Homotopy Type Theory Electronic Seminar Talks
the 4^{th} of December 2025

Different descriptions of the semantics of computation axioms

parts of the talk are based on joint work with Daniël Otten
(University of Amsterdam)

speaker Matteo Spadetto
(University of Nantes)

Content, *roughly*

How to *express* this:

$$\frac{\vdash A : \textsc{Type}}{\begin{array}{c} x, x' : A \vdash x = x' : \textsc{Type} \\ x : A \vdash r(x) : x = x \end{array}} \qquad \frac{\begin{array}{c} \vdash A : \textsc{Type} \\ x, x' : A;\ p : x = x' \vdash C(x, x', p) : \textsc{Type} \\ x : A \vdash q(x) : C(x, x, r(x)) \end{array}}{\begin{array}{c} x, x' : A;\ p : x = x' \vdash \mathsf{J}(q, x, x', p) : C(x, x', p) \\ x : A \vdash \mathsf{J}(q, x, x, r(x)) \equiv q(x) \end{array}}$$

in categorical structures.

The semantics of a dependent type theory can be seen as the class of copies of that theory, i.e. the categorical structures that can *express*, in this sense, the theory. This encoding is typically done using the arrows of these categorical structures.

The type theoretic language of a category

If $\mathcal{C}$ is a category with finite limits then:

- Objects Γ of $\mathcal{C}$ are contexts.

The type theoretic language of a category

If $\mathcal{C}$ is a category with finite limits then:

- Objects Γ of $\mathcal{C}$ are contexts.
- (Some, possibly all) arrows of codomain Γ are types A in context Γ

The type theoretic language of a category

If $\mathcal{C}$ is a category with finite limits then:

- Objects Γ of $\mathcal{C}$ are contexts.
- (Some, possibly all) arrows of codomain Γ are types A in context Γ and are usually denoted as $\Gamma.A \to \Gamma$.

The type theoretic language of a category

If $\mathcal{C}$ is a category with finite limits then:

- Objects Γ of $\mathcal{C}$ are contexts.
- (Some, possibly all) arrows of codomain Γ are types A in context Γ and are usually denoted as $\Gamma.A \to \Gamma$.
- Sections of $\Gamma.A \to \Gamma$ are terms of A in context Γ.

Seely, *Locally cartesian closed categories and type theory*, 1983.

Substitution

If we are given $\Delta \xrightarrow{f} \Gamma$ and $\Gamma.A \to \Gamma$ then the judgement:

$$\Delta \vdash A[f] : \textsc{Type}$$

is represented by the pullback:

$$\Delta.A[f] \to \Delta$$

of $\Gamma.A \to \Gamma$ along f.

Substitution

If we are given $\Delta \xrightarrow{f} \Gamma$ and $\Gamma.A \to \Gamma$ then the judgement:

$$\Delta \vdash A[f] : \textsc{Type}$$

is represented by the pullback:

$$\Delta.A[f] \to \Delta$$

of $\Gamma.A \to \Gamma$ along f.

If we are given a section $\Gamma \xrightarrow{t} \Gamma.A$ of $\Gamma.A \to \Gamma$ then the judgement:

$$\Delta \vdash t[f] : A[f]$$

is represented by the unique section of $\Delta.A[f] \to \Delta$ such that:

Substitution

If we are given $\Delta \xrightarrow{f} \Gamma$ and $\Gamma.A \to \Gamma$ then the judgement:

$$\Delta \vdash A[f] : \textsc{Type}$$

is represented by the pullback:

$$\Delta.A[f] \to \Delta$$

of $\Gamma.A \to \Gamma$ along f.

If we are given a section $\Gamma \xrightarrow{t} \Gamma.A$ of $\Gamma.A \to \Gamma$ then the judgement:

$$\Delta \vdash t[f] : A[f]$$

is represented by the unique section of $\Delta.A[f] \to \Delta$ such that:

$$\begin{array}{ccc} \Delta & \xrightarrow{\;f\;} & \Gamma \\ \downarrow & & \downarrow{\scriptstyle t} \\ \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{\;f\;} & \Gamma \end{array}$$

commutes.

Approaches to identify a model

In a categorical structure (with enough stuff) one can use this language to formulate type constructors of a theory and hence ask if this structure is or is not a model of a given inference rule.

These are some of the approaches to formulate models:

- the **syntactic** approach, *encoding type constructors into a model in alignment with the syntax*

Approaches to identify a model

In a categorical structure (with enough stuff) one can use this language to formulate type constructors of a theory and hence ask if this structure is or is not a model of a given inference rule.

These are some of the approaches to formulate models:

- the **syntactic** approach, *encoding type constructors into a model in alignment with the syntax*
- the **(higher) categorical** approach, *characterising type constructors via (higher) universal properties*

Approaches to identify a model

In a categorical structure (with enough stuff) one can use this language to formulate type constructors of a theory and hence ask if this structure is or is not a model of a given inference rule.

These are some of the approaches to formulate models:

- the **syntactic** approach, *encoding type constructors into a model in alignment with the syntax*
- the **(higher) categorical** approach, *characterising type constructors via (higher) universal properties*
- the **homotopy theoretic** approach, that *relies on a primitive notion of weak equivalence to phrase the type formers*

Intensional type constructors (with computation rules)

Intensional identity types

$$\frac{\vdash A : \text{Type}}{\begin{array}{c} x, x' : A \vdash x = x' : \text{Type} \\ x : A \vdash r(x) : x = x \end{array}}$$

$$\frac{\begin{array}{c} \vdash A : \text{Type} \\ x, x' : A;\ p : x = x' \vdash C(x, x', p) : \text{Type} \\ x : A \vdash q(x) : C(x, x, r(x)) \end{array}}{\begin{array}{c} x, x' : A;\ p : x = x' \vdash \mathsf{J}(q, x, x', p) : C(x, x', p) \\ x : A \vdash \quad \mathsf{J}(q, x, x, r(x)) \equiv q(x) \end{array}}$$

Dependent sum types

$$\frac{\begin{array}{c} \vdash A : \text{Type} \\ x : A \vdash B(x) : \text{Type} \end{array}}{\begin{array}{c} \vdash \Sigma_{x:A} B(x) : \text{Type} \\ x : A, y : B(x) \vdash \langle x, y \rangle : \Sigma_{x:A} B(x) \end{array}}$$

$$\frac{\begin{array}{c} \vdash A : \text{Type} \\ x : A \vdash B(x) : \text{Type} \\ u : \Sigma_{x:A} B(x) \vdash C(u) : \text{Type} \\ x : A;\ y : B(x) \vdash c(x, y) : C(\langle x, y \rangle) \end{array}}{\begin{array}{c} u : \Sigma_{x:A} B(x) \vdash \mathsf{split}(c, u) : C(u) \\ x : A;\ y : B(x) \vdash \quad \mathsf{split}(c, \langle x, y \rangle) \equiv c(x, y) \end{array}}$$

Axiomatic type constructors[1] (with computation *axioms*)

