

Scaling mathlib: tooling and automation for an ever-growing mathematics library

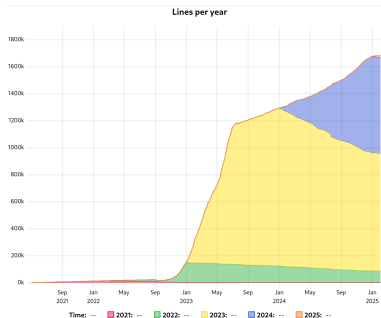
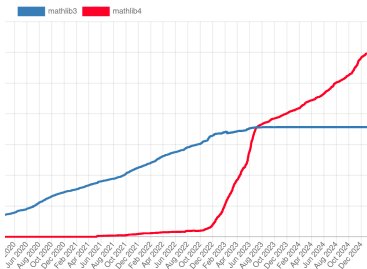
Michael B. Rothgang (he/him)

Formalised mathematics group
Universität Bonn



Lean together 2025
January 16, 2025

What is this all about?



Left: Number of files in mathlib and mathlib4 over time

Right: Lines in mathlib4 per year (without #aligns)

Up-to-date graphs: see

https://leanprover-community.github.io/mathlib_stats.html resp.

https://plugh.de/tmp/mathlib4_years_adjusted.html

What does mathlib's growth *really* mean?

Challenges for users: lots of churn

(e.g. renamed lemmas, import changes, breaking changes to proofs)

What does mathlib's growth *really* mean?

Challenges for users: lots of churn

(e.g. renamed lemmas, import changes, breaking changes to proofs)

Challenges for maintainers (more later):

- churn also applies
- keeping technical debt in check
- catch systemic issues: linters
- reviewing contributions

What does mathlib's growth *really* mean?

Challenges for users: lots of churn
(e.g. renamed lemmas, import changes, breaking changes to proofs)

Challenges for maintainers (more later):

- churn also applies
- keeping technical debt in check
- catch systemic issues: linters
- **reviewing contributions**

leanprover-community / mathlib4 Public

Code Issues (240) Pull requests (1.2k) Actions Projects (11) Wiki Security Insights

ispr is open

Labels (43) Milestones (0) New pull request

1,172 Open ✓ 19,104 Closed

Author	Label	Projects	Milestones	Reviews	Assignee	Sort
chore: address some porting notes about coercions of equiv	porting notes	tech debt				1
chore(Analysis/Normed/Ring/WithAbs): make equiv a ring equiv	test suite	WIP				1
chore(Algebra/Group/Equiv): split into Defs and Basic	test suite	test suite				1
test leanprover/lean4#6627 against #20532 (downstream cache)	test suite					1
chore: move import of Tactic.Common downstream from Pairwise	test suite	test suite				3

Today's topic: what can mathlib tooling do for you?

- as a user: dealing with churn
- as a contributor: linting to catch common mistakes
- as a reviewer
- as a maintainer

Before we begin

This is joint work with many people, including



Johan Commelin, Mario Carneiro and Damiano Testa (left to right)

Tools for mathlib users

- Dealing with mathlib churn
 - deprecation warnings for deleted/renamed lemmas
 - files moved of split: add `import Mathlib` and use `#min_imports`
 - Can be automated: prototypes
 - lake exe update_deprecations (Damiano Testa),
 - lake exe bump-imports (R.)
- coming soon: `lake exe cache miniget`

Tools for mathlib users: finding lemmas

- strict naming convention: `https://leanprover-community.github.io/contribute/naming.html`
- lemma-finding tactics: `exact?`, `apply?` [`using h`], `rw?`
- loogle (Joachim Breitner): structured theorem search
- moogle, leansearch: natural language search
- generated documentation: rendered in the browser, clickable

Tooling for mathlib contributors: overlooked aspects

- open development, live collaboration (github/zulip)
- welcoming community
- newcomer-friendly: great docs, focus on good tools
- code review: quality control and **teaching venue**
- continuous integration; not rocket science rule (Graydon Hoare)

Scaling mathlib: tools for contributors

- `add_deprecations` script (Damiano Testa):
deprecate renamed lemmas automatically
- linters: automate style, consistency, robustness, correctness checks
many improvements last year
- mathlib cache

Scaling mathlib: tools for contributors

- `add_deprecations` script (Damiano Testa):
deprecate renamed lemmas automatically
- linters: automate style, consistency, robustness, correctness checks
many improvements last year
- mathlib cache
- `shake` (Mario Carneiro): find superfluous imports
- `lean4checker` (Kim Morrison):
ensure no meta-programming “tamperers” with the environment
- `lake exe refactor` (Mario Carneiro):
framework for large refactorings
- proof automation: e.g. `aesop`, `omega`, `fun_prop`,
`bv_decide`, `grind` (in progress)

