ActiveState

The Python End of Life Trap: **Avoiding Legacy Open Source Risks in Your Software Supply Chain**

Agenda

- State of Python 3.7 EOL
- Strategies for Upgrading to Supported Versions
- Another Route: Extended Support
- Cloudera Machine Learning Application

Introductions

Dana Crane

Product Marketing Mgr
ActiveState

Evan Cole

Sr. Solutions Engineer,
ActiveState

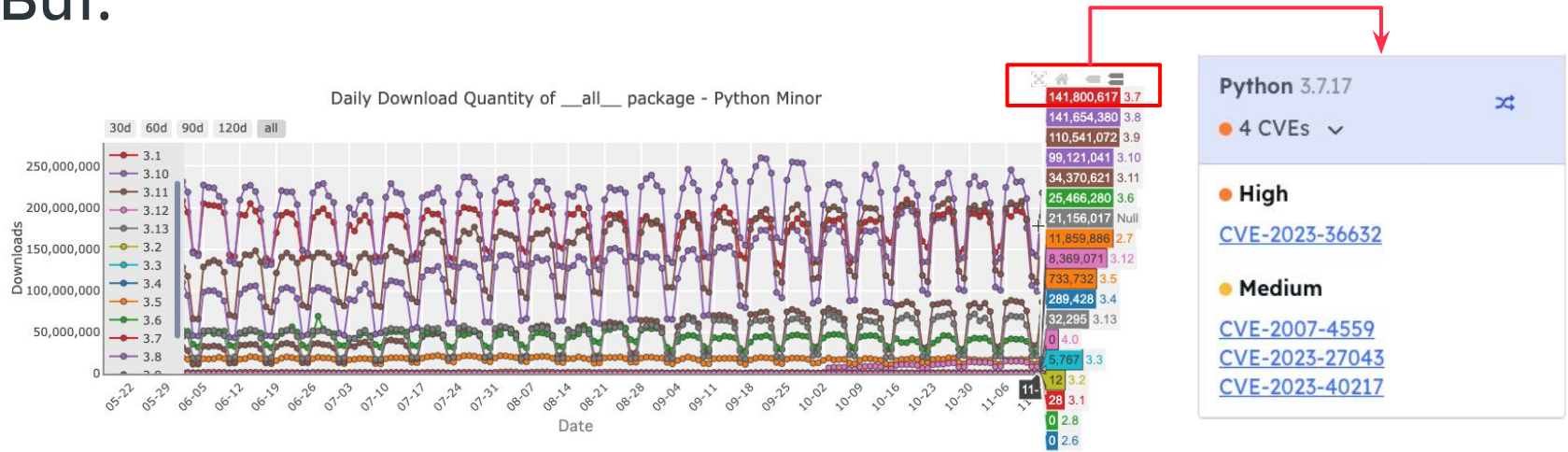
ActiveState

State of Python 3.7

Python 3.7

- Released 26 June 2018
- Security support ended 27 June 2023

But:



> 70% of the time, developers never update third-party libraries after including them in a codebase.

— Veracode State of Software Security

To Update or Not?

Typical Considerations:

3.7.x -> 3.7.y

- Break the build
- Dependency hell
- Tech debt

EOL Considerations:

