Towards elementary ∞-toposes

Michael Shulman[1]

[1](University of San Diego)

September 13, 2018
Vladimir Voevodsky Memorial Conference
Institute for Advanced Study

Outline

Voevodsky's most important contribution

As Dan Grayson told us on Tuesday, Vladimir himself said that his main accomplishment wasn't univalence, but rather this:

From the Foundations Coq library (now part of UniMath)

```
Definition iscontr T := Σcntr:T, Πt:T, t=cntr.

Fixpoint isofhlevel (n:nat) (X:UU) : UU
:= match n with
   | O => iscontr X
   | S m => Πx:X, Πx':X, (isofhlevel m (x=x'))
   end.

Definition isweq {X Y:UU} (f:X -> Y) : UU :=
  Πy:Y, iscontr (hfiber f y).
```

Why? Consider the history. . .

Hofmann–Streicher, 1996

"The question whether UIP [Uniqueness of Identity Proofs] is valid in intensional Martin-Löf type theory was open for a while. . . we answer the question of derivability of UIP in pure type theory in the negative by exhibiting a counter model. . . . every type will be a *groupoid*, i.e. a category with isomorphisms only. A posteriori this justifies a view of propositional equality in type theory as a notion of isomorphism."

– Hofmann and Streicher, *The groupoid interpretation of type theory*, 1996

Baez–Dolan, 2006

"…given objects x, y in an n-category, there is an $(n-1)$-category called $\hom(x, y)$…A couple of objects in $\hom(x, y)$ give an $(n-2)$-category, and so on.…

If x and y are parallel n-morphisms in an n-category, then $\hom(x, y)$ is a (-1)-category.…if $x = y$ there is one object in $\hom(x, y)$, otherwise there's none.…Thus, there are just two (-1)-categories. You could think of them as the 1-element set and the empty set.… We can also call them … 'True' and 'False.'…

If we have two parallel $(n+1)$-morphisms in an n-category, they are both identities…so they're equal.…So there's just one (-2)-category. …the 1-element set, or 'True'."

– John Baez, *Lectures on n-categories and cohomology*, 2006

Awodey–Warren, 2007

"… we show that a form of Martin-Löf type theory can be soundly modelled in any model category. … Because Martin-Löf type theory is, in one form or another, the theoretical basis for many of the computer proof assistants currently in use … this promise of applications is of a practical, as well as theoretical, nature.

"… the idea underlying the interpretation of type theory… is *Fibrations as Types*… the natural interpretation of the identity type $\mathrm{Id}_A(a, b)$ … should be a path object for A. … this interpretation soundly models a form of type theory with identity types…"

– Awodey and Warren, *Homotopy-theoretic models of identity types*, 2009 (preprint 2007)

Hofmann–Streicher again, 1996

"... we want to make an extension of type theory taking account of the fact that propositional equality on a universe is isomorphism. ... We write $Iso(A, B)$ for the set

$$\Sigma([f : A \to B]\Sigma([g : B \to A]Id(g \circ f, id) \times Id(f \circ g, id)))$$

... Now let U be a universe of discrete groupoids ... if $A, B : U$ then the interpretations of $Iso(A, B)$ and $Id(U, A, B)$ are isomorphic. One direction of the isomorphism is syntactically definable as `id_iso`... Like in the case of functional extensionality we can now syntactically postulate an inverse to the function `id_iso`... By analogy to functional extensionality we refer to this extension by *universe extensionality*."

– Hofmann and Streicher, *The groupoid interpretation of type theory*, 1996

What was missing?

Hofmann and Streicher's *universe extensionality* is literally the univalence axiom... except that their definition of isomorphism:

$$Iso(A, B) \stackrel{\text{def}}{=} \Sigma([f : A \to B]\Sigma([g : B \to A]Id(g \circ f, id) \times Id(f \circ g, id)))$$

is no longer the right thing to use when the types in U are no longer *discrete* groupoids. In fact:

Theorem (HoTT Book Exercise 4.6)

It is inconsistent to have two nested universes $U_0 : U_1$ both satisfying Hofmann–Streicher's universe extensionality.

