



The Archive of Formal Proofs

Manuel Eberl

Outline

- 1. The Isabelle Universe**
- 2. Size and Comparison**
- 3. The Life of an AFP Entry**
- 4. Technical Details**
- 5. Issues and Challenges**
- 6. The Future of Isabelle and AFP**

What is Isabelle?



- An interactive theorem prover / proof assistant (like Rocq, Lean, Mizar)
- Like most of the big systems, quite old (first created in 1986)
- Can do multiple logics; in practice mostly Higher-Order Logic (HOL)
- Simple, less expressive logic (compared to Rocq and Lean)
- Focus on good automation and high performance
- Big focus on ‘readable’, structured proofs
- Large library of mathematics (especially analysis)
- No longer *just* an ITP, but a whole ecosystem of software and components.

The Isabelle Universe

The Distribution

- **Distribution** = what you get when you download Isabelle
- Contains various 'sessions': Pure, ZF, HOL, HOL-Library, HOL-Algebra, HOL-Computational_Algebra, HOL-Combinatorics, HOL-Number_Theory, HOL-Analysis, HOL-Complex_Analysis
- Mostly curated general-purpose library material
- In a Mercurial repo. Few people have push access (≈ 15 active maintainers).
- No formal process for external contributions, but people do post material to include on the mailing list.

The AFP

The [Archive of Formal Proofs](#) was created in 2004 by Nipkow and Klein

- Purposes:
 - to allow people to contribute and find proof developments
 - to give Isabelle developers a feeling of how people use the system
 - to immediately see repercussions of proposed changes to the system
- Entries can (and do) use each other
- Permissive licencing: BSD-style licence (default) or LGPL (< 3%)

Nowadays:

- 8 editors and 1 technical administrator (Fabian Huch)
- 929 articles by 566 authors, 300k lemmas, 5300 kLOC
- Mathematics, logic, computer science, physics, tools

Why the distinction?

- Distribution is a controlled space of ‘canonical’ general-purpose formalisations and tools.
- On the other hand: AFP solicits diverse submissions from everyone
- Authors have a lot of freedom and independence (more later)
- Current wisdom is to put more things in the AFP rather than the distribution.

Example: Basic complex analysis goes in the distribution. The Riemann ζ function goes in the AFP.

But: The distinction is not rigid. Stuff gets moved in both directions all the time.

- Landau symbols (‘Big-O’) from AFP to distribution
- Bernoulli numbers from distribution to AFP

The Visible Isabelle Universe

Visible Universe: Distribution + AFP

Dark Matter: Public, separate projects (e.g. seL4, IsaFoL, IsaFoR/CeTA)

The Beyond: Private projects we know little about (e.g. by Apple, AWS)

- When we make changes, we look at the effects on the Visible Universe and make sure it still works (continuous integration).
- Projects outside it have to take care of this themselves.
- **History shows:** Developments that not in the AFP quickly succumb to 'bit rot' unless there is a company or big research group behind them.
- One sad example: Avigad's formalisation of the Prime Number Theorem

Size and Comparison

Comparison

Comparing the AFP to other similar collections is comparing apples and oranges.

Let's try nevertheless:

- AFP arguably the largest *centralised* repository of formal proofs by any metric.
- 92 of Freek Wiedijk's 'top 100' theorems formalised in Isabelle.
(HOL Light: 89, Lean: 82, Rocq: 79)
- 2–3 times bigger than Lean's mathlib – but Lean is catching up fast.
- The collection of all Rocq packages is *much* bigger, but not centralised, not intercompatible, and with much duplication.

Huch's PhD thesis draft has a more in-depth comparison.

