Code Repositories and Packages

Brigitta Sipőcz

DIRAC Fellow, University of Washington

@bsipocz @AstroBrigi

What is a repository

- A place to keep your work

- A place to share your work and collaborate with others

- Good things to have:
    - Version controll
    - Regular backups

What is

- A place t

- A place t
thers

- Good thi

"FINAL".doc
FINAL.doc!
FINAL_rev.2.doc
FINAL_rev.6.COMMENTS.doc
FINAL_rev.8.comments5. CORRECTIONS.doc
track changes
FINAL_rev.18.comments7. corrections9.MORE.30.doc
FINAL_rev.22.comments49. corrections.10.#@$%WHYDID ICOMETOGRADSCHOOL????.doc
JORGE CHAM © 2012
WWW.PHDCOMICS.COM

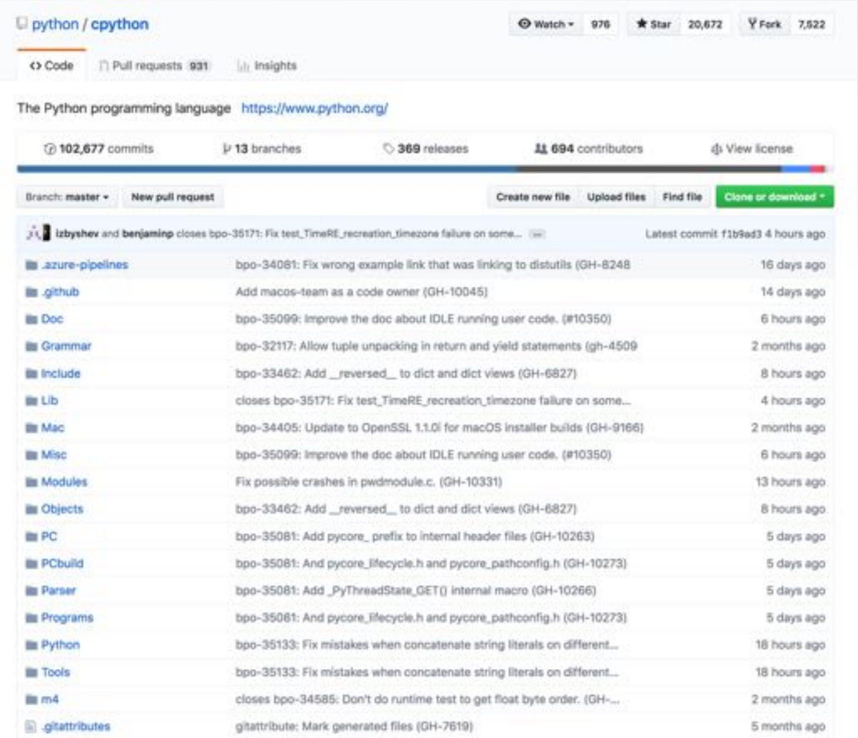
Hosting services for repositories

Atlassian
Bitbucket

SOURCEFORGE

Repositories can be...

- From a pile of short scripts to complex libraries



The Python programming language https://www.python.org/

102,677 commits 13 branches 369 releases 694 contributors View license

izbyshev and **benjaminp** closes bpo-35171: Fix test_TimeRE_recreation_timezone failure on some... Latest commit f1b9ad3 4 hours ago

.azure-pipelines	bpo-34081: Fix wrong example link that was linking to distutils (GH-8248	16 days ago
.github	Add macos-team as a code owner (GH-10045)	14 days ago
Doc	bpo-35099: Improve the doc about IDLE running user code. (#10350)	6 hours ago
Grammar	bpo-32117: Allow tuple unpacking in return and yield statements (gh-4509	2 months ago
Include	bpo-33462: Add __reversed__ to dict and dict views (GH-6827)	8 hours ago
Lib	closes bpo-35171: Fix test_TimeRE_recreation_timezone failure on some...	4 hours ago
Mac	bpo-34405: Update to OpenSSL 1.1.0i for macOS installer builds (GH-9166)	2 months ago
Misc	bpo-35099: Improve the doc about IDLE running user code. (#10350)	6 hours ago
Modules	Fix possible crashes in pwdmodule.c. (GH-10331)	13 hours ago
Objects	bpo-33462: Add __reversed__ to dict and dict views (GH-6827)	8 hours ago
PC	bpo-35081: Add pycore_ prefix to internal header files (GH-10263)	5 days ago
PCbuild	bpo-35081: And pycore_lifecycle.h and pycore_pathconfig.h (GH-10273)	5 days ago
Parser	bpo-35081: Add _PyThreadState_GET() internal macro (GH-10266)	5 days ago
Programs	bpo-35081: And pycore_lifecycle.h and pycore_pathconfig.h (GH-10273)	5 days ago
Python	bpo-35133: Fix mistakes when concatenate string literals on different...	18 hours ago
Tools	bpo-35133: Fix mistakes when concatenate string literals on different...	18 hours ago
m4	closes bpo-34585: Don't do runtime test to get float byte order. (GH-...	2 months ago
.gitattributes	gitattribute: Mark generated files (GH-7619)	5 months ago

Repositories can be...

- From a pile of short scripts to complex libraries

- Publicly shared or private (T&Cs vary)
    - Educational / not-for-profit options exists

From repository to package

- Make it easily installable by others (makefiles, upload to indexing services e.g. PyPI, conda-forge, etc…)

- Good things to have:

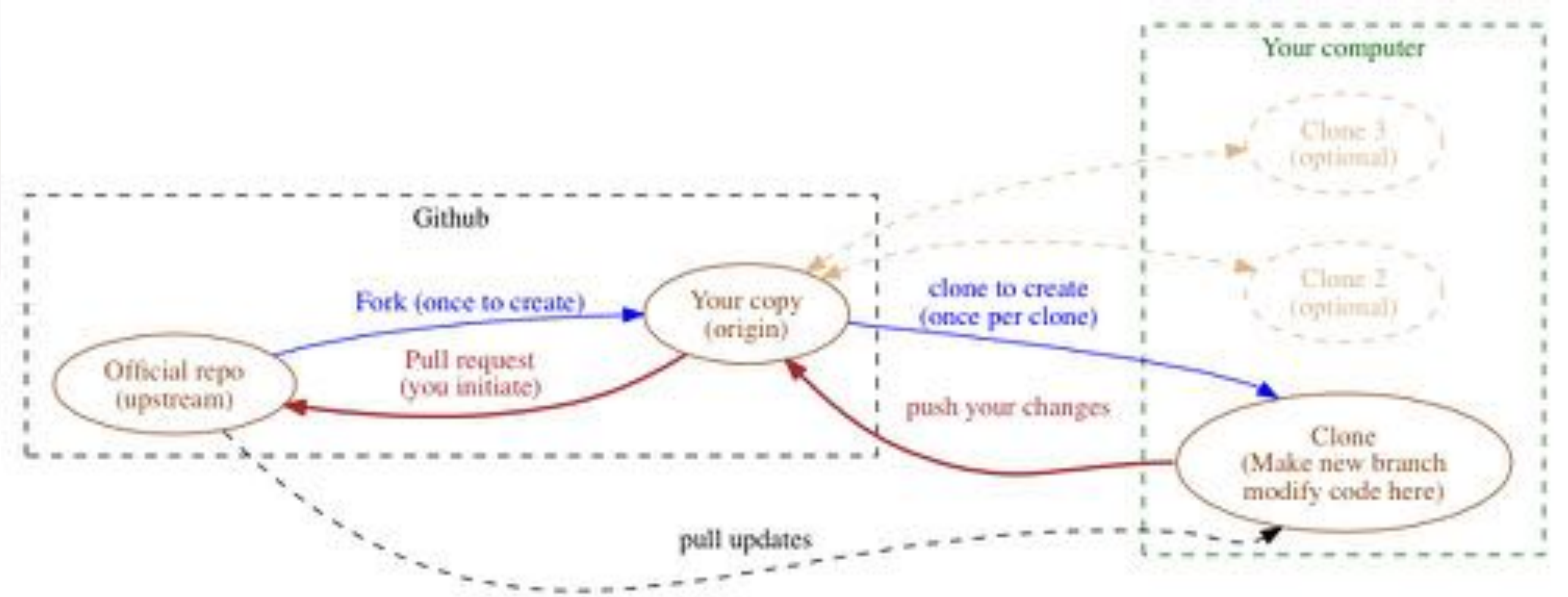
  - Modularity, and well structured overall layout

  - Sensible versioning

The case of Astropy

The Astropy Project is a **community effort** to develop a common *core package* for Astronomy in Python and foster an **ecosystem of interoperable** packages.

GitHub workflow

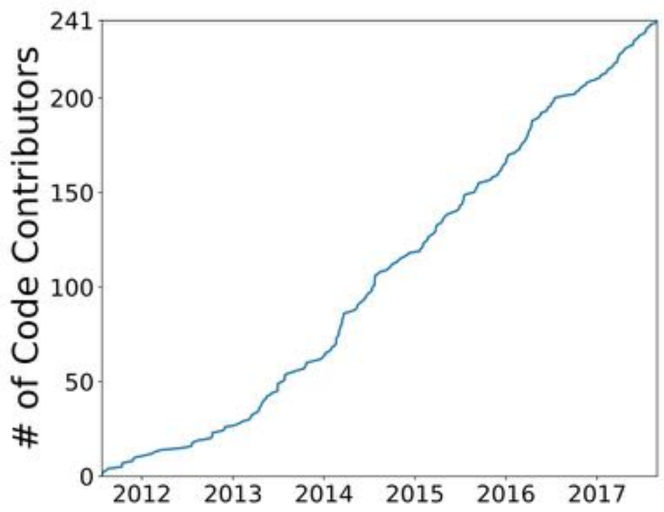


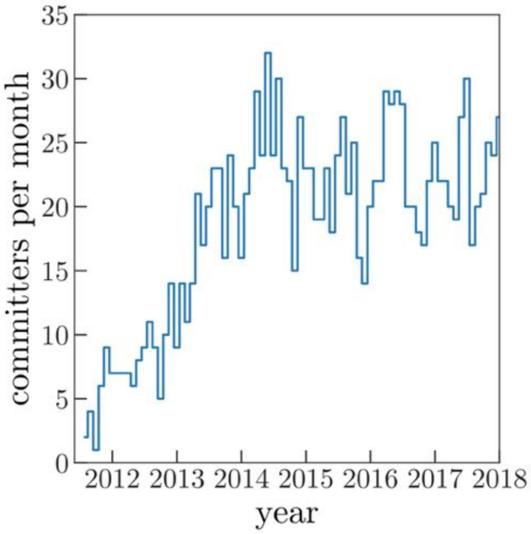
By Matt Craig

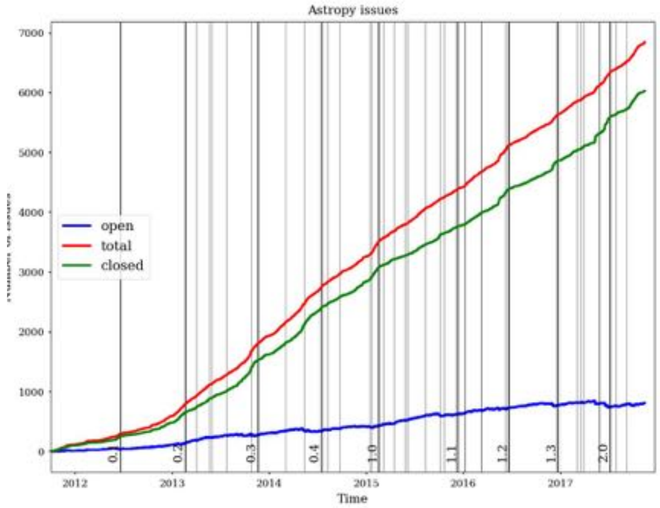
- Learn and practice your branches:
  https://learngitbranching.js.org/

Case study of Astropy

- Key to success: open development
- Community consensus for adding new features
- Well **tested**, **documented**, and **stable** code

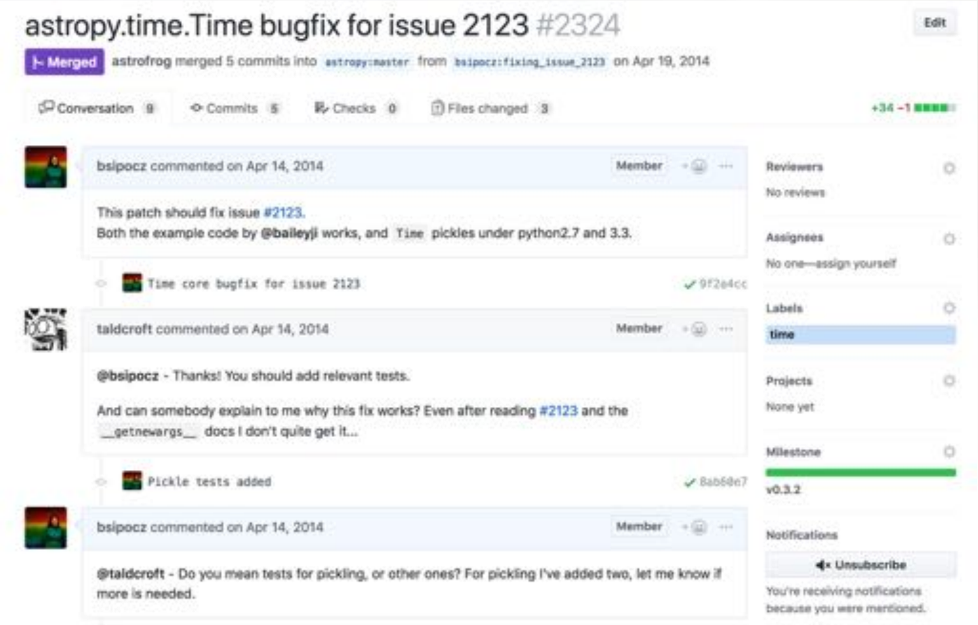






Case study of Astropy

- Wide variety of contributions, from one liners...