What's wrong with isomorphisms?

$$\mathsf{isIso}(f : A \to B) \overset{\mathsf{def}}{=} \sum_{g:B\to A} \Big(\mathsf{Id}(g \circ f, \mathsf{id}_A) \times \mathsf{Id}(f \circ g, \mathsf{id}_B)\Big)$$

Consider the case when f is $\mathsf{id}_A : A \to A$. Then

$$\mathsf{isIso}(\mathsf{id}_A) \equiv \sum_{g:A\to A} \Big(\mathsf{Id}(g, \mathsf{id}_A) \times \mathsf{Id}(g, \mathsf{id}_B)\Big)$$

So a proof that id_A is an "isomorphism" consists of an endomorphism together with *two* homotopies relating it to id_A. Any such triple (g, p, q) is homotopic to $(\mathsf{id}_A, \mathsf{refl}_{\mathsf{id}_A}, p^{-1} \cdot q)$, but the superfluous data of an auto-homotopy $p^{-1} \cdot q$ of id_A cannot be eliminated.

The state of the field pre-Voevodsky

Some snippets from a conversation on the *n*-Category Café blog in December 2009, before any of us were aware of Voevodsky's work:

John Baez:. . . ultimately mathematics will be founded on ∞-categories, with sets appearing as a simple *special case*.

Me: . . . it may turn out that . . . intensional type theory . . . [is] a theory of ∞-groupoids . . . which would reduce to ordinary set theory when we restrict attention to the ∞-groupoids in which all hom-spaces are empty or contractible.

Peter Lumsdaine: . . . asking a hom-set to be contractible involves arbitrary high dimensions, and I'm not sure what kind of language . . . would let you talk about that. . . while a type may have infinitely high non-trivial structure, internally you can only work with it via its finite-dimensional approximations.

https://golem.ph.utexas.edu/category/2009/11/feferman_set_theory.html

The state of the field pre-Voevodsky

Some snippets from a conversation on the *n*-Category Café blog in December 2009, before any of us were aware of Voevodsky's work:

John Baez: … ultimately mathematics will be founded on ∞-categories, with sets appearing as a simple *special case*.

Me: … it may turn out that … intensional type theory … [is] a theory of ∞-groupoids … which would reduce to ordinary set theory when we restrict attention to the ∞-groupoids in which all hom-spaces are empty or contractible.

Peter Lumsdaine: … asking a hom-set to be contractible involves arbitrary high dimensions, and I'm not sure what kind of language … would let you talk about that … while a type may have infinitely high non-trivial structure, internally you can only work with it via its finite-dimensional approximations.

https://golem.ph.utexas.edu/category/2009/11/feferman_set_theory.html

Voevodsky equivalences

$$\mathsf{isContr}(A) \stackrel{\text{def}}{=} \sum_{x:A} \prod_{y:A} \mathsf{Id}(x,y)$$

$$\mathsf{isEquiv}(f : A \to B) \stackrel{\text{def}}{=} \prod_{y:B} \mathsf{isContr}(\sum_{x:A} \mathsf{Id}(f(x),y))$$

$$\mathsf{Equiv}(A,B) \stackrel{\text{def}}{=} \sum_{f:A\to B} \mathsf{isEquiv}(f)$$

I remember coming to CMU in February 2010 to hear Vladimir present his ideas. (Especially because of the snowstorm that trapped us all there for several extra days!)

In the pub afterwards, we were all talking excitedly, not about univalence, but about this definition of equivalence. That conversation led to a proof that $f : A \to B$ is a Voevodsky equivalence if and only if it is a Hofmann–Streicher isomorphism. But the types $\mathsf{isIso}(f)$ and $\mathsf{isEquiv}(f)$ are not equivalent.

