Build a reliable math library

An introduction to Lean 4 and Mathlib - 13.09.2025

Cheni Yuki Yang
Universität Heidelberg

What is Lean?

Dependent type and logic

It is long known by computer scientists and mathematicians, that **dependent type theory** is a well-behaved logic base for constructive mathematics.

Lean 4 is a programming language aiming to achieve this goal.

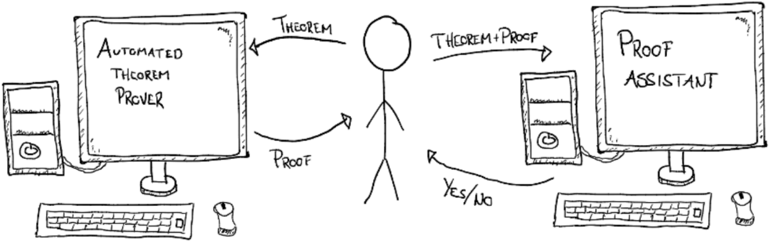
More precisely, Lean is an interactive theorem prover.

Lean 4 abandons the support for **Homotopy Type Theory** recently, making it impossible to extend a cubical type theory like other theorem provers(agda, aya). This is due to the stability demand of Mathlib development, which in turns to have double sides on Lean 4 development.

Lean is a choice by time, many active researchers in HoTT, including Floris van Doorn, are now more devoted to Mathlib, making Voevodsky's dream less reachable.

Interactive Theorem prover

What is a theorem prover?



A theorem prover is not an automatic intelligence! There's some automation, but most codes are written by hand(and in a foreseeable future, not by LLM).

Machines give a goal, and people write tactic or construction to solve it.

It should be powerful enough to encode all mathematical concepts, in some sense, matches what Leibniz called *Characteristica universalis*.

How to write Lean?

A huge advantage of Lean is it's just like the daily used mathematical language. You give a statement, and Lean returns a goal.

Once you hit some tactics, Lean's type checker will work and return you a new goal, or a human readable error.

Currently this type checker is still not fast enough.

It's interactive and is suitable for writing a textbook or give exercise sheets. Thus we need a functional math library in college level!

There are many tutorials and textbooks:

- Natural number games
- Mathematics in Lean
- Imperial College course on formalizing mathematics

What is Mathlib?

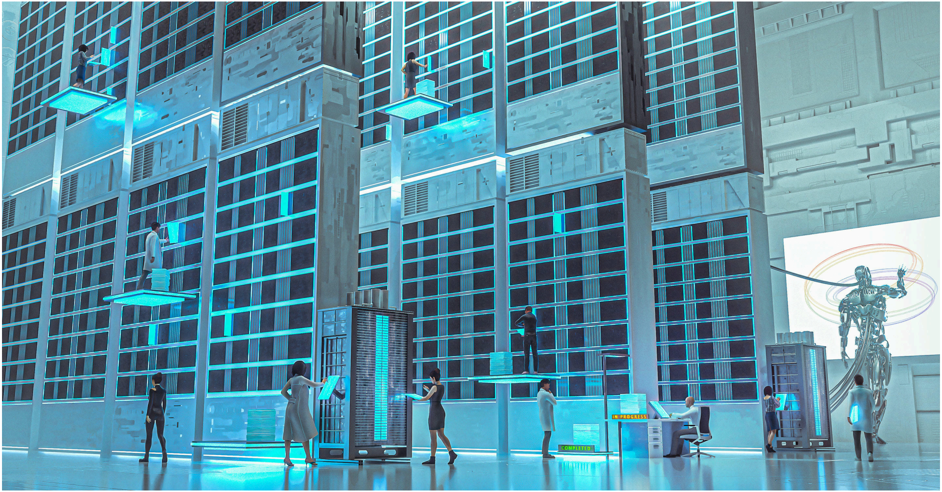
Traditional math library

A traditional mathematical library consists of journal papers, textbooks, manuscript and preprints.

- Mathematics is old, thus they are vast and heterogenous.
- It takes time to explore, find and learn them.
- The math library is still growing very fast and hard to keep in state of art.
- Digitalization helps a lot, but still not enough.

Future math library

How does a future math library look like?



Source: Quanta Magazine, illustrated by BakaArts

Future math library (ii)

Borges once described a library without boundaries, divided with hexagon units and can be accessed from many other units. Think each theorem as a unit, Mathlib wants to build the Babel library of mathematical knowledge.

- Internally highly linked, indexed
- easy accessible from any start point
- shared by humankind and machines
- proven correct
- a large scale collaboration(>20 typically)

