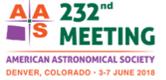


REPLACING THE IRAF/PYRAF CODE-BASE AT STSCI: THE ADVANCED CAMERA FOR SURVEYS

Ray A. Lucas, Tyler D. Desjardins, and Norman A. Grogin

Space Telescope Science Institute, Baltimore, MD



ABSTRACT

IRAF/PyRAF are no longer viable on the latest hardware often used by HST observers, therefore STScI no longer actively supports IRAF or PyRAF for most purposes. STScI instrument teams are in the process of converting all of our data processing and analysis code from IRAF/PyRAF to Python, including our calibration reference file pipelines and data reduction software. This is exemplified by our latest ACS Data Handbook, version 9.0, which was recently published in February 2018. Examples of IRAF and PyRAF commands have now been replaced by code blocks in Python, with references linked to documentation on how to download and install the latest Python software via Conda and AstroConda. With the temporary exception of the ACS slitless spectroscopy tool aXe, all ACS-related software is now independent of IRAF/PyRAF. A concerted effort has been made across STScI divisions to help the astronomical community transition from IRAF/PyRAF to Python, with tools such as Python Jupyter notebooks being made to give users workable examples. In addition to our code changes, the new ACS data handbook discusses the latest developments in charge transfer efficiency (CTE) correction, bias de-stripping, and updates to the creation and format of calibration reference files among other topics.

Introduction:

- IRAF/PyRAF was “black box” code; written by astronomers, but often no immediate insight into what it was actually doing
- IRAF/PyRAF no longer maintained; few/no resources for help; documentation no longer updated
- IRAF/PyRAF has some known/advertised problems/bugs in basic tasks which will not be fixed, and possibly unknown ones, too → Example: imstat: sums of exponentially large numbers of pixel values may encounter roundoff errors, significant when standard deviation is small compared to the mean
- IRAF/PyRAF increasingly difficult to compile on machines popular with users, and may eventually become impossible to compile
- IRAF/PyRAF has small user base of only astronomers which means little funding for maintenance and growth of capabilities
- Python is open source; much larger user community for many applications; critical astronomical coding can still be done by astronomers, but much of more general utility can be done by programmers
- Python more interoperable with new concepts such as downloadable interactive Jupyter notebooks (make changes to code and immediately see effects); more effective teaching tool by its nature
- Python more interoperable with new data visualization tools like Glueviz, etc. which can link different datasets, plot quantities from them all, rotate in 3-D space, etc.; also more active documentation



Using IRAF/PyRAF, and START using Python!

Ultimate Goal:

- Simplify/streamline documentation
- Reduce code blocks embedded in documents such as ACS Data Handbook
- Move code to interactive notebooks linked from the ACS Data Handbook and other on-line ACS pages and documentation

ACS Updates to User-Facing Documentation

- ACS websites now provide links to practical, Python-based ACS examples
- New ACS Data Handbook v9.0 now has scripting/software code examples in Python in all places except (for now) aXe grism software.
- aXe code still to be replaced

Some sample Python code from STAK Notebooks

```
# Astronomy Specific Imports
from astropy.table import Table

# Plotting Imports/Setup
import matplotlib.pyplot as plt
import numpy as np

# Using the weight column of our example table
n, bins, patches = plt.hist(ex_table['data'])
plt.xlabel('fwhm')
plt.ylabel('fwhm of stars')
plt.show()
```

ACS Functional Work and acstools

- ACS group internal scripts using IRAF/PyRAF are now in Python:
- New pixel area map (PAM) code 100% in Python generates PAMs on the fly, and no longer calls AstroDrizzle
- Superbias/Superdark/Sinkfile code uses Python
- Updated code for creating other reference files now uses Python
- acstools gives users access to CALACS and other useful tools in Python as part of Astroconda (see URL below, at bottom of poster)

ACS Data Handbook Updates, Code Examples, and STScI ACS code-base

- New ACS Data Handbook v9.0 also has CTE correction updates, updated info about new calibration files such as sinkfiles, bias de-stripping of subarrays, etc.
- Python used in new scripts for:
 - Bias de-stripping of subarrays
 - Creation and new format of some ACS calibration reference files
- Python used in generic tools for working with HST data such as updating wcs, DrizzlePac (AstroDrizzle, etc.) software

ACS group internal calibration software (now in Python) and new associated Jupyter STAK notebooks form basis for the standalone software provided to users

Some parts of astropy and other Python tools for astronomy and working on HST data are developed in-house here at STScI – wcstools, all the DrizzlePac software, etc.

Ultimate Benefits to Users:

- Modern, actively well-maintained code that compiles and runs on modern machines
- Larger user base with open source code; more people able to create more widely useful tools while still allowing specialists to create specialty tools
- Better insight into what code is actually doing; can interact with data, interface with databases, run web content, do calibration, data analysis, link to software that can make complex, advanced plots from individual or linked datasets or databases enabling new and better ways of looking at data
- Bugs/problems continue to be fixed; innovations/improvements continue to be made
- Code repositories like Github make software available to all for modification as needed for their own purposes and for contribution of new software
- Documentation is more up-to-date and can even be embedded with code where needed

Ultimately: Along with standard calibrations and tailored re-running of CALACS by users, these scripts continue the tradition of users being able to also fine-tune other aspects of the data reduction and data analysis to their own needs, with more insight, incorporating modern interactive means of learning how to use and modify the code for various purposes, and with more flexibility and interoperability with not only the data itself, but with databases, web content, advanced graphics and plotting routines using linked datasets and/or databases,

Questions?

- HST Help: <http://hsthelp.stsci.edu>
- ACS: <http://www.stsci.edu/hst/acs>
- <http://astroconda.readthedocs.io>
- STAK Notebooks: <http://stak-notebooks.readthedocs.io>