Axiomatic identity types

$$\frac{\vdash A : \textsc{Type}}{\begin{array}{c} x, x' : A \vdash x = x' : \textsc{Type} \\ x : A \vdash r(x) : x = x \end{array}}$$

$$\frac{\begin{array}{c} \vdash A : \textsc{Type} \\ x, x' : A;\ p : x = x' \vdash C(x, x', p) : \textsc{Type} \\ x : A \vdash q(x) : C(x, x, r(x)) \end{array}}{\begin{array}{c} x, x' : A;\ p : x = x' \vdash \mathsf{J}(q, x, x', p) : C(x, x', p) \\ x : A \vdash \mathsf{H}(q, x) : \mathsf{J}(q, x, x, r(x)) = q(x) \end{array}}$$

Axiomatic dependent sum types

$$\frac{\begin{array}{c} \vdash A : \textsc{Type} \\ x : A \vdash B(x) : \textsc{Type} \end{array}}{\begin{array}{c} \vdash \Sigma_{x:A} B(x) : \textsc{Type} \\ x : A, y : B(x) \vdash \langle x, y \rangle : \Sigma_{x:A} B(x) \end{array}}$$

$$\frac{\begin{array}{c} \vdash A : \textsc{Type} \\ x : A \vdash B(x) : \textsc{Type} \\ u : \Sigma_{x:A} B(x) \vdash C(u) : \textsc{Type} \\ x : A;\ y : B(x) \vdash c(x, y) : C(\langle x, y \rangle) \end{array}}{\begin{array}{c} u : \Sigma_{x:A} B(x) \vdash \mathsf{split}(c, u) : C(u) \\ x : A;\ y : B(x) \vdash \sigma(c, x, y) : \mathsf{split}(c, \langle x, y \rangle) = c(x, y) \end{array}}$$

[1] Also known as *weak*, *objective*, *propositional* theory.

How semantics works

We said that in an appropriate structure, e.g. a **display map category**, there are some arrows (display maps, that we can denoted as $\Gamma.A \to \Gamma$) that interpret type judgements $\Gamma \vdash A : \textsc{Type}$; and that term judgements $\Gamma \vdash t : A$ are interpreted as sections $\Gamma \to \Gamma.A$ of the corresponding display map.

How semantics works

We said that in an appropriate structure, e.g. a **display map category**, there are some arrows (display maps, that we can denoted as $\Gamma.A \to \Gamma$) that interpret type judgements $\Gamma \vdash A : \textsc{Type}$; and that term judgements $\Gamma \vdash t : A$ are interpreted as sections $\Gamma \to \Gamma.A$ of the corresponding display map.

To formulate a model of a type constructor:

- In the **syntactic approach** one copies the type constructor into a display map category by means of choice functions in the language of the display map category.

How semantics works

We said that in an appropriate structure, e.g. a **display map category**, there are some arrows (display maps, that we can denoted as $\Gamma.A \to \Gamma$) that interpret type judgements $\Gamma \vdash A : \textsc{Type}$; and that term judgements $\Gamma \vdash t : A$ are interpreted as sections $\Gamma \to \Gamma.A$ of the corresponding display map.

To formulate a model of a type constructor:

- In the **syntactic approach** one copies the type constructor into a display map category by means of choice functions in the language of the display map category.
- In the **category theoretic approach** one looks for a 1-dimensional categorical property to give to display maps that *characterises* the type constructor, allowing *a choice function as in the syntactic approach to be induced* by this property.

How semantics works

We said that in an appropriate structure, e.g. a **display map category**, there are some arrows (display maps, that we can denoted as $\Gamma.A \to \Gamma$) that interpret type judgements $\Gamma \vdash A : \textsc{Type}$; and that term judgements $\Gamma \vdash t : A$ are interpreted as sections $\Gamma \to \Gamma.A$ of the corresponding display map.

To formulate a model of a type constructor:

- In the **syntactic approach** one copies the type constructor into a display map category by means of choice functions in the language of the display map category.

- In the **category theoretic approach** one looks for a 1-dimensional categorical property to give to display maps that *characterises* the type constructor, allowing *a choice function as in the syntactic approach to be induced* by this property. The **homotopy theoretic approach** is similar, but the emphasis is on the family of weak equivalences.

An example: identity types

- **Syntactic approach**.
  For every display map $P_A : \Gamma.A \to \Gamma$, there is a choice of:
  - (*Form Rule*) a display map $\Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}$;

An example: identity types

- **Syntactic approach.**
  For every display map $P_A : \Gamma.A \to \Gamma$, there is a choice of:
  - (*Form Rule*) a display map $\Gamma.A.A^{\blacktriangledown}.\operatorname{id}_A \to \Gamma.A.A^{\blacktriangledown}$;
  - (*Intro Rule*) a section:
    $$\mathsf{refl}_A : \Gamma.A \to \Gamma.A.\operatorname{id}_A[v_A]$$
    of the display map $\Gamma.A.\operatorname{id}_A[v_A] \to \Gamma.A$.

An example: identity types

- **Syntactic approach.**
  For every display map $P_A : \Gamma.A \to \Gamma$, there is a choice of:
  - (*Form Rule*) a display map $\Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}$;
  - (*Intro Rule*) a section:
    $$\mathsf{refl}_A : \Gamma.A \to \Gamma.A.\mathrm{id}_A[v_A]$$
    of the display map $\Gamma.A.\mathrm{id}_A[v_A] \to \Gamma.A$.

  For every display map $P_A : \Gamma.A \to \Gamma$, every display map:
  $$P_C : \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A$$
  and every section
  $$c : \Gamma.A \to \Gamma.A.C[v_A^{\bullet}\mathsf{refl}_A]$$
  of $P_{C[v_A^{\bullet}\mathsf{refl}_A]}$, there is a choice of:
  - (*Elim Rule*) a section:
    $$\mathsf{J}_c : \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C$$
    of the display map $\Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A$;

An example: identity types

- **Syntactic approach.**
  For every display map $P_A : \Gamma.A \to \Gamma$, there is a choice of:
  - (*Form Rule*) a display map $\Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}$;
  - (*Intro Rule*) a section:
    $$\mathsf{refl}_A : \Gamma.A \to \Gamma.A.\mathrm{id}_A[v_A]$$
    of the display map $\Gamma.A.\mathrm{id}_A[v_A] \to \Gamma.A$.

  For every display map $P_A : \Gamma.A \to \Gamma$, every display map:
  $$P_C : \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A$$
  and every section
  $$c : \Gamma.A \to \Gamma.A.C[v_A^{\bullet}\mathsf{refl}_A]$$
  of $P_{C[v_A^{\bullet}\mathsf{refl}_A]}$, there is a choice of:
  - (*Elim Rule*) a section:
    $$\mathsf{J}_c : \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C$$
    of the display map $\Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A$;
  - (*Comp Axiom*) a section:
    $$\mathsf{H}_c : \Gamma.A \to \Gamma.A.\mathrm{id}_{C[\mathsf{r}_A]}[\mathsf{J}_c[\mathsf{r}_A]; c]$$
    of the display map $\Gamma.A.\mathrm{id}_{C[\mathsf{r}_A]}[\mathsf{J}_c[\mathsf{r}_A]; c] \to \Gamma.A$.

