Imperial College London

Formalizing Mathematics in Lean

Conferencias de Posgrado UCM

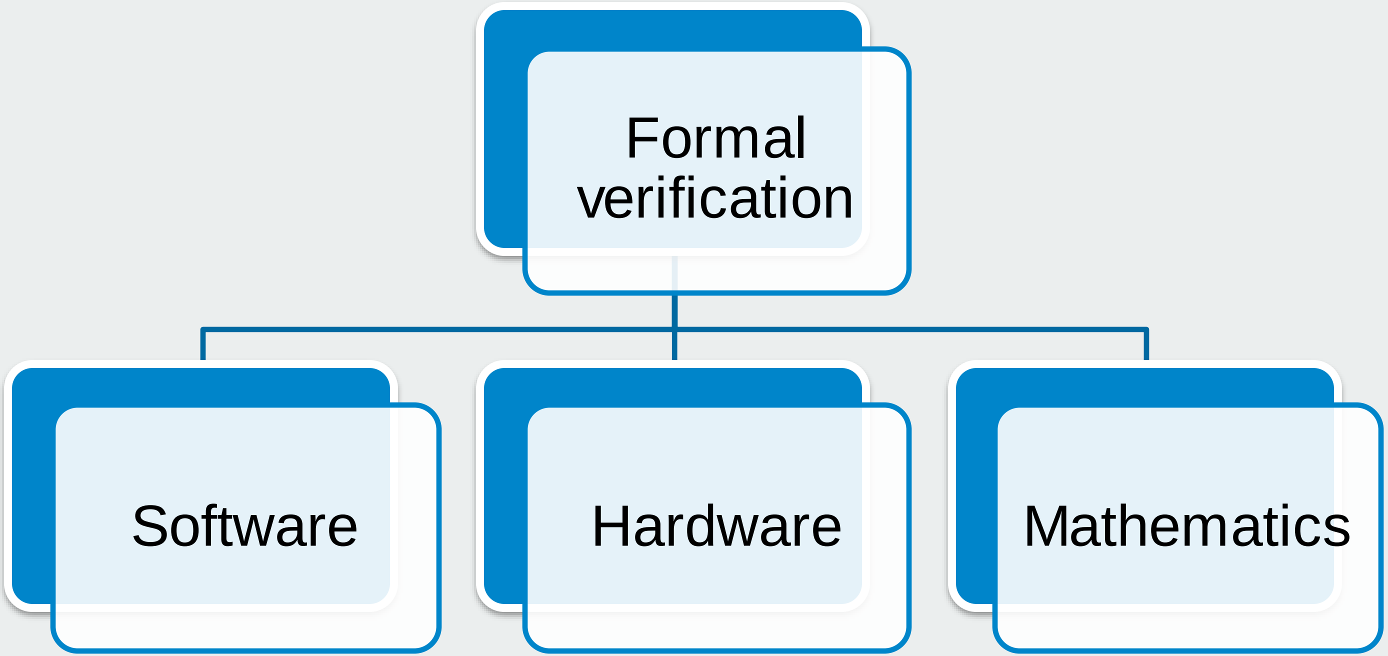
María Inés de Frutos Fernández

What?
- Formal Mathematics

Why?
- Research
- Teaching

How?
- Lean
- Mathlib

Formal Mathematics


Formal verification
Software
Hardware
Mathematics

Computers in Mathematics

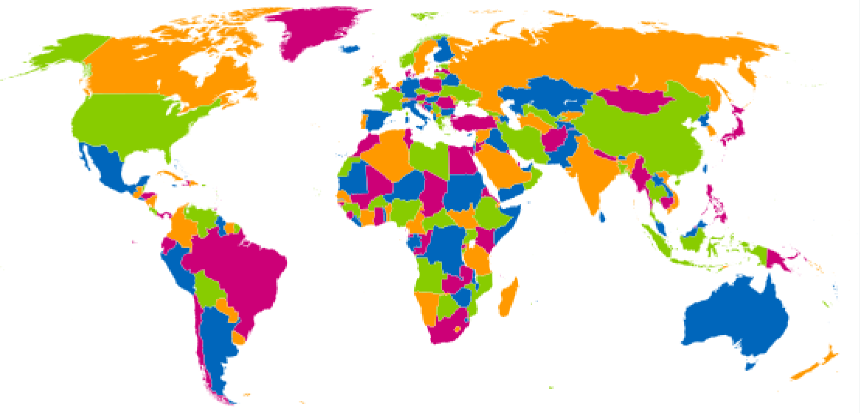
Computation

- Used for decades
- Not what this talk is about

Reasoning

- Check proofs
- Finding proofs/counterexamples?