Statistics

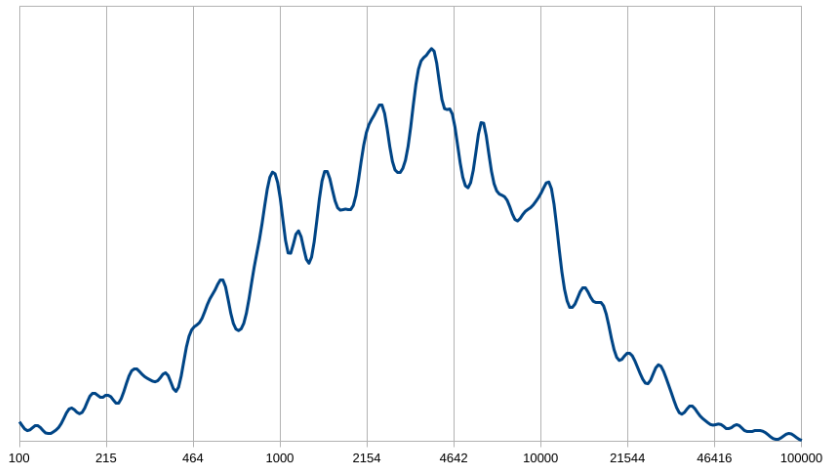
Isabelle-2025 distribution:

- In total: ≈ 900 kLOC, 90 % thereof HOL
- Base HOL has 123 kLOC, HOL-Library another 68 kLOC
- Biggest HOL session: HOL-Analysis with 176 kLOC

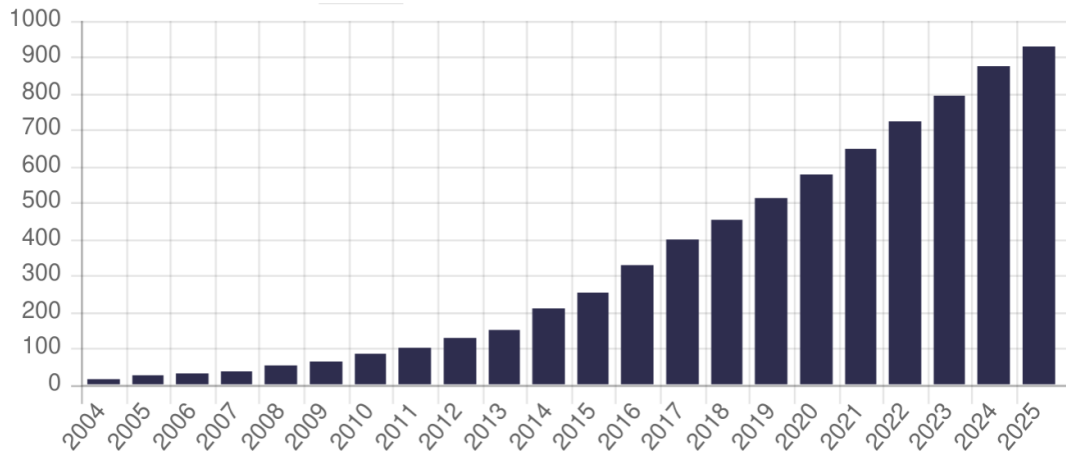
AFP-2025:

- 5300 kLOC total (average 6 kLOC, median 3.2 kLOC, $\sigma = 9$ kLOC)
- heavily right-skewed, high kurtosis (most entries fairly small, some very big)
- 50 % have 1200–7000 lines, 98 % have < 30000 lines
- 60 % proofs, 20 % lemma statements, 10 % definitions
- Most entries are not used by any other entries.

LOC Per Entry Histogram




AFP Growth



The Life of an AFP Entry

Submission

- Entries typically submitted to latest release version of the AFP via the **submission website** .
- Exceptionally, submission via email and to the afp-devel is possible.
- Submissions are built in a sandbox before they can be sent to the editors.
- Automated style feedback using the Isabelle linter
- Submissions consist of
 - Isabelle 'theory' files
 - ROOT file specifying session information (e.g. timeouts, base session)
 - LaTeX/BibTeX for PDF generation
 - metadata (author information, abstract, topics)
 - rarely: additional data, e.g. ML code, SAT proofs to import, etc.
 - can also export result data (e.g. generated code)

Reviewing

- Editors are informed of new entries via email and can 'claim' submissions via a web interface
- Reviewing is almost always done by the editor themselves
- Sometimes submissions do get rejected, but generally requirements are low:
 - honesty (i.e. accurately formalise what you claim to have formalised)
 - significance (i.e. no trivialities – but we do have some very short entries)
 - follow style guidelines (e.g. avoid brittle proof patterns, \rightsquigarrow linter)
 - good structure and explanations
- Degree of enforcement depends on the taste of the editor.
- Reviewing typically takes a few days to several weeks.
- There is often some back-and-forth with authors concerning formatting, metadata, or brittle proof steps.

