STScI | SPACE TELESCOPE SCIENCE INSTITUTE

Python Data-Analysis Tools

Harry Ferguson

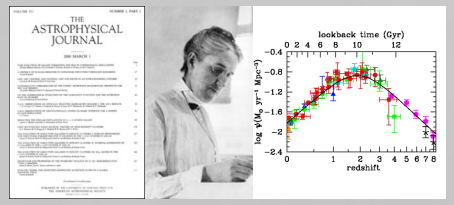
JWST Proposal Planning Workshop

18 May 2017

Motivation

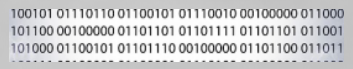
Data Analysis Tools

- Enable rigorous scientific analysis, fast
- These tools are where the astronomer interacts with the data.
- Many common tasks across missions & observatories & scientific disciplines:
  - Convert, combine, measure, model, visualize
- Must be built on rich, robust scientific libraries
- Must be flexible, powerful and user-friendly
- Must be well documented



Knowledge

Analysis Tools



Data

Who uses Python as their go-to language for data analysis?

Who doesn't now but thinks they might in the near future?

Tiers and key dependencies

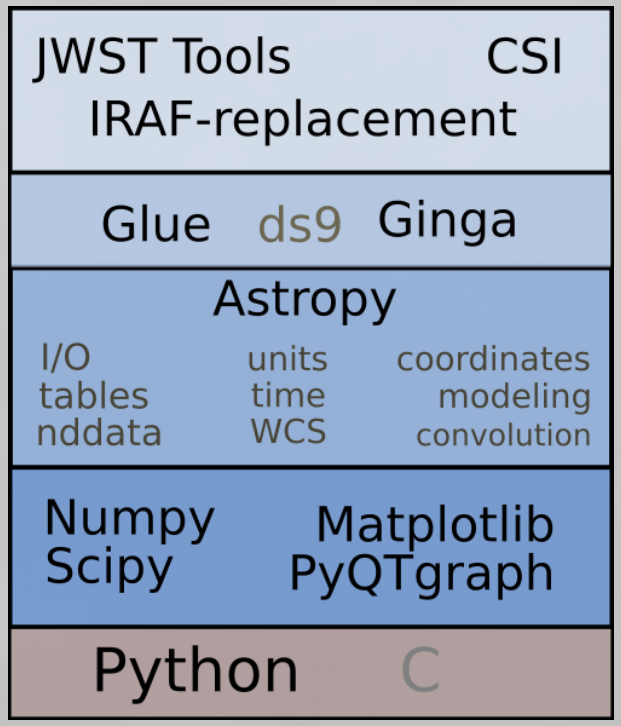
STScI-led Data Analysis Tools

Interactive visualization tools

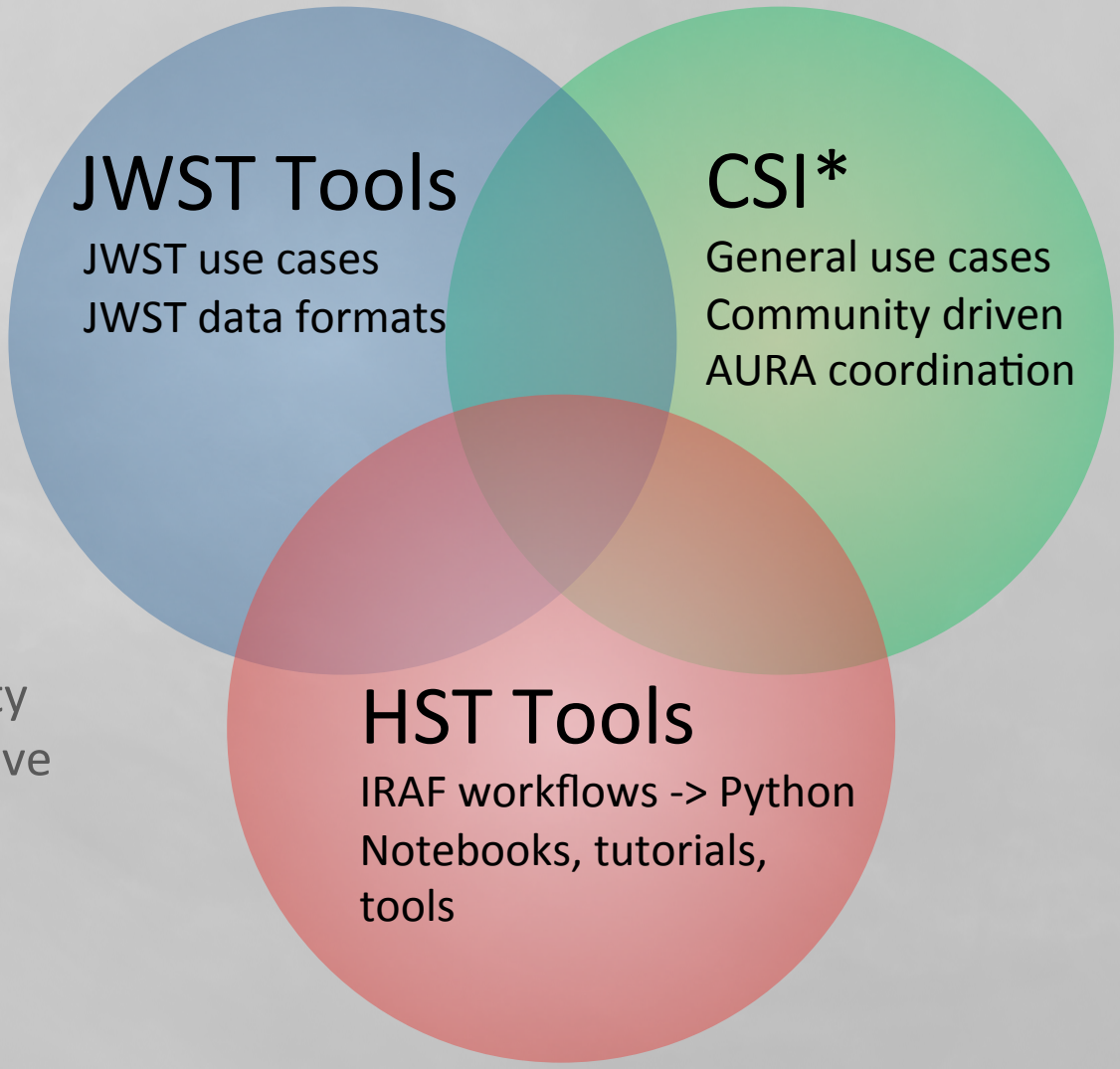
Astronomical libraries

Scientific & numerical libraries

Languages



DAT Efforts

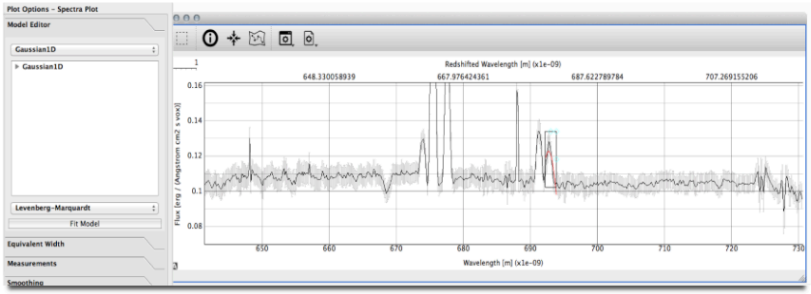


*CSI: Community Software Initiative

News

AstroConda: New Release of Science Software
News Feature • May 2016

Data Analysis Toolbox



Together with the broader astronomical community, STScI is building the next generation of software data-analysis tools to facilitate the translation of data into scientific knowledge. The user-customizable JWST pipeline will produce science-ready data. The tools referenced on this page pick up where the pipeline leaves off, allowing interactive data analysis and model fitting. These tools are being built primarily in the Python programming language and in conjunction with the community-led Astropy Project. Software development in this area is entirely open-source and community involvement is encouraged at every level: from providing feature requests, to commenting on code and documentation, to providing code or algorithms. The JWST Data Analysis Tools Development Forum (JDox) provides an avenue for getting involved. Links at the bottom of the page provide access to code and other resources.

The development effort covers a broad spectrum of software tools. On the one hand, "bread and butter" kinds of analyses need to be easy to accomplish. Examples include – circle a source on an image and measure its flux or point-spread function, or mask out the sources and fit a smooth sky background to this image. Such day-to-day tools will work with JWST data formats and provide facilities for dealing with units, coordinate systems, uncertainties and data-quality. On the other hand development is needed in areas where current tools in IRAF or elsewhere are not particularly mature – e.g. visualization and analysis of 3D IFU data, or fitting models to data using a combination of the rectified co-added frames for source detection and the un-rectified "raw" frames for the model fitting, to take advantage of the largely uncorrelated errors.