Why was this not obvious?

$$\mathsf{isContr}(A) \stackrel{\mathsf{def}}{=} \sum_{x:A} \prod_{y:A} \mathsf{Id}(x, y)$$

- From the standard propositions-as-types point of view, this says that "A is a singleton": it has an element x such that every other element y is equal to x.
- But it was not obvious to us that in the homotopical interpretation this means full contractibility.
- Looked at wrongly, it seems to say "there is a point x such that every other point y is connected to x by a path", which is a statement of connectedness rather than contractibility.
- It works because the homotopical $\prod$ is a type of continuous functions. So the paths connecting all the y's to x vary continuously with y, giving a contracting homotopy.

Other notions of equivalence

Today we know many other equivalent ways to define $\mathsf{isEquiv}(f)$, such as:

$$\Big(\sum_{g:B\to A}\mathsf{Id}(g\circ f,\mathsf{id}_A)\Big)\times\Big(\sum_{h:B\to A}\mathsf{Id}(f\circ h,\mathsf{id}_B)\Big)$$

$$\sum_{g:B\to A}\sum_{p:\mathsf{Id}(g\circ f,\mathsf{id}_A)}\sum_{q:\mathsf{Id}(f\circ g,\mathsf{id}_B)}\mathsf{Id}(\mathsf{ap}_f(p),q_f)$$

$$\Big(\sum_{g:B\to A}\mathsf{Id}(f\circ g,\mathsf{id}_B)\Big)$$
$$\times\Big(\prod_{x,y:A}\sum_{h:\mathsf{Id}(f(x),f(y))\to\mathsf{Id}(x,y)}\mathsf{Id}(\mathsf{ap}_f\circ h,\mathsf{id})\Big)$$

As often happens, the hard part is solving a problem for the first time; it's much easier to find additional solutions afterwards.

More about Voevodsky equivalences

Clearly the naive definition of isomorphism doesn't work, but how can we tell that Voevodsky's definition (and the others) is correct? Well. . .

1. The resulting univalence axiom holds in models.
2. The type isEquiv(f) is a *proposition*, or (-1)-groupoid. In Voevodsky's terminology, we have

```
Definition isaprop := isofhlevel 1.
Theorem isapropisweq {X Y:UU} (f:X->Y): isaprop (isweq f).
```

Today I want to talk about another viewpoint on this correctness, and an entire new subject that it makes possible.

Outline

Categorical foundations for mathematics

- Lawvere (1964): The basic rules of set theory can be expressed as properties of the category **Set** of sets and functions.
- Therefore: Insofar as mathematics can be coded into set theory, it can also be coded into any category with properties like **Set**.
- Lawvere and Tierney (late 1960s): Grothendieck's "toposes of sheaves" satisfy almost all the properties of **Set**. This is also enough to code mathematics into them!

"Definition"

An elementary topos is a category with all the finitary properties of the category **Set** except

- Excluded middle: monos need not have complements.
- The axiom of choice: epis need not split.
- Well-pointedness: the terminal object need not be a generator.

(Finitary makes it *first-order*, hence axiomatic like ZFC.)

So what?

Why is this useful?

1. If we prove a substantial theorem, then code it into a concrete topos of sheaves, we get a nontrivial result about sheaves for free (internal languages).
2. By carefully constructing new toposes of sheaves, we can construct new universes of mathematics that satisfy new axioms (forcing).

Representability

More precisely:

Definition

An elementary topos is a category with finite limits, finite colimits, exponentials, and a subobject classifier.

These are all representability properties: they say that some functor or other is representable.

- Products: $\mathcal{E}(-, A) \times \mathcal{E}(-, B) : \mathcal{E}^{\mathrm{op}} \to \mathbf{Set}$ is representable.
- Coproducts: $\mathcal{E}(A, -) \times \mathcal{E}(B, -) : \mathcal{E} \to \mathbf{Set}$ is representable.
- Similarly, other finite limits and colimits.
- Exponentials: $\mathcal{E}(- \times A, B) : \mathcal{E}^{\mathrm{op}} \to \mathbf{Set}$ is representable.
- Subobject classifier: $\mathrm{Sub} : \mathcal{E}^{\mathrm{op}} \to \mathbf{Set}$ is representable.

