_{\sigma\tau}/\sigma, \tau \textit{ finite types}\}, \{\otimes_{\sigma\tau}/\sigma, \tau \textit{ finite types}\},$ $\{\pi^j_{\sigma\tau}/\sigma, \tau \textit{ finite types}\}, \{i^j_{\sigma\tau}/\sigma, \tau \textit{ finite types}, j = 1, 2\} >$ *such that :*

1. *$\forall\sigma, M_\sigma$ has a botton element $\bot_\sigma$.*
2. *M_{EMP} has as its unique element the botton element $\bot_{EMP}$.*
3. *$\odot_{\sigma\tau} : M_{\sigma\to\tau} \times M_\sigma \longrightarrow M_\tau$ is a strict map.*
4. *$\otimes_{\sigma\tau} : M_\sigma \times M_\tau \longrightarrow M_{\sigma\times\tau}$ is a strict map.*
5. *$\pi^1_{\sigma\tau} : M_{\sigma\times\tau} \longrightarrow M_\sigma$ is a strict map.*
   $\pi^2_{\sigma\tau} : M_{\sigma\times\tau} \longrightarrow M_\tau$ is a strict map.
6. *$\forall a \in M_\sigma, \forall b \in M_\tau \pi^1_{\sigma\tau}(a \otimes_{\sigma\tau} b) = a, \pi^2_{\sigma\tau}(a \otimes_{\sigma\tau} b) = b$*
7. *$i^1_{\sigma\tau} : M_\sigma \longrightarrow M_{\sigma+\tau}$ is a strict map.*
   $i^2_{\sigma\tau} : M_\tau \longrightarrow M_{\sigma+\tau}$ is a strict map.

Definition A.3 A *total ordinal operator structure over* Λ *(Λ-OOS)* is a pair $< M, * >$ such that M is a total type structure, M_0 is a limit ordinal $\geq \Lambda \cup \bot_0$ *or* $\Lambda = On \cup \bot_0$, and $*$ is a map associating with every term $t \in \lambda\text{-}ND_\tau$ $\tau \neq EMP$ and every sequence $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$ of different variables containing the free variables of t a strict map $t^* : M_{\sigma_1} \times \ldots \times M_{\sigma_n} \longrightarrow M_\tau$ satisfying the below conditions. If t is of type EMP then $*$ associates to t the botton element $\bot_{EMP}$.[7]

1. *t^* is independent of the naming and ordering of the free variables of t.*

[7]This means that : 1) $\forall t^{EMP}$ $t^*(c_1, \ldots, c_n) = \bot_{EMP}$. 2) $\forall t^\tau$ $\forall\sigma$ $t^*(c_1, \ldots, c_n, \bot_\sigma) = \bot_\tau$.

2. $(a^\sigma)^*\ (c) = c \in M_\sigma,\ \forall\sigma \neq EMP.$

3. • *If u is not a case-term then*
$(uv)^*(c_1,\ldots,c_n) = u^*(c_1,\ldots,c_n) \odot_{\sigma\tau} v^*(c_1,\ldots,c_n)$
• $((\oplus(a,b)(r,t,s))v)^*(c_1,\ldots,c_n) = (\oplus(a,b)(r,tv,sv))^*(c_1,\ldots,c_n)$

4. • *If v is not a case-term then*
$(\pi^j v)^*(c_1,\ldots,c_n) = \pi^j_{\sigma\tau}(v^*(c_1,\ldots,c_n))$
• $(\pi^j(\oplus(a,b)(r,t,s)))^*(c_1,\ldots,c_n) = (\oplus(a,b)(r,\pi^j t,\pi^j s))^*(c_1,\ldots,c_n)$

5. $(s\otimes t)^*(c_1,\ldots,c_n) = s^*(c_1,\ldots,c_n)\otimes t^*(c_1,\ldots,c_n)$

6. • $(\lambda a^\sigma.t)^*(c_1,\ldots,c_n) \odot_{\sigma\tau} d = t^*(c_1,\ldots,c_n,d)$ *for every* $d \in M_\sigma$.
• *If* $\forall d \in M_\sigma\ \ t^*(c_1,\ldots,c_n,d) = s^*(c_1,\ldots,c_n,d)$ *then*
$(\lambda x^\sigma t)^*(c_1,\ldots,c_n) = (\lambda x^\sigma s)^*(c_1,\ldots,c_n)$

7. $0^* = 0 \in M_0$.

8. $S^* \odot_{0\to 0\,0} \alpha = \alpha + 1$ *for every* $\alpha \in M_0$.

9. *If* $t^*(c_1,\ldots,c_n,c_{n+1})$ *is defined, then*
$(\sup_{x<\alpha} t)^*(c_1,\ldots,c_n) = \sup_{\beta<\alpha} t^*(c_1,\ldots,c_n,\beta) \in M_0$.