Software is available from github as it is being developed and will generally be released as Astropy affiliated packages when a stable version is available. By the time JWST launches, installation of a full suite of software will be possible. At the current time, interested users must install the packages individually.

Resources

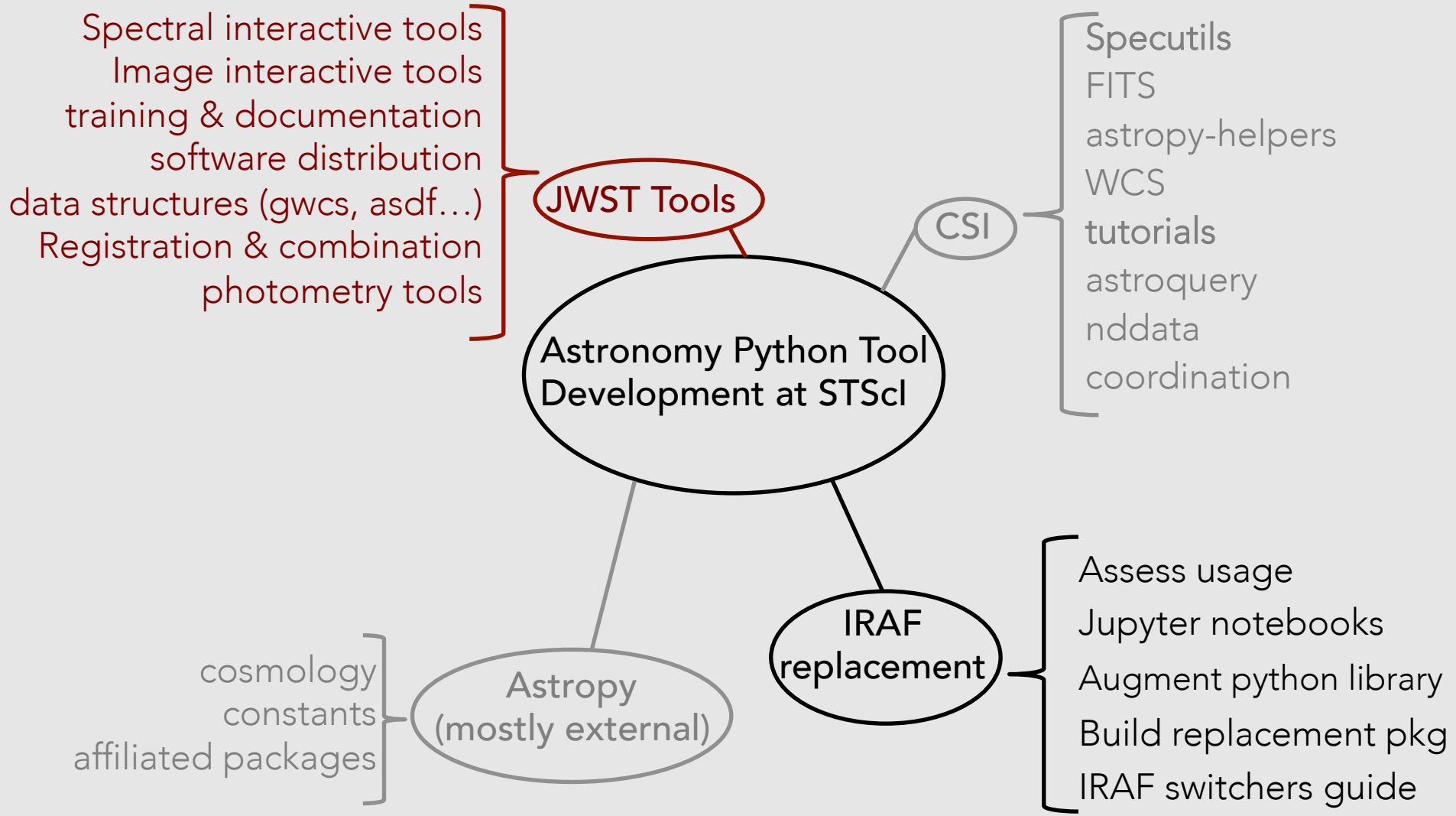
- Software installation using AstroConda
- JWST Data Analysis Tools Development Forum (JDox)
- User Training in JWST Data Analysis Workshop 2016
- 2015 Scientific Python Course
- Astropy software
- Astropy affiliated packages
- Python for Astronomers website
- glue - multidimensional data exploration
- Ginga image viewer
- STScI open-source software on github