An example: identity types

- **Syntactic approach.**
  For every display map $P_A : \Gamma.A \to \Gamma$, there is a choice of:
  - (*Form Rule*) a display map $\Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}$;
  - (*Intro Rule*) a section:
    $$\mathsf{refl}_A : \Gamma.A \to \Gamma.A.\mathrm{id}_A[v_A]$$
    of the display map $\Gamma.A.\mathrm{id}_A[v_A] \to \Gamma.A$.

  For every display map $P_A : \Gamma.A \to \Gamma$, every display map:
  $$P_C : \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A$$
  and every section
  $$c : \Gamma.A \to \Gamma.A.C[v_A^{\bullet}\mathsf{refl}_A]$$
  of $P_{C[v_A^{\bullet}\mathsf{refl}_A]}$, there is a choice of:
  - (*Elim Rule*) a section:
    $$\mathsf{J}_c : \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C$$
    of the display map $\Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A.C \to \Gamma.A.A^{\blacktriangledown}.\mathrm{id}_A$;
  - (*Comp Axiom*) a section:
    $$\mathsf{H}_c : \Gamma.A \to \Gamma.A.\mathrm{id}_{C[\mathsf{r}_A]}[\mathsf{J}_c[\mathsf{r}_A]; c]$$
    of the display map $\Gamma.A.\mathrm{id}_{C[\mathsf{r}_A]}[\mathsf{J}_c[\mathsf{r}_A]; c] \to \Gamma.A$.

An example: identity types

- **Categorical approach.**
  If the identity types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$, the arrow $v_A : \Gamma.A \to \Gamma.A.A^{\blacktriangledown}$ (obtained by factoring the pair $(1_{\Gamma.A}, 1_{\Gamma.A})$ through $\Gamma.A.A^{\blacktriangledown}$) is isomorphic to a display map.

An example: identity types

- **Categorical approach.**
  If the identity types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$, the arrow $v_A : \Gamma.A \to \Gamma.A.A^\blacktriangledown$ (obtained by factoring the pair $(1_{\Gamma.A}, 1_{\Gamma.A})$ through $\Gamma.A.A^\blacktriangledown$) is isomorphic to a display map.

  If the identity types are intensional/axiomatic. More complicated.

An example: identity types

- **Categorical approach.**
  If the identity types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$, the arrow $v_A : \Gamma.A \to \Gamma.A.A^{\blacktriangledown}$ (obtained by factoring the pair $(1_{\Gamma.A}, 1_{\Gamma.A})$ through $\Gamma.A.A^{\blacktriangledown}$) is isomorphic to a display map.

  If the identity types are intensional/axiomatic. More complicated.

Another example: Dependent sum types.

- **Syntactic approach.**
  As before, rewriting the inference rules.

An example: identity types

- **Categorical approach.**
  If the identity types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$, the arrow $v_A : \Gamma.A \to \Gamma.A.A^{\blacktriangledown}$ (obtained by factoring the pair $(1_{\Gamma.A}, 1_{\Gamma.A})$ through $\Gamma.A.A^{\blacktriangledown}$) is isomorphic to a display map.

  If the identity types are intensional/axiomatic. More complicated.

Another example: Dependent sum types.

- **Syntactic approach.**
  As before, rewriting the inference rules.
- **Categorical approach.**
  If the dependent sum types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$ and every display map $P_B : \Gamma.A.B \to \Gamma.A$, the composition $P_A P_B$ is isomorphic to a display map.

An example: identity types

- **Categorical approach.**
  If the identity types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$, the arrow $v_A : \Gamma.A \to \Gamma.A.A^{\blacktriangledown}$ (obtained by factoring the pair $(1_{\Gamma.A}, 1_{\Gamma.A})$ through $\Gamma.A.A^{\blacktriangledown}$) is isomorphic to a display map.

  If the identity types are intensional/axiomatic. More complicated.

Another example: Dependent sum types.

- **Syntactic approach.**
  As before, rewriting the inference rules.
- **Categorical approach.**
  If the dependent sum types are extensional. For every display map $P_A : \Gamma.A \to \Gamma$ and every display map $P_B : \Gamma.A.B \to \Gamma.A$, the composition $P_A P_B$ is isomorphic to a display map.

  If the dependent sum types are intensional/axiomatic. More complicated.

For **extensional** type theory, the categorical approach is clear and conceptually simple to formulate.

This is not the case for **intensional**, and **axiomatic**, type theory: there aren't obvious categorical properties to characterise intensional and axiomatic inference rules.

For **extensional** type theory, the categorical approach is clear and conceptually simple to formulate.

This is not the case for **intensional**, and **axiomatic**, type theory: there aren't obvious categorical properties to characterise intensional and axiomatic inference rules.

Higher categorical approach. In order to characterise intensional type constructors, we can use **higher dimensional models**, *that still can be converted into ordinary models according to the syntactic approach*, and higher dimensional - e.g. weakly universal - categorical properties.

For **extensional** type theory, the categorical approach is clear and conceptually simple to formulate.

This is not the case for **intensional**, and **axiomatic**, type theory: there aren't obvious categorical properties to characterise intensional and axiomatic inference rules.

Higher categorical approach. In order to characterise intensional type constructors, we can use **higher dimensional models**, *that still can be converted into ordinary models according to the syntactic approach*, and higher dimensional - e.g. weakly universal - categorical properties.

A line of research: *how to adapt this approach to axiomatic type constructors, and hence identify a higher dimensional structure with natural categorical conditions that allow to interpret axiomatic theory.*

Finitely complete categories

Seely proved that, as display map categories, finitely complete categories are *almost* models of extensional identity types and extensional dependent sum types.

Finitely complete categories

Seely proved that, as display map categories, finitely complete categories are *almost* models of extensional identity types and extensional dependent sum types.

If we are given $\Omega \xrightarrow{g} \Delta \xrightarrow{f} \Gamma$ and A in context Γ and t of type A, then:

- $\Omega.A[f][g] \cong \Omega.A[fg]$ and not necessarily $\Omega.A[f][g] \equiv \Omega.A[fg]$

Finitely complete categories

Seely proved that, as display map categories, finitely complete categories are *almost* models of extensional identity types and extensional dependent sum types.

If we are given $\Omega \xrightarrow{g} \Delta \xrightarrow{f} \Gamma$ and A in context Γ and t of type A, then:

- $\Omega.A[f][g] \cong \Omega.A[fg]$ and not necessarily $\Omega.A[f][g] \equiv \Omega.A[fg]$
- $$\begin{array}{ccc} & \overset{t[f][g]}{\nearrow} & \Omega.A[f][g] \\ \Omega & & \wr\Vert \\ & \underset{t[fg]}{\searrow} & \Omega.A[fg] \end{array}$$ and not necessarily $t[f][g] \equiv t[fg]$

Finitely complete categories

Seely proved that, as display map categories, finitely complete categories are *almost* models of extensional identity types and extensional dependent sum types.