astropy.time.Time bugfix for issue 2123 #2324

Merged astrofrog merged 5 commits into `astropy:master` from `bsipocz:fixing_issue_2123` on Apr 19, 2014

bsipocz commented on Apr 14, 2014 — Member

This patch should fix issue #2123.
Both the example code by **@baileyji** works, and `Time` pickles under python2.7 and 3.3.

`Time core bugfix for issue 2123` ✓ 9f2e4cc

taldcroft commented on Apr 14, 2014 — Member

@bsipocz - Thanks! You should add relevant tests.

And can somebody explain to me why this fix works? Even after reading #2123 and the `__getnewargs__` docs I don't quite get it...

`Pickle tests added` ✓ 8ab68e7

bsipocz commented on Apr 14, 2014 — Member

@taldcroft - Do you mean tests for pickling, or other ones? For pickling I've added two, let me know if more is needed.

Reviewers: No reviews

Assignees: No one—assign yourself

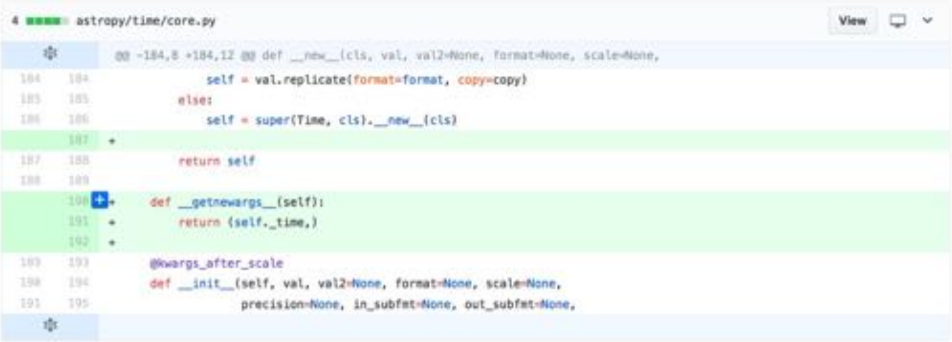
Labels: time

Projects: None yet

Milestone: v0.3.2

Notifications

You're receiving notifications because you were mentioned.

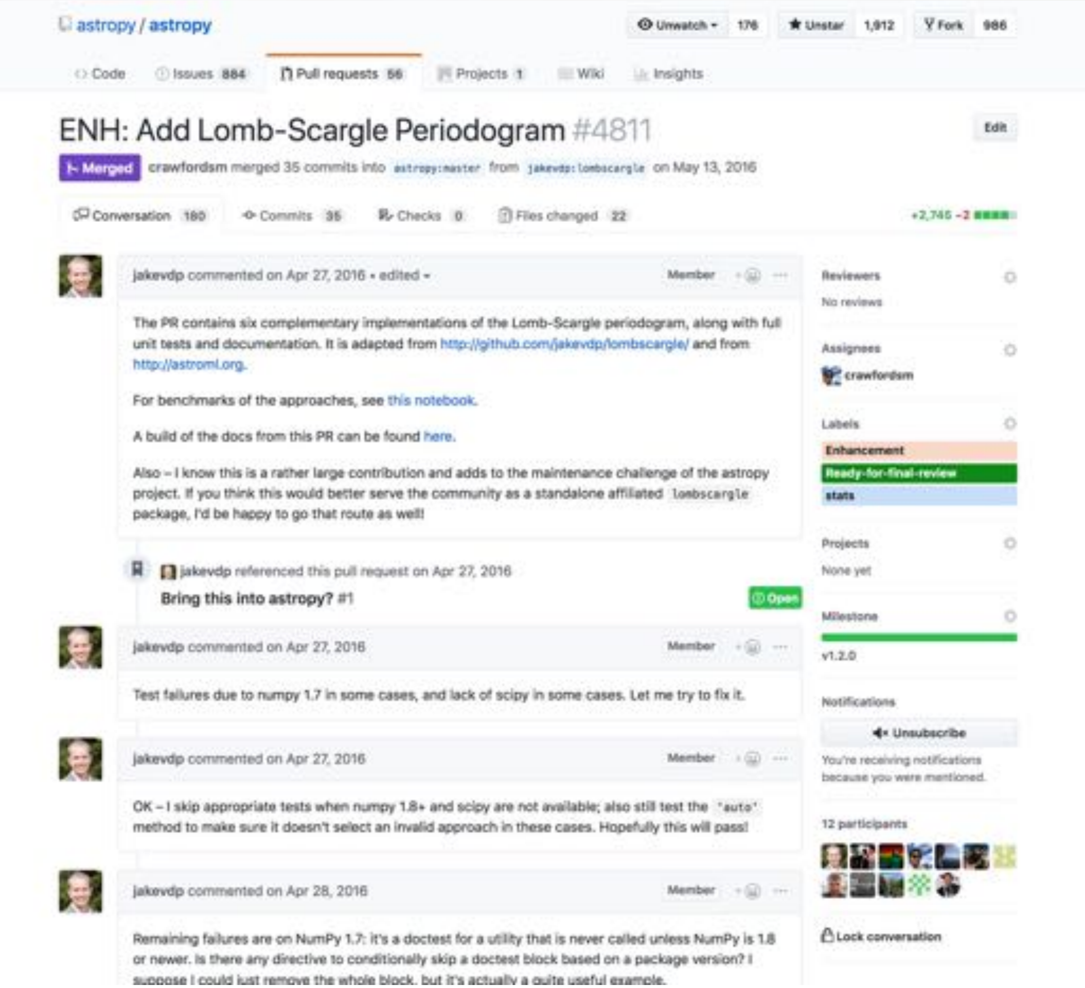


```
astropy/time/core.py
@@ -184,8 +184,12 @@ def __new__(cls, val, val2=None, format=None, scale=None,
                 self = val.replicate(format=format, copy=copy)
             else:
                 self = super(Time, cls).__new__(cls)
+
             return self

+        def __getnewargs__(self):
+            return (self._time,)
+
         @kwargs_after_scale
         def __init__(self, val, val2=None, format=None, scale=None,
                      precision=None, in_subfmt=None, out_subfmt=None,
```

Case study of Astropy

- ...to functionality from established libraries



ENH: Add Lomb-Scargle Periodogram #4811

Merged crawfordsm merged 35 commits into `astropy:master` from `jakevdp:lombscargle` on May 13, 2016

jakevdp commented on Apr 27, 2016 • edited

The PR contains six complementary implementations of the Lomb-Scargle periodogram, along with full unit tests and documentation. It is adapted from http://github.com/jakevdp/lombscargle/ and from http://astroml.org.

