

Coqdev

Software engineering supporting Coq evolution

Théo Zimmermann

Inria

September 21st, 2020

Personal introduction

- Member of the core development team of Coq:
 - First contribution to Coq in 2015 (8.5).
 - Participated in the release management since 2017 (8.7).
 - Maintainer of the documentation since 2018 (8.8).
 - Founder of the coq-community organization.
 - Main developer of coqbot, a bot supporting Coq development.
 - Co-organizer of the “Coq Users in Paris” Meetup.
- Defended a PhD at Université de Paris in 2019:

“Challenges in the collaborative evolution of a proof language and its ecosystem”
- Inria employee (funded thanks to Nomadic Labs) since 2020.

Coqdev project: overview

- Ongoing four year project.
- Funds my 3-year fixed-term Inria position (started Jan 2020).

Objectives:

- Accessibility of the Coq ecosystem.
- Community organization and development.
- Making Coq evolution robust.

“Scaling Coq by tapping into the incredible resources of its user community.”

Methods:

- Serving as release manager, community manager, maintainer...
- Open source tools to help Coq developers and Coq projects.
- Empirical studies and literature search.

How is Coq developed today?

- Core development team: currently 10 Inria employees (in Paris, Nantes and Sophia-Antipolis).
- Development in the open:
 - Written communication on GitHub and other publicly accessible platforms.
 - Coq Calls every week open to everyone and notes posted on the wiki.
- All code changes must be reviewed (pull requests).
- Maintainer teams for each component (more than 30 maintainers).

1 Done

- Improving communication channels (with C. Cohen)
- Coq workshop 2020 (with E. Gallego & H. Herbelin)
- New options for testing Coq projects (with E. Martin-Dorel)
- Julien Coolen's internship on coqbot development

2 Ongoing

- 8.12 release management (with E. Gallego)
- Restructuring the reference manual (with J. Fehrle)
- Coq platform (with M. Soegtrop)
- Twitter (with E. Gallego & A. Trunov)
- Research on community organizations for package maintenance

3 Planned

- Online Meetup (with E. Gallego)
- Automatic extraction of test cases from CI failures (with J. Gross)
- More automation for coq-community projects (with K. Palmkog)
- Research on collaboration and community involvement

① Done

Improving communication channels (with C. Cohen)

Coq workshop 2020 (with E. Gallego & H. Herbelin)

New options for testing Coq projects (with E. Martin-Dorel)

Julien Coolen's internship on coqbot development

② Ongoing

③ Planned

① Done

Improving communication channels (with C. Cohen)

Coq workshop 2020 (with E. Gallego & H. Herbelin)

New options for testing Coq projects (with E. Martin-Dorel)

Julien Coolen's internship on coqbot development

② Ongoing

③ Planned

Leaving Gitter, joining Zulip

The development team was using Gitter since 2017:

- single-threaded,
- high traffic,
- lacking space for beginners' questions.

Zulip:

- Slack-like but open source,
- flexible stream / topic mechanism,
- increasingly popular (adopted by Lean, HoTT, Isabelle, and now Agda).

Migration and import from Gitter mainly handled by Cyril Cohen.

State of the discussion platforms 2020

- Discourse forum (adopted in 2019 following OCaml example):
 - structured and easily browsable Q&A / discussion
 - other languages than English accepted
 - mailing list mode
 - **409 users**
 - administrator: myself
 - 5 moderators (speak Chinese, Spanish, French, German, Russian).
- Zulip chat:
 - casual and high traffic discussions
 - **346 users**
 - streams for projects related to Coq (CertiCoq, Elpi, Equations, jsCoq, math-comp, MetaCoq, Mtac2, SerAPI, VsCoq, etc.)
 - integrations announcing new **Discourse** posts and **Stack Overflow** questions
 - administrators: Cyril Cohen and myself.