History

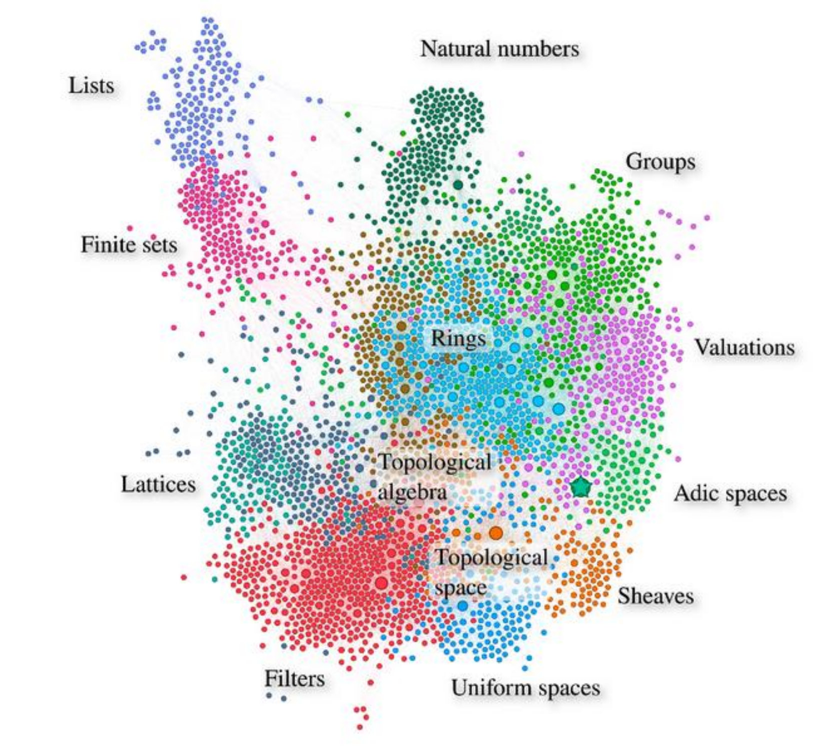


G. Gonthier (2005), A computer-checked proof of the four colour theorem.

T. Hales et. al. (2017), A formal proof of the Kepler conjecture.

Formalizing perfectoid spaces

- K. Buzzard, J. Commelin, and P. Massot (2020) "Formalising perfectoid spaces". https://doi.org/10.1145/3372885.3373830
- More than 3000 definitions or statements used in this definition



The Liquid Tensor Experiment

2019
Condensed Mathematics (Clausen, Scholze)

Dec. 2020
Liquid Tensor Experiment challenge (Scholze)

June 2021
Key technical lemma formalized (Commelin et. al.)

July 2022
Liquid Tensor Experiment completed (Commelin, Topaz et. al.)

Liquid tensor experiment

Posted on December 5, 2020 by xenaproject

This is a guest post, written by Peter Scholze, explaining a liquid real vector space mathematical formalisation challenge. For a pdf version of the challenge, see here. For comments about formalisation, see section 6. Now over to Peter.

Quanta magazine

Physics | Mathematics | Biology | Computer Science | Topics | Archive

PROOFS

Proof Assistant Makes Jump to Big-League Math

7

Mathematicians using the computer program Lean have verified the accuracy of a difficult theorem at the cutting edge of research mathematics.