For benchmarks of the approaches, see this notebook.

A build of the docs from this PR can be found here.

Also – I know this is a rather large contribution and adds to the maintenance challenge of the astropy project. If you think this would better serve the community as a standalone affiliated `lombscargle` package, I'd be happy to go that route as well!

jakevdp referenced this pull request on Apr 27, 2016

Bring this into astropy? #1

jakevdp commented on Apr 27, 2016

Test failures due to numpy 1.7 in some cases, and lack of scipy in some cases. Let me try to fix it.

jakevdp commented on Apr 27, 2016

OK – I skip appropriate tests when numpy 1.8+ and scipy are not available; also still test the `'auto'` method to make sure it doesn't select an invalid approach in these cases. Hopefully this will pass!

jakevdp commented on Apr 28, 2016

Remaining failures are on NumPy 1.7: it's a doctest for a utility that is never called unless NumPy is 1.8 or newer. Is there any directive to conditionally skip a doctest block based on a package version? I suppose I could just remove the whole block, but it's actually a quite useful example.

Reviewers
No reviews

Assignees
crawfordsm

Labels
Enhancement
Ready-for-final-review
stats

Projects
None yet

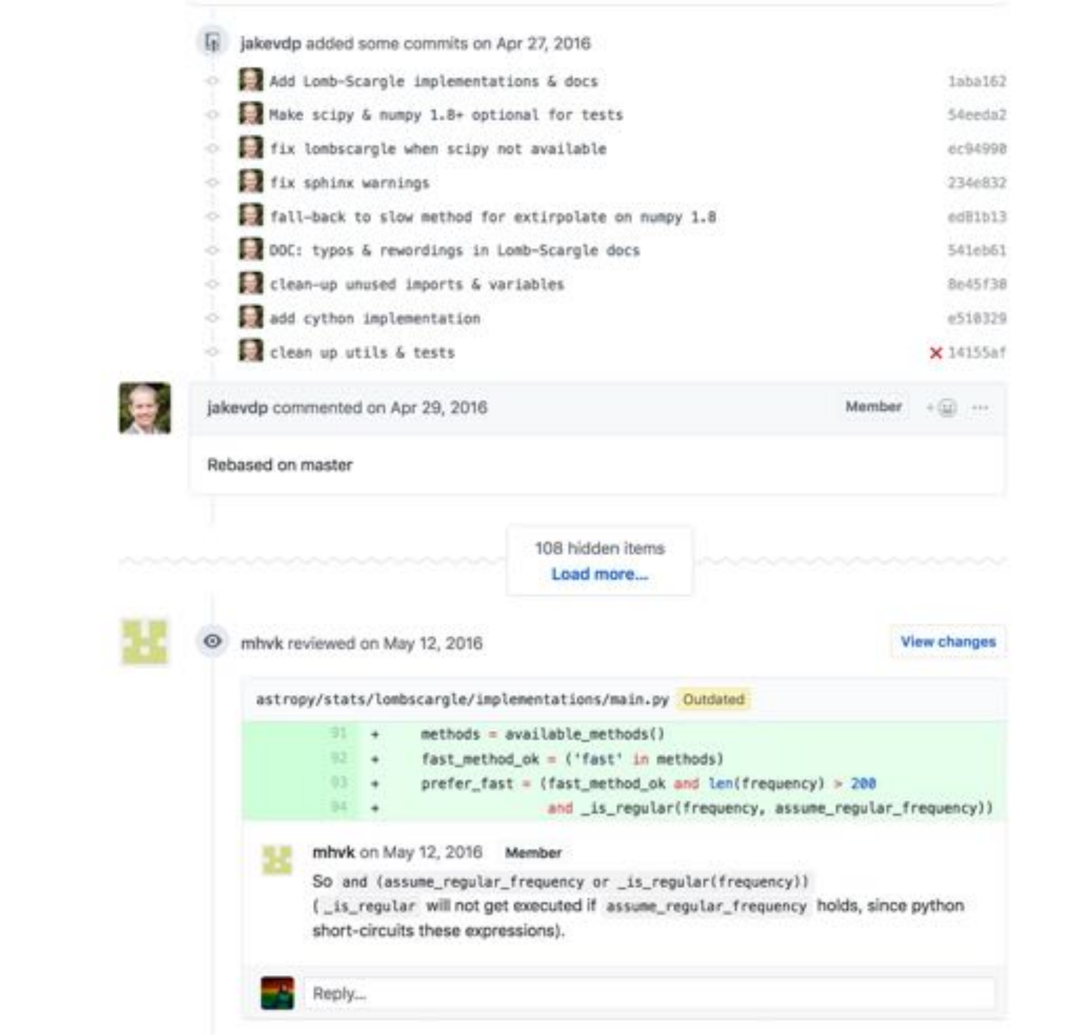
Milestone
v1.2.0

Notifications
You're receiving notifications because you were mentioned.

12 participants

Case study of Astropy

- Code review and cycles of iterations on details



jakevdp added some commits on Apr 27, 2016

```
Add Lomb-Scargle implementations & docs                        1aba162
Make scipy & numpy 1.8+ optional for tests                     54eeda2
fix lombscargle when scipy not available                       ec94990
fix sphinx warnings                                            234e832
fall-back to slow method for extirpolate on numpy 1.8          ed81b13
DOC: typos & rewordings in Lomb-Scargle docs                   541eb61
clean-up unused imports & variables                            8e45f38
add cython implementation                                      e510329
clean up utils & tests                                       ✕ 14155af
```

jakevdp commented on Apr 29, 2016 — Member

Rebased on master

108 hidden items

Load more...