10. $(i^j t)^*(c_1,\ldots,c_n) = i^j_{\sigma\tau}(t^*(c_1,\ldots,c_n))$

11. •
$$(\oplus(a,b)(r^{\sigma+\tau},t[a^\sigma],s[b^\tau]))^*(c_1,\ldots,c_n) = \begin{cases} t^*(c_1,\ldots,c_n,u^*(c_1,\ldots,c_n)) & \text{if } r^{\sigma+\tau} = i^1 u \\ s^*(c_1,\ldots,c_n,u^*(c_1,\ldots,c_n)) & \text{if } r^{\sigma+\tau} = i^2 u \end{cases}$$
• $(\oplus(a,b)(\varepsilon^{\sigma+\tau} r, t^\alpha, s^\alpha))^*(c_1,\ldots,c_n) = (\varepsilon^\alpha r)^*(c_1,\ldots,c_n)$
• $(\oplus(a',b')((\oplus(a,b)(r,t,s)),t',s'))^*(c_1,\ldots,c_n) =$
$(\oplus(a,b)(r,(\oplus(a',b')(t,t',s')),(\oplus(a',b')(s,t',s'))))^*(c_1,\ldots,c_n)$
• If $(t[a])^*(c_1,\ldots,c_n) = (t'[a])^*(c_1,\ldots,c_n)$,
$(s[b])^*(c_1,\ldots,c_n) = (s'[b])^*(c_1,\ldots,c_n)$,
$r^*(c_1,\ldots,c_n) = r'^*(c_1,\ldots,c_n)$ then
$(\oplus(a,b)(r,t[a],s[b]))^*(c_1,\ldots,c_n) = (\oplus(a,b)(r',t'[a],s'[b]))^*(c_1,\ldots,c_n)$

12. • $(\varepsilon^\sigma t)^*(c_1,\ldots,c_n) = \bot_\sigma$
• $(\varepsilon^\sigma(\oplus(a,b)(r,t,s)))^*(c_1,\ldots,c_n) = (\oplus(a,b)(r,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(c_1,\ldots,c_n)$

Definition A.4 A *canonical total model of λ-ND* is a Λ-*OOS* $< M, * >$ satisfying the following requeriments :

1. *If* $t[a^\sigma] \in \lambda\text{-}ND_\tau, s \in \lambda\text{-}ND_\sigma, a^\sigma$ *not free in s then,*
$(t[s])^*(c_1,\ldots,c_n) = t^*(c_1,\ldots,c_n,s^*(c_1,\ldots,c_n))$

2. *If* $\alpha < \Lambda$ limit ordinal, then $(\sup_{x<\alpha} t)^*(c_1, \ldots, c_n) = t^*(c_1, \ldots, c_n, \alpha)$

3. $+^* \odot \alpha \odot \beta = \alpha + \beta \in M_0$ *for all* $\alpha, \beta \in M_0$.

4. $R^*_\sigma \odot c \odot d \odot 0 = c$,
$R^*_\sigma \odot c \odot d \odot (\alpha + 1) = +^*_\sigma \odot (R^*_\sigma \odot c \odot d \odot \alpha) \odot (d \odot (R^*_\sigma \odot c \odot d \odot \alpha) \odot \alpha)$
for every $\alpha \in M_0$.

Lemma A.5 *Let* $< M, * >$ *be a canonical total model of* λ*-ND.*
Then, for t, $s \in \lambda$*-ND : If* $t \vdash s$ *then* $t^* = s^*$

Proof : By inspection of the reduction rules. The only cases to be considered are:

1. $(\lambda a.t[a])s \vdash t[s]$. *If* σ, τ *are not EMP then, the proof is as in [Pap85].*
If $\sigma = EMP$ *then,*
$((\lambda a^{EMP}.t^\tau)s^{EMP})^*(c_1, \ldots, c_n) =$
$(\lambda a.t)^*(c_1, \ldots, c_n) \odot_{\sigma\tau} s^*(c_1, \ldots, c_n) =$
$\perp_\tau = (t[s])^*(c_1, \ldots, c_n)$

2. $\pi^1(\varepsilon^{\sigma\times\tau}t) \vdash \varepsilon^\sigma t$.
$(\pi^1(\varepsilon^{\sigma\times\tau}t))^*(c_1, \ldots, c_n) =$
$\pi^1_{\sigma\tau}((\varepsilon^{\sigma\times\tau}t)^*(c_1, \ldots, c_n)) =$
$\pi^1_{\sigma\tau}(\perp_{\sigma\times\tau}) = \perp_\sigma =$
$(\varepsilon^\sigma t)^*(c_1, \ldots, c_n)$

3. $\pi^2(\varepsilon^{\sigma\times\tau}t) \vdash \varepsilon^\tau t$. *As in case 2 above.*

4. $(\varepsilon^{\sigma\to\tau}\ t)\ u \vdash \varepsilon^\tau t$.
$((\varepsilon^{\sigma\to\tau}t)u)^*(c_1, \ldots, c_n) =$
$(\varepsilon^{\sigma\to\tau}t)^*(c_1, \ldots, c_n) \odot_{\sigma\tau} u^*(c_1, \ldots, c_n) =$
$\perp_{\sigma\to\tau} \odot_{\sigma\to\tau} u^*(c_1, \ldots, c_n) = \perp_\tau = (\varepsilon^\tau t)^*(c_1, \ldots, c_n)$

5. $\varepsilon^\alpha(\varepsilon^{EMP}t) \vdash \varepsilon^\alpha t$
$(\varepsilon^\alpha(\varepsilon^{EMP}t))^*(c_1, \ldots, c_n) = \perp_\alpha =$
$(\varepsilon^\alpha t)^*(c_1, \ldots, c_n)$ □

Proposition A.6 *Let* $< M, * >$ *be a canonical total model of* λ*-ND. Then, for* $t,s \in \lambda$*-ND*
If $t \leadsto s$ *then* $t^* = s^*$

Proof : By induction on $\leadsto$. *We only need to consider case 8 :*
$(\varepsilon^\alpha t)^*(c_1, \ldots, c_n) = \perp_\alpha = (\varepsilon^\alpha t')^*(c_1, \ldots, c_n)$ □

A.2 Reviewing Ptyxes

Each category PT^σ will now have a new element, different from the others objects, this element will be the least element of the category.