(This definition is actually redundant, but that's irrelevant.)

Fiberwise structure

Theorem (The fundamental theorem of topos theory)

If $\mathcal{E}$ is an elementary topos, then so is any slice category $\mathcal{E}/X$.

(objects = arrows $A \to X$; morphisms = commutative triangles)

Why is this important?

- In set theory, we can talk about sets of sets, and perform set-operations on them.

$$\{ A_i \mid i \in X \} \quad \{ B_i \mid i \in X \} \quad \rightsquigarrow \quad \{ A_i \times B_i \mid i \in X \}$$

- In topos theory, an X-indexed family is an arrow $A \to X$, i.e. an object of $\mathcal{E}/X$. We think of "A_i" as the "fiber over $i \in X$".

$$A \to X \qquad B \to X \qquad \rightsquigarrow \qquad A \times_X B \to X$$

So $\mathcal{E}/X$ needs to have all the same structure that $\mathcal{E}$ does.

Local cartesian closure

In particular, $\mathcal{E}$ is locally cartesian closed:

$$\mathrm{Hom}_{\mathcal{E}/X}\left(\begin{array}{c} A \times_X B \\ \downarrow \\ X \end{array}, \begin{array}{c} C \\ \downarrow \\ X \end{array} \right) \cong \mathrm{Hom}_{\mathcal{E}/X}\left(\begin{array}{c} A \\ \downarrow \\ X \end{array}, \begin{array}{c} C_X^B \\ \downarrow \\ X \end{array} \right)$$

("Local" in topos theory means "in every slice".)

This is equivalent to: for any $f : Y \to X$, the pullback functor

$$f^* : \mathcal{E}/X \to \mathcal{E}/Y$$

has a right adjoint, denoted f_* or Π_f.
(It always has a left adjoint, $f_!$ or Σ_f, given by composition with f.)

Outline

The subobject classifier

Definition

A subobject classifier Ω is a representing object for

$$\begin{array}{rcl} \mathrm{Sub} : \mathcal{E}^{\mathrm{op}} & \to & \mathbf{Set} \\ X & \mapsto & \{\text{isomorphism classes of monos } U \rightarrowtail X\} \end{array}$$

In particular, it has a universal subobject $\top : 1 \rightarrowtail \Omega$, whose domain turns out to be terminal.

Examples

- In **Set**, $\Omega = \{\, \bot, \top \,\}$ is the set of truth values.
  The universal property is about characteristic functions.
- In sheaves on a space X, Ω is the sheaf of open subsets of X.

The internal language of propositions

A subobject classifier allows us to "talk internally" about propositions or truth values. Using exponentials, we can therefore also talk internally about subsets (via characteristic functions).

Example

We can define the "object of Dedekind reals": the classifier for pairs of subobjects of $\mathbb{Q}$ satisfying the usual axioms. In sheaves on a space X, this is the sheaf of continuous real-valued functions on X.

However, often we want to talk about sets, not just subsets.

Coarse object classifiers

Definition

A coarse object classifier is a representing object for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & \mathbf{Set} \\ X & \mapsto & \{\text{isomorphism classes of objects of } \mathcal{E}/X\} \end{array}$$

Such things do occasionally exist, but rarely.

Slogan

When things can be isomorphic in more than one way, passing to isomorphism classes loses too much information.

So we keep around at least the isomorphisms.

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & \textbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & \textbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Um. . . but now $\mathcal{E}(X, U)$ has to be a groupoid rather than a set. . .

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & \textbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Um. . . but now $\mathcal{E}(X, U)$ has to be a groupoid rather than a set. . .
. . . which means $\mathcal{E}$ has to be a 2-category. . .

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & 2\text{-}\mathbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Um...but now $\mathcal{E}(X, U)$ has to be a groupoid rather than a set...