If we are given $\Omega \xrightarrow{g} \Delta \xrightarrow{f} \Gamma$ and A in context Γ and t of type A, then:

- $\Omega.A[f][g] \cong \Omega.A[fg]$ and not necessarily $\Omega.A[f][g] \equiv \Omega.A[fg]$
- $$\begin{array}{ccc} & \overset{t[f][g]}{\nearrow} & \Omega.A[f][g] \\ \Omega & & \wr\!\Vert \\ & \underset{t[fg]}{\searrow} & \Omega.A[fg] \end{array}$$ and not necessarily $t[f][g] \equiv t[fg]$

However $\Omega \vdash A[f][g] \equiv A[f[g]]$ and $\Omega \vdash t[f][g] \equiv t[f[g]]$ are derivable: in this sense we do not necessarily have a genuine model.

Hofmann's coherence result

However, in:

Hofmann, *On the Interpretation of Type Theory in Locally Cartesian Closed Categories*, 1994.

every finitely complete category is shown to be equivalent to a *split* display map category (still endowed with extensional $=$, and Σ), where 'split' means that there is a choice of pullback squares and $A[f][g] \equiv A[f[g]]$.

Hofmann's coherence result

However, in:

Hofmann, *On the Interpretation of Type Theory in Locally Cartesian Closed Categories*, 1994.

every finitely complete category is shown to be equivalent to a *split* display map category (still endowed with extensional $=$, and Σ), where 'split' means that there is a choice of pullback squares and $A[f][g] \equiv A[f[g]]$.

In general, we have a **right-adjoint splitting theorem**: the inclusion:

$$\{\text{split display map categories over } \mathcal{C}\} \hookrightarrow \{\text{display map categories over } \mathcal{C}\}$$

has a right adjoint that:

Hofmann's coherence result

However, in:

Hofmann, *On the Interpretation of Type Theory in Locally Cartesian Closed Categories*, 1994.

every finitely complete category is shown to be equivalent to a *split* display map category (still endowed with extensional $=$, and Σ), where 'split' means that there is a choice of pullback squares and $A[f][g] \equiv A[f[g]]$.

In general, we have a **right-adjoint splitting theorem**: the inclusion:

$$\{\text{split display map categories over } \mathcal{C}\} \hookrightarrow \{\text{display map categories over } \mathcal{C}\}$$

has a right adjoint that:

- maps every given display map category into a split one which is equivalent to the given one;
- (under some pseudo-stability condition) preserves extensional $=$ and Σ structure.

Hofmann's coherence result

However, in:

Hofmann, *On the Interpretation of Type Theory in Locally Cartesian Closed Categories*, 1994.

every finitely complete category is shown to be equivalent to a *split* display map category (still endowed with extensional $=$, and Σ), where 'split' means that there is a choice of pullback squares and $A[f][g] \equiv A[f[g]]$.

In general, we have a **right-adjoint splitting theorem**: the inclusion:

$$\{\text{split display map categories over } \mathcal{C}\} \hookrightarrow \{\text{display map categories over } \mathcal{C}\}$$

has a right adjoint that:

- maps every given display map category into a split one which is equivalent to the given one;
- (under some pseudo-stability condition) preserves extensional $=$ and Σ structure.

Warren, *Homotopy Theoretic Aspects of Constructive Type Theory*, 2008.

Streicher, *Fibred categories à la Jean Bénabou*, 2018.

Hofmann's coherence result

Clairambault & Dybjer, and Maietti proved that there exists a biequivalence between:

- the 2-category of finitely complete categories
- an appropriate 2-category for the *contextual models of extensional* $=$-*types* (+ 1 and Σ)

Hofmann's coherence result

Clairambault & Dybjer, and Maietti proved that there exists a biequivalence between:

- the 2-category of finitely complete categories
- an appropriate 2-category for the *contextual models of extensional =-types* (+ 1 and Σ)

There's a **left-adjoint splitting theorem** too: the inclusion:

$$\{\text{split display map categories over } \mathcal{C}\} \hookrightarrow \{\text{cloven display map categories over } \mathcal{C}\}$$

has left adjoint. It:

Hofmann's coherence result

Clairambault & Dybjer, and Maietti proved that there exists a biequivalence between:

- the 2-category of finitely complete categories
- an appropriate 2-category for the *contextual models of extensional* $=$*-types* ($+$ 1 and Σ)

There's a **left-adjoint splitting theorem** too: the inclusion:

$$\{\text{split display map categories over } \mathcal{C}\} \hookrightarrow \{\text{cloven display map categories over } \mathcal{C}\}$$

has left adjoint. It:

- maps every cloven display map category into a split one which is equivalent to the given one;
- (under some weak-stability condition) preserves the semantic intensional $=$ and Σ structure.

Lumsdaine, Warren, *The local universes model*, 2015.

One may ask if there exists a coherence result for axiomatic identity types as well.

One may ask if there exists a coherence result for axiomatic identity types as well.

Some reasons to study axiomatic type theory: broader concept of semantics, conservativity.

Path categories i.e. non-genuine models of axiomatic identity types

A path category $\mathcal{C}$ is a category with a terminal object, a class of fibrations and a class of weak equivalences such that the following properties are satisfied:

1. The composition of two fibrations is a fibration as well.
2. Every pullback of a fibration exists and is a fibration as well.
3. Every pullback of an acyclic fibration is a trivial fibration as well.
4. Weak equivalences satisfy 2-out-of-six.
5. Every isomorphism is a trivial fibration and every trivial fibration has a section.
6. For every object X of $\mathcal{C}$ there is an object PX, called path object on X, together with a weak equivalence $X \xrightarrow{r} PX$ and a fibration $PX \xrightarrow{\langle s,t\rangle} X \times X$ such that $(X \xrightarrow{r} PX \xrightarrow{\langle s,t\rangle} X \times X) = \delta_X$.
7. Every arrow of target a terminal object is a fibration.

Path categories i.e. non-genuine models of axiomatic identity types

A path category $\mathcal{C}$ is a category with a terminal object, a class of fibrations and a class of weak equivalences such that the following properties are satisfied:

1. The composition of two fibrations is a fibration as well.
2. Every pullback of a fibration exists and is a fibration as well.
3. Every pullback of an acyclic fibration is a trivial fibration as well.
4. Weak equivalences satisfy 2-out-of-six.
5. Every isomorphism is a trivial fibration and every trivial fibration has a section.
6. For every object X of $\mathcal{C}$ there is an object PX, called path object on X, together with a weak equivalence $X \xrightarrow{r} PX$ and a fibration $PX \xrightarrow{\langle s,t \rangle} X \times X$ such that $(X \xrightarrow{r} PX \xrightarrow{\langle s,t \rangle} X \times X) = \delta_X$.
7. Every arrow of target a terminal object is a fibration.