Definition A.7 *1. PT^{EMP} is the 1-category, with the botton element $\bot_{EMP}$, and the identity morphism $E^{EMP}_{\bot_{EMP}}$.*

2. *PT^0 is ON^+; more precisely, Pt^0 is $On \cup \{\bot_0\}$, and $I^0(x,y) = I(x,y)$ when $x, y \in On$. For the remaining cases, we have*
    - $I(\bot_0, \bot_0) = \{E_{\bot_0}\}$
    - $E^0_{\bot_0 \alpha} \in I(\bot_o, \alpha) \forall \alpha \in On$
    - $I(\alpha, \bot_0) = \emptyset \forall \alpha \in On$
3. *$PT^{\sigma\to\tau}$ is such that $Pt^{\sigma\to\tau}$ is the class of all functors from PT^σ to PT^τ preserving direct limits and pull-back, and the botton element $\bot_{\sigma\to\tau}$. If A, B are in $Pt^{\sigma\to\tau}$, then $I^{\sigma\to\tau}(A,B)$ is the set of all natural transformations from A to B. Besides, we have $E_{\sigma\to\tau}$, and $E^{\sigma\to\tau}_{\bot_{\sigma\to\tau} A}$ for all A in $Pt^{\sigma\to\tau}$.*
4. *$PT^{\sigma\times\tau}$ is the product of categories PT^σ and PT^τ : $Pt^{\sigma\times\tau}$ is the class of all pairs (A,B) such that A is in Pt^σ and B is in Pt^τ; we call $\bot_{\sigma\times\tau} = (\bot_\sigma, \bot_\tau)$. $I^{\sigma\times\tau}((A,A'),(B,B'))$ consist of pairs (T,T') such that T is in $I^\sigma(A,B)$ and T' is in $I^\tau(A',B')$.*
5. *$PT^{\sigma+\tau}$ is the sum (or coproduct) of the categories PT^σ and PT^τ : $Pt^{\sigma+\tau}$ consist of pairs (i,A) with $i=1$ and $A \in Pt^\sigma$ or $i=2$ and $A \in Pt^\tau$, and the botton element $\bot_{\sigma+\tau}$; $I^{\sigma+\tau}((i,A),(j,B))$ is void when $i \neq j$, and consists of pairs $(1,T)$ such that T is in $I^\sigma(A,B)$ if $i=j=1$ or consists of pairs $(2,T)$ such that T is in $I^\tau(A,B)$ if $i=j=2$. . We also have $E^{\sigma+\tau}_{\bot_{\sigma+\tau}}$ and $E^{\sigma+\tau}_{\bot_{\sigma+\tau} A}$ for all A in $Pt^{\sigma+\tau}$.*

Definition A.8 Let be $MPT = < \{PT^\sigma\}, \odot_{\sigma\tau}, \otimes_{\sigma\tau}, \pi^j_{\sigma\tau}, i^j_{\sigma\tau} >$ *for σ, τ finite types and* $j = 1,2$ where

- *$\odot_{\sigma\tau}$ is the strict application of Ptyxes and of morphisms of corresponding types (this means that $A^{\sigma\to\tau} \odot_{\sigma\tau} \bot_\sigma = \bot_{\sigma\to\tau} \odot_{\sigma\tau} A^\sigma = \bot_\tau$, and $F^{\sigma\to\tau} \odot_{\sigma\tau} E_{\bot_\sigma} = E_{\bot_{\sigma\to\tau}} \odot_{\sigma\tau} F^\sigma = E_{\bot_\tau}$ for A Ptyx and F morphism).*
- *$\otimes_{\sigma\tau}$ is the strict pairing of Ptyxes and of morphism of corresponding types.*
- *$\pi^j_{\sigma\tau}$ are the strict unpairing of Ptyxes and of morphisms of corresponding types.*
- *$i^j_{\sigma\tau}$ are the strict injections of Ptyxes and of morphisms of corresponding types.*

Thus, MPT is a total type structure.

Definition A.9 The Λ-OOS^+ is the pair $< MPT, * >$ where M_0 *is* ON^+, $*$ is a strict map (both in Ptyxes and in morphisms) associating to each term $t \in \lambda\text{-}ND_\tau$ $\tau \neq EMP$ and every sequence $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$ of different variables containing the free variables of t a strict multifunctor $t^* : PT^{\sigma_1} \times \ldots \times PT^{\sigma_n} \longrightarrow PT^\tau$; more precisely, assume that $t(a_1^{\sigma_1}, \ldots, a_n^{\sigma_n})$ is a λ-term of type τ whose only free variables are contained in the sequence $a_1^{\sigma_1}, \ldots, a_n^{\sigma_n}$. Then, we define for all Ptyxes $A_1, \ldots, A_n$ of respective types $\sigma_1, \ldots, \sigma_n$ a Ptyx $t^*(A_1, \ldots, A_n)$ *of type* τ; and for every sequence of morphisms $F_1 \in I^{\sigma_1}(A_1, A_1'), \ldots, F_n \in I^{\sigma_n}(A_n, A_n')$ we define a morphism $t^*(F_1, \ldots, F_n) \in I^\sigma(t^*(A_1, \ldots, A_n), t^*(A_1', \ldots, A_n'))$. *If t is the type* EMP,$t^*(A_1, \ldots, A_n)$ *is* $\perp_{EMP}$, *and* $t^*(F_1, \ldots, F_n)$ *is* $E^{EMP}_{\perp_{EMP}}$.

We need that the requeriments of the definition of Λ-OOS and the equations (eq) of section 4 are satisfied.