Reviewing

Caveats:

- We rarely use external reviewers and there is no anonymity in reviewing.
- Work in the AFP does not have to be particularly novel or significant.
- There is no guarantee that the approach chosen is the best or most elegant.
- Due to lack of time and expertise, there is not even a guarantee of ‘honesty’ (as defined before)


The AFP **does not see itself as a scientific journal**, but rather as a data repository like Zenodo or the Software Heritage Foundation.

But: Unlike these, we do guarantee that all our entries work and continue to work.

Acceptance

- Accepted articles are added to the latest afp-release repo.
- They appear on the website immediately and are announced on the isabelle-users mailing list.
- An editor (usually Gerwin Klein) adds them to afp-devel some time after. This often breaks things, but the breakage is usually minor enough that someone will quickly take care of it.
- Entries submitted directly to afp-devel only appear on the afp-devel website until the next release.

Maintenance

- The distribution and the AFP live in separate Mercurial repos.
- The AFP used to be on Bitbucket, now on **Heptapod** .
- Authors receive full push access upon request to improve their entries and metadata as they see fit.
- Changes must be 'monotonic', but this is neither checked nor enforced.
- Changing other people's entries is possible, but it is good etiquette to ask for permission before changing anything substantial
- People who add substantial material to existing entries can ask to be credited as *contributors* in the metadata

Maintenance

Supporting infrastructure:

- First, there was isatest/afptest.
- Lars Hupel introduced continuous integration tests via Jenkins in 2015.
- Since 2025: Isabelle Build Manager (more later)
- Build hardware consists of some big servers at the Nipkow group in Munich.

How it works:

- CI runs on every push to distribution and AFP (except `very_slow`).
- There is also a way to test work *before* making a public commit.
- Breaking the build happens and is typically not considered a huge issue.
- But: *If you break it, you have to fix it.*

Usually within at most a week or two, ideally much earlier.

Maintenance

This model seems to work very well. We have not had issues with AFP authors.

The price: Any small change to the distribution can potentially break many things in the AFP, which potentially stifles innovation.

If we want to keep all entries running, this is probably unavoidable.


Expecting all authors to keep their own entries in sync with Isabelle development indefinitely would be unrealistic.

Release

- One Isabelle release approx. every 9 months
- Some weeks before, Makarius announces a schedule with several RCs
- After some RCs, Makarius forks a new `isabelle-release` repo off from `isabelle-devel` (and similarly Gerwin for AFP)
- This begins a period of consolidation: no big changes are to be made and both the distribution and the AFP should be in working order at all time
- A few days after the final Isabelle release, Gerwin Klein makes a corresponding AFP release and things go back to normal
- After release, the Isabelle release repos are always frozen.
- The AFP release repos are also frozen except for metadata changes and new entries. Only editors have write access.

Technical Details

Website

- Since 2022, website is completely statically generated via Hugo.
- When accepting a new entry, editors regenerate the HTML and publish it on the website.
- Submission system implemented in Isabelle/Scala.
- MathJax allows using LaTeX-like maths formulas in abstracts.
- Can search through entry metadata and also formal content using FindFacts .
- Theories can be browsed with full syntax highlighting and clickable constants, theorem names, etc.
- The latter is achieved using Isabelle/Scala and a Hugo component upon site generation.
- All of this infrastructure is publicly available in the AFP repo.

Build Manager

- By 2024, running `distribution ∪ AFP \ (slow ∪ very_slow)` often took 1–2 hours even on a high-end server.
- This is painfully long for on-push CI and on-demand tests.
- In 2024, Fabian Huch and Makarius Wenzel introduced the [Isabelle Build Manager](#)
- Drop-in replacement for the normal Isabelle build command
- Distributed builds on heterogeneous clusters
- Sophisticated scheduling takes dependencies into account
- Using 8 machines: 10 min for `distribution`, 45 min for `AFP \ very_slow`. And often much less using partial builds.

