

GNU Radio Releases

- ▶ 3.8.0.0: 2019-08-09
- ▶ 3.7.14.0: 2020-02-15
- ▶ 3.8.1.0: 2020-04-08
- ▶ 3.8.2.0: 2020-08-20

- ▶ 3.9.0.0: 2021-01-17

Realizations:

- ▶ If you fit a 4th order polynomial, you can see that by GRCon'21, we'll have GNU Radio 4.3.8.6 (we're slightly behind on that schedule)
- ▶ Release cadence hasn't become stable
 - ▶ too many external, human factors
- ▶ Development is indeed steady; no real slowdown



What's new in GNU Radio 3.9?

cont.

Programming Languages

- ▶ Python \geq 3.6.5, **no** Python2
 - ▶ Greatly reduces OOT pitfalls
 - ▶ Reduces maintenance effort
 - ▶ Can use Py3 features
- ▶ C++14
 - ▶ Better smart pointer handling, lambdas, constexpr
 - ▶ Makes for potentially cleaner code

Language Binding

- ▶ SWIG \rightarrow pybind11
- ▶ Requires touching of OOT code :(
- ▶ Most bindings can be generated, but readable

Dependencies

- ▶ VOLK not build as submodule anymore
 - ▶ More compatible with many Linux distro packaging guidelines
 - ▶ Benefit: stable VOLK packaging
 - ▶ VOLK is a software project in its own right, treat it as such
- ▶ For C++14, need one of
 - ▶ GCC \geq 8.3
 - ▶ clang \geq 11
 - ▶ MSVC \geq 1910 (Visual Studio 2017 15.0)
- ▶ Boost, numpy, mako bump (modest)

Should I Be Switching to GNU Radio 3.9?

Well, yes! But.

Users

- ▶ Your Linux distro offers GNU Radio 3.8: that's great, stick with it. You can get 3.9 when you update your Linux distro in half a year.
- ▶ If it offers 3.7: Update it! (you can also use our PPA if it's Ubuntu and not ancient)

OOT Developers

- ▶ Add your currently supported versions to your Manifest:
`gr_supported_versions: v3.7,v3.8`
- ▶ `maint-3.X` branches make it clear which branch to build for which GNU Radio version
- ▶ `pybind11` is the main difference, we've got `bindtool`!

Software Distributors / People with legacy systems

- ▶ Not dropping support for 3.8 soon, and not touching 3.7
- ▶ Consider migrating to 3.9 for Py2 reasons

Keeping Release Series Alive

Problem:

- ▶ Development happens on `master` branch
- ▶ Problem: Backporting to three release series now.
- ▶ Solution: Be selective

Solution:

- ▶ Contributor opens PR, ideally notes whether backporting seems reasonable
- ▶ Core contributors tag accordingly:
 - ▶ `no tag`: Things that break API, make no sense on backport
 - ▶ `backport-3.9`: Bugfixes (all), Features where sensible
 - ▶ `backport-3.8`: Bugfixes (all that don't hurt), Features on request
 - ▶ `backport-3.7`: Probably won't apply (wrong version of everything). Extreme caution to not break existing system
- ▶ after backporting PR got merged, `ported-x.y` tag is added
- ▶ regular checks for PRs with `backport-`, but without `ported-` tag

