From the nLab to the HoTT Book

Michael Shulman

University of San Diego

January 7, 2016

(Some of) my experiences with digital mathematics

1. The *n*Lab — a wiki for higher-categorical mathematics, now with 11k pages by 1.5k authors
2. Homotopy Type Theory / Univalent Foundations — a new foundational system for mathematics, closely tied to computer formalization
3. The Homotopy Type Theory book — a "research textbook" about homotopical type theory and univalent foundations, 600 pages written in 6 months by 30 authors

Some of my conclusions

1. Record partial work and then improve it.
2. Invite strangers to solve your problems.
3. Choose appropriate project structure.
4. Embrace synthetic mathematics.

Too long; didn't listen

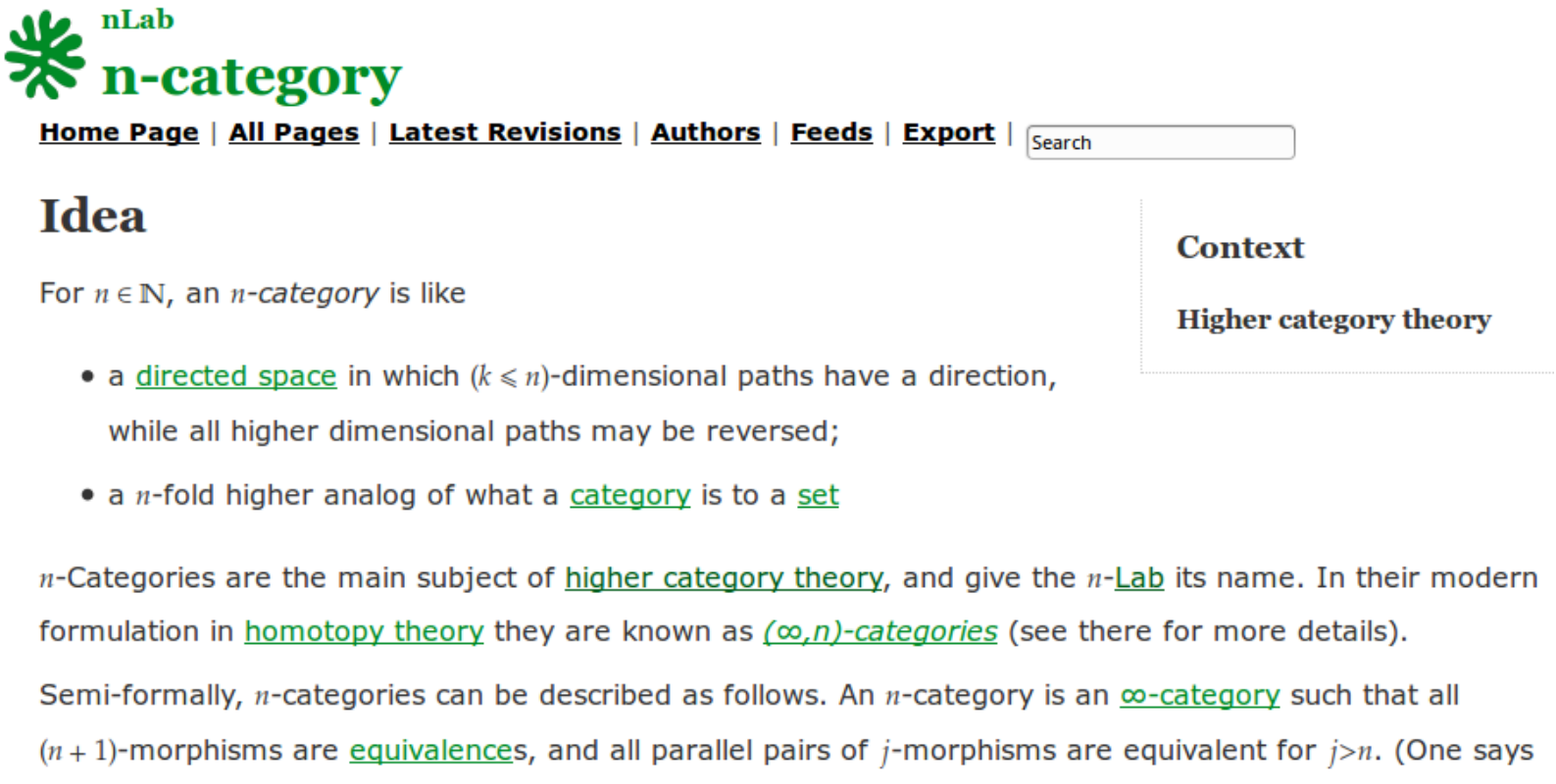
One size doesn't fit all.

