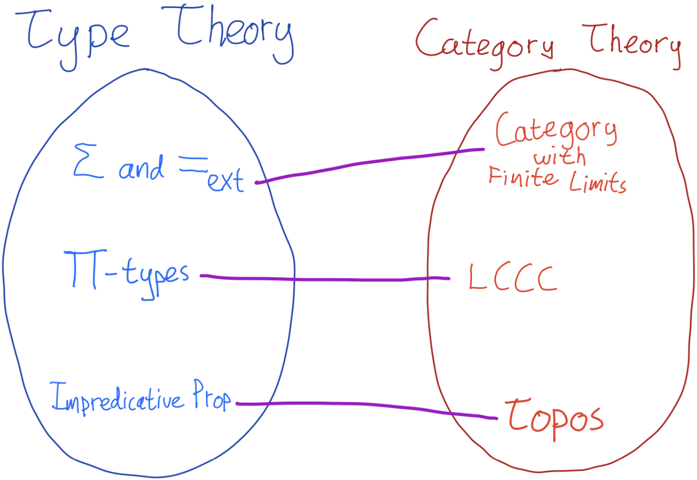
The Internal Language of Univalent Categories

Niels van der Weide

June 13, 2024

Type Theory and Category Theory



Internal Language Theorems

Theorem (Theorem 6.1 in Clairambault&Dybjer 2014[1])

We have a biequivalence betweeen the bicategories

- $\mathsf{CwF}_{\mathsf{dem}}^{\sum,=_{\mathsf{ext}}}$*: democratic comprehension categories with extensional identity types and sigma types*
- FinLim*: finitely complete categories*

This biequivalence can be extended to $\prod$*-types and LCCCs*

[1] Clairambault, Pierre, and Peter Dybjer. "The biequivalence of locally cartesian closed categories and Martin-Löf type theories.

Internal Language Up To Isomorphism

Final sentence of the paper by Clairambault and Dybjer:

> *So we can ask whether Martin-Löf type theory with extensional identity types, $\sum$- and $\prod$-types is an internal language for lcccs? And we can answer, yes, it is an internal language 'up to isomorphism'.*

Internal Language **Up To Isomorphism**

Final sentence of the paper by Clairambault and Dybjer:

*So we can ask whether Martin-Löf type theory with extensional identity types, $\sum$- and $\prod$-types is an internal language for lcccs? And we can answer, yes, it is an internal language '***up to isomorphism***'.*

Category Theory in Univalent Foundations

Recall from 1 hour ago:

- In **univalent foundations**, there are two notions of category: **univalent** categories and **strict** categories
- We can thus consider **internal language theorems** for both notions of category
- For **strict categories**: we can follow Clairambault and Dybjer verbatim
- For **univalent categories**: this is more interesting and subtle

This Talk

Goal: what is the internal language of univalent categories?

Theorem

We have a biequivalence betweeen the bicategories

- DFLCompCat*: univalent democratic comprehension categories that support unit types, equalizer types, binary product types, and strong* $\sum$*-types*
- FinLim*: univalent finitely complete categories*

We can extend this biequivalence to

- $\prod$*-types and LCCCs*
- *pretoposes,* $\prod$*-pretoposes*
- *elementary toposes*

Note: the proof is formalized using UniMath

The Remainder

I will comment on two things in the proof

- Why do I use comprehension categories?
- How is univalence used in the proof?

Reject Discreteness

- In a CwF, we have a **presheaf** of types
- So: for every context Γ, we have a **set** of types in Γ
- However, in UF **the type of sets is not a set**: it is a **groupoid**
- We thus do not have a CwF where the types in the empty context are sets

Reject Discreteness

- In a CwF, we have a **presheaf** of types
- So: for every context Γ, we have a **set** of types in Γ
- However, in UF **the type of sets is not a set**: it is a **groupoid**
- We thus do not have a CwF where the types in the empty context are sets

Note:

- This is also the basis for the talk "Coherent Categories with Families" by Altenkirch and Kaposi
- One could use a different notion of set (iterative sets) and obtain a CwF of iterative sets ("The Category of Iterative Sets in Homotopy Type Theory and Univalent Foundations" by Gratzer, Gylterud, Mörtberg, Stenholm)

Accept Higher Categories

- We need to use **higher categorical structure**
- We want a **pseudofunctor** of type: for every context Γ, a **category** of types in Γ

How do we represent such pseudofunctors?

- Algebraic style: we have to deal with **coherence** manually
- Alternative: use **universal properties** and coherence comes for free
- So, we use **fibrations** and **comprehension categories**

Univalence

Definition

A category is **univalent** if the map sending identities $x = y$ to isomorphisms $x \cong y$ is an equivalence of types.

Univalence

Definition

A category is **univalent** if the map sending identities $x = y$ to isomorphisms $x \cong y$ is an equivalence of types.

Definition

A bicategory is **univalent** if

- the map sending identities $x = y$ to adjoint equivalence $x \simeq y$ is an equivalence of types
- the map sending identities $f = g$ to invertible 2-cells $f \cong g$ is an equivalence of types

Univalence

Definition
A category is **univalent** if the map sending identities $x = y$ to isomorphisms $x \cong y$ is an equivalence of types.

Definition
A bicategory is **univalent** if

- the map sending identities $x = y$ to adjoint equivalence $x \equiv y$ is an equivalence of types
- the map sending identities $f = g$ to invertible 2-cells $f \cong g$ is an equivalence of types

We can show that all categories and bicategories in this talk are univalent

Univalence is nice

Univalence simplifies proofs of statements like

for all x and y and for all equivalences $e : x \cong y$, we have $P(e)$

By equivalence induction, we can assume that e is the identity

Univalence is nice

Univalence simplifies proofs of statements like

for all x and y and for all equivalences $e : x \cong y$, we have $P(e)$

By equivalence induction, we can assume that e is the identity
We use this to:

- transport properties/structure along equivalences
- characterize adjoint equivalences, e.g.
  - to prove that pointwise pseudonatural adjoint equivalences are adjoint equivalences
  - to characterize adjoint equivalences of comprehension categories

Characterizing Adjoint Equivalences

Often we want to show that some pseudofunctor reflects adjoint equivalences

$$e : x \to y \qquad\qquad \mathcal{B}_2$$

$$\Big\downarrow P$$

$$Pe : Px \simeq Py \qquad\qquad \mathcal{B}_1$$

Characterizing Adjoint Equivalences

This works better if we use **displayed bicategories**

$\bar{e} : \bar{x} \to_e \bar{y}$ $\qquad\qquad \mathcal{D}$

$e : x \simeq y$ $\qquad\qquad \mathcal{B}$

By induction on e: we only have to consider morphisms over identities

And there's more

There are more interesting features of the proof

- usage of displayed biequivalence (see Bicategories in univalent foundations)
- local properties (based on Modular correspondence between dependent type theories and categories including pretopoi and topoi by Maietti)

Conclusion

- We gave versions of the theorem by Clairambault and Dybjer for univalent categories, and we extended it to toposes
- We used comprehension categories instead of CwFs, since we don't want the types to form a set
- Univalence also helped us to simplify parts of the proof (transporting structure/properties along equivalences, characterizing adjoint equivalences)
- The results in this talk are formalized: `https://github.com/UniMath/UniMath/tree/master/UniMath/Bicategories/ComprehensionCat`