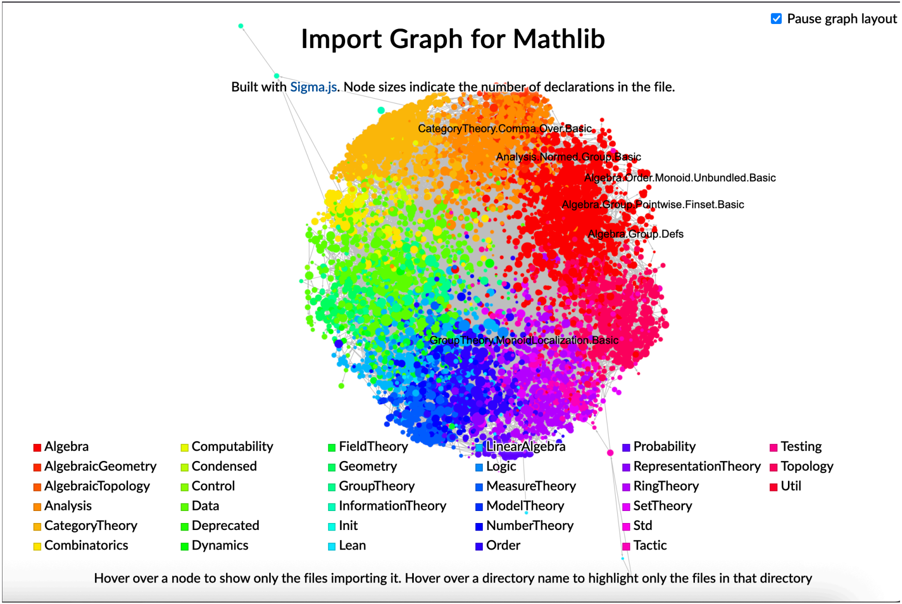
Mathlib was developed in 2017 in Lean 3 and ported to Lean 4 in 2023. So far, there are still many migrating and compatibility issues however.

Theorems in Mathlib

Currently, Mathlib consists of 112301 definitions and 230091 theorems. Most of them are in algebra, category theory and topology. Analysis and geometry parts are still in early stage.

Below is a import graph of Mathlib

Theorems in Mathlib (ii)



How do they work together?

Toolchains

Loogle

Loogle searches Lean and Mathlib definitions and theorems. It is developed by Lean FRO, a non-profit organization founded by Leo de Moura and Sebastian Ulrich, both active maintainers of Lean compiler.

LeanSearch

LeanSearch finds theorems in Mathlib using natural language query, originally written by AI4M Team in BICMR.

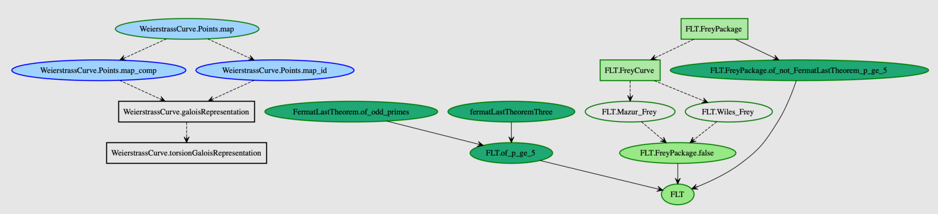
Aesop

Aesop (Automated Extensible Search for Obvious Proofs) is a proof search tactic for Lean 4. It is broadly similar to Isabelle's auto. A PhD project by Jannis Limperg in LMU Müchen.

Toolchains (ii)

Lean Blueprint

Lean Blueprint is a plasTeX plugin allowing to write blueprints for Lean 4 blueprints.



Community

There is a Zulip channel for discussing Mathlib and Lean 4 development. Though Mathlib is community-driven, many features and bug fixes of Lean 4 itself are driven by the needs of Mathlib development.

Most of topics in Zulip channel are public and there is a very friendly atmosphere welcoming all mathematicians, computer scientists and more theorem proving enthusiastic.

Many courses are provided on formal mathematics in Europe, America and China, including summer school in PKU and undergraduate courses in Imperial College.

Active research seminars around university, plus yearly Lean Together meeting lasting two days contributes to the energy of community.

Why you should contribute

Mathlib has 26 maintainers and 653 contributors. There're still more than 2000 pull request unmerged!

It will surprise many, but a lot of fundamental results in mathematics are not formalized, for example, the Cauchy's theorem for complex integral!

You do not need to learn a lot of math to start to contribute, thanks to team collaboration and toolchains, many tasks are now a bunch of subgoals.

Lean and Mathlib lacks nowadays contributors with a rich experience in computer and software engineering.

This year, three student projects in a undergraduate course in Universität Heidelberg are already merged to Mathlib, including Kuratowski's theorem and a theorem over PIDs.

How does Lean 4 prove itself useful?

Liquid tensor project

In 2020, Fields medalist Peter Scholze challenged the Leanprover community to check the correctness of one of his most recent theorems, in the newly-created field of **Condensed Mathematics**. He did so via a blog post. Only 18 months later, the challenge was completed. The findings of the project became the topic of an article in Nature.

Theorem 1.

(Clausen-Scholze) Let $0 < p < p' \leq 1$ be real numbers, let S be a profinite set, and let V be a p-Banach space. Let $\mathcal{M}_{p'}(S)$ be the space of p'-measures on S. Then

$$\mathrm{Ext}^i_{\mathrm{Cond(Ab)}}\left(\mathcal{M}_{p'}(S), V\right) = 0$$

for $i \geq 1$.

It proves Lean is a strong help of active researches.

Kevin Buzzard's dream

- A five year EPSRC-grant was issued in 2024 for proving this diamond in number theory.
- It is a large-scale collaborative project across Europe.
- The goal is to formalize the famous annual paper by Andrew Wiles.

Theorem 2: Taniyama-Shimura Conjecture.

(Wiles) Every semistable elliptic curve is modular, i.e. the L-function of its associated Galois representation comes from a cusp form of weight 2.

- Kevin told me he anticipates the formalization of Fermat's last theorem will last for more 30 years, and it is the last happy thing he shall see.

Write lean is actually a job!

Many PhD jobs are available in IC, Bonn, INRIA, Utrecht and Heidelberg.

Recently, Lean FRO receives a funding with 100 million dollars for research on automated reasoning.

Thanks for Listening!