nature

Explore content | About the journal | Publish with us | Subscribe

nature > news > article

NEWS | 18 June 2021

Mathematicians welcome computer-assisted proof in 'grand unification' theory

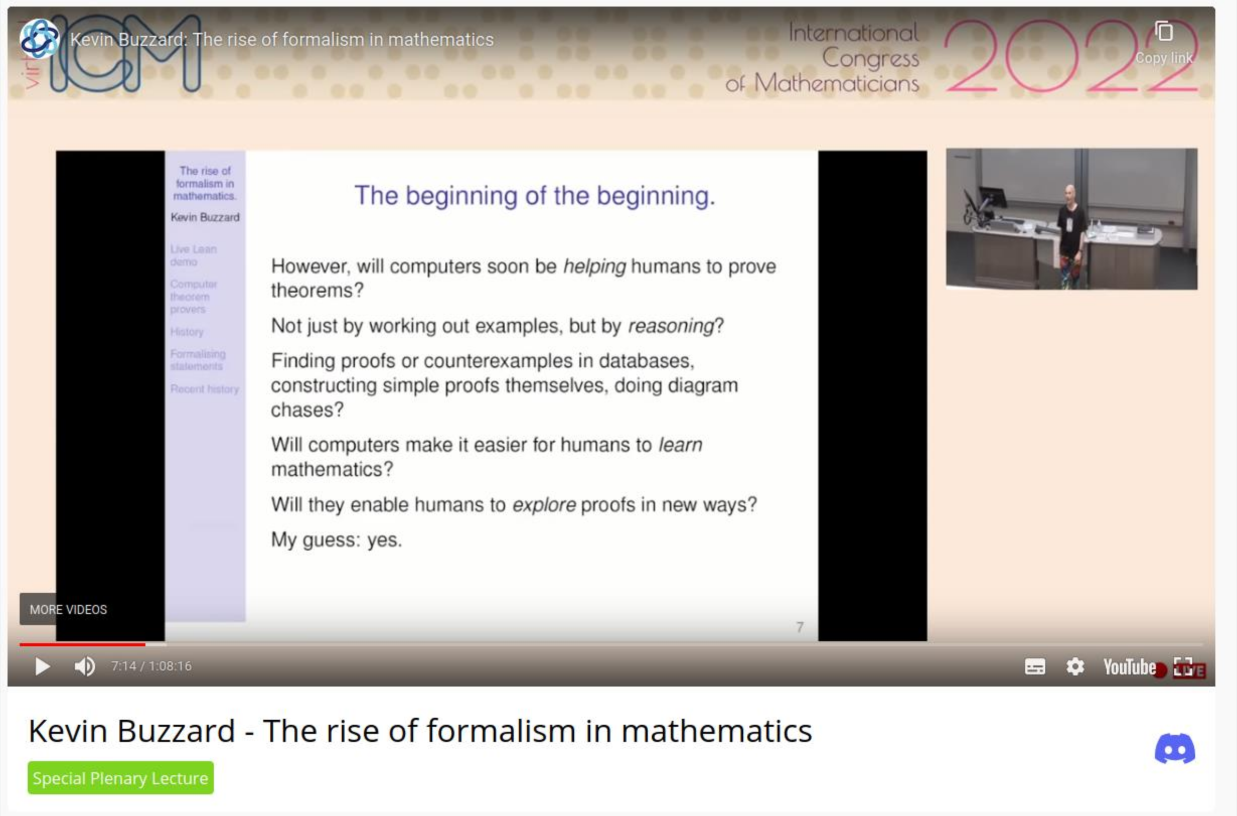
Proof-assistant software handles an abstract concept at the cutting edge of research, revealing a bigger role for software in mathematics.

