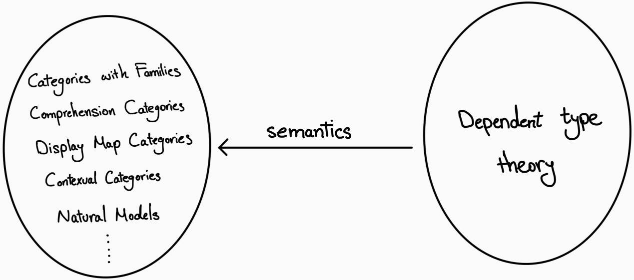
Categorical Semantics for the Extrinsic View of Typing

Niyousha Najmaei
with Ambroise Lafont

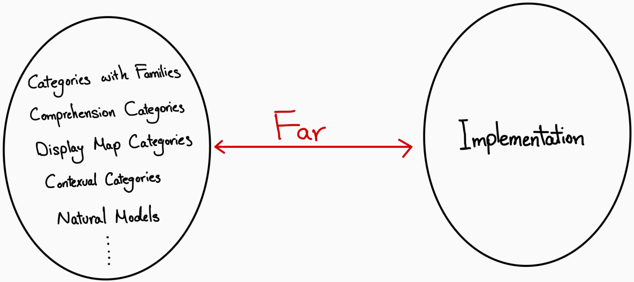
LHC Days, Palaiseau
4 June, 2025

Overview



Categories with Families
Comprehension Categories
Display Map Categories
Contexual Categories
Natural Models
⋮
semantics
Dependent type theory

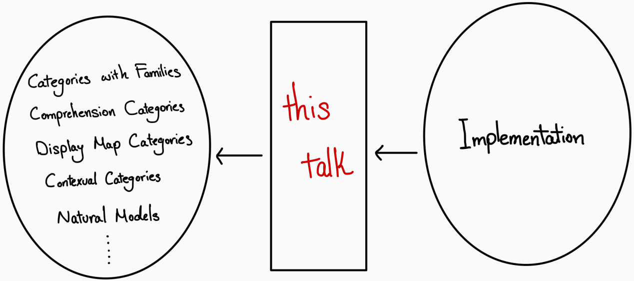
Overview



1. Terms are typed by construction

1. Raw syntax
2. Typing judgments
3. Typed syntax

Overview



This talk: ongoing work about the semantics of the extrinsic view of typing

Outline

Motivation

Extrinsic vs Intrinsic Typing

- **Intrinsic presentation:**
  - Syntax is typed by construction
  - Ill-formed terms are not expressible
  - Easier to reason about mathematically
  - Aka types à la Church
- **Extrinsic presentation:**
  - Start with untyped syntax
  - Typing judgments generated by inference rules
  - Ill-typed terms are ruled out externally
  - Aka types à la Curry

Intrinsic Semantics is Well-Studied

- Many categorical frameworks for the intrinsic view:
  - Categories with Families (CwFs)
  - Comprehension Categories
  - Natural Models
  - Contextual Categories
  - ...
- This is far from practical implementations where we start from untyped syntax.

Motivation

Semantics for untyped syntax and typing: Why?

- Categorically prove syntactic results that are not provable in the traditional semantics, e.g. that reduction is compatible with typing
- Check that the implementation corresponds to the initial model
- Translation of syntactic models for e.g. proving consistency of the source type theory

$$\mathsf{CwF} \longleftarrow \mathsf{ExtCwF}$$

Intrinsic Semantics

Intrinsic Semantics

Many categorical frameworks have been developed:

- Categories with Families (CwFs)
- Comprehension Categories
- Natural Models
- Contextual Categories
- ...

Categories with Families

Definition (Dybjer 1996)

A category with families consists of:

1. A category $\mathcal{C}$ with a terminal object of contexts and substitutions
2. A presheaf $\mathsf{Ty} : \mathcal{C}^{\mathsf{op}} \to \mathsf{Set}$
3. A presheaf $\mathsf{Tm} : (\int_{\mathcal{C}} \mathsf{Ty})^{\mathsf{op}} \to \mathsf{Set}$
4. A context extension operation giving $\Gamma.A$, $\mathsf{proj} : \Gamma.A \to \Gamma$, and ...

Extrinsic Semantics

Extrinsic Semantics

Raw Syntax

Untyped syntax

- A functor $\mathrm{pTm} : \mathrm{Nat} \to \mathrm{Set}$

$$\mathrm{pTm} : n \mapsto \text{preterms with } n \text{ free variables}$$

- A functor $\mathrm{pTy} : \mathrm{Nat} \to \mathrm{Set}$

$$\mathrm{pTy} : n \mapsto \text{pretypes with } n \text{ free variables}$$

And we want to be able to do substitution of raw terms.