Now, we defined these multifunctors t^* by induction on the complexity of the term t as in [Gir].

Definition A.10 **1.variables** $(a^\sigma)^*(A) = A$ *and* $(a^\sigma)^*(F) = F$ $\forall \sigma \neq EMP$

This satisfies the required conditions.

2.constants $(0)^*$ *is the Ptyx 0 of type 0 (the ordinal 0).*

$(S)^*$ *is the Ptyx Id + 1 of type* $0 \to 0$ *(dilator) [Gir81]*

$(+)^*$ *is the functor sum (bilator) of type* $0 \to 0 \to 0$ *[Gir81]*

$(R_\sigma)^*$ *is define as in [Gir] and [Pap85].*

3.app-terms *We want to defined* $(tu)^*$

1. *Assume t is not a case-term. Then, we define :*
   $(tu)^*(A_1, \ldots, A_n) = t^*(A_1, \ldots, A_n)(u^*(A_1, \ldots, A_n))$
   $(tu)^*(F_1, \ldots, F_n) = t^*(F_1, \ldots, F_n)(u^*(F_1, \ldots, F_n))$
   This satisfies the required conditions.

2. *If t is a case-term then, we have:*
   $((\oplus(a,b)(r^{\sigma+\tau}, t^{\alpha\to\delta}, s^{\alpha\to\delta}))v^\alpha)^*(A_1, \ldots, A_n) = (\oplus(a,b)(r, tv, sv))^*(A_1, \ldots, A_n)$
   $((\oplus(a,b)(r^{\sigma+\tau}, t^{\alpha\to\delta}, s^{\alpha\to\delta}))v^\alpha)^*(F_1, \ldots, F_n) = (\oplus(a,b)(r, tv, sv))^*(F_1, \ldots, F_n)$
   We shall only check the conditions when r is $\varepsilon^{\sigma+\tau}u^{EMP}$

   (a) $((\oplus(a,b)(\varepsilon^{\sigma+\tau}u, s, t))v)^*(E_{A_1}, \ldots, E_{A_n}) =$
   $(\oplus(a,b)(\varepsilon^{\sigma+\tau}u, tv, sv))^*(E_{A_1}, \ldots, E_{A_n}) =$
   $(\varepsilon^\delta u)^*(E_{A_1}, \ldots, E_{A_n}) =$
   $E^\delta_{\perp_\delta} = E^\delta_{(\varepsilon^\delta u)^*(A_1, \ldots, A_n)} =$
   $E^\delta_{(\oplus(a,b)(\varepsilon^{\sigma+\tau}u, tv, sv))^*(A_1, \ldots, A_n)} =$
   $E^\delta_{((\oplus(a,b)(\varepsilon u^{\sigma+\tau}, t, s))v)^*(A_1, \ldots, A_n)}$

(b) $((\oplus(a,b)(r,t,s))v)^*(F_1,\ldots,F_n) \circ ((\oplus(a,b)(r,t,s))v)^*(F'_1,\ldots,F'_n) =$
$(\oplus(a,b)(r,tv,sv))^*(F_1,\ldots,F_n) \circ (\oplus(a,b)(r,tv,sv))^*(F'_1,\ldots,F'_n) =$
$(\varepsilon^\delta u^{EMP})^*(F_1,\ldots,F_n) \circ (\varepsilon^\delta u^{EMP})^*(F'_1,\ldots,F'_n) =$
$E^\delta_{\perp_\delta} \circ E^\delta_{\perp_\delta} = E^\delta_{\perp_\delta} =$
$(\varepsilon^\delta u)^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n) =$
$(\oplus(a,b)(\varepsilon^{\sigma+\tau},t,s))^*(F_1 \circ F'_1,\ldots,F_n \circ F'_n)$

(c) *As for property 2.*

(d) $(((\oplus(a,b)(r,t,s))v)^*(A_1,\ldots,A_n),((\oplus(a,b)(r,t,s))v)^*(F_{1_i},\ldots,F_{n_i})) =$
$((\oplus(a,b)(r,tv,sv))^*(A_1,\ldots,A_n),(\oplus(a,b)(r,tv,sv))^*(F_{1_i},\ldots,F_{n_i})) =$
$((\varepsilon^\delta u)^*(A_1,\ldots,A_n),(\varepsilon^\delta u)^*(F_{1_i},\ldots,F_{n_i})) =$
$(\perp_\delta, E^\delta_{\perp_\delta}) =$
$\lim_{i\in I}(\perp_\delta, E^\delta_{\perp_\delta}) =$
$\lim_{i\in I}((\varepsilon^\delta u)^*(A_{1_i},\ldots,A_{n_i}),(\varepsilon^\delta u)^*(F_{1_{ij}},\ldots,F_{n_{ij}})) =$
$\lim_{i\in I}((\oplus(a,b)(\varepsilon^{\sigma+\tau}u,tv,sv))^*(A_{1_i},\ldots,A_{n_i}),(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,tv,sv))^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

4.λ-terms $(\lambda a^\sigma t)^*(A_1,\ldots,A_n)(A_{n+1}) = t^*(A_1,\ldots,A_n;A_{n+1})$
$(\lambda a^\sigma t)^*(A_1,\ldots,A_n)(F_{n+1}) = t^*(E_{A_1},\ldots,E_{A_n},F_{n+1})$
$(\lambda a^\sigma t)^*(F_1,\ldots,F_n)(A_{n+1}) = t^*(F_1,\ldots,F_n,E_{A_{n+1}})$