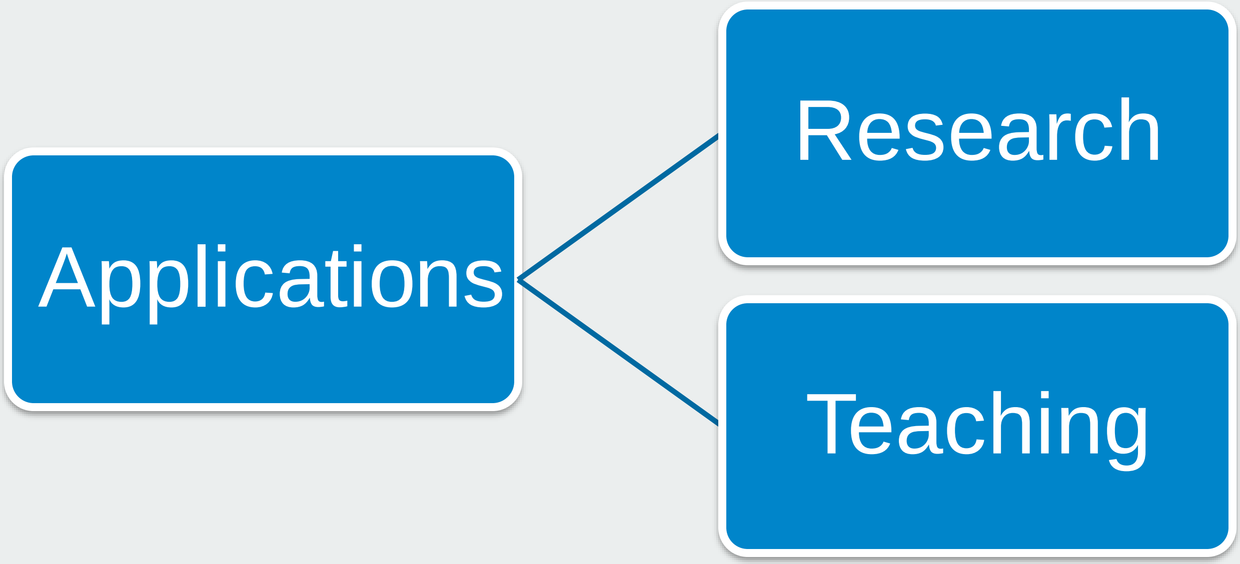
Davide Castelvecchi

Completion of the Liquid Tensor Experiment

Mathlib community — 2022-07-15 15:00 — Source

We are proud to announce that as of 15:46:13 (EST) on Thursday, July 14 2022 the Liquid Tensor Experiment has been completed. A year and a half after the challenge was posed by Peter Scholze we have finally formally verified the main theorem of liquid vector spaces using the Lean proof assistant. The blueprint for the project can be found here and the formalization itself is available on GitHub.


Kevin Buzzard: The rise of formalism in mathematics
International Congress of Mathematicians 2022
Copy link
The rise of formalism in mathematics.
Kevin Buzzard
Live Lean demo
Computer theorem provers
History
Formalising statements
Recent history
The beginning of the beginning.
However, will computers soon be *helping* humans to prove theorems?
Not just by working out examples, but by *reasoning*?
Finding proofs or counterexamples in databases, constructing simple proofs themselves, doing diagram chases?
Will computers make it easier for humans to *learn* mathematics?
Will they enable humans to *explore* proofs in new ways?
My guess: yes.
7
MORE VIDEOS
7:14 / 1:08:16
YouTube
Kevin Buzzard - The rise of formalism in mathematics
Special Plenary Lecture


Applications
Research
Teaching

Checking proofs

Big proofs

Technical parts

Consistency at all scales

Creating mathematics

Routine steps	Assumptions/ dependencies
Better understanding	AI

Solving (Some) Formal Math Olympiad Problems

We built a neural theorem prover for Lean that learned to solve a variety of challenging high-school olympiad problems, including problems from the AMC12 and AIME competitions, as well as two problems adapted from the IMO.[1] The prover uses a language model to find proofs of formal statements. Each time we find a new proof, we use it as new training data, which improves the neural network and enables it to iteratively find solutions to harder and harder statements.

These problems are not standard math exercises, they are used to let the best high-school students from the US (AMC12, AIME) or the world (IMO) compete against each other.

OpenAI

Semantic Search Engines for Mathematics

FAB STRACTS

View the Project on GitHub
formalabstracts/formalabstracts

Formal Abstracts

About the project

The *Formal Abstracts* project was initiated by Thomas Hales in 2017. See his talk Big conjectures from the Big Proof meeting in Cambridge.

A **formal abstract**, or **fabstract** for short, is a formalization of the main results (constructions, definitions, proofs, conjectures) of a piece of informal mathematics, such as a research paper. There is no requirement that the entire text be formalized. Proofs of statements are omitted. A formal abstract is *not* the formalization of the abstract itself.

A vision

The Formal Abstracts (FAbstracts) project will establish a formal abstract service that will express the results of mathematical publications in a computer-readable form that captures the semantic content of publications.