Untyped Syntax: Binding

$$\mathrm{pTm} : \mathrm{Nat} \to \mathrm{Set} \quad , \quad \mathrm{pTy} : \mathrm{Nat} \to \mathrm{Set}$$

- Substituting term variables in terms

$$\frac{f : n \to \mathrm{pTm}(m)}{\mathrm{bind}_{\mathrm{Tm}}(f) : \mathrm{pTm}(n) \to \mathrm{pTm}(m)}$$

- Substituting term variables in types

$$\frac{f : n \to \mathrm{pTm}(m)}{\mathrm{bind}_{\mathrm{Ty}}(f) : \mathrm{pTy}(n) \to \mathrm{pTy}(m)}$$

Untyped Syntax: Binding

$$\mathsf{pTm} : \mathsf{Nat} \to \mathsf{Set} \quad , \quad \mathsf{pTy} : \mathsf{Nat} \to \mathsf{Set}$$

- Substituting term variables in terms

$$\frac{f : n \to \mathsf{pTm}(m)}{\mathsf{bind}_{\mathsf{Tm}}(f) : \mathsf{pTm}(n) \to \mathsf{pTm}(m)}$$

- Substituting term variables in types

$$\frac{f : n \to \mathsf{pTm}(m)}{\mathsf{bind}_{\mathsf{Ty}}(f) : \mathsf{pTy}(n) \to \mathsf{pTy}(m)}$$

pTm is an ι-relative monad and pTy is a module over pTm.

Extrinsic Semantics

Typed Syntax

Well-typed Syntax: Terms, Types and Typing Derivations

We want to express well-typed terms and types in a context Γ with n free variables.

Such diagrams in Set:

$$\mathsf{pTm}(n) \xleftarrow{\mathsf{tm}_\Gamma} \mathsf{wTm}_\Gamma \xrightarrow{\Sigma_\Gamma} \mathsf{wTy}_\Gamma \xrightarrow{\mathsf{ty}_\Gamma} \mathsf{pTy}(n)$$

Intuitively:

- $\mathsf{pTm}(n)$ ($\mathsf{pTy}(n)$) is the set of raw terms (types)

Well-typed Syntax: Terms, Types and Typing Derivations

We want to express well-typed terms and types in a context Γ with n free variables.

Such diagrams in Set:

$$\mathrm{pTm}(n) \xleftarrow{\mathrm{tm}_\Gamma} \mathrm{wTm}_\Gamma \xrightarrow{\Sigma_\Gamma} \mathrm{wTy}_\Gamma \xrightarrow{\mathrm{ty}_\Gamma} \mathrm{pTy}(n)$$

Intuitively:

- $\mathrm{pTm}(n)$ ($\mathrm{pTy}(n)$) is the set of raw terms (types)
- wTm (wTy) is the set of derivations for well-typedness for terms (types)

Well-typed Syntax: Terms, Types and Typing Derivations

We want to express well-typed terms and types in a context Γ with n free variables.

Such diagrams in Set:

$$\mathrm{pTm}(n) \xleftarrow{\mathrm{tm}_\Gamma} \mathrm{wTm}_\Gamma \xrightarrow{\Sigma_\Gamma} \mathrm{wTy}_\Gamma \xrightarrow{\mathrm{ty}_\Gamma} \mathrm{pTy}(n)$$

$$w_0 \longmapsto A$$

Intuitively:

- $\mathrm{pTm}(n)$ ($\mathrm{pTy}(n)$) is the set of raw terms (types)
- wTm (wTy) is the set of derivations for well-typedness for terms (types)
- We have $w_0 : \Gamma \vdash A$ type if $\exists w_0 \in \mathrm{wTy}_\Gamma$ s.t. $\mathrm{ty}_\Gamma(w_0) = A$

Well-typed Syntax: Terms, Types and Typing Derivations

We want to express well-typed terms and types in a context Γ with n free variables.