...which means $\mathcal{E}$ has to be a 2-category...

...which means the core of $\mathcal{E}/X$ is also a 2-groupoid...

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & 2\text{-}\mathbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Um... but now $\mathcal{E}(X, U)$ has to be a groupoid rather than a set...
...which means $\mathcal{E}$ has to be a 2-category...
...which means the core of $\mathcal{E}/X$ is also a 2-groupoid...
...which means $\mathcal{E}(X, U)$ has to be a 2-groupoid too...

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & 3\text{-}\mathbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Um... but now $\mathcal{E}(X, U)$ has to be a groupoid rather than a set...

...which means $\mathcal{E}$ has to be a 2-category...

...which means the core of $\mathcal{E}/X$ is also a 2-groupoid...

...which means $\mathcal{E}(X, U)$ has to be a 2-groupoid too...

...which means $\mathcal{E}$ has to be a 3-category...

...la la la la la...

Object classifiers

Definition (almost)

An object classifier is a representing object U for

$$\begin{array}{rcl} \mathrm{Fam} : \mathcal{E}^{\mathrm{op}} & \to & \infty\text{-}\mathbf{Gpd} \\ X & \mapsto & \text{the maximal sub-groupoid of } \mathcal{E}/X \end{array}$$

Um. . . but now $\mathcal{E}(X, U)$ has to be a groupoid rather than a set. . .

. . . which means $\mathcal{E}$ has to be a 2-category. . .

. . . which means the core of $\mathcal{E}/X$ is also a 2-groupoid. . .

. . . which means $\mathcal{E}(X, U)$ has to be a 2-groupoid too. . .

. . . which means $\mathcal{E}$ has to be a 3-category. . .

. . . la la la la la. . .

. . . I suppose $\mathcal{E}$ has to be an ∞-category!

$(\infty, 1)$-categories

Actually we only need $(\infty, 1)$-categories, where the homs are groupoids rather than categories. As Emily described them yesterday, $(\infty, 1)$-categories are the world of abstract homotopy theory, with many equivalent analytic presentations: quasicategories, Rezk spaces, fibration categories (like tribes), model categories, as well as synthetic approaches.

Today I'm one of the people who wants to be "model-independent", but when pressed retreats to quasicategories. I'd love to be able to say all of this in an ∞-cosmos or in simplicial HoTT, but what I have to say hasn't been written in those languages yet.

$(\infty, 1)$-toposes

The $(\infty, 1)$-analogue of Grothendieck's sheaf toposes is by now well-known:

Definition (Lurie)

An $(\infty, 1)$-topos is an $(\infty, 1)$-category of $(\infty, 1)$-sheaves.

(For a very generalized notion of "$(\infty, 1)$-site" to take sheaves on.)

Theorem (Rezk, Lurie)

A locally presentable $(\infty, 1)$-category is an $(\infty, 1)$-topos iff

1. *It is locally cartesian closed, and*
2. *It has "sufficiently many" object classifiers.*

How many object classifiers?

We can't expect a classifier for all objects (Russellian paradoxes).

Definition

An object classifier is an object U together with a natural transformation

$$\mathcal{E}(X, U) \to \Big(\text{the } \infty\text{-groupoid of arrows } A \to X\Big)$$

which is an embedding, i.e. an equivalence onto a union of connected components (the families with "small fibers").

Theorem (Rezk, Lurie)

A locally presentable $(\infty, 1)$-category is an $(\infty, 1)$-topos iff

1. *It is locally cartesian closed, and*
2. *It has a classifier for the κ-compact objects, for all regular cardinals κ.*

Outline

What is an elementary $(\infty, 1)$-topos?

What is the "finitary" structure of Lurie's $(\infty, 1)$-toposes?