Teaching

Pros	Cons
• Proof understanding • Precision • Immediate feedback	• Barrier to entry • Lack of graphical inferface/documentation • Tradition

A new kind of mathematical document

1.2 Preliminaries

In this section, E is a real vector space with (finite) dimension d. We'll need the Carathéodory lemma:

Lemma 1.4 (Carathéodory's lemma)✓# L∃∀N

> If a point x of E lies in the convex hull of a set P, then x belongs to the convex hull of a finite set of affinely independent points of P.

Proof ▸

Lean declarations ✖

convex_hull_eq_union

By assumption points t_i in P and weights f_i such that $x = \sum f_i t_i$, each f_i is non- Choose such a set of points of minimum cardinality. We argue by c set must be affinely independent.

Thus suppose that there is some vanishing combination $\sum g_i t_i$ with $\sum g_i = 0$ and not all g_i vanish. Let $S = \{i | g_i > 0\}$. Let i_0 in S be an index minimizing f_i / g_i. We shall obtain our contradiction by showing that x belongs to the convex hull of the set $\{t_i | i \neq i_0\}$, which has cardinality strictly smaller than $\{t_i\}$.

We thus define new weights $k_i = f_i - g_i f_{i_0} / g_{i_0}$. These weights sum to $\sum f_i - (\sum g_i) f_{i_0} / g_{i_0} = 1$ and $k_{i_0} = 0$. Each k_i is non-negative, thanks to the choice of i_0 if i

Imperial College London

Lean and mathlib

Theorem provers

- Coq
- Isabelle/HOL
- HOL Light
- Agda
- Metamath
- Mizar
- Lean
- ...

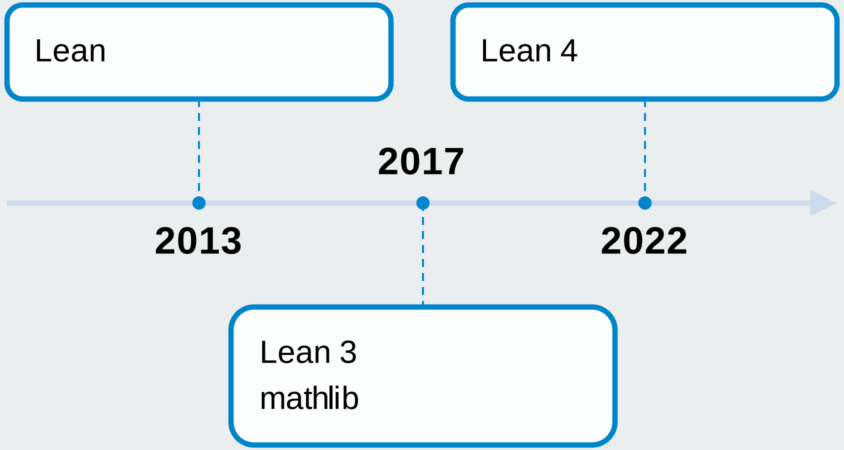
Lean

Interactive Theorem Prover

- Dependent type theory
- Proof irrelevance

Microsoft Research

- Leonardo de Moura



Dependent type theory

Dependent type theory

- Prop, Type, Type 1, ... , Type*
- t : T

Notation

- λ x, f x
- π, ∞ , ⨂, ∀, ∃, . . .

```
def square : ℝ → ℝ := λ x, x^2
```

```
lemma mp (p q : Prop) :
  p → (p → q) → q :=
λ hp hpq, hpq hp
```

Type class inference

- We can declare **type classes** and **instances**.

```
class inhabited' (α : Type*) :=
(default : α)
instance : inhabited' ℕ := ⟨1⟩
```

- **Variables** can be:

(x) explicit

{x} implicit

[x] type class inference

```
variables {G : Type*} [comm_group G]

example (g h : G) : g*h = h*g :=
mul_comm g h
```

Tactics

Basic	Math-specific	Search
• intro • apply • rw • simp • ...	• continuity • linarith • ring • polyrith • ...	• library_search • suggest • hint

Example

src > test.lean

```
import tactic.basic

lemma modus_ponens (p q : Prop) : p → (p → q) → q :=
begin
  
end
```

▼ test.lean:5:2

▼ Tactic state

1 goal

```
p q : Prop
⊢ p → (p → q) → q
```

▶ All Messages (1)