Such diagrams in Set:

$$\begin{array}{ccccccc} \mathrm{pTm}(n) & \xleftarrow{\mathrm{tm}_\Gamma} & \mathrm{wTm}_\Gamma & \xrightarrow{\Sigma_\Gamma} & \mathrm{wTy}_\Gamma & \xrightarrow{\mathrm{ty}_\Gamma} & \mathrm{pTy}(n) \\ t & \longleftarrow\!\shortmid & w_1 & & \longmapsto & & A \end{array}$$

Intuitively:

- $\mathrm{pTm}(n)$ ($\mathrm{pTy}(n)$) is the set of raw terms (types)
- wTm (wTy) is the set of derivations for well-typedness for terms (types)
- We have $w_0 : \Gamma \vdash A$ type if $\exists w_0 \in \mathrm{wTy}_\Gamma$ s.t. $\mathrm{ty}_\Gamma(w_0) = A$
- We have $w_1 : \Gamma \vdash t : A$ if $\exists w_1 \in \mathrm{wTm}_\Gamma$ s.t. $\mathrm{tm}_\Gamma(w_1) = t$ and $\mathrm{ty}_\Gamma(\Sigma_\Gamma(w_1)) = A$

Well-typed Syntax: Substitution

A morphism from Γ to Δ is a triple $\sigma = (\sigma_0, \sigma_{\mathsf{Tm}}, \sigma_{\mathsf{Ty}})$:

$$\begin{tikzcd}
\Gamma \arrow[d, "\sigma"'] & \mathsf{pTm}(n) \arrow[d, "\mathsf{bind}_{\mathsf{Tm}}(\sigma_0)"'] & \mathsf{wTm}_\Gamma \arrow[l, "\mathsf{tm}_\Gamma"'] \arrow[r, "\Sigma_\Gamma"] \arrow[d, "\sigma_{\mathsf{Tm}}"] & \mathsf{wTy}_\Gamma \arrow[r, "\mathsf{ty}_\Gamma"] \arrow[d, "\sigma_{\mathsf{Ty}}"] & \mathsf{pTy}(n) \arrow[d, "\mathsf{bind}_{\mathsf{Ty}}(\sigma_0)"] \\
\Delta & \mathsf{pTm}(m) & \mathsf{wTm}_\Delta \arrow[l, "\mathsf{tm}_\Delta"] \arrow[r, "\Sigma_\Delta"'] & \mathsf{wTy}_\Delta \arrow[r, "\mathsf{ty}_\Delta"'] & \mathsf{pTy}(m)
\end{tikzcd}$$

Intuitively: Morphisms capture substitution.

This gives a category $\mathcal{J}_{\mathsf{pTm}}$ with an initial object:

$$\begin{tikzcd}
\mathsf{pTm}(0) & 0 \arrow[l, "\mathsf{var}_{\mathsf{Tm}}"'] \arrow[r, equal] & 0 \arrow[r, "\emptyset"] & \mathsf{pTy}(0)
\end{tikzcd}$$

Putting Things Together

A discrete subcategory of $\mathcal{J}_{\mathsf{pTm}}$ called $\mathcal{J}_M$ with objects:

$$\mathsf{pTm}(n) \xleftarrow{\mathsf{var}_{\mathsf{Tm}}} n = n \xrightarrow{\Gamma} \mathsf{pTy}(n)$$

Intuitively: this is like a precontext, n variables and their types (a list of length n of types).

Putting Things Together

A discrete subcategory of $\mathcal{J}_{\mathsf{pTm}}$ called $\mathcal{J}_M$ with objects:

$$\mathsf{pTm}(n) \xleftarrow{\mathsf{var}_{\mathsf{Tm}}} n = n \xrightarrow{\Gamma} \mathsf{pTy}(n)$$

Intuitively: this is like a precontext, n variables and their types (a list of length n of types).

A type theory (extrinsically) is a relative monad $M : \mathcal{J}_M \to \mathcal{J}_{\mathsf{pTm}}$ such that

$$\begin{tikzcd}
\mathcal{J}_M \arrow[rr, "M"] \arrow[dr, "\pi"'] & & \mathcal{J}_{\mathsf{pTm}} \arrow[dl, "\pi"] \\
& \mathsf{Kl}(\mathsf{pTm}) &
\end{tikzcd}$$

and some other coherences with $\mathsf{Kl}(\mathsf{pTm})$.

Semantics for the Extrinsic View

Definition

An ExtrinsicCwF consists of:

1. A relative monad $\mathsf{pTm} : \mathsf{Nat} \to \mathsf{Set}$
2. A relative module $\mathsf{pTy} : \mathsf{Nat} \to \mathsf{Set}$ over pTm
3. A discrete subcategory of $\mathcal{J}_{\mathsf{pTm}}$ called $\mathcal{J}_M$ 'with certain sort of objects'
4. A relative monad $M : \mathcal{J}_M \to \mathcal{J}_{\mathsf{pTm}}$ 'over' $\mathsf{Kl}(\mathsf{pTm})$

such that

5. 'empty context' is in $\mathcal{J}_M$
6. $\mathcal{J}_M$ is closed under 'context extension'

Semantics for the Extrinsic View

Definition

An ExtrinsicCwF consists of:

1. A relative monad $\mathsf{pTm} : \mathsf{Nat} \to \mathsf{Set}$
2. A relative module $\mathsf{pTy} : \mathsf{Nat} \to \mathsf{Set}$ over pTm
3. A discrete subcategory of $\mathcal{J}_{\mathsf{pTm}}$ called $\mathcal{J}_M$ 'with certain sort of objects'
4. A relative monad $M : \mathcal{J}_M \to \mathcal{J}_{\mathsf{pTm}}$ 'over' $\mathsf{Kl}(\mathsf{pTm})$

such that

5. 'empty context' is in $\mathcal{J}_M$
6. $\mathcal{J}_M$ is closed under 'context extension'

Context Extension

For $\Gamma \in \mathcal{J}_M$:

$$\mathrm{pTm}(n) \xleftarrow{\mathrm{var}_{\mathrm{Tm}}} n = n \xrightarrow{\Gamma} \mathrm{pTy}(n)$$

and $A \in \mathrm{Im}(\Gamma)$ we have $\Gamma.A \in \mathcal{J}_M$:

$$\mathrm{pTm}(n+1) \xleftarrow{\mathrm{var}_{\mathrm{Tm}_{n+1}}} n+1 = n+1 \xrightarrow{[\Gamma, A]} \mathrm{pTy}(n) \xrightarrow{\mathrm{pTy}(\iota_I)} \mathrm{pTy}(n+1)$$

Some Notation

Let $(\mathrm{pTm}, \mathrm{pTy}, \mathcal{J}_M, M)$ be a ExtCwF, and $\Gamma, \Delta \in \mathcal{J}_M$

1. We write $\pi : \Gamma \vdash_M A$ type for $\exists \pi \in \mathrm{wTy}_{M(\Gamma)}$ such that $\mathrm{ty}_{M(\Gamma)}(\pi) = A$
2. We write $\pi : \Gamma \vdash_M t : A$ for $\exists \pi \in \mathrm{wTm}_{M(\Gamma)}$ such that $\mathrm{tm}_{M(\Gamma)}(\pi) = t$ and $\mathrm{ty}_{M(\Gamma)}(\Sigma_{M(\Gamma)}(\pi)) = A$
3. We write $\sigma : \Gamma \rightarrow_M \Delta$ for a morphism $\sigma : \Gamma \rightarrow \Delta$ in $\mathrm{Kl}(M)$

$$\mathrm{pTm}(n) \xleftarrow{\mathrm{tm}_\Gamma} \mathrm{wTm}_\Gamma \xrightarrow{\Sigma_\Gamma} \mathrm{wTy}_\Gamma \xrightarrow{\mathrm{ty}_\Gamma} \mathrm{pTy}(n)$$

Substitution in ExtCwF

Proposition

In an ExtCwF we have:

$$\frac{\pi : \Gamma \vdash_M A \text{ type} \quad \sigma : \Gamma \to_M \Delta}{\pi[\sigma] : \Delta \vdash_M A[\sigma] \text{ type}} \qquad \frac{\pi : \Gamma \vdash_M t : A \quad \sigma : \Gamma \to_M \Delta}{\pi[\sigma] : \Delta \vdash_M t[\sigma] : A[\sigma]}$$

Proposition

In an ExtCwF we have:

$$\frac{\sigma' : \Gamma.A \to_M \Delta}{\sigma : \Gamma \to_M \Delta \quad \pi : \Delta \vdash_M t : A[\sigma]}$$

Bridging the Two Views

From Extrinsic to Intrinsic

We skip defining the (1-)category ExtCwF.

We want to build a functor:

$$\mathsf{CwF} \longleftarrow \mathsf{ExtCwF}$$

Here, I only discuss the action on objects.

From ExtCwF to CwF

We start from an ExtCwF $(\mathsf{pTm}, \mathsf{pTy}, \mathcal{J}_M, M)$ and build a CwF $(\mathcal{C}, \mathsf{Type}, \mathsf{Term}, \mathsf{ext})$.

- $\mathcal{C}$ is $\mathsf{Kl}(M)^{\mathsf{op}}$ quotiented by equality of the first component of morphisms

From ExtCwF to CwF

We start from an ExtCwF $(\mathrm{pTm}, \mathrm{pTy}, \mathcal{J}_M, M)$ and build a CwF $(\mathcal{C}, \mathrm{Type}, \mathrm{Term}, \mathrm{ext})$.