Theorem. Path categories are contextual display map categories with extensional 1 and Σ types and axiomatic $=$ types, and vice versa.

Path categories i.e. non-genuine models of axiomatic identity types

A path category $\mathcal{C}$ is a category with a terminal object, a class of fibrations and a class of weak equivalences such that the following properties are satisfied:

1. The composition of two fibrations is a fibration as well.
2. Every pullback of a fibration exists and is a fibration as well.
3. Every pullback of an acyclic fibration is a trivial fibration as well.
4. Weak equivalences satisfy 2-out-of-six.
5. Every isomorphism is a trivial fibration and every trivial fibration has a section.
6. For every object X of $\mathcal{C}$ there is an object PX, called path object on X, together with a weak equivalence $X \xrightarrow{r} PX$ and a fibration $PX \xrightarrow{\langle s,t \rangle} X \times X$ such that $(X \xrightarrow{r} PX \xrightarrow{\langle s,t \rangle} X \times X) = \delta_X$.
7. Every arrow of target a terminal object is a fibration.

Theorem. Path categories are contextual display map categories with extensional 1 and Σ types and axiomatic $=$ types, and vice versa.

This statement extends to a result à la Clairambault & Dybjer.

Rough statement

There exists a *biequivalence* between:

- the 2-category of the *contextual models of axiomatic =-types* (+ other constructors)

Rough statement

There exists a *biequivalence* between:

- the 2-category of the *contextual models of axiomatic =-types* (+ other constructors)
- the 2-category of *path categories*

Rough statement

There exists a *biequivalence* between:

- the 2-category of the *contextual models of axiomatic =-types* (+ other constructors)
- the 2-category of *path categories*

in the manner of the biequivalence between:

- the 2-category of the *contextual models of extensional =-types* (+ other constructors)
- the 2-category of finitely complete categories

studied by Seely, Hofmann, Clairambault & Dybjer, and Maietti.

The precise statement

Semantics of $=_{\mathrm{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$

$$\simeq$$

$$\{\text{models of } =_{\mathrm{ext}} \text{ (and } \Sigma_{\mathrm{ext}}, 1_{\mathrm{ext}})\}$$

The precise statement

Semantics of $=_{\text{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ext}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}})\}$$

Semantics of $=_{\text{ax}}$

$$\{\text{path categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ax}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}})\}$$

The precise statement

Universal data are pseudo-unique

The precise statement

Universal data are pseudo-unique

Homotopy universal data are weakly-unique

The precise statement

Semantics of $=_{\text{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ext}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}})\}$$

Semantics of $=_{\text{ax}}$

$$\{\text{path categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ax}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}})\}$$

The precise statement

Semantics of $=_{\text{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$

$$\simeq$$

$$\{\text{models of } =_{\text{ext}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}\text{) and data pseudo-preserving morphisms}\}$$

Semantics of $=_{\text{ax}}$

$$\{\text{path categories and structure preserving functors}\}$$

$$\simeq$$

$$\{\text{models of } =_{\text{ax}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}\text{)}\}$$

The precise statement

Semantics of $=_{\text{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ext}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}\text{) and data pseudo-preserving morphisms}\}$$

Semantics of $=_{\text{ax}}$

$$\{\text{path categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ax}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}\text{) and data weakly-preserving morphisms}\}$$

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify.*

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify*.

Finitely complete categories induce *pseudo-stable data*.

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify*.

Finitely complete categories induce *pseudo-stable data*.

$\Longrightarrow$

These data can be preserved by the ***right***-*adjoint splitting*.

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify*.

Finitely complete categories induce *pseudo-stable data*.

$\Longrightarrow$

These data can be preserved by the ***right***-*adjoint splitting*.

Path categories induce *weakly-stable data*.

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify*.

Finitely complete categories induce *pseudo-stable data*.

$\Longrightarrow$

These data can be preserved by the ***right****-adjoint splitting*.

Path categories induce *weakly-stable data*.

$\Longrightarrow$

These data are only preserved by the ***left****-adjoint splitting*.

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify*.

Finitely complete categories induce *pseudo-stable data*.

$\Longrightarrow$

These data can be preserved by the ***right****-adjoint splitting*.

Path categories induce *weakly-stable data*.

$\Longrightarrow$

These data are only preserved by the ***left****-adjoint splitting*.

$\Longrightarrow$

Models in input need to be *cloven*.

The precise statement

In principle, finitely complete categories and path categories are *pre-models that we need to strictify*.

Finitely complete categories induce *pseudo-stable data*.
$\Longrightarrow$
These data can be preserved by the ***right***-*adjoint splitting*.

Path categories induce *weakly-stable data*.
$\Longrightarrow$
These data are only preserved by the ***left***-*adjoint splitting*.
$\Longrightarrow$
Models in input need to be *cloven*. Additionally, they need to have *function variable contexts* (LF condition).

The precise statement

Semantics of $=_{\text{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ext}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}\text{) and data pseudo-preserving morphisms}\}$$

Semantics of $=_{\text{ax}}$

$$\{\text{path categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ax}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}\text{) and data weakly-preserving morphisms}\}$$

The precise statement

Semantics of $=_{\text{ext}}$

$$\{\text{finitely complete categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ext}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}) \text{ and data pseudo-preserving morphisms}\}$$

Semantics of $=_{\text{ax}}$

$$\{\text{LF cloven path categories and structure preserving functors}\}$$
$$\simeq$$
$$\{\text{models of } =_{\text{ax}} \text{ (and } \Sigma_{\text{ext}}, 1_{\text{ext}}) \text{ and data weakly-preserving morphisms}\}$$

- In particular, we obtained a *coherence result* for path categories.

- In particular, we obtained a *coherence result* for path categories.
- These results can be specialised to include other type formers (axiomatic Π types) and generalised to drop extensional ones (display map path categories).

- In particular, we obtained a *coherence result* for path categories.
- These results can be specialised to include other type formers (axiomatic Π types) and generalised to drop extensional ones (display map path categories).
- There is also a *categorical counterpart*, where models are formulated using *higher category theoretic* data rather than *homotopy theoretic data*.

- In particular, we obtained a *coherence result* for path categories.
- These results can be specialised to include other type formers (axiomatic Π types) and generalised to drop extensional ones (display map path categories).
- There is also a *categorical counterpart*, where models are formulated using *higher category theoretic* data rather than *homotopy theoretic data*.

In every path category, a higher-dimensional category is hidden.

[Den Besten] A 2-morphism between parallel morphisms $f, g : \Delta \to \Gamma$ is a morphism $h : \Delta \to P\Gamma$ that constitutes a homotopy $f \simeq g$.

In every path category, a higher-dimensional category is hidden.

[Den Besten] A 2-morphism between parallel morphisms $f, g : \Delta \to \Gamma$ is a morphism $h : \Delta \to P\Gamma$ that constitutes a homotopy $f \simeq g$.

Analogously, since homotopies are also 1-morphisms, we can talk about homotopies between homotopies and so on. For the sake of simplicity, let us stop to dimension 2.