Example

src > test.lean

```
import tactic.basic

lemma modus_ponens (p q : Prop) : p → (p → q) → q :=
begin
  intro hp,
end
```

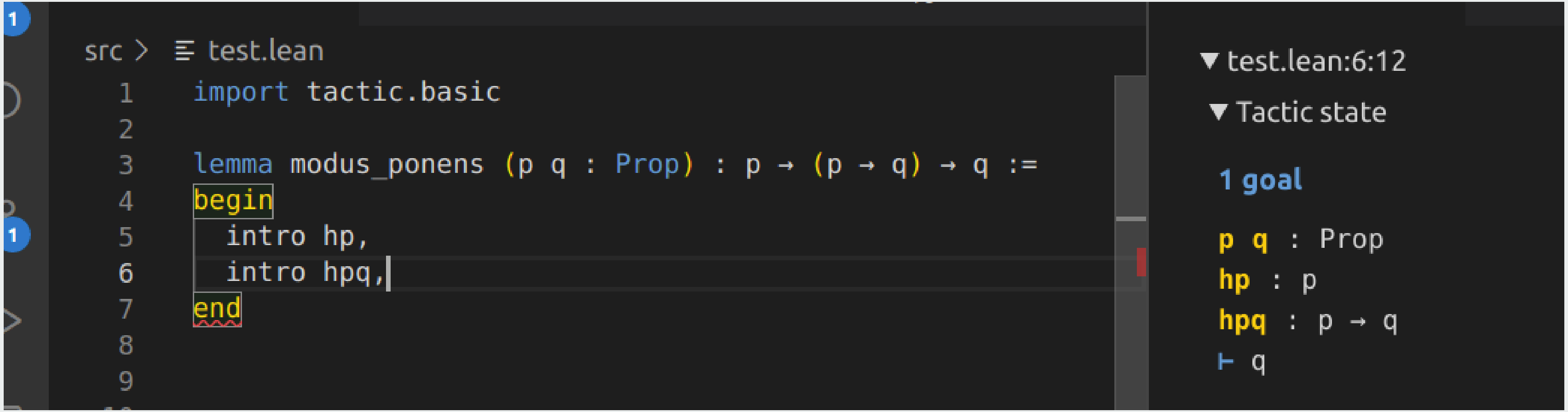
▼ test.lean:5:11

▼ Tactic state

1 goal

```
p q : Prop
hp : p
⊢ (p → q) → q
```

Example


src > test.lean
import tactic.basic
lemma modus_ponens (p q : Prop) : p → (p → q) → q :=
begin
intro hp,
intro hpq,
end
test.lean:6:12
Tactic state
1 goal
p q : Prop
hp : p
hpq : p → q
⊢ q

Example

src > test.lean

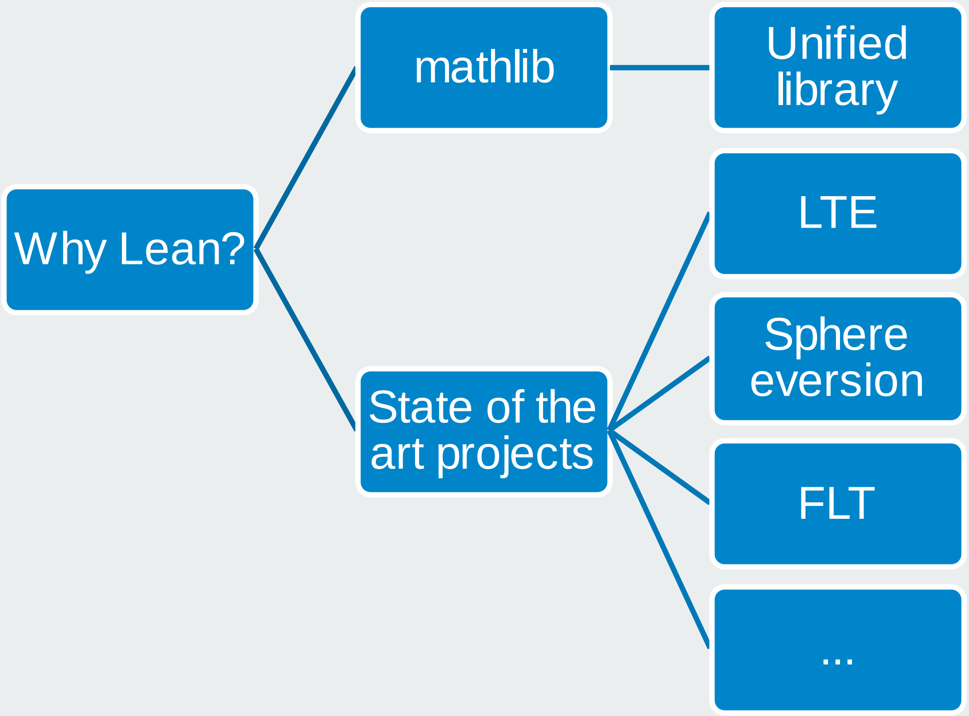
```
import tactic.basic

lemma modus_ponens (p q : Prop) : p → (p → q) → q :=
begin
  intro hp,
  intro hpq,
  exact hpq hp,
end
```