Scaling mathlib: some technical challenges

- fast core system (see FRO talk)
- keeping up with core changes
- speeding up mathlib, for example
 - local profiling (easy)
 - benchmarking and performance tracking
 - refactor FunLike hierarchy (Anne Baanen, $\approx 20\%$ speed-up)
 - tweaking TC synthesis (Yuyang Zhao, in progress)
- keeping technical debt in check
- import refactoring and reduction
parallelism; less recompilation; easier minimisation

Tooling for reviewers

- finding good reviewers: auto-labelling

PR summary [1643d5e952](#)

Import changes for modified files

No significant changes to the import graph

► Import changes for all files

Declarations diff

No declarations were harmed in the making of this PR! 🎉

► You can run this locally as follows

The doc-module for `script/declarations_diff.sh` contains some details about this script.

▼ Decrease in tech debt: (relative, absolute) = (4.17, 4.55)

Current number	Change	Type
1	-4	backwards compatibility flags
1417	-2	enw
11	-6	maxHeartbeats modifications

Current commit [1643d5e952](#)

Reference commit [b1178f81d](#)

- sticky summary comment (Damiano Testa):
show renamings, import changes, technical debt change
- emoji-bot: signal on zulip if a PR has already been reviewed/merged

Scaling reviews of PRs

- everybody can review, don't be shy!
guidelines: <https://leanprover-community.github.io/contribute/pr-review.html>
- *maintainers* have final merge rights: currently 28
- *reviewers* group: experienced members (currently 49)
PR approval notifies maintainers directly
- helps share the load: 1/3 PRs gets reviewer-approved first
4986 merged PRs > 15000; 1659 got maintainer-merged, 28 twice or more
- make it enjoyable: regular review/triage meeting

Editorial tooling for mathlib (Commelin-R.)

leanprover-community / mathlib4 Public

Notifications Fork 361 Star 1.7k

<> Code Issues 240 Pull requests 1.2k Actions Projects 11 Wiki Security Insights

is:pr is:open Labels 63 Milestones 0 New pull request

1,172 Open ✓ 19,104 Closed

Author	Label	Projects	Milestones	Reviews	Assignee	Sort
#20714 opened 21 minutes ago by Vierkantor	porting-notes tech debt					1
#20713 opened 33 minutes ago by MichaelStollBayruth	analysis WIP					1
#20712 opened 1 hour ago by Vierkantor	awaiting CI algebra					1
#20711 opened 2 hours ago by kim-em	large-import					1
	maintainer-merge tdata					3

Editorial tooling for mathlib (Commelin-R.)

leanprover-community / mathlib4 Public

Notifications Fork 361 Star 1.7k

Code Issues 240 Pull requests 1.2k Actions Projects 11 Wiki Security Insights

is:pr is:open Labels 63 Milestones 0 New pull request

1,172 Open 19,104 Closed

	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
chore: address some porting notes about coercions of equivs	#20714 opened 21 minutes ago by Vierkantor	porting-notes tech debt					1
chore(Analysis/Normed/Ring/WithAbs): make equiv a ring equiv	#20713 opened 33 minutes ago by MichaelStollBayreuth	analysis WIP					1
chore(Algebra/Group/Equiv): split into Defs and Basic	#20712 opened 1 hour ago by Vierkantor	swating CI algebra					1
chore: test leanprover/lean4#6627 against #20532 (downstream cache)	#20711 opened 2 hours ago by kim-em	large-import					1
chore: move import of Tactic.common downstream from Pairwise		maintainer-merge tdata					3

Goal

Keep track of all open pull requests, and ensure each gets a timely response.

Editorial tooling for mathlib (Commelin-R.)

Overall goal

Keep track of all open pull requests, and ensure each PR gets a timely response.

Editorial tooling for mathlib (Commelin-R.)

Overall goal

Keep track of all open pull requests, and ensure each PR gets a timely response.

Specific aims

- track the review queue, filtered e.g. by subject area
- “leave no PR behind”
- better “last updated” information; total time in review
- different target groups need different information

A review and triage dashboard for mathlib

Mathlib review and triage dashboard

Welcome to the mathlib review and triage webpage! There are many ways to help, what are you looking for in particular?

