New Python versions in Fedora

- Before 3.9: New Python every 1.5 year: ~3 Fedora releases
- Now: New Python each year: ~2 Fedora releases
- Python schedule adapted to match Fedora:
  - new Python in odd releases
- Development of new Python version takes 1 year
- Fedora integrates soon, not late
  - upstream involvement
  - getting hands dirty
  - lots of breakages
  - Fedora is the driver of fixes for many upstreams
Fedora offers multiple Python versions

- Mostly for Python developers, our primary target audience
  - Python 2.7 (dead but not missing, RHEL 7 devs)
  - Python 3.6 (security only, RHEL 8 devs)
  - Python 3.7 (security only)
  - Python 3.8 (security only)
  - Python 3.9 (security only)
  - Python 3.10
  - Python 3.11 (development preview)
  - also PyPy, MicroPython
The “main” Python

- One Python version is more important than others
  - Fedora 29, 30 & 31: Python 3.7
  - Fedora 32: Python 3.8 (originally planned for 31)
  - Fedora 33 & 34: Python 3.9
  - Fedora 35 & 36: Python 3.10
  - Fedora 37 & 38: Python 3.11

- Stack of packages (thousands)
- Fedora runs on this Python (DNF, Anaconda, packaging tools...)
- Critical component
- /usr/bin/python3 (and /usr/bin/python)
- Many users only use this Python
How many Python 3 packages are in Fedora
Packages require a specific Python version

- Majority of Python packages require Python X.Y specifically
  - Python libraries installed to /usr/lib(64)/pythonX.Y/
  - Applications embedding Python linking to libpythonX.Y.so
  - The bytecode cache (.pyc files) is compiled for Python X.Y

- Upgrading “main” Python == rebuilding ~4-5 thousands packages
  - in proper order 😂
  - packages don’t always build 💔
  - new Python is not backwards compatible 🐍
    - 3.10 to 3.11 is a “major” upgrade (not semver)
October 2021: Python 3.11.0a1

- Package as new Python version for testing
- Fork the Python 3.10 package and adapt macros (variables)
- Rebase Fedora patches
- Remove cruft
  - old Provides/Obsoletes
  - patches that have been merged upstream
- Change proposal
  - usually the first one for that Fedora
- Package can be built --with main_python
- Since Fedora 33, there is no “python3“ source package
  - “python3“ is built from “python3.11“
### Build Monitor

Warning: This is a large project with many packages and/or builds - we can not hold all the requested data on one page (as we usually do), please use the pagination buttons below the table.

<table>
<thead>
<tr>
<th>Package</th>
<th>Fedora rawhide</th>
</tr>
</thead>
<tbody>
<tr>
<td>2ping</td>
<td>succeeded</td>
</tr>
<tr>
<td>389-ds-base</td>
<td>succeeded</td>
</tr>
<tr>
<td>abiword</td>
<td>succeeded</td>
</tr>
<tr>
<td>abrt</td>
<td>succeeded</td>
</tr>
<tr>
<td>abrt-server-info-page</td>
<td>succeeded</td>
</tr>
<tr>
<td>academic-admin</td>
<td>succeeded</td>
</tr>
<tr>
<td>accarciser</td>
<td>succeeded</td>
</tr>
<tr>
<td>adapt</td>
<td>succeeded</td>
</tr>
<tr>
<td>adb-enhanced</td>
<td>succeeded</td>
</tr>
<tr>
<td>afflib</td>
<td>succeeded</td>
</tr>
</tbody>
</table>
Copr repo: @python/python3.11

- “Bootstrap” the very initial package set 🎁 x86_64 only
- Rebuild everything in dependency order 🔨
- Triage hundreds of build failures:
  - (few) missing dependencies 🎁
  - Python 3.11 incompatibilities 🐍
  - Python 3.11 regressions 🐞
  - fails to build for unrelated reason 🥱🐞
    - sometimes in copr only 🥱🐞
- Mass rebuild ~4k packages with every new Python version
- Keep up with Rawhide
  - automation
Past: Bootstrapping the initial package set

GitHub: sclorg/rpm-list-builder ❤
- YAML with build order and RPM conditionals (%bconds)
- Initially created from Python 3.8 data, adapting (rewriting)
- Automated with human operator

~500 critical packages
- setuptools, pip, pytest, sphinx, cython
- rpm, dnf, anaconda, abrt, freeipa
- koji, bodhi, fedpkg, pungi
- boost, numpy, scipy, matplotlib, notebook
- clang, gdb
Past: Bootstrapping the initial package set

```python
python39:
  packages:
    - python3.9:
      macros:
        _with_bootstrap: 1
        _without_rpmwheels: 1
        _without_tests: 1
        _without_optimizations: 1
    - gdb # not blocking
    - python-setuptools:
      macros:
        _with_bootstrap: 1
        _without_tests: 1
    - python-wheel:
      macros:
        _with_bootstrap: 1
    - python-pip:
      macros:
        _without_tests: 1
        _without_doc: 1
    - python-setuptools:
      macros:
        _without_tests: 1
    - pyparsing:
      macros:
        _without_doc: 1
    - python3.9  # wait for gdb, skip test_io in copr
```
Past: Rebuilding everything else until it does

- Determine the full list of packages (dnf repoquery)
- Replace rpm-list-builder with copr automation
- Define packages in copr to build from Fedora sources (dist-git)
- Set up build automation (hooks)
  - every git push to Fedora Rawhide triggers a build
  - even Pull Requests trigger builds (isolated)
  - manual builds possible as well
- Build packages in waves
  - every successful build may be another failed build’s remedy
- Triaging and reporting failures
- Keep the package list up to date
Present: Resolve the dependency order