In every path category, a higher-dimensional category is hidden.

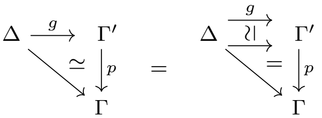
[Den Besten] A 2-morphism between parallel morphisms $f, g : \Delta \to \Gamma$ is a morphism $h : \Delta \to P\Gamma$ that constitutes a homotopy $f \simeq g$.

Analogously, since homotopies are also 1-morphisms, we can talk about homotopies between homotopies and so on. For the sake of simplicity, let us stop to dimension 2.

In this setting:

Proposition. Path categories are enriched in groupoids [den Besten] and:

- fibrations $p : \Gamma' \to \Gamma$ are cloven isofibrations:



- path objects are homotopy arrow objects: $(\mathcal{C}/\Gamma)(\Delta, \Gamma')^{\to} \simeq (\mathcal{C}/\Gamma)(\Delta, P_\Gamma \Gamma')$;
- pullbacks are also 2-pullbacks.

In every path category, a higher-dimensional category is hidden.

If we are given a type judgement $\Gamma \vdash A : \textsc{Type}$ and substitutions $\Delta \vdash f : \Gamma$ and $\Delta \vdash g : \Gamma.A$ (i.e. g is given by $g_1 : \Gamma \, , \, g_2 : A[g_1]$) with a context identity proof $\Delta \vdash p : f = g_1$, then the lifted 1-cell is the substitution:

$$\Delta \vdash f : \Gamma \, , \, p^* g_2 : A[f]$$

and the lifted homotopy is the context identity proof $\Delta \vdash f \, , \, p^* g_2 \quad = \quad g_1 \, , \, g_2$ provided by the list:

$$\begin{aligned} \Delta \vdash \quad & p : & f = g_1 \\ \Delta \vdash & \mathsf{r}(p^* g_2) : & p^* g_2 = p^* g_2 \end{aligned}$$

of identity proofs. Now, if $f \equiv g_1$ and $p \equiv \mathsf{r}(g_1)$, then the lifted 1-cell is:

$$\Delta \vdash g_1 : \Gamma \, , \, \mathsf{r}(g_1)^* g_2 : A[g_1]$$

and the lifted homotopy is the list:

$$\begin{aligned} \Delta \vdash \quad & \mathsf{r}(g_1) : & g_1 = g_1 \\ \Delta \vdash & \mathsf{r}(\mathsf{r}(g_1)^* g_2) : & \mathsf{r}(g_1)^* g_2 = \mathsf{r}(g_1)^* g_2 \end{aligned}$$

hence in general the former will not be g and the latter will not be the identity 2-cell: with *axiomatic* identity types we can infer that $\mathsf{r}(g_1)^* g_2 = g_2$ (it is a fragment of the computation axiom for $=$-types) but not that $\mathsf{r}(g_1)^* g_2 \equiv g_2$.

In every path category, a higher-dimensional category is hidden.

If we are given a type judgement $\Gamma \vdash A : \text{Type}$ and substitutions $\Delta \vdash f : \Gamma$ and $\Delta \vdash g : \Gamma.A$ (i.e. g is given by $g_1 : \Gamma\,,\, g_2 : A[g_1]$) with a context identity proof $\Delta \vdash p : f = g_1$, then the lifted 1-cell is the substitution:

$$\Delta \vdash f : \Gamma\,,\, p^* g_2 : A[f]$$

and the lifted homotopy is the context identity proof $\Delta \vdash f\,,\, p^* g_2 \;=\; g_1\,,\, g_2$ provided by the list:

$$\begin{aligned} \Delta \vdash \quad p : \quad & f = g_1 \\ \Delta \vdash \mathsf{r}(p^* g_2) : \; & p^* g_2 = p^* g_2 \end{aligned}$$

of identity proofs. Now, if $f \equiv g_1$ and $p \equiv \mathsf{r}(g_1)$, then the lifted 1-cell is:

$$\Delta \vdash g_1 : \Gamma\,,\, \mathsf{r}(g_1)^* g_2 : A[g_1]$$

and the lifted homotopy is the list:

$$\begin{aligned} \Delta \vdash \quad \mathsf{r}(g_1) : \quad & g_1 = g_1 \\ \Delta \vdash \mathsf{r}(\mathsf{r}(g_1)^* g_2) : \; & \mathsf{r}(g_1)^* g_2 = \mathsf{r}(g_1)^* g_2 \end{aligned}$$

hence in general the former will not be g and the latter will not be the identity 2-cell: with *axiomatic* identity types we can infer that $\mathsf{r}(g_1)^* g_2 = g_2$ (it is a fragment of the computation axiom for $=$-types) but not that $\mathsf{r}(g_1)^* g_2 \equiv g_2$.
This is precisely why the display map associated to the type A is a cloven isofibration but not necessarily a normal isofibration.

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

1. The class of display maps is closed under **2-dimensional re-indexing**.

$$
\begin{array}{ccc}
 & & \Gamma.A \\
 & & \downarrow \\
\Delta & \xrightarrow{f} & \Gamma
\end{array}
\quad \Rightarrow \quad
\begin{array}{ccc}
\Delta.A[f] & \longrightarrow & \Gamma.A \\
\downarrow & \lrcorner & \downarrow \\
\Delta & \xrightarrow{f} & \Gamma
\end{array}
$$

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

1. The class of display maps is closed under **2-dimensional re-indexing**.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array}$$

2. Every display map is a **cloven isofibration**.

$$\begin{array}{ccc} \Delta & \longrightarrow & \Gamma.A \\ & \searrow \;\Rightarrow & \downarrow \\ & & \Gamma \end{array} \quad = \quad \begin{array}{ccc} \Delta & \overset{\longrightarrow}{\underset{\longrightarrow}{\Uparrow}} & \Gamma.A \\ & \searrow \;= & \downarrow \\ & & \Gamma \end{array}$$

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

1. The class of display maps is closed under **2-dimensional re-indexing**.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array}$$

2. Every display map is a **cloven isofibration**.

$$\begin{array}{ccc} \Delta & \longrightarrow & \Gamma.A \\ & \Rightarrow & \downarrow \\ & & \Gamma \end{array} \quad = \quad \begin{array}{ccc} \Delta & \overset{\longrightarrow}{\underset{\longrightarrow}{\Uparrow}} & \Gamma.A \\ & = & \downarrow \\ & & \Gamma \end{array}$$

3. Every display map has a **homotopy arrow object**.

$$\begin{array}{ccc} \Omega & \overset{\longrightarrow}{\underset{\longrightarrow}{\Downarrow}} & \Gamma.A \\ & & \downarrow \\ & & \Gamma \end{array} \quad \text{s.t.} \quad \begin{array}{c} \hom_\Gamma(\Delta, \Omega) \\ \wr\| \\ \hom_\Gamma(\Delta, \Gamma.A)^{\to} \end{array}$$

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

1. The class of display maps is closed under **2-dimensional re-indexing**.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array}$$