State of the discussion platforms 2020

Mailing lists:

- Coqdev
 - virtually not used anymore
 - TODO: make it a mirror of the Coq dev category of Discourse
- Coq-Club:
 - still active
 - QUESTION: how to reduce the need for cross-posting / encourage migration to Discourse.

① Done

Improving communication channels (with C. Cohen)

Coq workshop 2020 (with E. Gallego & H. Herbelin)

New options for testing Coq projects (with E. Martin-Dorel)

Julien Coolen's internship on coqbot development

② Ongoing

③ Planned

Coq workshop 2020

Organizers: Emilio Gallego, Hugo Herbelin and myself.

- Associated with IJCAR-FSCD 2020 (a merger including ITP)
- Initially planned as a physical meeting in Paris Nord.

For the first time:

- Held virtually due to Covid-19.
- Spread over two days (as planned).
- Included two panels in addition to the usual program.

Tools: Zoom webinar + Zulip for questions + YouTube livestream

Coq workshop 2020: stats

- 312 registered participants (top IJCAR-FSCD workshop)
- 100 people joined Zulip stream
- about 50 to 100 simultaneous viewers on Zoom + YouTube
- 14 talks + 10 panelists (not counting Coq devs)
- 7 moderators or session chairs
- 10 PC members (including the organizers)

① Done

Improving communication channels (with C. Cohen)

Coq workshop 2020 (with E. Gallego & H. Herbelin)

New options for testing Coq projects (with E. Martin-Dorel)

Julien Coolen's internship on coqbot development

② Ongoing

③ Planned

coq-community templates

Since 2018, coq-community provides templates for setting up CI:

- initially Travis CI
 - building with both Nix and opam (Docker-Coq)
 - both solutions **avoid recompiling Coq** (speed up feedback)
- now also CircleCI (contribution by Yishuai Li)
- and GitHub Actions.

Templates play an educational and normative role, beyond their usage in coq-community projects.

GitHub Actions

- CI/CD solution introduced just last year (Aug. 2019 in beta)
- Yet immensely popular because:
 - well integrated with GitHub
 - generous free tier
 - Action Marketplace
- Docker-Coq-Action:
 - relies on Docker-Coq images to build with multiple versions of Coq & OCaml;
 - annotates changed files with warnings and errors (thanks to Jason Gross).

Easy CI configuration with Docker-Coq-Action

```
name: CI

on:
  push:
  pull_request:

jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: coq-community/docker-coq-action@v1
        with:
          opam_file: 'my-project.opam'
```

① Done

Improving communication channels (with C. Cohen)

Coq workshop 2020 (with E. Gallego & H. Herbelin)

New options for testing Coq projects (with E. Martin-Dorel)

Julien Coolen's internship on coqbot development

② Ongoing

③ Planned

History of coqbot

- **Software** created (during my PhD, in 2018) to **automate** processes and **simplify** collaborative development (the coqbot GitHub account is older than this).
- First main task: **synchronize pull requests** on GitHub with GitLab (for CI) and report results back (also used by other projects like math-comp).
- Second main task: help the backporting process.

Technology

- OCaml server with `cohttp-lwt-unix`.
- Type-safe requests with GitHub's GraphQL API and `reasonml-community/graphql-ppx`.
- Deployed to Heroku.

Julien Coolen's internship

- Officially started June 22nd (should have started one month before but Covid-19 created administrative issues).
- Ended August 31st.
- Funded thanks to Nomadic Labs.

Achievements:

- Infrastructure:
 - Simplified the installation process (GitHub App).
 - Simplified & automated build & deployment thanks to Docker.
- Features:
 - Support merging with coqbot (reduce barriers for onboarding new maintainers).
 - Support calling coqbot to run the bug minimizer (with Jason Gross).

