The Gemini Data Reduction Package – Now and Beyond

Kathleen Labrie, Jen Holt, Emma Hogan, Craig Allen, James Turner, & Nelson Zarate

Gemini Observatory

ABSTRACT

In 2005, Gemini started to move towards Python for data reduction. This was motivated by PyRAF emerging as a natural path from our IRAF CL/SPP code base to a modern, supported data processing platform. The users of the Gemini Data Reduction Package will soon benefit from the effort Gemini set aside to that end in recent years.

PyRAF is a combined environment containing the Python shell plus the IRAF CL, offering the users backward IRAF compatibility in addition to a Python programming base. Using the PyRAF module, IRAF tasks can be called from Python scripts much like a function library, thus making it easy to build custom tools without having to invest time reinventing tools already present in IRAF.

Python is becoming widely used as a modern scientific scripting language in the scientific community. Python is an easily comprehensible language that allows a novice to produce useful work in a few days. Moreover, Python is suitable for IDL- or Matlab-style interactive use as well as for developing complex programs.

Many of the advantages of Python from the developers' perspective will directly impact our users. Those advantages include more robust error handling, faster development cycle, and easier ways to automate tools for pipeline processing.

Here we report on the status of the Gemini Data Reduction Package and the new tools coming down the production line.

"Cut to the chase. When's the next release?"

GEMINI IRAF v1.10 – July 2009

The upcoming release of the Gemini IRAF package, v1.10, will be **fully compatible with PyRAF** (but will not *require* PyRAF). The software is undergoing pre-packaging review and testing. Based on the current schedule, packaging and final testing is expected to happen in June 2009, with the release shortly after. This next version will include major improvements to the GMOS tools, in particular support for propagation of the variance and data quality planes, and significantly improved support for the **nod-&-shuffle** mode. Also, for the **near-infrared spectroscopy** data, the application of the s-distortion and the wavelength calibration (the *transform* step) has been streamlined. Finally, some **new mid-IR tasks**, and several bug fixes and improvements throughout the package will also be included.

"What's cooking?"

gemini_python
A package for the Python era

The new Python development will be packaged and released in the **`gemini_python`** package. We are adopting a release model similar to the one used by STScI. We will release the **`GEMINI`** package, a continuation of the current IRAF package, which will contain the IRAF tasks and PyRAF interfaces (the interfaces that provide an IRAF look-and-feel to Python tools), and the **`gemini_python`** package for the new Python tools. The Data Processing Software Group is working on streamlining the package and dependency installation procedures, at least on the platforms most commonly used by our users.

"What have you been up to?"

Recent activities fall into the following three categories:

- Migration to PyRAF and Python
- Automation tools for the Pipeline Project
- Support for new instruments, namely NICI (on this poster) and Flamingos 2.

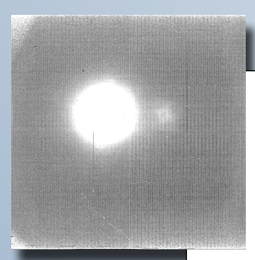
The primary goal of the Migration to PyRAF Project was to ensure that the Gemini IRAF package not only functions in the PyRAF environment, but that it generates products that are scientifically equivalent to the ones obtained from the CL environment. This project is completed. Release v1.10 (see elsewhere on this poster) of the package will be compatible with PyRAF. Also, as of May 2009, PyRAF is the official platform for operations at Gemini.

The automation of the data reduction software is done through the Recipe System and AstroData. Both projects are described elsewhere on this poster, along with the future gemini_python package.

In support of all that development, we set up a Regression Test Framework (RTF) and added literally hundreds of tests. Once the unit tests and reference input and output data have been defined, the RTF can automatically confirm that a new version of the software works as expected in all areas tested. This system adds robustness against change to our software and it will help us speed up our release cycle.

Finally, our group has grown! We have gone from 3 in 2007 to 6 today. And there is enough work to keep us all very busy!

NICI
Data reduction software for non-campaign users



NICI is the first instrument for which data reduction software is designed and implemented in Python. In fact, only one occasional calibration requires the use of IRAF components (which are accessed as functions from Python).

The software initially supports point source observation. It can process observations obtained for Angular and Spectral Differential Imaging (ASDI), the latest planet-detection technique. Subsequent versions will reduce data from extended sources and include other general improvements.

The code is being documented and tested prior to an initial standalone release. Eventually, it will be integrated to the package and released as part of v1.0 of the new gemini_python package.

AstroData
A smart way to access our data

In a nutshell, the AstroData class is built on top of PyFITS and allows instrument-agnostic access to data. For example, a little script to print the name of the filter used would look like this:

```
#!/usr/bin/env python

import astrodata

inputfile = sys.argv[1]
ad = astrodata.open(inputfile)
print "The name of the filter is ", ad.filtername
```

This script will work for data from any instruments defined in the AstroData facility. To put this in perspective, figuring out the name of the filter used for a NIRI observation requires probing three header keywords and analyzing all the combinations. With AstroData, the developer or the PI wishing to write a script does not need to know anything about how the filter information is stored in the FITS header for a particular instrument.

Additional features include a general rule-based mechanism for classifying datasets, eg as spectra, and a convenient method of grouping and iterating over multiple image extensions within FITS files.

The code is currently at the alpha testing stage and will be released through the up-coming gemini_python package.

Recipe System
Automating our data reduction tools

For the Pipeline Project, the Gemini data reduction tools must be automated and controllable. There is also a requirement for making the creation of new data reduction "recipes" astronomer-friendly.

The Recipe System is built on AstroData. AstroData recognizes the type of data being fed into the Recipe System. A Recipe Manager will then select automatically the most appropriate Recipe from a Recipe Library, and launch a reduction. A Recipe is a list of scientifically meaningful instructions like "biasCorrect" or "measureIQ". Those instructions we call Primitives. The Primitives are controllable Python scripts calling the data reductions tools (Python or CL).

"I need help!"

Please submit your questions, bug reports, praises through the HelpDesk system.

http://www.gemini.edu/sciops/helpdesk/

Your request will first be assigned to someone at your partner country's National Gemini Office (Tier 1). If the request cannot be solved at that level, it will be sent to Tier 2 which is handled by the Gemini Data Processing Team (that's us!).