Definition

An elementary $(\infty, 1)$-topos is an $(\infty, 1)$-category with

1. Finite (homotopy) limits
2. Finite (homotopy) colimits
3. Locally cartesian closed exponentials
4. A subobject classifier
5. "Enough" object classifiers.
   - We can't say "κ-compact objects"; that's not finitary.
   - Instead: every $A \to X$ is classified by some object classifier U whose classified morphisms are closed under the other structure.

What is an elementary $(\infty, 1)$-topos?

What is the "finitary" structure of Lurie's $(\infty, 1)$-toposes?

Definition

An elementary $(\infty, 1)$-topos is an $(\infty, 1)$-category with

1. Finite (homotopy) limits
2. Finite (homotopy) colimits
3. Locally cartesian closed exponentials
4. A subobject classifier
5. "Enough" object classifiers.
   - We can't say "κ-compact objects"; that's not finitary.
   - Instead: every $A \to X$ is classified by some object classifier U whose classified morphisms are closed under the other structure.

Anyone could have written this down after reading Lurie. The real question is, is it enough to allow us to "code mathematics" into such a category?

A reason to worry

Lots of constructions that are finitary for 1-categories are no longer finitary for $(\infty, 1)$-categories, since there are now infinitely many dimensions of morphisms to deal with. This includes:

- Splitting idempotents
- Quotients of equivalence relations
- The image of a morphism

So can we do these things in an "elementary $(\infty, 1)$-topos"?

Actually, it's even worse

The very definition of $(\infty, 1)$-category involves infinitely many cells of arbitrary dimension, and at least *a priori* all the axioms are actually infinitely many! So can it really be called "elementary"?

Example

"Finite products" means equivalences of ∞-groupoids

$$\mathcal{E}(X, A \times B) \to \mathcal{E}(X, A) \times \mathcal{E}(X, B)$$

which says something about their cells at all dimensions.

Well, maybe it's not all that bad

Fortunately, there are solutions, such as:

- A synthetic theory of ∞-categories might help.
- Use a 1-categorical presentation (like a tribe, or a model category) that strictifies the universal properties, thus requiring only finitely many axioms.

This works well for limits, exponentials, etc. But the object classifiers are trickier, because their universal property involves a subgroupoid of $\mathcal{E}$ itself, rather than its hom-groupoids.

Could it be not bad at all?

Let $U_1 \to U$ be the universal object of an object classifier, and let $f, g : X \to U$ classify $f^* U_1, g^* U_1$ in $\mathcal{E}/X$. We want a way to assert that $\mathcal{E}(X, U)(f, g) \simeq \mathsf{Iso}(\mathcal{E}_{/X})(f^* U_1, g^* U_1)$.

- Homotopies between f, g are lifts of $(f, g) : X \to U \times U$ along $\Delta : U \to U \times U$.
- If we could construct an "equivalence classifier" $E \to U \times U$, such that lifts of (f, g) to it correspond to equivalences $f^* U_1 \simeq g^* U_1$, then we could assert $U \simeq E$ over $U \times U$.
- By local cartesian closure, we have a *morphism* classifier $F \to U \times U$, such that lifts of (f, g) to it correspond to *morphisms* $f^* U_1 \to g^* U_1$.
- We need to cut out a "subobject of equivalences" from F.

A classification problem

A question

Given a map $g : A \to B$ in $\mathcal{E}/X$, can we form the "subset"

$$\{\, i \in X \mid g_i \text{ is an equivalence} \,\} \quad ?$$

In categorical language, is the subfunctor

$$\{\, f : Y \to X \mid f^*(g) \text{ is an equivalence} \,\} \rightarrowtail \mathcal{E}(Y, X)$$

representable?

A classification answer

1-Question

Is $\left[Y \mapsto \{ f : Y \to X \mid f^*(g) \text{ is an isomorphism } \} \right]$ representable?

1-Answer

In a 1-topos, sure.

1. The local exponential $A_X^B \to X$ classifies morphisms $f^*B \to f^*A$.
2. Use equalizers to ensure that this generic morphism $f^*B \to f^*A$ composes to the identity with $f^*(g)$ on both sides, obtaining E_g.
3. Since inverses are unique when they exist, the induced map $E_g \to X$ is mono.