[Review queue](#)
[For maintainers \(quick\)](#)
[Help out](#)
[Triage dashboard](#)
[Why is my PR not on the queue? Can I see all my PRs?](#)

Stale new contributor PRs

 entries per page

 Search:

Number	Author	Title	Labels	+/-		Assignee(s)	Updated	Last status change	total time in review
14060	YnirPaz	feat(SetTheory.Ordinal.C Clubs): define club sets and prove basic properties	new-contributor t-logic blocked-by-other-PR	315/3	6	93 nobody	2025-01-04 21:33 (10 days ago)	2025-01-04 21:33:47+00:00 (10 days ago)	92 days
13685	niklasmohr	feat(Combinatorics): add definitions for network flows	new-contributor t-combinatorics merge-conflict awaiting-author	241/0	3	34 nobody	2024-12-10 18:25 (1 month ago)	2024-12-10 18:25:48+00:00 (1 month ago)	85 days
14313	grhkm21	feat(RepresentationTheory.FdRep): FdRep is a full subcategory of Rep	new-contributor t-algebra t-category-theory	47/8	1	19 nobody	2024-10-30 11:59 (2 months ago)	2024-10-30 11:59:51+00:00 (2 months ago)	81 days

Stale unassigned PRs on the review queue

 entries per page

 Search:

Number	Author	Title	Labels	+/-		Assignee(s)	Approvals(s)	Updated	Last status change	total time in review
19440	Bergschaf	feat(Order/Nucleus): Nucleus	t-order RFC	115/0	2	8 nobody	none	2025-01-01 17:53 (14 days ago)	2025-01-01 17:53:15+00:00 (14 days ago)	48 days
20366	joelriou	chore(CategoryTheory): move Functor.IsWellOrderContinuous	t-category-theory	86/57	4	1 nobody	none	2024-12-31 19:08 (15 days ago)	2024-12-31 16:00:54+00:00 (15 days ago)	15 days
19325	madvorak	style(Computability/ContextFreeGrammar/reverse): injective and surjec...	t-computability	4/4	1	6 nobody	none	2024-12-31 00:48 (15 days ago)	2024-12-31 00:48:08+00:00 (15 days ago)	55 days
20340	alreadydone	feat(LocalizedModule): expand API	t-algebra	94/31	4	3 nobody	none	2024-12-30 18:34 (16 days ago)	2024-12-30 18:34:15+00:00 (16 days ago)	16 days

Some technical details

- statically generated page: Python writes HTML (with some CSS/JS)
- backend: download metadata for all github PRs, near-real time
- some surprises/lessons learned
 - local testing is good
 - beware of different time-zones
 - github actions: run at most every 5min, and no guarantees
 - prepare for outliers, e.g. PRs with *huge* metadata
 - pagination
 - did it really update? Github's "last update" is incomplete
 - expect errors: network, intermittent, push races, ...
 - github's search results are sometimes outdated

An unexpected application

Yael Dillies' upstreaming dashboard for LeanCamCombi

<https://yaeldillies.github.io/LeanCamCombi/upstreaming>

“Which open mathlib PRs modify file ‘X’?”

Files ready to upstream

The following files are `sorry`-free and do not depend on any other LeanCamCombi, meaning they can be readily PRed to Mathlib.

- `LeanCamCombi.Multipartite`
- `LeanCamCombi.MetricBetween`
- `LeanCamCombi.Util`
- `LeanCamCombi.GrowthInGroups.LinearLowerBound`
- `LeanCamCombi.GrowthInGroups.ConstructibleSetData`
- `LeanCamCombi.Kneser.MulStab`
- `LeanCamCombi.ConvexityRefactor.StdSimplex`
- `LeanCamCombi.GraphTheory.ExampleSheet2`
- `LeanCamCombi.Corners.CombiDegen`
- `LeanCamCombi.PHD.VCDim.AddVCDim`
- `LeanCamCombi.Mathlib.RingTheory.FinitePresentation`
 - `feat: induction principle for finitely presented ring homs #20069`
 - `refactor: allow non-unital AlgEquiv #8686`
- `LeanCamCombi.Mathlib.Order.BooleanSubalgebra`
 - `feat(Aesop): Improve SetLike ruleset #20477`
 - `refactor(SetLike): Introduce OrderedSetLike and remove SetLike.instPartialOrder #20638`
- `LeanCamCombi.Mathlib.Topology.MetricSpace.MetricSeparated`
 - `chore: review induction principles for Set.Finite #20444`
- `LeanCamCombi.Mathlib.Probability.ProbabilityMassFunction.Constructions`
 - `Chore: Replace open scoped Classical with (open scoped Classical in) or (classical) #20501`
- `LeanCamCombi.Mathlib.Data.Prod.Lex`
 - `feat: definition of \lt on $\alpha \times \beta$ in partial orders #20067`

Summary

- ① It takes a village to create a big library.
- ② Empower and mentor new contributors.
- ③ Build tooling to automate as much as you can.
- ④ Custom tooling takes effort, but can be worth it!

Ask not what `mathlib tooling` can do for you —
ask what you can do for `mathlib tooling`!

Summary

- ① It takes a village to create a big library.
- ② Empower and mentor new contributors.
- ③ Build tooling to automate as much as you can.
- ④ Custom tooling takes effort, but can be worth it!

Ask not what **mathlib tooling** can do for you —
ask what you can do for **mathlib tooling**!

Thanks for listening! Any questions?