mhvk reviewed on May 12, 2016

astropy/stats/lombscargle/implementations/main.py — Outdated

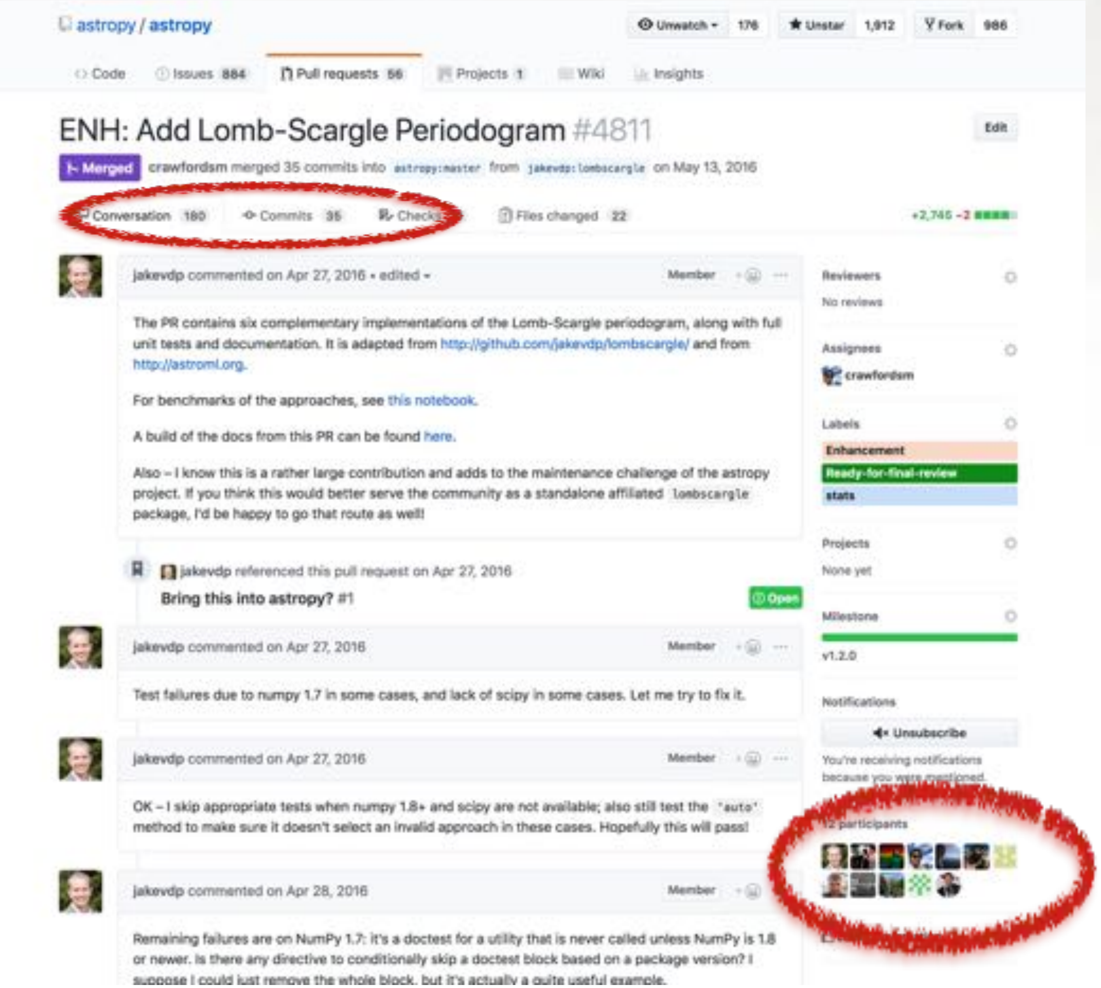
```
+     methods = available_methods()
+     fast_method_ok = ('fast' in methods)
+     prefer_fast = (fast_method_ok and len(frequency) > 200
+                    and _is_regular(frequency, assume_regular_frequency))
```

mhvk on May 12, 2016 — Member

So `and (assume_regular_frequency or _is_regular(frequency))` (`_is_regular` will not get executed if `assume_regular_frequency` holds, since python short-circuits these expressions).

Case study of Astropy

- Code review and cycles of iterations on details



ENH: Add Lomb-Scargle Periodogram #4811

crawfordsm merged 35 commits into `astropy:master` from `jakevdp:lombscargle` on May 13, 2016

jakevdp commented on Apr 27, 2016 • edited

The PR contains six complementary implementations of the Lomb-Scargle periodogram, along with full unit tests and documentation. It is adapted from http://github.com/jakevdp/lombscargle/ and from http://astroml.org.

For benchmarks of the approaches, see this notebook.

A build of the docs from this PR can be found here.

Also – I know this is a rather large contribution and adds to the maintenance challenge of the astropy project. If you think this would better serve the community as a standalone affiliated `lombscargle` package, I'd be happy to go that route as well!

jakevdp referenced this pull request on Apr 27, 2016

Bring this into astropy? #1

jakevdp commented on Apr 27, 2016

Test failures due to numpy 1.7 in some cases, and lack of scipy in some cases. Let me try to fix it.

jakevdp commented on Apr 27, 2016

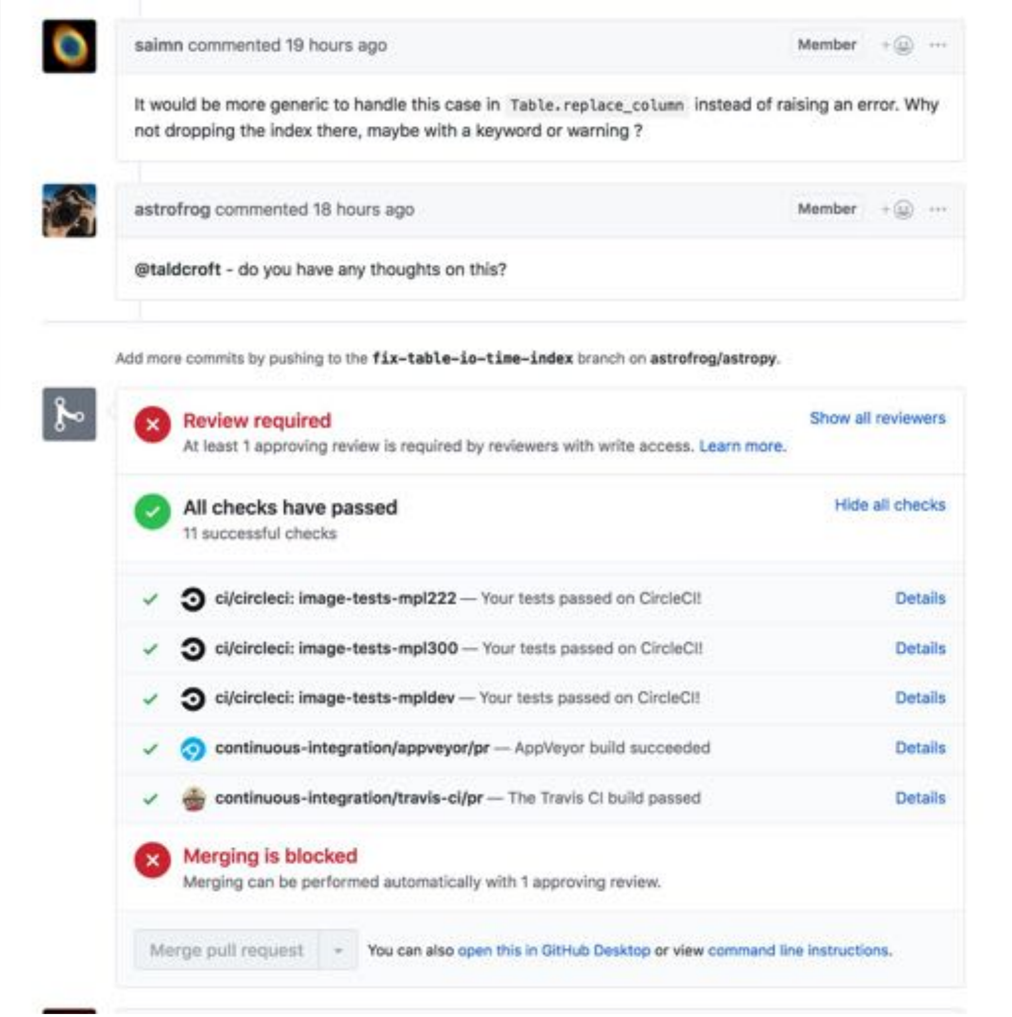
OK – I skip appropriate tests when numpy 1.8+ and scipy are not available; also still test the `'auto'` method to make sure it doesn't select an invalid approach in these cases. Hopefully this will pass!