① Done

② Ongoing

8.12 release management (with E. Gallego)

Restructuring the reference manual (with J. Fehrle)

Coq platform (with M. Soegtrop)

Twitter (with E. Gallego & A. Trunov)

Research on community organizations for package maintenance

③ Planned

① Done

② Ongoing

8.12 release management (with E. Gallego)

Restructuring the reference manual (with J. Fehrle)

Coq platform (with M. Soegtrop)

Twitter (with E. Gallego & A. Trunov)

Research on community organizations for package maintenance

③ Planned

8.12 release dates

- 8.12+beta1: June 18th
- 8.12.0: July 27th
- 8.12.1: ~September

① Done

② Ongoing

8.12 release management (with E. Gallego)

Restructuring the reference manual (with J. Fehrle)

Coq platform (with M. Soegtrop)

Twitter (with E. Gallego & A. Trunov)

Research on community organizations for package maintenance

③ Planned

A better structure for the manual

- The manual had too many chapters and a long “Addendum”.
- New organization discussed with Matthieu Sozeau:
 - In 8.12, we gathered those in just 4 parts and 8 chapters, divided into multiple pages.
 - Still ongoing: some old chapters are going to be split in 8.13 (Vernacular commands, Tactics) or possibly even later.
- Call for external contributions to get new material (e.g., on proof engineering).

Other 8.12 improvements to the manual

- Fixing the documented syntax (huge effort relying on tooling to extract the actual syntax from source files and manual labor to insert it, document it and review it).
- More consistency in the presentation of syntax.
- Better documentation of attributes.
- Introduction of a glossary.

① Done

② Ongoing

8.12 release management (with E. Gallego)

Restructuring the reference manual (with J. Fehrle)

Coq platform (with M. Soegtrop)

Twitter (with E. Gallego & A. Trunov)

Research on community organizations for package maintenance

③ Planned

Problem

Coq has a **rich ecosystem** which has scaled much beyond what Inria alone could maintain.

It is important to **ease access** to this ecosystem so that beginners, students, industrial users, academics, etc. can take **full advantage** of Coq.

Goals

- Easy to install on all major systems (Windows, macOS, Ubuntu, ...).
- Comes with all the important packages of the ecosystem.
- Same packages and same versions on all the systems.
- Stable set of stable packages.

① Done

② Ongoing

8.12 release management (with E. Gallego)

Restructuring the reference manual (with J. Fehrle)

Coq platform (with M. Soegtrop)

Twitter (with E. Gallego & A. Trunov)

Research on community organizations for package maintenance

③ Planned

① Done

② Ongoing

8.12 release management (with E. Gallego)

Restructuring the reference manual (with J. Fehrle)

Coq platform (with M. Soegtrop)

Twitter (with E. Gallego & A. Trunov)

Research on community organizations for package maintenance

③ Planned

The origin of coq-community

The main goal of coq-community is the **long-term maintenance of Coq packages**.

Method: ease the transition between maintainers by hosting packages under a shared organization.

Inspiration: elm-community, an organization with similar goals in the Elm ecosystem.

Since then: coq-community has influenced the ocaml-community organization which was created shortly after.

A model of community organization ...

- Instead of taking responsibility of forking alone, a fork is created under a **shared organization** (associated to a specific ecosystem).
- Packages still get a **principal maintainer** (avoid diluting responsibility).
- But if a maintainer drops out, it is **easy to replace them** (all that matters is to keep several active organization admins).

A model of community organization ...

- Instead of taking responsibility of forking alone, a fork is created under a **shared organization** (associated to a specific ecosystem).
- Packages still get a **principal maintainer** (avoid diluting responsibility).
- But if a maintainer drops out, it is **easy to replace them** (all that matters is to keep several active organization admins).

Consequences of the existence of such an organization:

- Forks that it hosts are **more trustworthy** (in particular regarding their sustainability).
- It provides an **exit strategy** for authors who do not want to maintain a package anymore (propose a transfer).

... emerging in many ecosystems

Organization	Ecosystem	Founded	# repos	# members
Vox Pupuli	Puppet	Sep. 2014	203	62
Sous Chefs	Chef	May 2015	82	19
elm-community	Elm	Nov. 2015	54	14
dlang-community	D	Dec. 2016	28	7
reasonml-community	ReasonML	~ 2017	31	13
coq-community	Coq	Jul. 2018	40	11
ocaml-community	OCaml	Aug. 2018	14	6

Organizations with similar objectives (long-term maintenance) with no explicit process for hard forking also exist.

