



# COQ DEVELOPMENT TEAM SESSION

**Coq Development Team**

CoqPL 2021

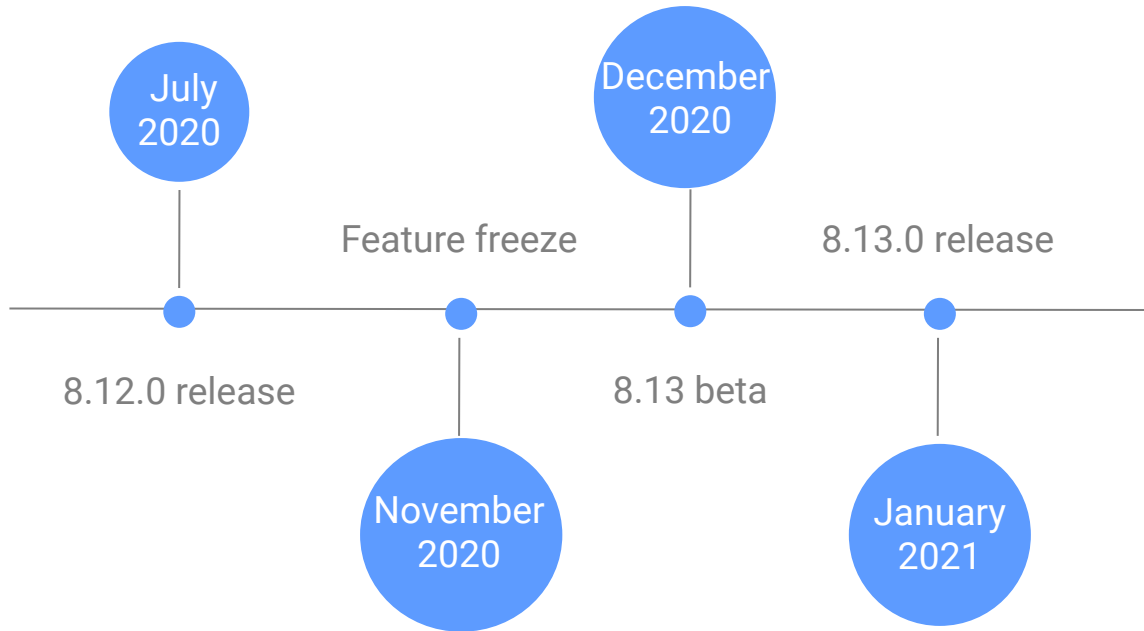
Online

January 19th, 2021

# OUTLINE

1. Coq 8.13, notes from the release manager
2. Coq 8.13, new features
3. Coq Future
4. Q & A

# Coq 8.13 Schedule



## Upgrading to 8.13 - Hint ...

- Hint ... raises a **warning** if no attribute is given
- **#[global]** Hint ... stops the warning, behaves as 8.12
- **#[export]** Hint ... the hint is active iff the enclosing module is Imported.

This is **not** backward compatible, but is preferable

- in 8.14 the warning will be fatal
- in 8.15 we hope to make **#[export]** the new default

[Migration HowTo \(by Pierre Marie Pédrot\)](#)

# Upgrading to 8.13 - Notations

Notation selection:

- more specific notation first
  - specific = matches a larger term
- order of Import matters, most recent wins

# The coq-native OPAM package

- lazy < vm\_compute < **native\_compute**
  - $\lambda$ -terms compiled using ocamlpt
  - Interesting to run reflexive tactics
- CEP#48 by Erik Martin Dorel & Pierre Roux
  - opt in: opam install coq-native
  - warning: requires more memory/time to compile .vo

# The [github.com/coq/platform](https://github.com/coq/platform) project

*A distribution of Coq packages*

Main objectives: easy, standard, tested

Output:

- scripts to setup/install on Win, OSX, Linux
- binary installers for Win, Linux, OSX (in progress)

[Coq Platform Charter by Michael Soegtrop](#)

# The release process

... 8.12 (coq) ... 8.13 (coq + platform) ... 8.14 (platform) ...

**user:** look at the platform scripts/installers

**library dev:** test against the platform

**platform package dev:** we will ask you “please tag”

**plugin dev:** put your plugin in Coq’s CI

[CEP#52 Release process for Coq 8.14 by Enrico Tassi](#)



## Coq 8.13 Features

<https://coq.inria.fr/refman/changes.html#version-8-13>

- Primitive persistent array type
- UIP for equality in SProp (with a caveat)
- Improvements of notations, implicit argument handling
- More consistent grammars in the reference manual, matching the implementation
- lia and zify enhancements to support boolean operators and the signed integers
- Fix an incompleteness bug in the treatment of cumulative inductive types

Active developer projects: Ltac 2.0, Coq-Elpi, SerAPI, Equations, JsCoq, ...

# In the Pipeline for Coq 8.14

- Change of case representation: more efficient and matching the user-level view. Updated meta-theory proof in MetaCoq
- Reworked document manager (easier integration with IDEs)
- Support for inductive-inductive types
- Primitive signed integers on top of the primitive unsigned ones
- Sized typing in the kernel (PR #12426)
- Deep “small-inversion” in pattern-matching compilation

# Coq Future

- Rewrite rules (T. Winterhalter, CEP PR#50)
- Eta-reduction and contravariant subtyping (H. Herbelin, M. Sozeau, CEP #47)
- Improved UI support (M. Dénès, E. Tassi, E J-G. Arias)
- Visual Ltac debugger (J. Fehrle, CEP PR #53)
- Website redesign (outsourced)

Longer term:

- Sort polymorphism (subsuming template-polymorphism):
  - Exceptional type theory
  - Setoid type theory
- Cubical type theory

# Development news

- Systematic use of PRs and CI infrastructure (now of critical size)
- Automation of PR review and merging process using coqbot by Théo Zimmermann
- coq-community project: <https://github.com/coq-community>
- Day-to-day communication: <https://coq.zulipchat.com>
- Discourse forum: <https://coq.discourse.group>

**Thank you!**

*inria*  
informatics mathematics