The Automation of Conda-Forge

Christopher J. “CJ” Wright
What is Conda-Forge
4 A Packaging Community

- 2000+ community members maintaining packages, troubleshooting issues and providing help
A Packaging Community

- 2000+ community members maintaining packages, troubleshooting issues and providing help
- Many institutional partners, including Anaconda, Azure, NumFOCUS, and others
A Packaging Community

- 2000+ community members maintaining packages, troubleshooting issues and providing help
- Many institutional partners, including Anaconda, Azure, NumFOCUS, and others
- 24 Core developers
7 Source of Conda Packages

- conda install -c conda-forge numpy
- conda config --add channels conda-forge
Source of Conda Packages

- conda install -c conda-forge numpy
- conda config --add channels conda-forge
- conda install -c conda-forge numpy
- conda config --add channels conda-forge
Build Infrastructure System

- Builds provided across Windows, OS X and Linux on x86_64, PowerPC and ARM chipsets
- Builds are automated across 5 CI services
Build Infrastructure System

- Builds provided across Windows, OS X and Linux on x86_64, PowerPC and ARM chipsets
- Builds are automated across 5 CI services
Build Infrastructure System

- Builds provided across Windows, OS X and Linux on x86_64, PowerPC and ARM chipsets
- Builds are automated across 5 CI services

![Number of Builds on Azure-Pipelines](chart.png)
A Consistent Ecosystem
A Consistent Ecosystem

- All packages are mutually compatible
- Packages are built in the open, with easy access to logs
- Most interactions are run through GitHub
Automation
Automation

- At our scale automation doesn’t just make things *easier*, it makes things *possible*

- Our Automation focuses on 3 major areas:
  1. Building packages
  2. Keeping everything up-to-date and consistent
  3. Administration and reducing maintenance burden
Building Packages

- To create conda packages you need all systems to support all users
- CF uses CI infrastructure to perform all package building, enabling us to:
  1. Provide logs of the package builds for transparency
  2. Build the packages for many more platforms and architectures
Building Packages

- To create conda packages you need all systems to support all users
- CF uses CI infrastructure to perform all package building, enabling us to:
  1. Provide logs of the package builds for transparency
  2. Build the packages for many more platforms and architectures

![Number of packages by architecture](image)
Building Packages

- To create conda packages you need all systems to support all users
- CF uses CI infrastructure to perform all package building, enabling us to:
  1. Provide logs of the package builds for transparency
  2. Build the packages for many more platforms and architectures
  3. Provide high throughput package building
  4. Switch between compute providers

Number of packages by architecture

![Graph showing the number of packages by architecture over time.](image-url)
CI templating

- Conda-Forge uses CI services for almost everything, so we’ve gotten good at templating CI configs.
- These templates can be used outside Conda-Forge, enabling best in class CI management.

- DEMO!
Everyone needs fixes, security patches, and features, so upstream code is constantly changing. These updates are provided by the regro-cf-autotick-bot (aka “the bot”).
Everyone needs fixes, security patches, and features, so upstream code is constantly changing. These updates are provided by the regro-cf-autotick-bot (aka “the bot”).
Everyone needs fixes, security patches, and features, so upstream code is constantly changing.

These updates are provided by the regro-cf-autotick-bot (aka “the bot”).
Making Things Just Work

- Many times new features require changes to a piece of software’s Application Binary Interface (ABI)
- When handled incorrectly this causes all sorts of inscrutable errors
Many times new features require changes to a piece of software’s Application Binary Interface (ABI).

When handled incorrectly this causes all sorts of inscrutable errors.
Many times new features require changes to a piece of software’s Application Binary Interface (ABI)
When handled incorrectly this causes all sorts of inscrutable errors
To maintain compatibility all downstream packages need to be recompiled
Many times new features require changes to a piece of software’s Application Binary Interface (ABI)
When handled incorrectly this causes all sorts of inscrutable errors
To maintain compatibility all downstream packages need to be recompiled
IANAL but the Bot is (almost)

- The bot even provides clean up of the package metadata
- Providing:
  - The latest python install commands
  - Yaml and Jinja2 formatting fixes
  - License files for packages that need them
IANAL but the Bot is (almost)

- The bot even provides clean up of the package metadata
- Providing:
  - The latest python install commands
  - Yaml and Jinja2 formatting fixes
  - License files for packages that need them
Keeping everything running

- CF wouldn’t be possible without admin bots
- Adding maintainers
- Updating CI scripts
- Uploading packages
The Automerge bot is a new addition to the admin bot team
Merges PRs once all builds pass and there is no lint
The Automerge bot is a new addition to the admin bot team
Merges PRs once all builds pass and there is no lint
The Automerge bot is a new addition to the admin bot team
Merges PRs once all builds pass and there is no lint
Future Work

- Expanding Automerge implementation
- Providing dependency update hints
- Creating wheels
A Call for Contributors (and Discussion)

- If you are interested in becoming a maintainer or working on infrastructure
- If you feel your ecosystem is underserved and conda-forge can help
- If you are having problems with packages
- Please let us know!
Thank you!

CONDA-FORGE

A community-led collection of recipes, build infrastructure and distributions for the conda package manager.