AstroConda: One-stop shopping

- New distribution mechanism for *all* public STScI software
  - Powered by Anaconda distribution tool
  - Single command install of the python ecosystem
  - Includes:
    - Data Analysis Tools/Astropy
    - HST pipelines
    - JWST pipelines
    - non-python: IRAF, ds9, cfitsio, freetds

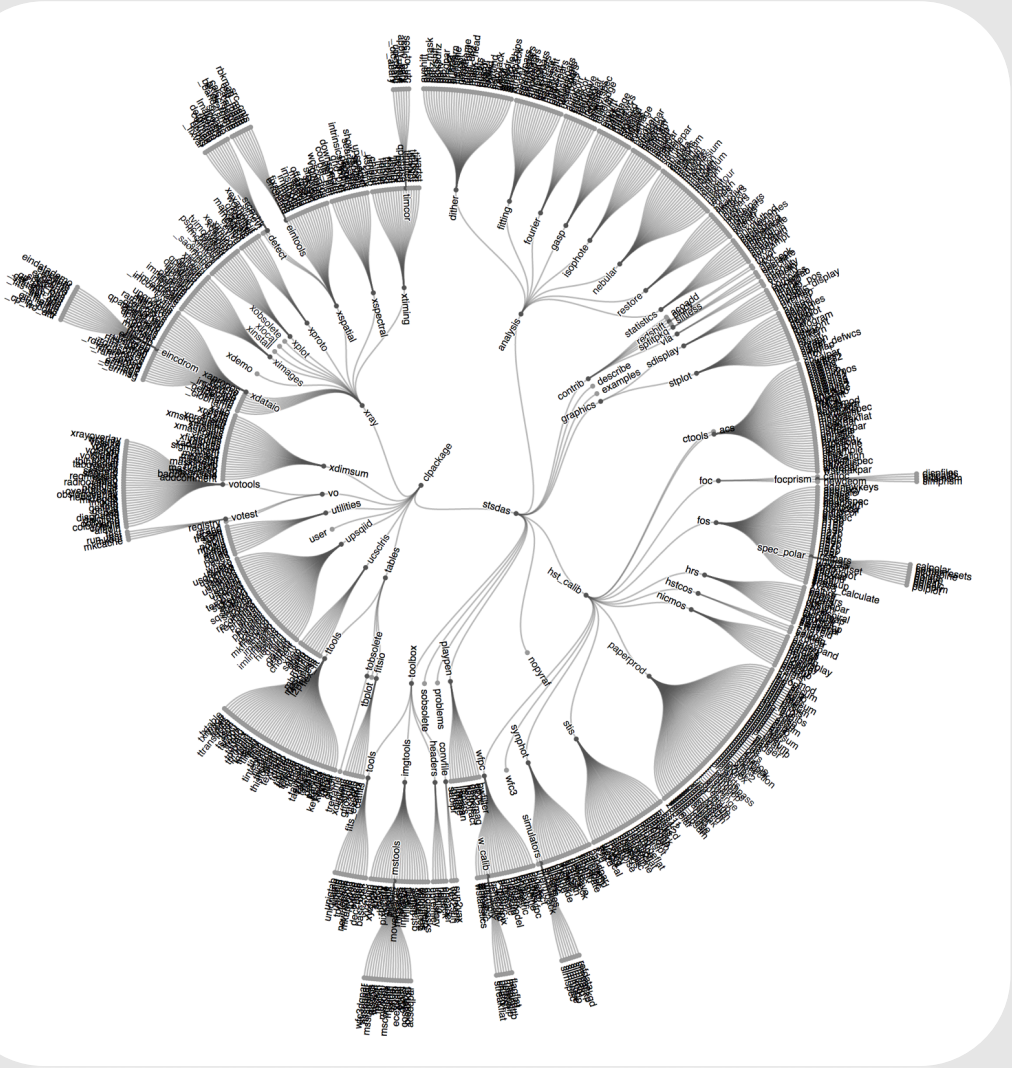
http://astroconda.readthedocs.io

JWST Data Analysis "Ecosystem"



IRAF Transition

- >2000 tasks in the IRAF distribution
  - ~1000 within current scope of project over a few year timeline
- HST funded work to transition away
  - Jupyter notebook workflows for simple tasks
  - Direct or improved replacements for critical tasks
- STAK project (not yet released)
  - Library for the replacement code
  - Home of the notebooks
  - http://stak.readthedocs.io
- Justin Ely, Sara Ogaz, …



