Advances in Chandra Data Analysis

Nancy RA. Wolk, Michael S. Noble, Stephen M. Doe

Harvard-Smithsonian Center for Astrophysics, 60 Garden Street, Cambridge, MA 02138

Abstract. With the December 2000 release of the Chandra Data Analysis System (CIAO), the interpreted programming language S-Lang was embedded into the system. This change ushered in a new phase of analysis within CIAO. S-Lang has increased the data reduction capability of CIAO applications and has provided a method for scientists and developers to quickly create new packages for specific analysis problems.

In this paper, we present a brief review of the CIAO applications ChIPS and Sherpa, and then discuss the recent advances in interactive analysis. The focus is on the S-Lang language and the benefits of this integration. We present a series of examples that demonstrate the power and flexibility of CIAO. Finally, we discuss plan for future enhancements.

1. Introduction

CIAO entered a new era of interactive analysis when the S-Lang scripting language was integrated with the system. Embedding S-Lang into CIAO applications opened new possibilities for the users including using S-Lang to inspect and easily manipulate their data. The addition of scripting allows scientists to add their own functions and packages to CIAO. Rapid prototyping of new procedures can be done with S-Lang which shortens development time.

The S-Lang language was developed by John Davis of the Center for Space Research at MIT. We have written a C++ interface layer, VARMM, as an access point for CIAO applications. This facilitates the merging of CIAO software and S-Lang without exposing application code to the internals of the scripting language and greatly reduces the development learning curve.

2. A Review of ChIPS and Sherpa

As of this release, S-Lang is used explicitly in two CIAO analysis applications: ChIPS and Sherpa.

The Chandra Imaging and Plotting System (ChIPS) is the visualization application of the CIAO system. Available as both an interactive application and a library, ChIPS is embedded within several CIAO applications including Sherpa, the Prism file browser, Filtwin, Obsvis, the summary page plotter and the verification and validation software.

The generalized fitting engine of CIAO is Sherpa. A mission and wavelength band independent analysis application, Sherpa enables the user to fit predefined or user defined models to their data.

3. Scripting in CIAO

The S-Lang scripting language is embedded in numerous popular Internet applications and is bundled with several Linux distributions. The VARMM (Variables, Math and Macros) layer (Doe *et al.*(2000)) interfaces S-Lang with the CIAO tools, introducing scripting to CIAO. VARMM provides a simple interface along with adding new i/o functionality. Both Sherpa and ChIPS are interfaced with VARMM providing access S-Lang functions and declarations at the command line.

Features of the S-Lang language include global and local interpreted variables, multidimensional arrays, built-in mathematical functions, application of functions on arrays, programmability with user-defined functions, and the availability of structures. These features are well suited to the astronomical community.

Scripting languages create an environment where rapid development of new analysis routines is encouraged. Specialized packages can be quickly prototyped in S-Lang and used within ChIPS and Sherpa. One example is the GUIDE spectral analysis package (Doe *et al.* 2000), now included in the current CIAO distribution. Scientific requirements quickly translate into functionality as scientists and developer alike use S-Lang to develop their own scripts to be imported into the system. This cross development between scientist and programmer is beneficial to the system as a whole. Although the science staff may not be familiar with S-Lang, its similarity to IDL helps ease the transition. These languages are similar enough that IDL procedures can be often translated easily into S-Lang.

The baseline functionality of CIAO applications can be augmented with user written functions in C, S-Lang, C++, or Fortran. These functions can be grouped into modules that are available to be imported into any S-Lang embedded application. ChIPS and Sherpa can import other S-Lang modules such as the Interactive Spectral Interpretation System (ISIS)[1], making the ISIS functions available within these applications. ChIPS is available as a S-Lang module that can be imported into other S-Lang applications. The simplicity of importing modules extends the application functionality without the need to write additional code.

4. CIAO analysis examples

In this section, we provide a few examples to demonstrate the abilities of CIAO with S-Lang integration. The CIAO thread page[2] provides more examples.