Outline

1. The *n*Lab
2. Homotopy type theory
3. The HoTT Book
4. Conclusions

The *n*Lab

- A wiki for higher category theory, created November 2008
- 11,000 pages by 1,500 authors
- Used as a reference by many mathematicians worldwide



nLab

n-category

Home Page | All Pages | Latest Revisions | Authors | Feeds | Export | Search

Idea

For $n \in \mathbb{N}$, an *n-category* is like

- a directed space in which $(k \leqslant n)$-dimensional paths have a direction, while all higher dimensional paths may be reversed;
- a n-fold higher analog of what a category is to a set

Context

Higher category theory

n-Categories are the main subject of higher category theory, and give the n-Lab its name. In their modern formulation in homotopy theory they are known as *(∞,n)-categories* (see there for more details).

Semi-formally, n-categories can be described as follows. An n-category is an ∞-category such that all $(n+1)$-morphisms are equivalences, and all parallel pairs of j-morphisms are equivalent for $j>n$. (One says

Wikis

- Any reader can edit any page through the browser
- All changes are recorded and can be reverted
- Simple syntax for formatting and inter-page links
- Wikipedia is probably the best-known wiki site.

The *n*Lab is not Wikipedia

Wikipedia says:

> *All encyclopedic content on Wikipedia must be written from a neutral point of view (NPOV), which means representing fairly, proportionately, and, as far as possible, without editorial bias, all of the significant views that have been published by reliable sources on a topic.*
>
> *NPOV is . . . one of Wikipedia's three core content policies; the other two are "Verifiability" and "No original research".*

The *n*Lab is not Wikipedia

The *n*Lab says:

> *. . . the nLab has a particular point of view, which we may call the nPOV or the n-categorical point of view. . . that category theory and higher category theory provide a point of view on Mathematics, Physics and Philosophy which is a valuable unifying point of view for the understanding of the concepts involved.*

The *n*Lab is not even an encyclopedia

The most apt analogy for the nLab is of a group lab book. A lab book for a research scientist is a place where they write down anything that they consider relevant for their work. . .

The key difference between the nLab and an "ordinary" lab book is that it is public. By making it public we hope to achieve two things:

1. *To enable others to benefit from our work while it is still being done.*
2. *To benefit from the work of others while we are doing it.*

Outline

1. The *n*Lab
2. Homotopy type theory
3. The HoTT Book
4. Conclusions

Homotopy Type Theory / Univalent Foundations

- In pre-HoTT type theories, "types" are like sets.
- In HoTT/UF, "types" are like homotopy types.
- The "type of types" is a "classifying space" for types.
- In particular, isomorphic objects can be identified (Voevodsky's univalence axiom).

Homotopy Type Theory / Univalent Foundations

- In pre-HoTT type theories, "types" are like sets.
- In HoTT/UF, "types" are like homotopy types.
- The "type of types" is a "classifying space" for types.
- In particular, isomorphic objects can be identified (Voevodsky's univalence axiom).

Consequences:

- For most of mathematics: none at all.
- Enables new mathematics: synthetic homotopy theory.
- Potentially some simplifications for formalized mathematics.

Synthetic homotopy theory

Consider a theorem like $\pi_1(S^1) = \mathbb{Z}$.

- Classically, S^1 is the set $\{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$ with a topology, eventually reducing to ZFC.
- In HoTT/UF, S^1 is a basic object of mathematics like the "sets" of ZFC.

Synthetic homotopy theory

Consider a theorem like $\pi_1(S^1) = \mathbb{Z}$.