5.⊗-terms $(t\otimes u)^*(A_1,\ldots,A_n) = (t^*(A_1,\ldots,A_n),u^*(A_1,\ldots,A_n))$
$(t\otimes u)^*(F_1,\ldots,F_n) = (t^*(F_1,\ldots,F_n),u^*(F_1,\ldots,F_n))$

6.π-terms *1. Assume that t is not a case-term. Then, we define:*
$(\pi^1 t)^*(A_1,\ldots,A_n) = \pi^1_{\sigma\tau}(t^*(A_1,\ldots,A_n))$
$(\pi^1 t)^*(F_1,\ldots,F_n) = \pi^1_{\sigma\tau}(t^*(F_1,\ldots,F_n))$
$(\pi^2 t)^*(A_1,\ldots,A_n) = \pi^2_{\sigma\tau}(t^*(A_1,\ldots,A_n))$
$(\pi^2 t)^*(F_1,\ldots,F_n) = \pi^2_{\sigma\tau}(t^*(F_1,\ldots,F_n))$

2. If t is an case-term then, we define:
$(\pi^i(\oplus(a,b)(r,t,s)))^*(A_1,\ldots,A_n) = (\oplus(a,b)(r,\pi^i t,\pi^i s))^*(A_1,\ldots,A_n)$
$(\pi^i(\oplus(a,b)(r,t,s)))^*(F_1,\ldots,F_n) = (\oplus(a,b)(r,\pi^i t,\pi^i s))^*(F_1,\ldots,F_n)$
We shall only verify the properties when r is $\varepsilon^{\sigma+\tau}u$

(a) $(\pi^i(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^{\alpha\times\delta})^*(E_{A_1},\ldots,E_{A_n}) =$
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^*(E_{A_1},\ldots,E_{A_n}) =$
$(\varepsilon^\alpha u)^*(E_{A_1},\ldots,E_{A_n}) =$
$E^\alpha_{\perp_\alpha} =$
$E^\alpha_{(\varepsilon^\alpha u)^*(A_1,\ldots,A_n)} =$
$E^\alpha_{(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^*(A_1,\ldots,A_n)}$

(b) $(\pi^i(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^{\alpha\times\delta})^*(F_1,\ldots,F_n) \circ (\pi^i(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^{\alpha\times\delta})^*(F'_1,\ldots,F'_n) =$

$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^{\alpha\times\delta\,*}(F_1,\ldots,F_n) \circ (\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^{\alpha\times\delta\,*}(F'_1,\ldots,F'_n) =$

$(\varepsilon^\alpha u)^*(F_1,\dots,F_n)\circ(\varepsilon^\alpha u)^*(F_1',\dots,F_n') =$
$E^\alpha_{\perp_\alpha}\circ E^\alpha_{\perp_\alpha} =$
$E^\alpha_{\perp_\alpha} =$
$(\varepsilon^\alpha u)^*(F_1\circ F_1',\dots,F_n\circ F_n') =$
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^*(F_1\circ F_1',\dots,F_n\circ F_n')$

(c) As for property 2.

(d) $((\pi^i(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s)))^*(A_1,\dots,A_n),(\pi^i(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s)))^*(F_{1_i},\dots,F_{n_i})) =$

$((\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^*(A_1,\dots,A_n),(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,\pi^i t,\pi^i s))^*(F_{1_i},\dots,F_{n_i})) =$

$((\varepsilon^\alpha u)^*(A_1,\dots,A_n),(\varepsilon^\alpha u)^*(F_{1_i},\dots,F_{n_i})) =$
$(\perp_\alpha, E^\alpha_{\perp_{\alpha}}) =$
$\lim_{i\in I}(\perp_\alpha, E^\alpha_{\perp_{\alpha}}) =$
$\lim_{i\in I}((\varepsilon^\alpha u)^*(A_{1_i},\dots,A_{n_i}),(\varepsilon^\alpha u)^*(F_{1_{ij}},\dots,F_{n_{ij}})) =$
$\lim_{i\in I}((\oplus(a,b)(\varepsilon^\alpha u,\pi^i t,\pi^i s))^*(A_{1_i},\dots,A_{n_i}),$
$(\oplus(a,b)(\varepsilon^\alpha u,\pi^i t,\pi^i s))^*(F_{1_{ij}},\dots,F_{n_{ij}}))$

7.sup-terms $(\sup_{x<\alpha} t)^*(A_1,\dots A_n) = t^*(A_1,\dots,A_n,\alpha)$
$(\sup_{x<\alpha} t)^*(F_1,\dots,F_n) = t^*(F_1,\dots,F_n,E^0_\alpha)$

8.⊕-terms *If* $\sigma,\tau\neq EMP$ *then,*
$(i^1t^\sigma)^*(A_1,\dots,A_n) = (1,t^*(A_1,\dots,A_n))$
$(i^1t^\sigma)^*(F_1,\dots,F_n) = (1,t^*(F_1,\dots,F_n))$
$(i^2t^\tau)^*(A_1,\dots,A_n) = (2,t^*(A_1,\dots,A_n))$
$(i^2t^\tau)^*(F_1,\dots,F_n) = (2,t^*(F_1,\dots,F_n))$
If $\sigma = EMP$ *then, we have* $(i^1t^{EMP})^{EMP+\sigma\,*}(A_1,\dots,A_n) = \perp_{EMP+\sigma}$ *,and*
$(i^1t^{EMP})^{EMP+\sigma\,*}(F_1,\dots,F_n) = E^{EMP+\sigma}_{\perp_{EMP+\sigma}}$.
And, if $\tau = EMP$ $(i^2t^{EMP})^{\sigma+EMP\,*}(A_1,\dots,A_n) = \perp_{\sigma+EMP}$ *,and*
$(i^2t^{EMP})^{\sigma+EMP\,*}(F_1,\dots,F_n) = E^{\sigma+EMP}_{\perp_{\sigma+EMP}}$