jakevdp commented on Apr 28, 2016

Remaining failures are on NumPy 1.7: it's a doctest for a utility that is never called unless NumPy is 1.8 or newer. Is there any directive to conditionally skip a doctest block based on a package version? I suppose I could just remove the whole block, but it's actually a quite useful example.

Case study of Astropy

- Test every commit, require review



saimn commented 19 hours ago · Member

It would be more generic to handle this case in `Table.replace_column` instead of raising an error. Why not dropping the index there, maybe with a keyword or warning ?

astrofrog commented 18 hours ago · Member

@taldcroft - do you have any thoughts on this?

Add more commits by pushing to the **fix-table-io-time-index** branch on **astrofrog/astropy**.

Review required — Show all reviewers

At least 1 approving review is required by reviewers with write access. Learn more.

All checks have passed — Hide all checks

11 successful checks

- ✓ **ci/circleci: image-tests-mpl222** — Your tests passed on CircleCI! — Details
- ✓ **ci/circleci: image-tests-mpl300** — Your tests passed on CircleCI! — Details
- ✓ **ci/circleci: image-tests-mpldev** — Your tests passed on CircleCI! — Details
- ✓ **continuous-integration/appveyor/pr** — AppVeyor build succeeded — Details
- ✓ **continuous-integration/travis-ci/pr** — The Travis CI build passed — Details

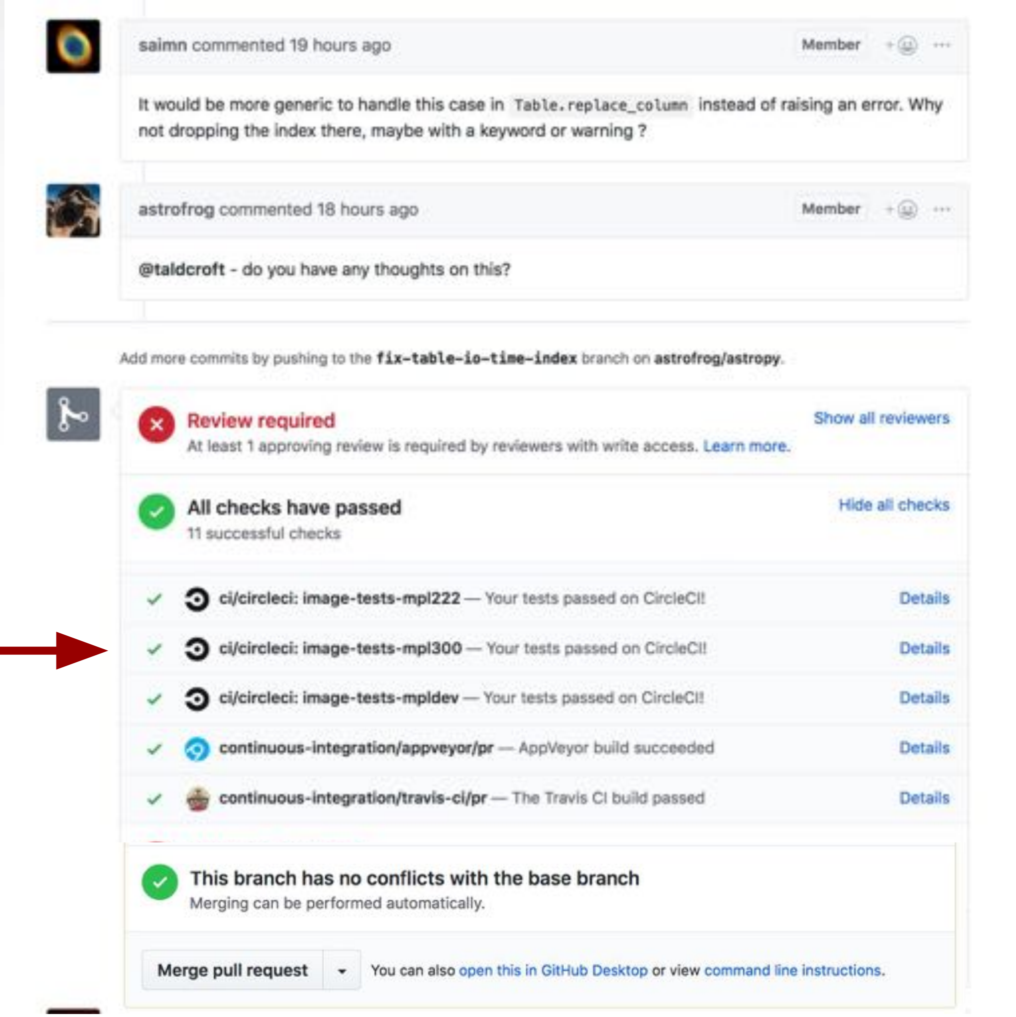
Merging is blocked

Merging can be performed automatically with 1 approving review.

Merge pull request ▾ You can also open this in GitHub Desktop or view command line instructions.

Case study of Astropy

- Test every commit, require review



saimn commented 19 hours ago

It would be more generic to handle this case in `Table.replace_column` instead of raising an error. Why not dropping the index there, maybe with a keyword or warning ?

astrofrog commented 18 hours ago

@taldcroft - do you have any thoughts on this?

Add more commits by pushing to the **`fix-table-io-time-index`** branch on **astrofrog/astropy**.

Review required

At least 1 approving review is required by reviewers with write access. Learn more.