- Classically, S^1 is the set $\{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$ with a topology, eventually reducing to ZFC.
- In HoTT/UF, S^1 is a basic object of mathematics like the "sets" of ZFC.

This makes the theorem:

- Potentially easier to prove.
- Closer to the metal.
- A different result.

HoTT/UF and formalization

Why is HoTT/UF is closely associated with computer-formalized mathematics?

- It's a type theory.
- It makes more things constructive and easier to formalize.
- Some of the same people are involved in both.
- Improves existing type theories . . . a bit.

HoTT/UF and formalization

Why is HoTT/UF is closely associated with computer-formalized mathematics?

- It's a type theory.
- It makes more things constructive and easier to formalize.
- Some of the same people are involved in both.
- Improves existing type theories . . . a bit.

In addition:

- Easier to do with help from a computer!
  - A "strange new world"
  - Handles bookkeeping
- A surprising amount of HoTT is done directly in a computer rather than first on paper and then being formalized.

A (perhaps) heretical opinion

It will be a long time before all mathematics is formalized.

- In general, formalizing a proof is a lot more work. (HoTT doesn't affect that much.)
- Even learning to use a proof assistant is a lot of work.
- Proof assistants have a long way to go in usability so that they "feel like mathematics" to mathematicians.
- We do mathematics for understanding, not just truth.
- For most mathematicians, the benefit isn't there yet.

A (perhaps) heretical opinion

It will be a long time before all mathematics is formalized.

- In general, formalizing a proof is a lot more work. (HoTT doesn't affect that much.)
- Even learning to use a proof assistant is a lot of work.
- Proof assistants have a long way to go in usability so that they "feel like mathematics" to mathematicians.
- We do mathematics for understanding, not just truth.
- For most mathematicians, the benefit isn't there yet.

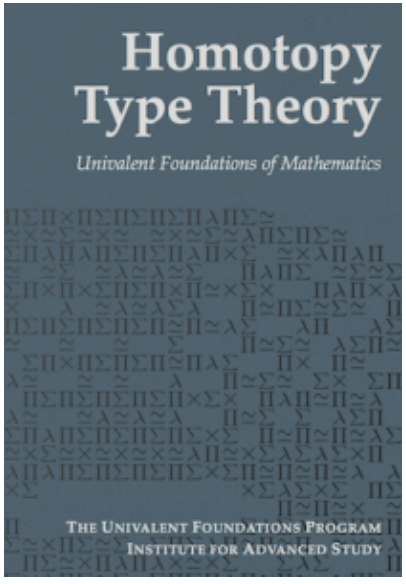
Conclusion: A new foundational system like HoTT/UF should have an "informal" version too.

Outline

1. The *n*Lab
2. Homotopy type theory
3. The HoTT Book
4. Conclusions

The HoTT Book

- 2012–2013: Special year at IAS on Homotopy Type Theory / Univalent Foundations (∼30 participants)
- We did not set out to write a book.
  1. Experiment with "informal type theory"
  2. Record results of special year
  3. Introduce newcomers to the field



- Many results had to be "unformalized" from proof assistants
- Written collaboratively with distributed version control
- Downloaded over 40,000 times, 1600 print-on-demand copies sold at cost

Distributed version control with Git/GitHub

- Designed for open-source software projects.
- Each "repository" stores the full history of changes:
  - Central repository on GitHub servers
  - Forked repositories on GitHub controlled by individuals
  - Personal repositories on individual workstations
- Workflow:
  - Pull other people's changes from Central to Personal
  - Edit and commit to Personal repository
  - Push from your Personal to your Forked repository
  - Make a pull request from your Forked repository
  - Someone approves the pull request, pulling the changes into the Central repository and resolving any conflicts
- Can branch to experiment with an idea before merging it in
- Also open issues (e.g. bug reports), like a threaded message board, and comment on lines of code.