9.case-terms *1. Assume that r is not an empty-term or a case-term. Then, we define*

$$(\oplus(a,b)(r^{\sigma+\tau},t^\alpha,s^\alpha))^{\alpha *}(A_1,\dots,A_n) = \begin{cases} t^*(A_1,\dots,A_n,u^*(A_1,\dots,A_n)) & \text{if } r = i^1u^\sigma \\ s^*(A_1,\dots,A_n,u^*(A_1,\dots,A_n)) & \text{if } r = i^2u^\tau \end{cases}$$

$$(\oplus(a,b)(r^{\sigma+\tau},t^\alpha,s^\alpha))^{\alpha *}(F_1,\dots,F_n) = \begin{cases} t^*(F_1,\dots,F_n,u^*(F_1,\dots,F_n)) & \text{if } r = i^1u^\sigma \\ s^*(F_1,\dots,F_n,u^*(F_1,\dots,F_n)) & \text{if } r = i^2u^\tau \end{cases}$$

2. If r is an empty-term then, we define
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u^{EMP},t^\alpha,s^\alpha))^*(A_1,\dots,A_n) = (\varepsilon^\alpha u)^*(A_1,\dots,A_n)$
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u^{EMP},t^\alpha,s^\alpha))^*(F_1,\dots,F_n) = (\varepsilon^\alpha u)^*(F_1,\dots,F_n)$
This satisfies the first condition of 3.2.11 and for the fourth one we have:
Suppose that $(t[a])^*(A_1,\dots,A_n) = (t'[a])^*(A_1,\dots,A_n)$

$(s[b])^*(A_1,\ldots,A_n) = (s'[b])^*(A_1,\ldots,A_n)$ *and*
$(\varepsilon^{\sigma+\tau}u)^*(A_1,\ldots,A_n) = (\varepsilon^{\sigma+\tau}u')^*(A_1,\ldots,A_n) = \bot_{\sigma+\tau}$
Suppose for the sake of contradiction that
$(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(A_1,\ldots,A_n) \neq (\oplus(a,b)(\varepsilon^{\sigma+\tau}u',t's'))^*(A_1,\ldots,A_n)$
Then, $(\varepsilon^{\sigma+\tau}u)^*(A_1,\ldots,A_n) \neq (\varepsilon^{\sigma+\tau}u')^*(A_1,\ldots,A_n)$ *This is a contradiction.*
Idem for $F_1,\ldots,F_n$.

(a) $(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(E_{A_1},\ldots,E_{A_n}) =$
$(\varepsilon^{\alpha}u)^*(E_{A_1},\ldots,E_{A_n}) = E^{\alpha}_{\bot_\alpha} =$
$E^{\alpha}_{(\varepsilon^\alpha u)^\alpha(A_1,\ldots,A_n)} =$
$E^{\alpha}_{(\oplus(a,b)(\varepsilon^{\sigma+\tau},t,s))^*(A_1,\ldots,A_n)}$
(b) $(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(F_1,\ldots,F_n) \circ (\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(F'_1,\ldots,F'_n) =$
$(\varepsilon^{\sigma+\tau}u)^*(F_1,\ldots,F_n) \circ (\varepsilon^{\sigma+\tau}u)^*(F'_1,\ldots,F'_n) =$
$E^{\alpha}_{\bot_\alpha} \circ E^{\alpha}_{\bot_\alpha} = E^{\alpha}_{\bot_\alpha} =$
$E^{\alpha}_{(\varepsilon^\alpha u)^*(F_1\circ F'_1,\ldots,F_n\circ F'_n)} =$
$E^{\alpha}_{(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(F_1\circ F'_1,\ldots,F_n\circ F'_n)}$
(c) . *Analogous to case b.*
(d) $((\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^{\alpha *}(A_1,\ldots,A_n),(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^{\alpha *}(F_{1_i},\ldots,F_{n_i})) =$
$((\varepsilon^\alpha u)^*(A_1,\ldots,A_n),(\varepsilon^\alpha u)^*(F_{1_i},\ldots,F_{n_i})) =$
$(\bot_\alpha, E^{\alpha}_{\bot_\alpha}) = \lim_{i\in I}(\bot_\alpha, E^{\alpha}_{\bot_\alpha}) =$
$\lim_{i\in I}((\varepsilon^\alpha u)^*(A_{1_i},\ldots,A_{n_i}),(\varepsilon^\alpha u)^*(F_{1_{ij}},\ldots,F_{n_{ij}})) =$
$\lim_{i\in I}((\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(A_{1_i},\ldots,A_{n_i}),(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

3. If t is a case-term then, we iterate the solutions given above.

10.empty-terms *1.* $(\varepsilon^\sigma t)^*(A_1,\ldots,A_n) = \bot_\sigma$
$(\varepsilon^\sigma t)^*(F_1,\ldots,F_n) = E^{\sigma}_{\bot_\sigma}$
We check that this definition satisfies the required conditions.