Community Software Initiative

PI

Lou Strolger

CoI(s)

Erik Tollerud, Harry Ferguson, Susan Kassin, Ivelina Momcheva, Joshua Peek, Jonathan Hargis, Justin Ely, Pey Lian Lim, Nadia Dencheva, Perry Greenfield, Megan Sosey

Abstract

This proposal is to cement the foundation upon which the next generation of astronomy data analysis software is being built. Data-intensive astronomy requires this investment, whether from the DRF or elsewhere. This project is an attempt to stimulate a community-wide and cross-AURA collaboration on this vital underpinning of modern research. At the same time, it will help to solidify STScI's leadership while ensuring that the software is aligned with both institutional research priorities and priorities for supporting our missions.

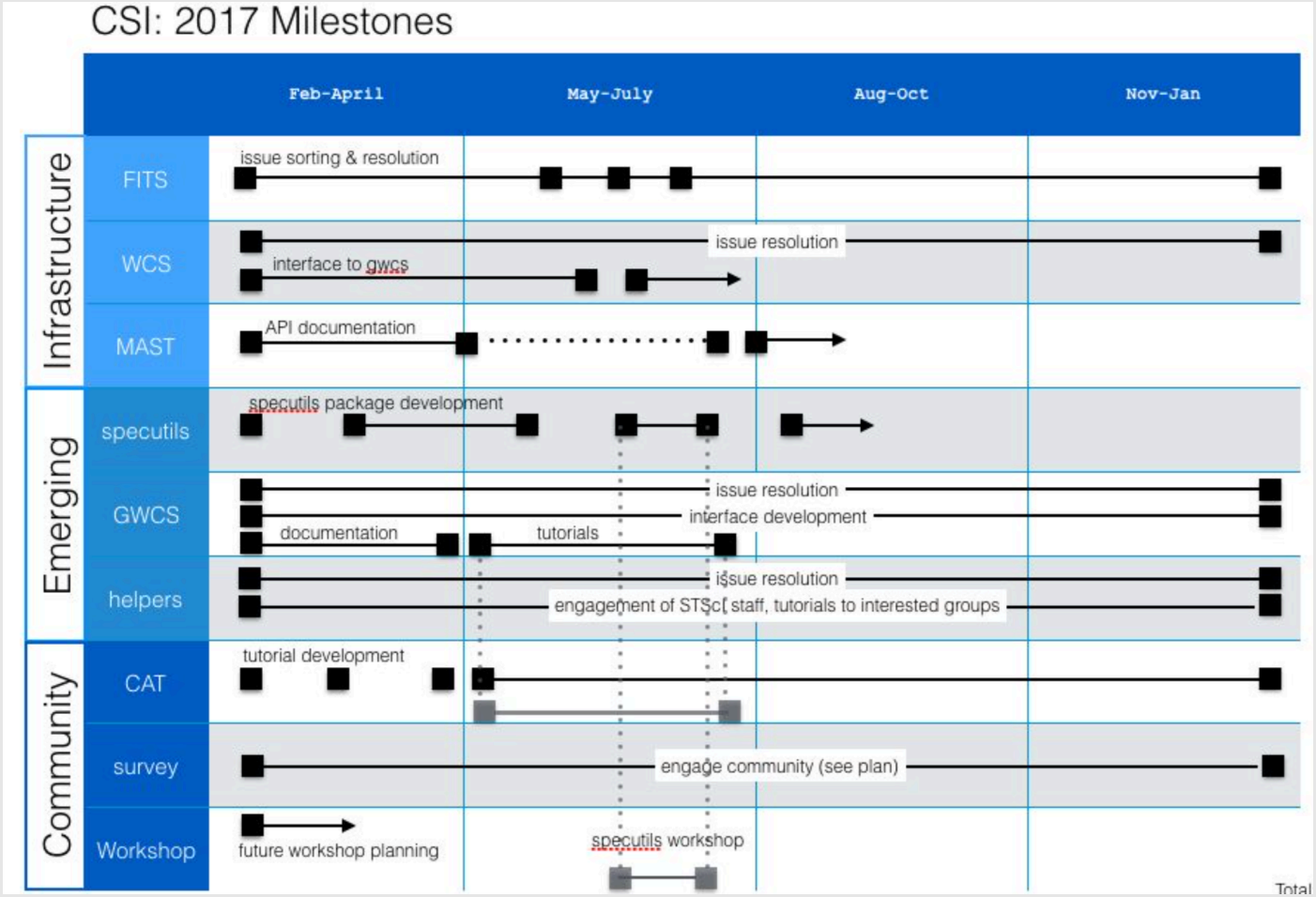
Dollar Amount

($902,500)