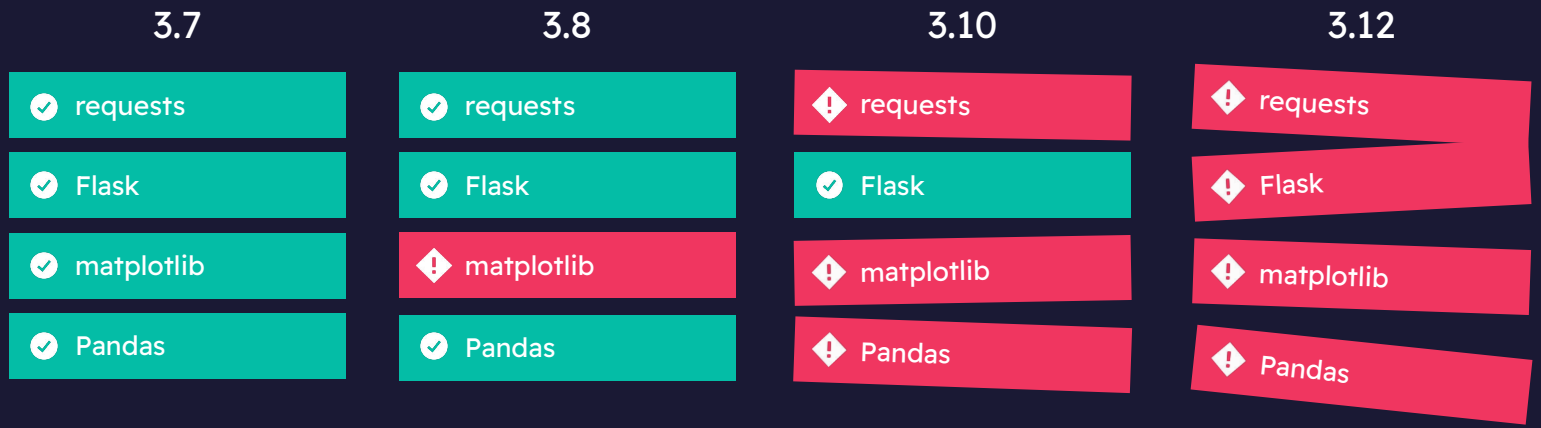
3.7 -> 3.8+

- Appetite for risk
- Patching the Python core
- Opportunity cost

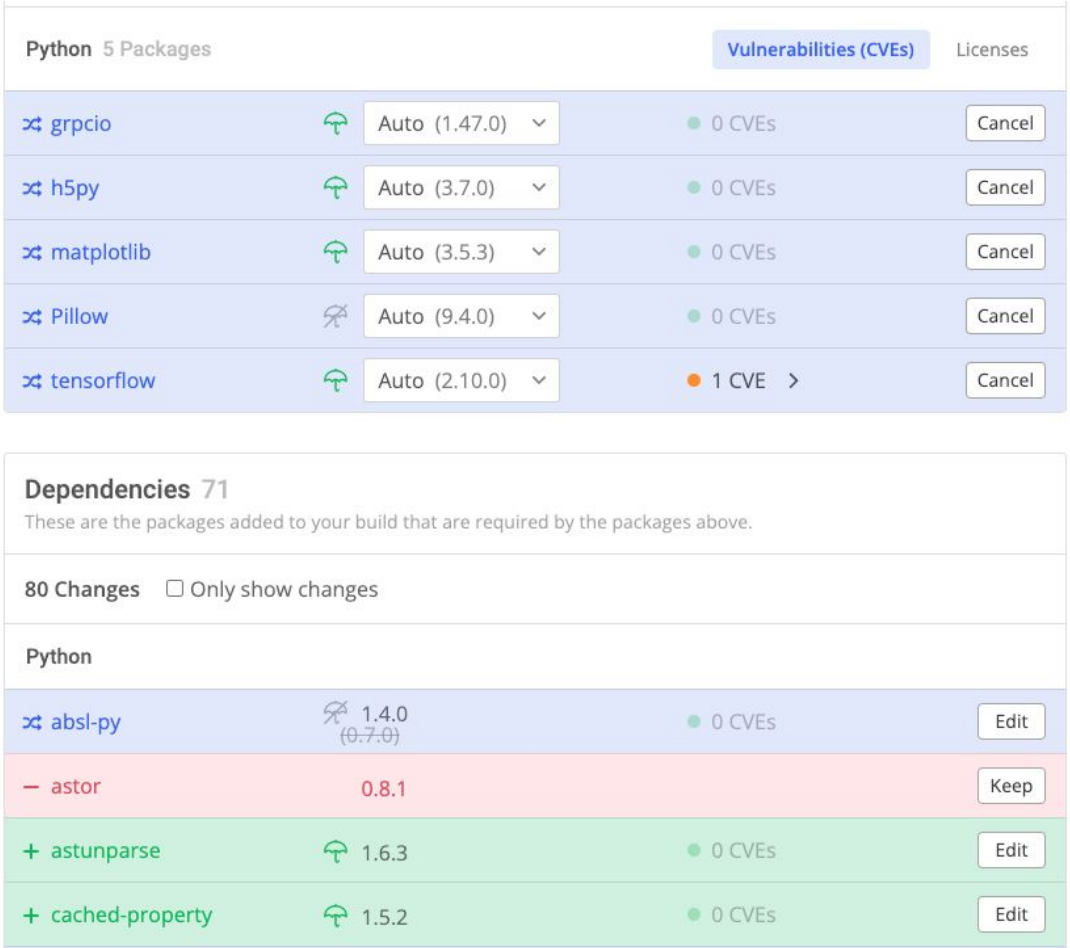
ActiveState

Strategies for Upgrading to Supported Versions

The further from Python 3.7 you upgrade, the more breaking changes are introduced from your dependencies...