- $(\varepsilon^\sigma t)^*(E_{A_1},\ldots,E_{A_n}) = E^{\sigma}_{\bot_\sigma} = E^{\sigma}_{(\varepsilon^\sigma t)^*(A_1,\ldots,A_n)}$
- $(\varepsilon^\sigma t)^*(F_1,\ldots,F_n) \circ (\varepsilon^\sigma t)^*(F'_1,\ldots,F'_n) = E^{\sigma}_{\bot_\sigma} \circ E^{\sigma}_{\bot_\sigma} = E^{\sigma}_{\bot_\sigma} =$
  $= E^{\sigma}_{(\varepsilon^\sigma t)^*(F_1\circ F'_1,\ldots,F_n\circ F'_n)}$
- *As above*
- $((\varepsilon^\sigma t)^*(A_1,\ldots,A_n),(\varepsilon^\sigma t)^*(F_{1_i},\ldots,F_{n_i})) =$
  $(\bot_\sigma, E^{\sigma}_{\bot_\sigma}) = \lim_{i\in I}(\bot_\sigma, E^{\sigma}_{\bot_\sigma}) = \lim_{i\in I}((\varepsilon^\sigma t)^*(A_{1_i},\ldots,A_{n_i}),(\varepsilon^\sigma t)^*(F_{1_{ij}},\ldots,F_{n_{ij}}))$

2. $(\varepsilon^\sigma(\oplus(a,b)(r,s,t)))^*(A_1,\ldots,A_n) = (\oplus(a,b)(r,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(A_1,\ldots,A_n)$
$(\varepsilon^\sigma(\oplus(a,b)(r,s,t)))^*(F_1,\ldots,F_n) = (\oplus(a,b)(r,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(F_1,\ldots,F_n)$
We check that this definition satisfies the required properties.

- – *If* $r^{\delta+\tau} = i^1u^\delta$ *then,*
  $(\varepsilon^\sigma(\oplus(a,b)(i^1u,t,s))^{EMP})^*(E_{A_1},\ldots,E_{A_n}) =$
  $(\oplus(a,b)(i^1u,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(E_{A_1},\ldots,E_{A_n}) =$
  $(\varepsilon^\sigma t)^*(E_{A_1},\ldots,E_{A_n},u^*(E_{A_1},\ldots,E_{A_n})) =$

$(\varepsilon^\sigma t)^*(E_{A_1}, \ldots, E_{A_n}, E^\delta_{u^*(A_1,\ldots,A_n)}) =$
$E^\sigma_{(\varepsilon^\sigma t)^*(A_1,\ldots,A_n,u^*(A_1,\ldots,A_n))} =$
$E^\sigma_{(\oplus(a,b)(i^1 u,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(A_1,\ldots,A_n)}$

- *If* $r^{\delta+\tau} = \varepsilon^{\delta+\tau} u^{EMP}$ *then,*
  $(\varepsilon^\sigma(\oplus(a,b)(\varepsilon^{\delta+\tau} u, t, s))^*(E_{A_1}, \ldots, E_{A_n}) =$
  $(\oplus(a,b)(\varepsilon^{\delta+\tau} u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(E_{A_1}, \ldots, E_{A_n}) =$
  $(\varepsilon^\sigma u)^*(E_{A_1}, \ldots, E_{A_n}) =$
  $E^\sigma_{(\varepsilon^\sigma u)^*(A_1,\ldots,A_n)} =$
  $E^\sigma_{(\oplus(a,b)(\varepsilon^{\delta+\tau} u,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(A_1,\ldots,A_n)}$

◦ – *If* $r^{\delta+\tau} = i^1 u^\delta$ *then,*
  $(\varepsilon^\sigma(\oplus(a,b)(i^1 u, t, s)))^*(F_1, \ldots, F_n) \circ (\varepsilon^\sigma(\oplus(a,b)(i^1 u, t, s)))^*(F'_1, \ldots, F'_n) =$
  $(\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_1, \ldots, F_n) \circ (\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F'_1, \ldots, F'_n) =$
  $(\varepsilon^\sigma t)^*(F_1, \ldots, F_n, u^*(F_1, \ldots, F_n)) \circ (\varepsilon^\sigma t)^*(F'_1, \ldots, F'_n, u^*(F'_1, \ldots, F'_n)) =$
  $(\varepsilon^\sigma t)^*(F_1, \circ F'_1, \ldots, F_n \circ F'_n, u^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)) =$
  $(\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)$
- *If* $r = \varepsilon^{\delta+\tau} u$ *then,*
  $(\varepsilon^\sigma(\oplus(a,b)(\varepsilon^{\delta+\tau} u, t, s)))^*(F_1, \ldots, F_n) \circ (\varepsilon^\sigma(\oplus(a,b)(\varepsilon^{\delta+\tau} u, t, s)))^*(F'_1, \ldots, F'_n) =$

  $(\oplus(a,b)(\varepsilon^{\delta+\tau} u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_1, \ldots, F_n) \circ (\oplus(a,b)(\varepsilon^{\delta+\tau} u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F'_1, \ldots, F'_n)$
  $= (\varepsilon^\sigma u)^*(F_1, \ldots, F_n) \circ (\varepsilon^\sigma u)^*(F'_1, \ldots, F'_n) =$
  $(\varepsilon^\sigma u)^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n) =$
  $(\oplus(a,b)(\varepsilon^{\delta+\tau} u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_1 \circ F'_1, \ldots, F_n \circ F'_n)$

◦ *As above.*

◦ – *If* $r = i^1 u$ *then,*
  $((\varepsilon^\sigma(\oplus(a,b)(i^1 u, t, s)))^*(A_1, \ldots, A_n), (\varepsilon^\sigma(\oplus(a,b)(i^1 u, t, s)))^*(F_{1_i}, \ldots, F_{n_i})) =$