2. Every display map is a **cloven isofibration**.

$$\begin{array}{ccc} \Delta & \longrightarrow & \Gamma.A \\ & \searrow \Rightarrow & \downarrow \\ & & \Gamma \end{array} \quad = \quad \begin{array}{ccc} \Delta & \overset{\longrightarrow}{\underset{\longrightarrow}{\Uparrow}} & \Gamma.A \\ & \searrow = & \downarrow \\ & & \Gamma \end{array}$$

3. Every display map has a **homotopy arrow object**.

$$\begin{array}{ccc} \Omega & \overset{\longrightarrow}{\underset{\longrightarrow}{\Downarrow}} & \Gamma.A \\ & \searrow & \downarrow \\ & & \Gamma \end{array} \quad \text{s.t.} \quad \begin{array}{c} \hom_\Gamma(\Delta, \Omega) \\ \wr\! \mid \\ \hom_\Gamma(\Delta, \Gamma.A)^{\rightarrow} \end{array}$$

4. The class of display maps is closed under composition, up to **equivalence**.

$$\begin{array}{ccc} \Gamma.A.B & & \\ \downarrow & & \\ \Gamma.A & \longrightarrow & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Gamma.A.B & \simeq & \Gamma.C \\ \downarrow & & \downarrow \\ \Gamma.A & \longrightarrow & \Gamma \end{array}$$

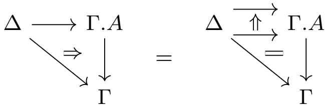
2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

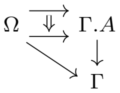
1. To substitute into types and terms.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \overset{f}{\dashrightarrow} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{\ f\ } & \Gamma \end{array}$$

2. Every display map is a **cloven isofibration**.



3. Every display map has a **homotopy arrow object**.



s.t.

$$\hom_\Gamma(\Delta, \Omega) \cong \hom_\Gamma(\Delta, \Gamma.A)^{\rightarrow}$$

4. The class of display maps is closed under composition, up to **equivalence**.

$$\begin{array}{ccc} \Gamma.A.B & & \\ \downarrow & & \\ \Gamma.A & \longrightarrow & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Gamma.A.B & \simeq & \Gamma.C \\ \downarrow & & \downarrow \\ \Gamma.A & \longrightarrow & \Gamma \end{array}$$

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

1. To substitute into types and terms.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \overset{f}{\dashrightarrow} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \overset{f}{\longrightarrow} & \Gamma \end{array}$$

2. Every display map is a **cloven isofibration**.

$$\begin{array}{ccc} \Delta & \longrightarrow & \Gamma.A \\ & \searrow \; \Rightarrow & \downarrow \\ & & \Gamma \end{array} \quad = \quad \begin{array}{ccc} \Delta & \overset{\longrightarrow}{\underset{\longrightarrow}{\Uparrow}} & \Gamma.A \\ & \searrow \; = & \downarrow \\ & & \Gamma \end{array}$$

3. To have identity types with *pseudo-elimination*.

$$\begin{array}{ccc} \Omega & \overset{\longrightarrow}{\underset{\longrightarrow}{\Downarrow}} & \Gamma.A \\ & \searrow & \downarrow \\ & & \Gamma \end{array} \quad \text{s.t.} \quad \begin{array}{c} \hom_\Gamma(\Delta, \Omega) \\ \wr\!\mid \\ \hom_\Gamma(\Delta, \Gamma.A)^{\rightarrow} \end{array} \qquad \begin{array}{c} \Gamma.A.A'.(x = x') \\ \shortparallel \\ \Omega \end{array}$$

4. The class of display maps is closed under composition, up to **equivalence**.

$$\begin{array}{ccc} \Gamma.A.B & & \\ \downarrow & & \\ \Gamma.A & \longrightarrow & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Gamma.A.B & \simeq & \Gamma.C \\ \downarrow & & \downarrow \\ \Gamma.A & \longrightarrow & \Gamma \end{array}$$

2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

1. To substitute into types and terms.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \overset{f}{\dashrightarrow} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{\;f\;} & \Gamma \end{array}$$

2. Every display map is a **cloven isofibration**.

$$\begin{array}{ccc} \Delta & \longrightarrow & \Gamma.A \\ & \searrow \;\Rightarrow & \downarrow \\ & & \Gamma \end{array} \quad = \quad \begin{array}{ccc} \Delta & \overset{\longrightarrow}{\underset{\longrightarrow}{\Uparrow}} & \Gamma.A \\ & \searrow \;= & \downarrow \\ & & \Gamma \end{array}$$

3. To have identity types with *pseudo-elimination*.

$$\begin{array}{ccc} \Omega & \overset{\longrightarrow}{\underset{\longrightarrow}{\Downarrow}} & \Gamma.A \\ & \searrow & \downarrow \\ & & \Gamma \end{array} \quad \text{s.t.} \quad \begin{array}{c} \hom_\Gamma(\Delta,\Omega) \\ \wr| \\ \hom_\Gamma(\Delta,\Gamma.A)^{\to} \end{array} \qquad \begin{array}{c} \Gamma.A.A'.(x = x') \\ \vdots\!\!| \\ \Omega \end{array}$$

4. To have dependent sum types with *pseudo-elimination*.

$$\begin{array}{c} \Gamma.A.B \\ \downarrow \\ \Gamma.A \longrightarrow \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Gamma.A.B & \simeq & \Gamma.C \\ \downarrow & & \downarrow \\ \Gamma.A & \longrightarrow & \Gamma \end{array} \qquad \begin{array}{c} \Sigma_A^B \\ \vdots\!\!| \\ C \end{array}$$

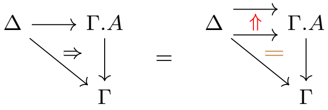
2-dimensional semantics of axiomatic theories

Display map 2-categories. (2,1)-dimensional categories with a specified class of 1-morphisms, called **display maps**, that satisfy the following conditions:

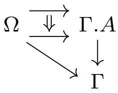
1. To substitute into types and terms.

$$\begin{array}{ccc} & & \Gamma.A \\ & & \downarrow \\ \Delta & \overset{f}{\dashrightarrow} & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Delta.A[f] & \longrightarrow & \Gamma.A \\ \downarrow & \lrcorner & \downarrow \\ \Delta & \xrightarrow{f} & \Gamma \end{array}$$

2. To strictify eliminations in 3-4 in change of producing computation *axioms*.



3. To have identity types with *pseudo-elimination*.



s.t. $\quad \mathrm{hom}_\Gamma(\Delta,\Omega) \cong \mathrm{hom}_\Gamma(\Delta,\Gamma.A)^{\rightarrow} \qquad \Gamma.A.A'.(x = x') \doteqdot \Omega$

4. To have dependent sum types with *pseudo-elimination*.

$$\begin{array}{ccc} \Gamma.A.B & & \\ \downarrow & & \\ \Gamma.A & \longrightarrow & \Gamma \end{array} \quad \Rightarrow \quad \begin{array}{ccc} \Gamma.A.B & \simeq & \Gamma.C \\ \downarrow & & \downarrow \\ \Gamma.A & \longrightarrow & \Gamma \end{array} \qquad \Sigma_A^B \doteqdot C$$