...requiring a greater portion of your application to be rewritten.



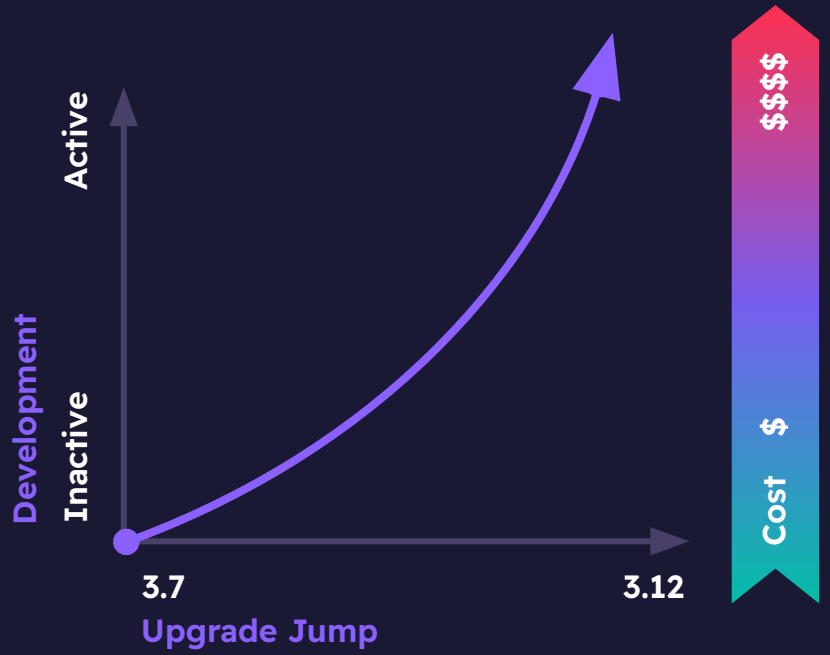
Python 3.7 -> 3.10

New versions

No longer required

New requirements

Minimizing Time, Complexity & Cost



Tips for minimizing costs

- Ensure adequate test coverage
- Setup CI for all versions
- Scale back or halt active development
- Reduce dependencies
- Preform incremental upgrades
  3.7 → 3.8 → 3.10 → 3.12
- **Use advanced environment management**

Identify Target Python

Generate List of Current Dependencies

Generate Runtimes for each minor Python Version

Establish repeatable development environment

CI/CD Integration

Team Training

Examine Code

Review Dependency usage and changes between versions

Dependency Upgrade

Address breaking changes

Repeat until target Python reached with working unit tests.

Poll:

How many tools/scripts/services/apps are still running on Python 3.7 in production and non-production?

- 0
- 1-2
- >2 but <10
- >10

ActiveState

Another Route: Extended Support

Upgrade Concerns

- Tight timelines
- Existing security concerns
- Critical static dependency on older software
- Internal capabilities
  - Opportunity cost

Lessons from Python 2.7

Solution domain is clear, but legacy codebases present unclear problem domains

- Upgrade timelines missed
- Upgrade difficulty underestimated
- $$$ expensive

ActiveState Extended Support

Adopt extended support versions of Python 3.7 from ActiveState

3.7.17.4

Upgrade retains full compatibility of existing 1st party applications.

Benefits

1. Cheaper than full migration
2. Faster than full migration upgrade
3. Removes risks of running EOL software
4. Requires no codebase expertise
5. Minimizes opportunity cost

Upstream versions are monitored for CVEs

Upstream CVEs are assessed for impact to legacy version

CVE fixes are backported from upstream versions

Indirect dependencies are updated as much as possible

Fixes are tested for effectiveness and backwards compatibility

Point released are delivered via the ActiveState platform

Changes can be tested using State Tool and rolled back quickly with ActiveState version control

How it works

Bespoke Support

We support more than the Python 3.7 interpreter. We tailor our converge to your open source package needs.

ActiveState

Cloudera Machine Learning Application

Legacy Software Holdup

Spark 2 Python 3.7 Dependency

Many Spark based data science workloads were written with Spark 2 relying on <=Python 3.7. Upgrading to Spark 3 requires a major rewrite of a lot of stable data science code.

ActiveState

Demo:
Bringing Python 3.7 to CML

ActiveState

Q&A

Next Steps

Schedule a consultation with our extended support team
www.activestate.com/solutions/contact-sales/

Learn more about our Python 3.7 EOL support
https://www.activestate.com/products/python/python-3-7/