  $((\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(A_1, \ldots, A_n), (\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_{1_i}, \ldots, F_{n_i}) =$

  $((\varepsilon^\sigma t)^*(A_1, \ldots, A_n, u^*(A_1, \ldots, A_n)), (\varepsilon^\sigma t)^*(F_{1_i}, \ldots, F_{n_i}, u^*(F_{1_i}, \ldots, F_{n_i})) =$
  $\lim_{i \in I}((\varepsilon^\sigma t)^*(A_{1_i}, \ldots, A_{n_i}, u^*(A_{1_i}, \ldots, A_{n_i})),$
  $(\varepsilon^\sigma t)^*(F_{1_{ij}}, \ldots, F_{n_{ij}}, u^*(F_{1_{ij}}, \ldots, F_{n_{ij}})) =$
  $\lim_{i \in I}((\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(A_{1_i}, \ldots, A_{n_i}),$
  $(\oplus(a,b)(i^1 u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_{1_{ij}}, \ldots, F_{n_{ij}}))$
- *If* $r = \varepsilon^{\delta+\tau} u$ *then,*
  $((\varepsilon^\sigma(\oplus(a,b)(\varepsilon^{\delta+\tau} u, t, s)))^*(A_1, \ldots, A_n),$
  $(\varepsilon^\sigma(\oplus(a,b)(\varepsilon^{\delta+\tau} u, t, s)))^*(F_{1_i}, \ldots, F_{n_i})) =$
  $((\oplus(a,b)(\varepsilon^{\delta+\tau} u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(A_1, \ldots, A_n),$
  $(\oplus(a,b)(\varepsilon^{\delta+\tau} u, \varepsilon^\sigma t, \varepsilon^\sigma s))^*(F_{1_i}, \ldots, F_{n_i}) =$
  $((\varepsilon^\sigma u)^*(A_1, \ldots, A_n), (\varepsilon^\sigma u)(F_{1_i}, \ldots, F_{n_i})) =$
  $(\perp_\sigma, E^\sigma_{\perp_\sigma}) =$
  $\lim_{i \in I}(\perp_\sigma, E^\sigma_{\perp_\sigma}) =$

$\lim_{i\in I}((\varepsilon^\sigma u)^*(A_{1_i},\ldots,A_{n_i}),(\varepsilon^\sigma u)(F_{1_{ij}},\ldots,F_{n_{ij}}))=$
$\lim_{i\in I}((\oplus(a,b)(\varepsilon^{\delta+\tau}u,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(A_1,\ldots,A_n),$
$(\oplus(a,b)(\varepsilon^{\delta+\tau}u,\varepsilon^\sigma t,\varepsilon^\sigma s))^*(F_{1_{ij}},\ldots,F_{n_{ij}})$ □

Lemma A.11 *If the free variables of $t^\tau \in \lambda\text{-}ND_\tau$ are among $a_1^{\sigma_1},\ldots,a_n^{\sigma_n},x^0$ and x^0 is d. var. of t then,*
$\forall\alpha\leq\beta\in On,\ t^*(E_{A_1}^{\sigma_1},\ldots,E_{A_n}^{\sigma_n},E_{\alpha\beta}^0)=E^\tau_{t^*(A_1,\ldots,A_n,\alpha)t^*(A_1,\ldots,A_n,\beta)}$

Proof : by induction on the term t. We shall only check 2 cases.

1. $(\varepsilon^\sigma t)^*(E_{A_1},\ldots,E_{A_n},E^0_{\alpha\beta})=E^\sigma_{\perp_\sigma}=E^\sigma_{\perp_\sigma\perp_\sigma}=E^\sigma_{(\varepsilon^\sigma t)^*(A_1,\ldots,A_n,\alpha)(\varepsilon^\sigma t)^*(A_1,\ldots,A_n,\beta)}$
2. $(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^{\delta *}(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta})=$
   $(\varepsilon^\delta u)^*(E_{A_1},\ldots,E_{A_n},E_{\alpha\beta})=$
   $E^\delta_{\perp_\delta\perp_\delta}=E^\delta_{(\varepsilon^\delta u)^*(A_1,\ldots,A_n,\alpha)(\varepsilon^\delta u)^*(A_1,\ldots,A_n,\beta)}=$
   $E^\delta_{(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(A_1,\ldots,A_n,\alpha)(\oplus(a,b)(\varepsilon^{\sigma+\tau}u,t,s))^*(A_1,\ldots,A_n,\beta)}$ □

Theorem A.12 *$< MPT,* >$ is a canonical total model.*
Proof: We shall only check 2 cases of the substitution property.

- $(\varepsilon^\sigma u[s])^*(A_1,\ldots,A_n)=\perp_\sigma=(\varepsilon^\sigma u)^*(A_1,\ldots,A_n,s^*(A_1,\ldots,A_n))$
- $((\oplus(c,b)(\varepsilon^{\sigma+\tau}w,u[a,c],v[a,b]))[s])^{\alpha *}(A_1,\ldots,A_n)=$
  $(\oplus(c,b)(\varepsilon^{\sigma+\tau}w,u[s/a,c],v[s/a,b]))^*(A_1,\ldots,A_n)=$
  $(\varepsilon^\alpha w)^*(A_1,\ldots,A_n)=\perp_\alpha=$
  $(\varepsilon^\alpha w)^*(A_1,\ldots,A_n,s^*(A_1,\ldots,A_n))=$
  $(\oplus(c,b)(\varepsilon^{\sigma+\tau}w,u,v))^*(A_1,\ldots,A_n,s^*(A_1,\ldots,A_n))$ □