- `github.com/hroncok/whatdoibuild`
- for each package, resolve its buildroot in current Rawhide
- repoquery too high-level, uses dnf Python API
- reports package readiness wrt. deps
- detects (some) loops
- bconds data hardcoded, resolved from scratchbuilds
- reports most significant blockers
- chokes on unresolvable packages
  - dist-git vs source repo
  - the “random architecture” BuildRequires problem (e.g. scipy and pythran)
Present: Resolve the dependency order

- Resolving 31 requirements... to 369 installs.
- module-build-service: 98 packages / 95 components relevant to our problem
  - python-kombu
    ✓ python3-kombu
  - python-alembic
    ✓ python3-alembic
  - python-constantly
    ✓ python3-constantly
  - python-stevedore
    ✓ python3-stevedore
  ...
- m2crypto
  ✗ python3-m2crypto
  - python-requests-gssapi
    ✓ python3-requests-gssapi
  ...

The 50 most commonly needed components are:
- 4 m2crypto
- 2 python-spnego
- 1 ...

The 20 most commonly last-blocking components are:
- 4 m2crypto
- 2 python-spnego
- 1 ...
Triaging and reporting failures 🐜

- Triaging build logs with regexes 🦇
- Build failed:
  - automatically check root.log for dependency issues
  - automatically check build.log for common errors
  - investigate the rest
- Reporting 🐜s
  - unrelated fails to build from source (very often, policies to the rescue)
    - block ☢️F36FTBFS, later ☢️F37FTBFS
  - Python 3.11 incompatibility
    - figure out why, offer help
    - block ☢️PYTHON3.11
  - fixes not possible without actual maintainers, thanks!
  - build failures often caused by dependencies without %check
~1200 bugzillas were opened for Python 3.11
November 2021: Python 3.11.0a2
December 2021: Python 3.11.0a3
January 2022: Python 3.11.0a4

- Update the Fedora package (official and copr)
- Check if basic packages still build in Copr
  - partial rebootstrap needed for arched packages
  - copr problems
    - we don’t bump releases, need to delete old builds
- Mass rebuild everything
February 2022: Fedora 36 branching

- Finally, Rawhide is Fedora 37 🎉

- Up until now, some don’t consider Fedora 37 🐜s urgent
  - some still don’t 😞
March to May: Repetition, repetition

- New alphas
- New mass rebuilds
- New Python incompatibilities
- New Rawhide breakages
- New dependency issues
May 2022: Python 3.11.0b1: The first beta

- No new features beyond beta 1
- Figure out whether Fedora 37 is ready (see copr, bugzilla)
  - it wasn’t, some critical packages were still broken
- 2nd beta - start building in Fedora
  - announce, announce, announce!
  - koji side tag not to break everything
  - reproduce 🛠
    - enhanced whatdoibuild with a Koji builder
  - mass build everything
  - new failures for different arches
Merging the side tag

- Ideally, when everything is built
  - That won’t ever happen
- Too soon? Too late?
- Mind the mass rebuild
- Point of no return, but is it?
- The merge “breaks” whatdoibuild 🙃

- Reality:
  - ~10 days
  - ~3700 builds merged
  - ~500 packages failed to build, nothing “important”
  - compose blocked after merge by anaconda/dracut bug, fixed in 1 day
Fails To Install bugzillas and policy

- Many maintainers don’t consider Python 3.11 bugs priority
- Many maintainers don’t respond to Bugzilla
- Nonresponsible maintainer policy does not scale
  - Many take it as a personal attack
- **Fails To Install** policy to the rescue
  - targets packages, not people
  - allows orphaning and retirement ☠
  - existing tooling, updated
- Still fighting the “I'll set it to ASSIGNED but won’t fix it“ approach
July: Revert! Revert! 🔄 ► ◀️ ◀️

- Too many upstream blockers
- “If not fixed, we’ll delay by 2 months“
- Possible ABI incompatibility after Fedora 37 GA
- Revert immediately?
  - Rawhide would go 3.10 ➔ 3.11 ➔ 3.10 ➔ 3.11
  - Triples the work
- Revert after F37 branching?
  - Rawhide would stay at 3.11
  - Complicates pre-beta testing
- Blockers actually fixed, never mind
July: New betas, checking ABI compat.

- New betas (3-5) upgraded in copr first and check ABI compat.
- No new features, but possible reverts
  - ABI changes can cause segfaults
  - bytecode cache format (SELinux), happened with Python 3.8 and 3.11
- Mini mass rebuilds possibly needed (~600 packages)
  - arched packages only

- Reality
  - the ABI stayed the same after 3.11.0b3
  - bytecode magic number was updated in 3.11.0b4
Bytecode magic number has changed

- Number encoded in headers of bytecode cache files (.pyc)
- Unique for each Python version, stable after rc
- Breaking change in 4th beta
- F37 mass rebuild took care of most packages
- ~60 packages failed during mass rebuild
- tracked in bugzilla
August: Python 3.11.0rc1, F37 Beta Freeze

- Python rc versions are pretty much the same as final
- Plan: Ship Fedora 37 Beta with at least 3.11.0rc1
- 85 packages still fail to build and hence fail to install
- Discover Python related runtime problems during beta
September/October: Finalizing

- September: Python 3.11.0rc2 (the last rc)
- early October: Fedora 37 Final Freeze
  - all packages built / retired before the freeze

- Freeze Exceptions:
  - Python 3.11.0 final (might happen after the freeze)
  - obsolete removed packages to unblock upgrade paths

- Python 3.12.0a1 released, let’s start over
fedoralovespython.org