All checks have passed

11 successful checks

✓ **ci/circleci: image-tests-mpl222** — Your tests passed on CircleCI! Details

✓ **ci/circleci: image-tests-mpl300** — Your tests passed on CircleCI! Details

✓ **ci/circleci: image-tests-mpldev** — Your tests passed on CircleCI! Details

✓ **continuous-integration/appveyor/pr** — AppVeyor build succeeded Details

✓ **continuous-integration/travis-ci/pr** — The Travis CI build passed Details

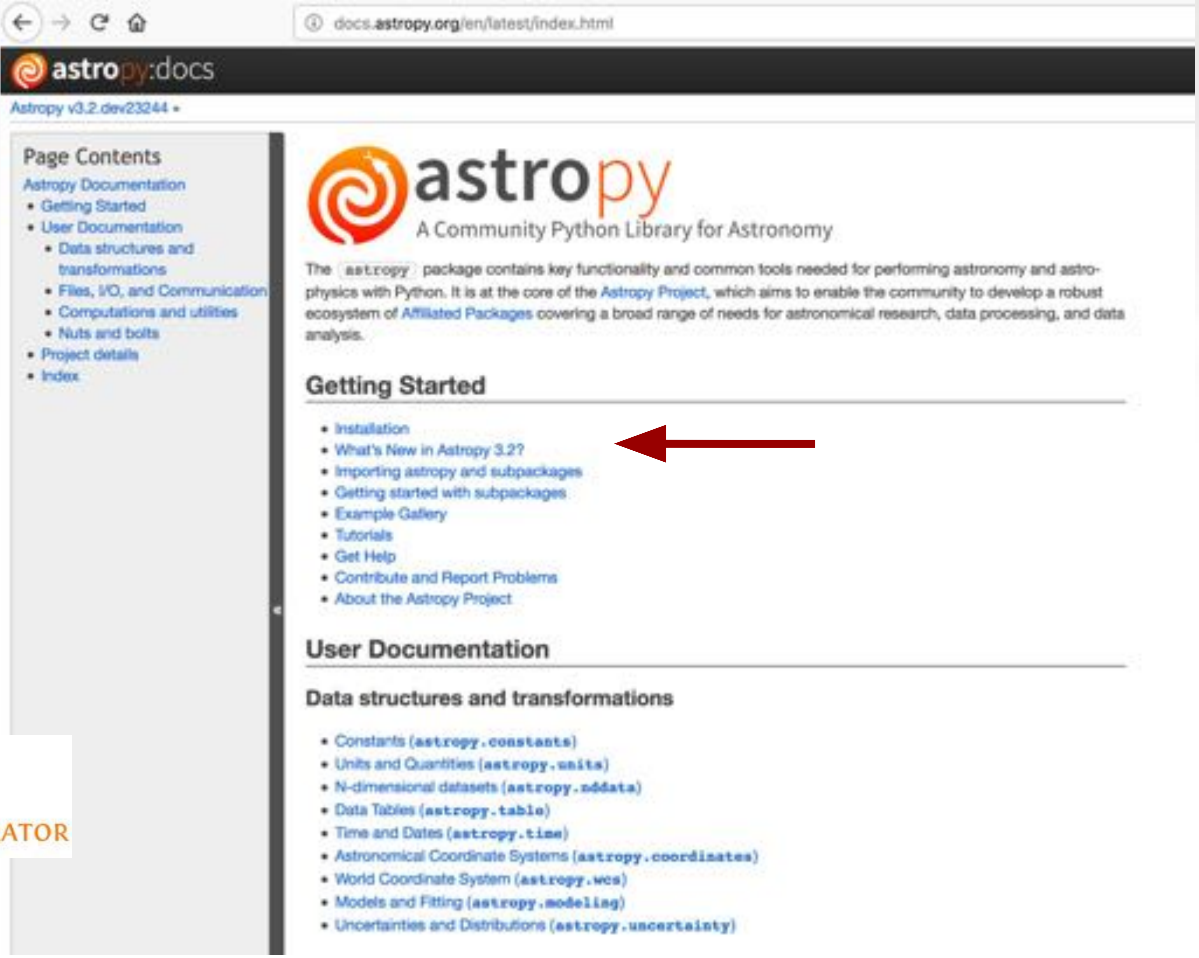
This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

Case study of Astropy

- Write documentation along with code rather than at release time



Page Contents

- Astropy Documentation
  - Getting Started
  - User Documentation
    - Data structures and transformations
    - Files, I/O, and Communication
    - Computations and utilities
    - Nuts and bolts
  - Project details
  - Index

astropy

A Community Python Library for Astronomy

The `astropy` package contains key functionality and common tools needed for performing astronomy and astrophysics with Python. It is at the core of the Astropy Project, which aims to enable the community to develop a robust ecosystem of Affiliated Packages covering a broad range of needs for astronomical research, data processing, and data analysis.

Getting Started

- Installation
- What's New in Astropy 3.2?
- Importing astropy and subpackages
- Getting started with subpackages
- Example Gallery
- Tutorials
- Get Help
- Contribute and Report Problems
- About the Astropy Project

User Documentation

Data structures and transformations

- Constants (`astropy.constants`)
- Units and Quantities (`astropy.units`)
- N-dimensional datasets (`astropy.nddata`)
- Data Tables (`astropy.table`)
- Time and Dates (`astropy.time`)
- Astronomical Coordinate Systems (`astropy.coordinates`)
- World Coordinate System (`astropy.wcs`)
- Models and Fitting (`astropy.modeling`)
- Uncertainties and Distributions (`astropy.uncertainty`)

SPHINX

PYTHON DOCUMENTATION GENERATOR

Python packaging jargon

- **Package:** the outer big one, e.g. `numpy, astropy, sklearn`

  E.g. outermost directory with a __init__.py file
- **Module**: a part of the namespace, can be from a single file

  E.g. `fourier` for the file `fourier.py`
- **Subpackage:** package within a package, e.g `astropy.io.fits`