The harder case

∞-Question

In an $(\infty, 1)$-topos, is the functor

$$\left[Y \mapsto \{ f : Y \to X \mid f^*(g) \text{ is an equivalence} \} \right]$$

representable?

∞-Problem

In an $(\infty, 1)$-category, equipping a morphism with a "homotopy inverse" is not a mere property, so the induced map $E_g \to X$ will no longer be mono, and E_g won't represent the right functor.

We can make it mono by adding infinitely many higher coherences. . . but then the construction is no longer finitary.

Voevodsky's answer

Remember this?

```
Definition iscontr T := Σcntr:T, Πt:T, t=cntr.

Definition isweq {A B:UU} (g:A -> B) : UU :=
  Πy:B, iscontr (hfiber g y).
```

Translated from type theory to category theory, this yields a construction of an E_g that works in an $(\infty, 1)$-topos:

$$\begin{tikzcd}
\Delta_A \arrow[r, mapsto] & \bullet \arrow[r, mapsto] & \bullet \arrow[r, mapsto] & E_g \\
\rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} \\
\mathcal{E}/(A \times_B A) \arrow[r, "\Pi_{\pi_1}"] & \mathcal{E}/A \arrow[r, "\Sigma_g"] & \mathcal{E}/B \arrow[r, "\Pi_{B \to X}"] & \mathcal{E}/X
\end{tikzcd}$$

(The other modern definitions also work.)

$(\infty, 1)$-categorical univalence

Applying this to the morphism classifier $F \to U \times U$, we get our equivalence classifier $E \to U \times U$. Thus we can express the object classifier "elementarily" as well.

This is essentially what is done by the univalence axiom in type theory...which was Vladimir's goal.

Outline

To NNO or not to NNO

A natural numbers object (NNO) is a topos $\mathcal{E}$ is an initial object of the category of objects N equipped with morphisms $0 : 1 \to N$ and $s : N \to N$.

- NB: this is a weaker "finitary" property than being the countable coproduct $\coprod_{\mathbb{N}} 1$. There are 1-toposes with an NNO but without countable coproducts.
- The basic definition of elementary 1-topos does not include an NNO. But reasonably often one needs to assume it, to encode "infinitary" mathematics.
- It's even harder to get off the ground in the $(\infty, 1)$-case without some kind of internal infinity. Fortunately...

To NNO

"Theorem"

Any elementary $(\infty, 1)$-topos has an NNO.

Proof.

- By assumption, we have an object classifier U that contains 1 and is closed under finite coproducts.

To NNO

"Theorem"

Any elementary $(\infty, 1)$*-topos has an NNO.*

Proof.

- By assumption, we have an object classifier U that contains 1 and is closed under finite coproducts.
- Using Ω, we can define the smallest subobject of U that contains 1 and is closed under finite coproducts. This is an internal version of $\coprod_{n \in \mathbb{N}} B\mathrm{Aut}([n])$.

To NNO

“Theorem”

Any elementary $(\infty, 1)$-topos has an NNO.

Proof.

- By assumption, we have an object classifier U that contains 1 and is closed under finite coproducts.
- Using Ω, we can define the smallest subobject of U that contains 1 and is closed under finite coproducts. This is an internal version of $\coprod_{n\in\mathbb{N}} B\mathrm{Aut}([n])$.
- Instead, let T be the classifier of totally ordered objects, and let N be the smallest subobject of T containing 1 and closed under finite coproducts. Since totally ordered finite sets are rigid, this N is a 0-type (a “set” internally).

To NNO

“Theorem”

Any elementary $(\infty, 1)$-topos has an NNO.

Proof.