- Work presented at SoHeal (an ICSE 2020 workshop).
- TODO: what makes such an organization successful?

① Done

② Ongoing

③ Planned

Online Meetup (with E. Gallego)

Automatic extraction of test cases from CI failures (with J. Gross)

More automation for coq-community projects (with K. Palmkog)

Research on collaboration and community involvement

① Done

② Ongoing

③ Planned

Online Meetup (with E. Gallego)

Automatic extraction of test cases from CI failures (with J. Gross)

More automation for coq-community projects (with K. Palmkog)

Research on collaboration and community involvement

① Done

② Ongoing

③ Planned

Online Meetup (with E. Gallego)

Automatic extraction of test cases from CI failures (with J. Gross)

More automation for coq-community projects (with K. Palmkog)

Research on collaboration and community involvement

The problem

- Currently, every pull request is tested against a test-suite + ~30 external Coq libraries.
- Pipelines take several hours of parallel build time and more than a dozen hours of sequential build time to run to conclusion.
- More than half of the time where an external library is broken, the test-suite is not.
- We do not learn from these cases to improve the test-suite.

The goal

- **Systematically** and mostly automatically extract a new (minimized) test case from a failing test of an external library.
- Significantly reduce the frequency when an incompatibility is discovered only in the external libraries.
- Do not always invoke the full pipeline.

① Done

② Ongoing

③ Planned

Online Meetup (with E. Gallego)

Automatic extraction of test cases from CI failures (with J. Gross)

More automation for coq-community projects (with K. Palmkog)

Research on collaboration and community involvement

Helping maintainers through automation

- coq-community templates: automated upgrades
- deployment of documentation
- preparation of releases
- some automated porting to new versions of Coq

① Done

② Ongoing

③ Planned

Online Meetup (with E. Gallego)

Automatic extraction of test cases from CI failures (with J. Gross)

More automation for coq-community projects (with K. Palmkog)

Research on collaboration and community involvement

Issue triaging

Issue tracker:

- currently ~2500 open issues
- status not always clear

Design and test a process to:

- keep the issue tracker up-to-date
- better prioritize
- clearly advertise “help wanted” issues
- involve the community in the process

Extending the teams of maintainers

Coq reviewing and merging process:

- based on code owners since 2018
- progressively introduced code owner teams
- all components switched to teams in 2020

Code owner teams bring benefits (to evaluate empirically):

- easier to join / leave a team (impact on the renewal process?)
- more people = better reactivity?
- impact on the components themselves?

How can we encourage even more people to join the maintainer teams?

RFCs / CEPs

- A process to propose, discuss and approve changes based on **design documents**.
- Increasingly popular: Python (PEPs), Rust, Coq (CEPs) , Nix, OCaml...
- Need a process:
 - what makes RFCs successful?
 - how to take decisions?
- For what purpose is it well suited?
- Can we encourage community involvement in language evolutions thanks to RFCs / CEPs?

Goal: study examples, draw conclusions and apply them to CEPs.

1 Done

- Improving communication channels (with C. Cohen)
- Coq workshop 2020 (with E. Gallego & H. Herbelin)
- New options for testing Coq projects (with E. Martin-Dorel)
- Julien Coolen's internship on coqbot development

2 Ongoing

- 8.12 release management (with E. Gallego)
- Restructuring the reference manual (with J. Fehrle)
- Coq platform (with M. Soegtrop)
- Twitter (with E. Gallego & A. Trunov)
- Research on community organizations for package maintenance

3 Planned

- Online Meetup (with E. Gallego)
- Automatic extraction of test cases from CI failures (with J. Gross)
- More automation for coq-community projects (with K. Palmkog)
- Research on collaboration and community involvement