Outline

1. The *n*Lab
2. Homotopy type theory
3. The HoTT Book
4. Conclusions

Record partial work and improve it

a.k.a. One size doesn't fit all for software

- Blogs, email, StackOverflow
  - Great for fast interaction, questions, conversation
  - Not well-organized or archived, hard to find later
- Published papers
  - Polished, organized carefully and logically
  - Take more time, perceived-importance filter, many things are never published
- Wikis and VC repositories
  - Easily and quickly record results of a conversation
  - Flexible and dynamic organization
  - Start with a stub and improve incrementally
  - Record things that could never be "published"

Invite strangers to solve your problems

a.k.a. One size doesn't fit all for contributors

Given enough eyeballs, all bugs are shallow.
– "Linus's Law", formulated by Eric S. Raymond

Treating your users as co-developers is your least-hassle route to rapid code improvement and effective debugging.
– Eric S. Raymond

Invite strangers to solve your problems

a.k.a. One size doesn't fit all for contributors

Given enough eyeballs, all bugs are shallow.
– "Linus's Law", formulated by Eric S. Raymond

Treating your readers as coauthors is your least-hassle route to rapid text improvement and effective updating.
– paraphrased from Eric S. Raymond

Invite strangers to solve your problems

a.k.a. One size doesn't fit all for contributors

> *Given enough eyeballs, all bugs are shallow.*
> – *"Linus's Law", formulated by Eric S. Raymond*

> *Treating your readers as coauthors is your least-hassle route to rapid text improvement and effective updating.*
> – *paraphrased from Eric S. Raymond*

So far the HoTT Book has ∼150 errata.

- A lot of mistakes?

Invite strangers to solve your problems

a.k.a. One size doesn't fit all for contributors

> *Given enough eyeballs, all bugs are shallow.*
> – *"Linus's Law", formulated by Eric S. Raymond*

> *Treating your readers as coauthors is your least-hassle route to rapid text improvement and effective updating.*
> – *paraphrased from Eric S. Raymond*

So far the HoTT Book has ∼150 errata.

- A lot of found mistakes!
- All fixed in the currently available version.
- Most found and fixed by readers via GitHub pull requests.
- A wiki or DVCS leverages the "drive-by contributor".

Choose appropriate structure

a.k.a. One size doesn't fit all for project structure

- If you want to achieve a coherent result, the core creators need a coherent goal and the means and willingness to guide the project in that direction.

Choose appropriate structure

a.k.a. One size doesn't fit all for project structure

- If you want to achieve a coherent result, the core creators need a coherent goal and the means and willingness to guide the project in that direction.
- The *n*Lab is not very coherent!
  - Large gaps in coverage: not encyclopedic
  - Varying style, quality, and content between pages

Choose appropriate structure

a.k.a. One size doesn't fit all for project structure

- If you want to achieve a coherent result, the core creators need a coherent goal and the means and willingness to guide the project in that direction.
- The *n*Lab is not very coherent!
  - Large gaps in coverage: not encyclopedic
  - Varying style, quality, and content between pages
  - . . . but that's not the point.

Choose appropriate structure

a.k.a. One size doesn't fit all for project structure

- If you want to achieve a coherent result, the core creators need a coherent goal and the means and willingness to guide the project in that direction.
- The *n*Lab is not very coherent!
  - Large gaps in coverage: not encyclopedic
  - Varying style, quality, and content between pages
  - . . . but that's not the point.
- Most people say the HoTT Book is surprisingly coherent.
  - A "technical dictator" enforced usage of semantic macros.
  - Not afraid to edit each other's writing for consistent style.
  - Pull requests allow more "gatekeeping" than a wiki.

Embrace synthetic mathematics

a.k.a. One size doesn't fit all for mathematical foundations

- ZFC is synthetic set theory: basic objects are sets.
- Ordinary type theory is hardly different in this.
- HoTT/UF is synthetic homotopy theory.

Embrace synthetic mathematics

a.k.a. One size doesn't fit all for mathematical foundations

- ZFC is synthetic set theory: basic objects are sets.
- Ordinary type theory is hardly different in this.
- HoTT/UF is synthetic homotopy theory.

But there can be other kinds of foundations too:

- Synthetic topology: basic objects are topological spaces.
- Synthetic computability theory: basic objects are computational datatypes.
- Cohesive HoTT: basic objects are topological *and* homotopical.