- **Repository:** everything that is under version control

- **Submodule**: a git repository embedded in another one

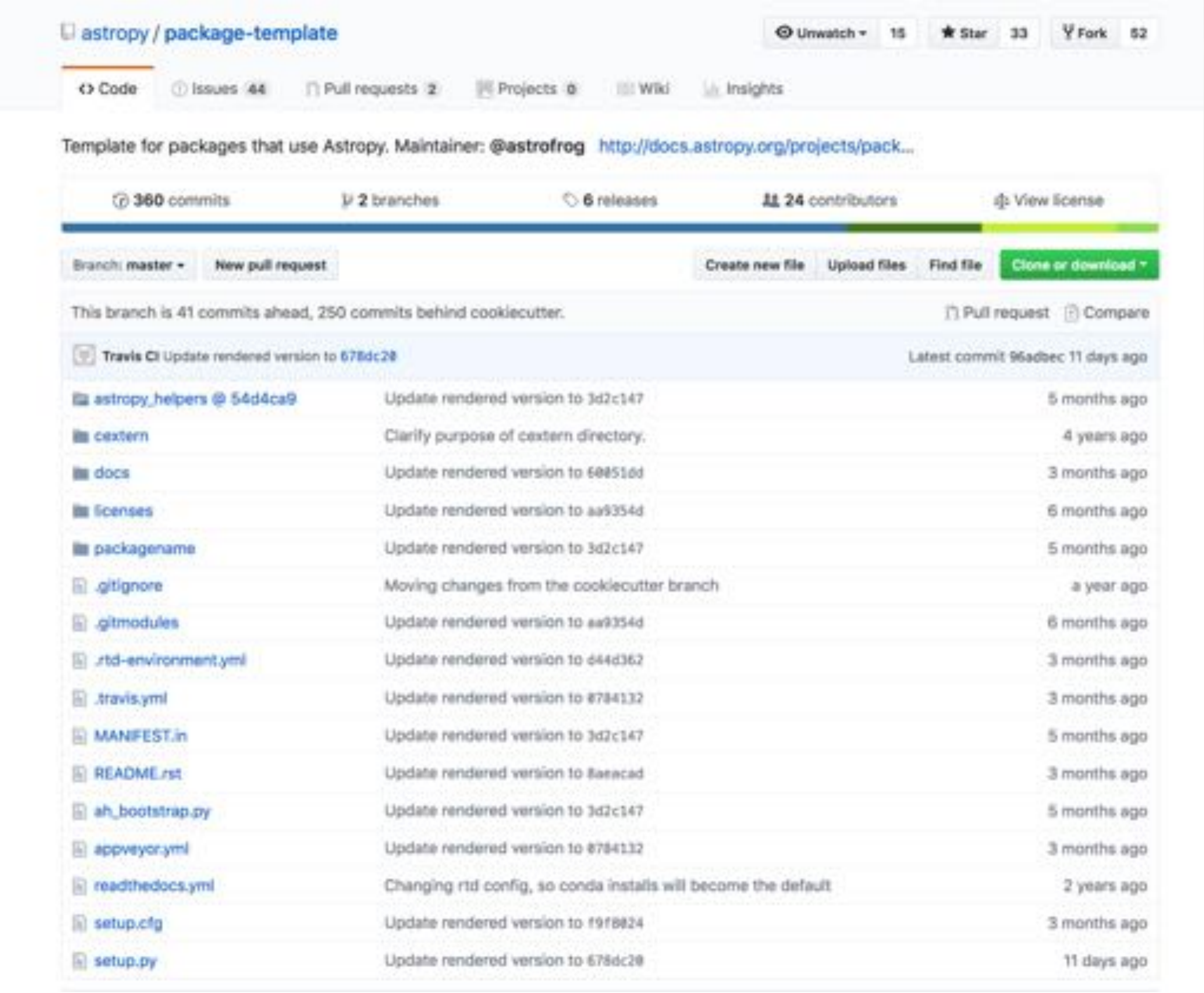
  *E.g. **astropy-helpers** used this way by many packages*

A simple package layout

- README
- LICENSE
- setup.py
- packagename/__init__.py
- packagename/moduleone.py
- packagename/moduletwo.py
- packagename/tests/
- packagename/subpackage/__init__.py
- packagename/subpackage/modulethree.py

```
>>> import mypackage
>>> from mypackage import moduleone, moduletwo
>>> import mypackage.subpackage
>>> from mypackage.subpackage import modulethree
```

Package template



Template for packages that use Astropy. Maintainer: @astrofrog http://docs.astropy.org/projects/pack...

This branch is 41 commits ahead, 250 commits behind cookiecutter.

Travis CI Update rendered version to 678dc20 — Latest commit 96adbec 11 days ago

astropy_helpers @ 54d4ca9	Update rendered version to 3d2c147	5 months ago
cextern	Clarify purpose of cextern directory.	4 years ago
docs	Update rendered version to 600516d	3 months ago
licenses	Update rendered version to aa9354d	6 months ago
packagename	Update rendered version to 3d2c147	5 months ago
.gitignore	Moving changes from the cookiecutter branch	a year ago
.gitmodules	Update rendered version to aa9354d	6 months ago
.rtd-environment.yml	Update rendered version to d44d362	3 months ago
.travis.yml	Update rendered version to 0784132	3 months ago
MANIFEST.in	Update rendered version to 3d2c147	5 months ago
README.rst	Update rendered version to 8aeacad	3 months ago
ah_bootstrap.py	Update rendered version to 3d2c147	5 months ago
appveyor.yml	Update rendered version to 0784132	3 months ago
readthedocs.yml	Changing rtd config, so conda installs will become the default	2 years ago
setup.cfg	Update rendered version to f9f0824	3 months ago
setup.py	Update rendered version to 678dc20	11 days ago

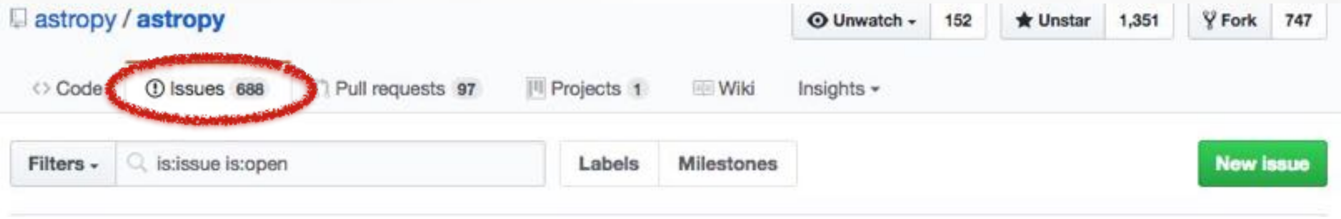
Astropy affiliated packages

- Total: 40 and growing (http://www.astropy.org/affiliated/)
- Functionality under development, may become part of core
  - reproject: astronomical image reprojection
  - astropy-healpix: A BSD-licensed HEALPix implementation
- More specialized functionality
  - **astroML:** statistical and machine learning tools for astronomy
  - **photutils:** source detection, photometry and related image-processing
  - **astroquery:** tools for astronomical data access (mainly for open data)
- Packages with incompatible licenses

How start contributing to existing libraries

- Look at the contributing guidelines, e.g. “CONTRIBUTING.md” files on GitHub repos, documentation pages.

- Look at the issues page



- Use the labels to navigate and find something to work on, e.g. Package-novice and good first issue are newcomer friendly.

- Open a PR with the fix. Expect to receive a code review and comment before your fix is merged.