External Connections

- Indexed by DBLP since 2012
- Twitter/Mastodon bots that announce new AFP entries based on an XML feed
- Entries can reference external publications (also via DOIs, which are checked)

Issues and Challenges

Structural Issues

- Every time you formalise *anything* you discover things that should be in the library but are not.
- Many entries have a theory like `More_X` or `X_Library` or material tagged with a `(* TODO Move *)` comment.
- Figuring out where each bit should go and moving it there is a lot of work.
- If it should go in the distribution, only a few people can do that and they tend to be busy with other things.
- Larry Paulson is the main person who regularly does this.

Structural Issues

If this material is *not* moved where it belongs:

- Other entries that need the same stuff either import otherwise unrelated entries just for their library (bad) or duplicate the material (worse).
- **Example:** Material that should be in the `Multivariate_Polynomials` entry is spread over a chain of half a dozen entries that *use* multivariate polynomials but are not *about* them and import each other.
- It's bad, but not bad enough for anyone to actually clean up the mess.

Refactoring tools could potentially make this easier in the future.

We also plan to encourage AFP authors to systematically point out material to be moved.

Stable URLs and DOIs

Unlike Zenodo, the AFP does not issue DOIs.

We made an attempt to do this a few years ago, but there were complications:

- Minting DOIs typically costs money
- DOIs are intended for documents that *never change*
- AFP entries, on the other hand, change all the time (typically just maintenance or monotonic changes)
- There was an idea to have one ‘concept DOI’ per entry and then a sub-DOI for every release.
- But that means minting *a lot* of DOIs on every release

We have not found another adequate solution yet – suggestions welcome.

For now, we simply attempt to keep the `isa-afp.org` entry URLs stable.

The Future of Isabelle and AFP

The Future

There are some hard questions about how to continue.

- Larry Paulson and Tobias Nipkow have both retired.
- Nipkow's group in Munich still exists but will be dissolved soon.
- Makarius is still going strong but has no institutional resources behind him.






Who will host, maintain, pay for the Isabelle + AFP infrastructure?

- This is a problem that affects all academic projects.
- **Idea:** Create some kind of non-profit organisation to take over.
- Possible source of money: Companies that use Isabelle in production
- Details, e.g. what *kind* of organisation, are still unclear.
- Suggestions for how to achieve this are welcome.

Acknowledgements

- Makarius Wenzel: main Isabelle ‘engineer’ for many years
- David Matthews: creator and maintainer of Poly/ML
- Fabian Huch: AFP technical administrator, author of *FindFacts*, build manager, and much more
- Max Haslbeck (the younger): creator of `afp_submit`
- Lars Hupel: first AFP sitegen in 2010 and CI in 2015
- Carlin MacKenzie (+ James Vaughan, Jacques Fleuriot): AFP website redesign in 2022 (MSc thesis)
- Yecine Megdiche: Isabelle linter (BSc thesis)
- T. Nipkow, G. Klein, L. Paulson, D. Traytel, A. Lochbihler, P. Lammich, R. Thiemann: fellow AFP editors
- The entire Isabelle community of course!

References

- J. Blanchette et al., *Mining the Archive of Formal Proofs* 
- C. MacKenzie et al., *An Evaluation of the Archive of Formal Proofs* 
- C. MacKenzie et al., *Re-imagining the Isabelle Archive of Formal Proofs* . 2022.
- F. Huch & M. Wenzel, *Distributed Parallel Build for the Isabelle Archive of Formal Proofs* . 2024.
- F. Huch, *FindFacts: A Scalable Theorem Search* . 2022.
- F. Huch, *Big Math in Interactive Theorem Provers*, PhD Thesis. To appear.

Conclusion

- AFP will probably hit 1000 entries next year and is still growing fast.
- Our infrastructure continuously has to improve just in order to catch up.
- Still, we are well-prepared for the future technologically.
- Organisational challenges (financing etc.) are more daunting, but I am optimistic.
- Dear Lean people, you are *not* the only game in town. ;)