▼ test.lean:7:15

▼ Tactic state

goals accomplished 🎉

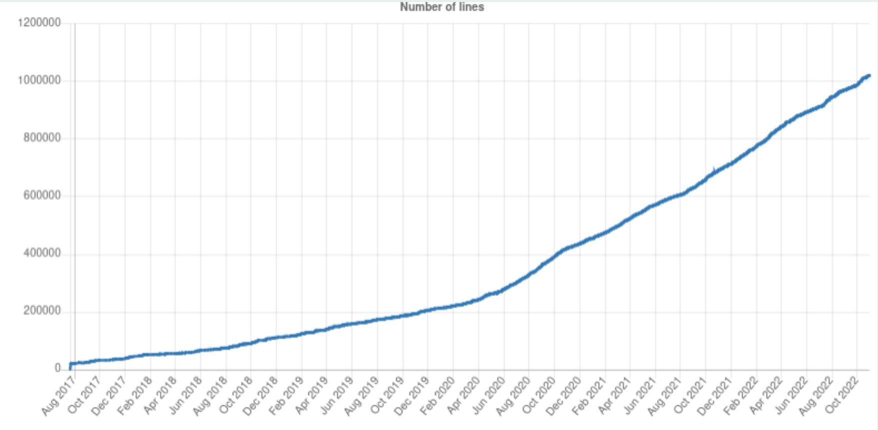
▶ All Messages (0)


Why Lean?
mathlib
Unified library
State of the art projects
LTE
Sphere eversion
FLT
...

mathlib

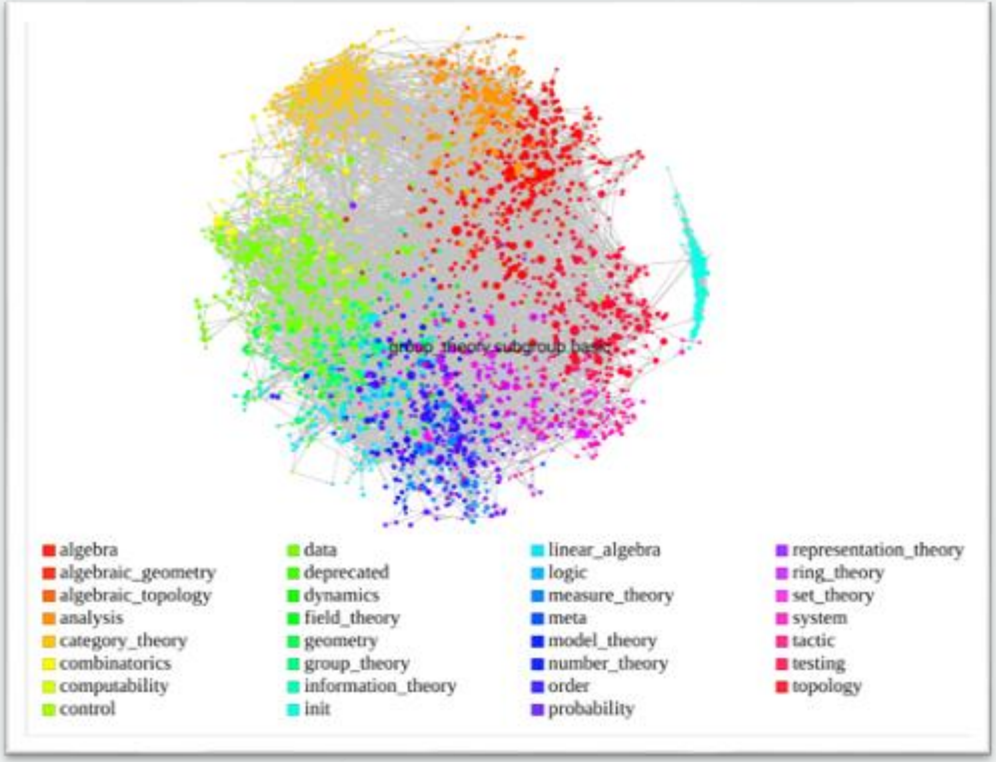
- Open source
- Decentralized
- Monolithic
- Overview