- $\mathcal{C}$ is $\mathrm{Kl}(M)^{\mathrm{op}}$ quotiented by equality of the first component of morphisms
- The presheaf $\mathrm{Type} : \mathcal{C}^{\mathrm{op}} \to \mathrm{Set}$ is:

$$\mathrm{Type} : \Gamma_n \mapsto \mathrm{Im}(\mathrm{ty}_{M\Gamma})$$

$$\mathrm{pTm}(n) \xleftarrow{\mathrm{tm}_\Gamma} \mathrm{wTm}_\Gamma \xrightarrow{\Sigma_\Gamma} \mathrm{wTy}_\Gamma \xrightarrow{\mathrm{ty}_\Gamma} \mathrm{pTy}(n)$$

From ExtCwF to CwF

We start from an ExtCwF $(\mathrm{pTm}, \mathrm{pTy}, \mathcal{J}_M, M)$ and build a CwF $(\mathcal{C}, \mathrm{Type}, \mathrm{Term}, \mathrm{ext})$.

- $\mathcal{C}$ is $\mathrm{Kl}(M)^{\mathrm{op}}$ quotiented by equality of the first component of morphisms
- The presheaf $\mathrm{Type} : \mathcal{C}^{\mathrm{op}} \to \mathrm{Set}$ is:

$$\mathrm{Type} : \Gamma_n \mapsto \mathrm{Im}(\mathrm{ty}_{M\Gamma})$$

- The presheaf $\mathrm{Term} : (\int_{\mathcal{C}} \mathrm{Type})^{\mathrm{op}} \to \mathrm{Set}$ is:

$$\mathrm{Term} : (\Gamma_n, A) \mapsto \text{The image under } \mathrm{tm}_{M\Gamma} \text{ of the preimage of } A$$

$$\mathrm{pTm}(n) \xleftarrow{\mathrm{tm}_\Gamma} \mathrm{wTm}_\Gamma \xrightarrow{\Sigma_\Gamma} \mathrm{wTy}_\Gamma \xrightarrow{\mathrm{ty}_\Gamma} \mathrm{pTy}(n)$$

From ExtCwF to CwF

We start from an ExtCwF $(\mathrm{pTm}, \mathrm{pTy}, \mathcal{J}_M, M)$ and build a CwF $(\mathcal{C}, \mathrm{Type}, \mathrm{Term}, \mathrm{ext})$.

- $\mathcal{C}$ is $\mathrm{Kl}(M)^{\mathrm{op}}$ quotiented by equality of the first component of morphisms
- The presheaf $\mathrm{Type} : \mathcal{C}^{\mathrm{op}} \to \mathrm{Set}$ is:

$$\mathrm{Type} : \Gamma_n \mapsto \mathrm{Im}(\mathrm{ty}_{M\Gamma})$$

- The presheaf $\mathrm{Term} : (\int_{\mathcal{C}} \mathrm{Type})^{\mathrm{op}} \to \mathrm{Set}$ is:

$$\mathrm{Term} : (\Gamma_n, A) \mapsto \text{The image under } \mathrm{tm}_{M\Gamma} \text{ of the preimage of } A$$

- ext follows from the intuitive context extension discussed before

Future Directions

Future Directions

- Define a notion of signature with an interpretation in both intrinsic and extrinsic views

$$\mathrm{Sig} \underset{\text{extrinsic model}}{\overset{\text{intrinsic model}}{\rightrightarrows}} \mathrm{Cat}$$

Future Directions

- Define a notion of signature with an interpretation in both intrinsic and extrinsic views
- Lift the construction discussed before to this notion of signature

$$\begin{tikzcd} \mathsf{Sig} \arrow[r, bend left, "\text{intrinsic model}"{name=U}] \arrow[r, bend right, "\text{extrinsic model}"'{name=D}] & \mathsf{Cat} \arrow[from=D, to=U, Rightarrow, shorten=2mm] \end{tikzcd}$$

Future Directions

- Define a notion of signature with an interpretation in both intrinsic and extrinsic views
- Lift the construction discussed before to this notion of signature
- Show that this passage from extrinsic to intrinsic preserves initial objects

$$\begin{tikzcd} \mathrm{Sig} \arrow[r, bend left, "\text{intrinsic model}"{name=U}] \arrow[r, bend right, "\text{extrinsic model}"'{name=D}] \arrow[from=D, to=U, Rightarrow, shorten=2mm] & \mathrm{Cat} \end{tikzcd}$$

Please Clap!

Questions?

Dybjer, Peter (1996). **"Internal type theory".** In: *Types for Proofs and Programs*. Springer Berlin Heidelberg, pp. 120–134. ISBN: 9783540707226. DOI: 10.1007/3-540-61780-9_66. URL: http://dx.doi.org/10.1007/3-540-61780-9_66.