- By assumption, we have an object classifier U that contains 1 and is closed under finite coproducts.
- Using Ω, we can define the smallest subobject of U that contains 1 and is closed under finite coproducts. This is an internal version of $\coprod_{n\in\mathbb{N}} B\mathrm{Aut}([n])$.
- Instead, let T be the classifier of totally ordered objects, and let N be the smallest subobject of T containing 1 and closed under finite coproducts. Since totally ordered finite sets are rigid, this N is a 0-type (a “set” internally).
- Prove the Peano axioms for N, which implies it is an NNO. Induction uses its definition as “the smallest subobject...”.

□

Internalizing infinity

Using the NNO, we can also “show”:

- (S., inspired by Lurie) While an “incoherent idempotent” need not split, if the “witness of idempotence” has one additional coherence datum then we can split it.
- (Kraus, van Doorn, Rijke) We have images of morphisms, and quotients of proposition-valued equivalence relations.
- (Rijke) We can also construct n-truncations (universal maps into “n-groupoidal objects”) for all n, and define a notion of “∞-equivalence relation” and construct their quotients.

Also, Rasekh has shown that local cartesian closure follows from the existence of the internal full subcategory on the object classifier.

So, indications are promising that this definition of elementary $(\infty, 1)$-toposes is enough to encode plenty of mathematics.

The internal language problem

But…actually, right now these things have only been proven in the type theory that is conjecturally the internal language of elementary $(\infty, 1)$-toposes.

Objects $A \in \mathcal{E}/X$	$\leftrightarrow$	Dependent types $x : X \vdash A(x) : \mathsf{Type}$
Right adjoint $\Pi_{f:B\to A}$	$\leftrightarrow$	Dependent function type $\prod_{x:A} B(x)$
Finite colimits	$\leftrightarrow$	Non-recursive HITs
Object classifiers	$\leftrightarrow$	Univalent universes

For short proofs, we can "manually" translate a type-theoretic argument to a category-theoretic one, or regard the type-theoretic syntax as "just a notation" for the category theory. But for longer arguments like these, this becomes increasingly infeasible and indefensible; we need a general theorem.

Categorical semantics of type theories

The initiality principle

For any kind of formal language or type theory, one can build a corresponding structured category whose objects are its derivable types and whose morphisms are its derivable terms, and this is the initial such structured category.

Corollary (Soundness and completeness)

1. *Anything derivable in the type theory is true in any category (just apply the unique functor out of the initial object).*
2. *Anything true in all categories is derivable in the type theory (the syntactic category is one such category).*

Type theories and their semantics

propositional logic	$\leftrightarrow$	Boolean/Heyting algebras
first-order logic	$\leftrightarrow$	Boolean/Heyting categories
simply typed λ-calculus	$\leftrightarrow$	cartesian closed categories
higher-order type theory	$\leftrightarrow$	elementary 1-toposes

In all cases the initiality theorem is proven by a straightforward induction over the construction of syntax. This is completely standard in categorical logic.

So what's the big deal?

1. For large and complicated type theories like HoTT, there are a lot of cases in that induction, and no one has actually written them all down.
2. There's a qualitative difference in difficulty for dependent type theories, because you can't induct "step-by-step" over types first, then terms, then equalities. Only one person (Streicher) has ever written down anything approaching a complete proof of initiality for a dependent type theory.
3. For homotopy type theories and $(\infty, 1)$-categories, there is an extra step of strictification, which we only know how to do fully in a few particular cases.

Opinions differ about the importance of (1) and (2), but the obstacle posed by (3) is indisputed.

Open Problems

1. Prove the internal language correspondence for elementary $(\infty, 1)$-toposes and a flavor of homotopy type theory.
2. In particular, this includes constructing an elementary $(\infty, 1)$-topos out of the syntax of type theory.
3. Other than this example and Lurie's Grothendieck ones, what elementary $(\infty, 1)$-toposes are there?
4. What do *recursive* Higher Inductive Types correspond to semantically in elementary $(\infty, 1)$-toposes?
5. What is a *logical functor* between $(\infty, 1)$-toposes? (A logical 1-functor preserves *the* subobject classifier.)