Definitions	Theorems	Contributors
42812	103024	282



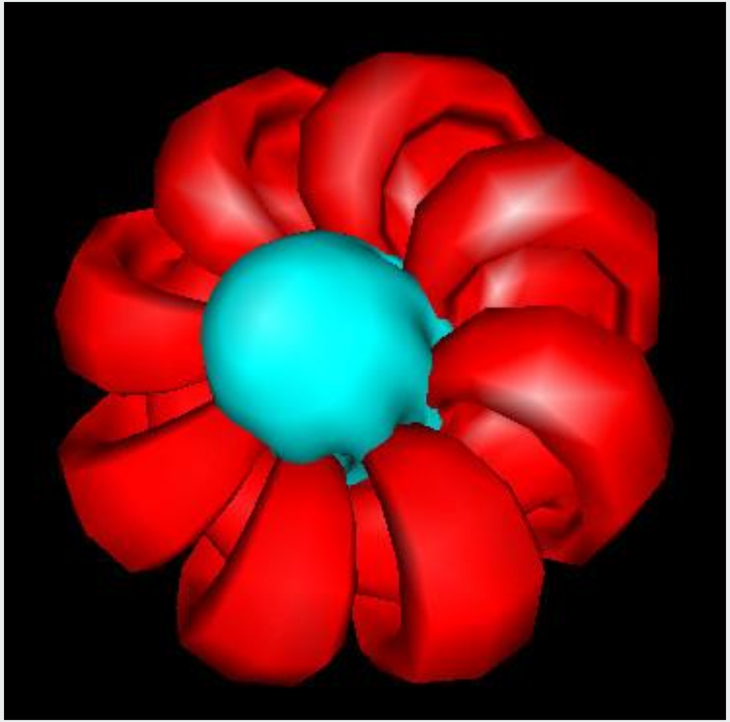
Mathlib's dependency graph:

- By Eric Wieser
- Interactive version



- Lean projects
  - LTE
  - Sphere Eversion
  - FLT
  - ...

Sphere eversion



- Led by Patrick Massot
- Turn a sphere inside-out in 3D space
- Differential topology
- Blueprint

Formalizing Number Theory: guiding goals

Fermat's Last Theorem

"If $x^n + y^n = z^n$ for n ≥ *3, then xyz = 0."*

- Formulated around 1637.
- Proven by Wiles and Taylor in 1995.
- Proof uses elliptic curves, modular forms, Galois representations, class field theory...

The Langlands Program

- Deep conjectures relating algebra and analysis, number theory and geometry.
- Largest research program in modern mathematics.

My Contributions

Formalized

- Adèles and idèles of global fields.
- Stating Global Class Field Theory.
- Extensions of norms.
- The p-adic complex numbers.

Ongoing

- Local Class Field Theory (with Filippo Nuccio).
- Divided powers (with Antoine Chambert-Loir).
- Fontaine's period rings.

Other Number Theory projects

Formalized

- p-adic numbers (R. Lewis).
- Witt vectors (J. Commelin, R.Lewis).
- Elliptic curves (K. Buzzard).
- Modular forms (C. Birkbeck).
- Galois cohomology (A. Livingston).

Ongoing

- FLT for regular primes (led by R. Brasca).
- Iwasawa Theory (A. Narayanan).
- Modularity Conjecture (K. Buzzard, M. Karatarakis).

Thank you! Questions? 🎉