2-dimensional semantics of axiomatic theories

Theorem. *Display map 2-categories are models of axiomatic dependent type theory.*

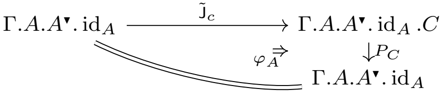
In detail

Under the hypotheses of the elimination rule of identity types, we are able to build a pseudo-term:

$$\begin{array}{ccc} \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A & \xrightarrow{\tilde{\jmath}_c} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A\,.C \\ & \varphi_A^{\Rightarrow} & \downarrow p_C \\ & \overset{}{=\!=\!=\!=\!=} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A \end{array}$$

In detail

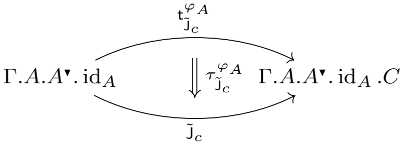
Under the hypotheses of the elimination rule of identity types, we are able to build a pseudo-term:



and, using the cloven isofibration structure on P_C, we obtain a section:

$$\mathsf{t}_{\tilde{\mathsf{J}}_c}^{\varphi_A} : \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A \to \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A\,.C$$

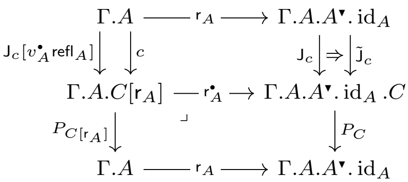
of P_C, at the cost of introducing an additional 2-cell:



We define $\mathsf{J}_c := \mathsf{t}_{\tilde{\mathsf{J}}_c}^{\varphi_A}$.

In detail

Now, referring to the diagram:



we obtain a 2-cell $\mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A] \Rightarrow c$.

In detail

Now, referring to the diagram:

$$\begin{array}{ccc}
\Gamma.A & \xrightarrow{\quad \mathsf{r}_A \quad} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A \\
{\scriptstyle \mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A]}\big\downarrow \;\; \big\downarrow{\scriptstyle c} & & {\scriptstyle \mathsf{J}_c}\big\downarrow \Rightarrow \big\downarrow{\scriptstyle \tilde{\mathsf{J}}_c} \\
\Gamma.A.C[\mathsf{r}_A] & \xrightarrow{\quad \mathsf{r}^{\bullet}_A \quad} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A\,.C \\
{\scriptstyle P_{C[\mathsf{r}_A]}}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle P_C} \\
\Gamma.A & \xrightarrow{\quad \mathsf{r}_A \quad} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A
\end{array}$$

we obtain a 2-cell $\mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A] \Rightarrow c$.

Remark. If P_C is normal, then:

$$\mathsf{J}_c\mathsf{r}_A = t^{\varphi_A * \mathsf{r}_A}_{\tilde{\mathsf{J}}_c\mathsf{r}_A} = t^{1_{\mathsf{r}_A}}_{\tilde{\mathsf{J}}_c\mathsf{r}_A} = \tilde{\mathsf{J}}_c\mathsf{r}_A$$

$$(\mathsf{J}_c \Rightarrow \tilde{\mathsf{J}}_c) * \mathsf{r}_A = \tau^{\varphi_A * \mathsf{r}_A}_{\tilde{\mathsf{J}}_c\mathsf{r}_A} = \tau^{1_{\mathsf{r}_A}}_{\tilde{\mathsf{J}}_c\mathsf{r}_A} = 1_{\tilde{\mathsf{J}}_c\mathsf{r}_A}$$

implying that $\mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A]$ is in fact c.

In detail

Now, referring to the diagram:

$$
\begin{array}{ccc}
\Gamma.A & \xrightarrow{\quad \mathsf{r}_A \quad} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A \\
{\scriptstyle \mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A]}\Big\downarrow \; \Big\downarrow{\scriptstyle c} & & {\scriptstyle \mathsf{J}_c}\Big\downarrow \Rightarrow \Big\downarrow{\scriptstyle \tilde{\mathsf{J}}_c} \\
\Gamma.A.C[\mathsf{r}_A] & \xrightarrow{\quad \mathsf{r}^{\bullet}_A \quad} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A\,.C \\
{\scriptstyle P_{C[\mathsf{r}_A]}}\Big\downarrow & \lrcorner & \Big\downarrow{\scriptstyle P_C} \\
\Gamma.A & \xrightarrow{\quad \mathsf{r}_A \quad} & \Gamma.A.A^{\blacktriangledown}.\,\mathrm{id}_A
\end{array}
$$

we obtain a 2-cell $\mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A] \Rightarrow c$.

Remark. If P_C is normal, then:

$$
\mathsf{J}_c \mathsf{r}_A = t^{\varphi_A * \mathsf{r}_A}_{\tilde{\mathsf{J}}_c \mathsf{r}_A} = t^{1_{\mathsf{r}_A}}_{\tilde{\mathsf{J}}_c \mathsf{r}_A} = \tilde{\mathsf{J}}_c \mathsf{r}_A
$$

$$
(\mathsf{J}_c \Rightarrow \tilde{\mathsf{J}}_c) * \mathsf{r}_A = \tau^{\varphi_A * \mathsf{r}_A}_{\tilde{\mathsf{J}}_c \mathsf{r}_A} = \tau^{1_{\mathsf{r}_A}}_{\tilde{\mathsf{J}}_c \mathsf{r}_A} = 1_{\tilde{\mathsf{J}}_c \mathsf{r}_A}
$$

implying that $\mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A]$ is in fact c.

However, if P_C is just cloven, then $\mathsf{J}_c[v^{\bullet}_A \mathsf{refl}_A]$ and c can remain different.

An application:

A revisitation of the groupoid model.

We consider the (2,1)-category GRPD of groupoids, functors, and natural transformations (i.e. natural isomorphisms) with **Grothendieck constructions of *pseudofunctors*** $\mathbf{\Gamma \to GRPD}$ as display maps over Γ.

An application:

A revisitation of the groupoid model.

We consider the (2,1)-category GRPD of groupoids, functors, and natural transformations (i.e. natural isomorphisms) with **Grothendieck constructions of *pseudofunctors* $\boldsymbol{\Gamma \to}$ GRPD** as display maps over Γ.
The model of axiomatic theory induced by this display map 2-category does not believe the judgemental computation rule.

An application:

A revisitation of the groupoid model.

We consider the (2,1)-category GRPD of groupoids, functors, and natural transformations (i.e. natural isomorphisms) with **Grothendieck constructions of *pseudofunctors* $\boldsymbol{\Gamma \to}$ GRPD** as display maps over Γ.
The model of axiomatic theory induced by this display map 2-category does not believe the judgemental computation rule.

In particular, the judgemental computation rule for intensional identity type constructor is independent of the axiomatic dependent type theory.

Thank you!