CSI Fills Urgent Needs

- Stopgap funding source for core infrastructure
  - FITS, World-coordinates, Astropy build tools
- Coordination between AURA centers (and beyond)
  - Generalized world-coordinate systems
  - General data structures for spectroscopy
- Better documentation and tutorials

CSI 2017 plans



SPECTROSCOPY TOOLS IN PYTHON WORKSHOP

INTRODUCTION

CSI sponsored workshop

This workshop is an effort orchestrated by the Astropy Spectroscopic Coordination Committee to establish the foundation for a unified spectroscopic framework in Astropy.

The workshop spans two days, the first of which will be discussion of the major structural concerns: a proposed Astropy Proposal for Enhancement, the first draft of the implementation, and the scope of the Python package containing the foundational pieces. We've coordinated with spectroscopy-inclined individuals in various astronomical fields to provide their input as potential users of the package, but we would also like to extend an invitation to anyone in the astronomical community who would like to participate. As part of the workshop, we have included time for some informal lightning talks on this day in the hope that demonstrating current workflows and tool sets will highlight the use cases this package ought to fulfill in order to be embraced by the community.

The second day will be a less formal hack day. The topics of the hack day will revolve around incorporating the designs discussed the previous day into current workflows, extending the implementation to include the results of the discussion day, or simply interacting with the package on a basic level: writing a custom spectrum reader, utilizing some basic analysis functionality, etc. In this way, users can walk away from the workshop having gained some experience and hopefully lower the bar for adopting the

Who uses Astropy regularly?

JWST Data Analysis Tools

Harry Ferguson, Justin Ely, Susan Kassin, Jay Anderson, Larry Bradley, Ivo Busko, Mihai Cara, Nicholas Earl, Jonathan Eisenhamer, Perry Greenfield, Pey-Lian Lim, Thomas Robitaille, Megan Sosey, Erik Tollerud, Matt Hill, Mario Gennaro, Gisella de Rosa, Martha Boyer & the JWST Data Analysis Development Forum

JWST Data Analysis: 5 Long Term Goals for Spring 2019

1. Enable analysis on un-resampled data, such as photometry, profile fitting, and spectral modeling of 1D/2D spectra and data cubes. (DMS-233, 630, 632, 633)

2. Build infrastructure to enable rapid development of analysis tools by the community.
   - Standards for data formats, in-memory data structures, classes, and methods (DMS-233)
   - Well-documented and well-integrated with all JWST tools

3. Provide interactive spectroscopic data analysis tools. (DMS 233)

4. Provide interactive image analysis tools. (DMS 233)

5. Provide support for tools for constructing JWST point and line-spread functions for a given observation. (DMS 634)

Who has used Ginga?

Who has used Glue?

JWST DAT 2017 Plans

This Year

Work area	%complete at end of FY17
Specviz	70
Specutils	50
MOSViz	50
Cubeviz	30
PSF-fitting photometry	60
photometry on dithered images	80
Image alignment	60
1D spectral alignment	80
2D spectral alignment	60
3D spectra alignment	60
grism alignment	30
JWST data registry (I/O)	100
ERS documentation	100
Cycle-1 documentation	100
Training	100
Architecture description	100
astroconda	80

2018+

gwcs	
asdf	
FITS	
Parameter files	50
Grism tools	0
pipeline helper tasks	5
ginga (maintenance + reqs gathering	30
optimal spectral extraction	5
ramp inspection & analysis	0
fitting dithered spectra	0
NRM/AMI (including alignment)	0
time-series	0
coronagraphy tools	0
coronagraphic alignment	0
alignment with different PSFs (image	10

How to get involved

- Contribute code or algorithms
- User testing of early releases
    - Specviz, Photutils, Glue & Ginga are high priorities for user feedback
- Contribute tutorials (writing or reviewing)
- Suggest tools, features
- Suggest tutorials
- Participate in Sprints

Who has used Photutils?

Who has tried SpecViz?

Videos

- SpecViz
- CubeViz
- MosViz
- Ginga plugin

What workflows do you wish were easier in Python?