---

[1] `http://space.mit.edu/ACIS/ISIS`

[2] `http://cxc.harvard.edu/ciao/documents_threads.html`

- Both ChIPS and Sherpa can be used as a command line calculator by directly evaluating S-Lang commands.

```
chips> sqr(12)
144
chips> sqrt(sqr(12))
12
```

- Powerful scripted I/O functions are available in CIAO. The VARMM module was written to read FITS, IRAF, and ASCII files, and to write FITS and ASCII outputs. With these functions, a single command can be used to read in a complex astronomical file and store the information in a S-Lang structure.
  Here is a 2-line ASCII to FITS conversion tool:

```
chips> ascii = readfile("ascii.dat");
chips> writefits("copy.fits",ascii,"duplicate")
```

  Along the same lines, here's 2-line QPOE to FITS converter:

```
chips> qpoe = readfile("table.qp");
chips> writefits("qpcopy.fits",qpoe,"duplicate")
```

- User defined functions are trivial to import and use. The GUIDE spectral analysis package is an example of a collection of user defined S-Lang functions. Writing this in S-Lang provided a rapid development cycle with a scientist directly involved with the software team.
- Sherpa can store information about the fitting session in a Model Descriptor List (MDL). The MDL can be stored in S-Lang structures for the exploration of the model and be written to a FITS file for later use. This example uses S-Lang to explore parts of a model before writing the file out.

```
sherpa> source = gauss[g]
sherpa> fit
sherpa> foo = sherpa_get_mdl() # get the MDL
sherpa> writeascii(stdout,foo.parname,foo.parvalue)
g.fwhm  32.3539
g.pos   807.863
g.ampl  117.825
sherpa> write mdl "mdl1.fits"   # write a FITS file
```

- In addition to user defined functions, user written models developed in S-Lang are accepted in Sherpa. These models can be used in conjunction or in place of the predefined models in a Sherpa fitting session.

5. New Features in CIAO

In addition to the S-Lang integration, we've added new features to CIAO including:

- The Prism file browser has cell-based table and image editing capabilities including row and column addition. It also can access the full ChIPS plotting functionality.
- The ChIPS undo/redo and store/restore commands function with S-Lang variables.

- An easier and more flexible mechanism for custom plotting is available through the use of state objects, specifically the ChIPS state object.

6. Future Enhancements

While the addition of S-Lang has transformed CIAO into a flexible and powerful analysis system, we are looking forward towards the next development cycle and further enhancements.

One goal is to expose more of ChIPS and Sherpa internals to the S-Lang interpreter. Current design allows the definition of S-Lang variables which can be passed into ChIPS and Sherpa commands. We wish to change these commands to accept S-Lang functions and declarations opening the possibility of extremely complex operations being completed with minimal command sets.

During the last development cycle, we created a prototype ChIPS GUI using Tcl/Tk. This approach led us to determine that a new design was necessary. As a result, we are exploring the encapsulation of the GTk widget set as a S-Lang module. This will enable users to create ad hoc GUIs, in addition to standard CIAO GUIs.

We are planning on adding S-Lang wrappers around additional libraries in the CIAO system. Proposed libraries are the fast Fourier transform (FFTW) library, the world coordinate system (WCS) library, the parameter interface library, the XPA set, and the convolution library. The wrappers would make the library functions accessible at the Sherpa and ChIPS command lines, further extending the power of data analysis in CIAO.

An enhancement under consideration is to merge the ChIPS and Sherpa applications into a single CIAO product. Since Sherpa uses the ChIPS libraries, we have an opportunity to bring together the grammars of these applications and to create a single, coherent interface for the user.

7. Conclusions

The integration of a scripting language with CIAO has created a system that is extremely flexible and powerful, without a large learning curve for the user. The choice of S-Lang produces an environment with rapid prototyping of code, user defined functionality and extensibility. Scientists are able to contribute significantly to the software development cycle. In examining future enhancements, we are actively pursuing a further integration of S-Lang and libraries into the CIAO system, producing an even more powerful analysis package.

Acknowledgments. This project is supported by the Chandra X-ray Center under NASA contract NAS8-39073.

References

Doe, S., Noble, M., & Smith, R. 2001,in ASP Conf. Ser., Vol. 238, Astronomical Data Analysis Software and Systems X, ed. F.R. Harnden, Jr., F.A. Primini, & H.E. Payne (San Francisco: ASP), 310