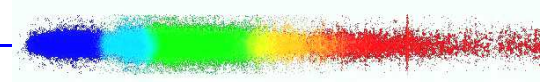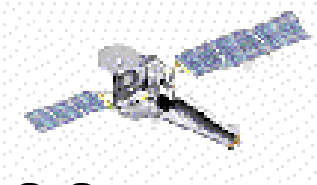## CXC Software for Grating Spectroscopy

David Huenemoerder (dph@space.mit.edu) 12 June 2003

A CXC goal is to provide a programmable analysis system for grating spectroscopy. ISIS has been serving as a stand-alone research and development package as well as an important pathfinder for features now appearing in Sherpa in CIAO 3.0.

There are two CXC-software options for grating spectroscopic analysis:

1. ISIS (plus optional Xspec module; CIAO 3 provides optional sherpa module).

2. Sherpa (including ISIS module for CIAO 2.3 on).

ISIS:

- Designed specifically for grating spectroscopy, for fitting list of 1D spectral histograms;

- Interface to AtomDB for atomic data, line, and continuum emissivities, browsing the database and computing plasma models;

- Slang interface only, designed for fully programmable control, but also with reasonably simple interactive commands.

Sherpa (CIAO 3.0)

- Generalized fitting, modeling; e.g. 1D, 2D, histogram, spatial, spectral.

- Designed as command-line-interpreter; SLang parser now coexists.

- S-Lang interfaces to Sherpa added starting in CIAO 3.0 (S-Lang parser and some functions in CIAO 2.0)
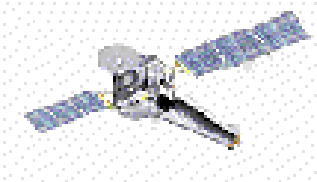
Important Features

- **ISIS**

  - **AtomDB interface, for line data, emissivities, plasma models;**
  - **Modular design (slang, cfitsio, pgplot); small source code volume (20,000 lines of C, 5,000 lines of S-Lang (including `cfitsio` and `xspec` modules).**
  - **Extensible, even at very low levels: kernel (entire "RMF" integrand; e.g., for pileup), slang or C user models, slang or C user statistic; "hook" functions;**
  - **Two optimization methods built-in; user method via C interface;**
  - **Two statistics built in; slang interface for user statistic;**
  - **Plotting (via pgplot);**
  - **A few key built in fit functions for spectroscopy (~ 10)**
  - **Xspec module, with all xspec models (including local and convolution);**
  - **Well documented (Introduction and Reference manual)**
  - **Dynamic binning and grouping (including rsp regridding)**
 - **Generic Monte-Carlo wrapper (for fitting or statistic)**
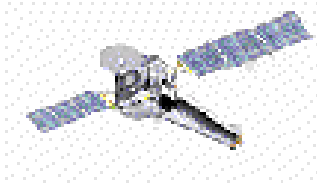
- Basic math functions (rebinning, interpolation, histograms, 1D FFT, random numbers);

- Easy to maintain, build, enhance;

- Portable (Linux, Sun Solaris, SunOS, DEC, FreeBSD, IBM AIX; MacOSX)

- Stable core; used for ~ 4 years;

- Maintained and developed by a single architect, with enhancements based on input from a small group conducting grating analysis.

- Low-resolution or other spectroscopic analysis automatically supported by hi-res functionality (ACIS imaging, RXTE, XMM, HST).
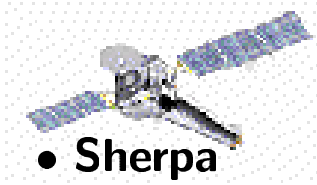
● **Sherpa**

– **Large volume of source code (100,000 lines)**

– **Extensible via S-Lang, optimization method and statistic via C interface;**

– **12 built-in optimization methods;**

– **9 built-in statistics; three statistical tests**

– **Portable (Solaris, Linux, MacOSX)**

– **Inherently multi-mission/discipline (no axis assumptions; Data-Model i/o)**

– **Multi-dimensional fitting.**

– **Access to AtomDB plasma models (via ISIS or Xspec modules)**

– **Xpsec models included (except local or convolution; Sherpa's convolution is more general)**

– **Well tested, documented (manuals, ahelp, threads), consistent.**

– **Long development cycle, improved by 3.0 infrastructure release**

– **Hybrid parser (command line interpreter plus slang); context sensitive commands**

– **Plotting (via SM, Chips)**

– ~ 6 **years development;**
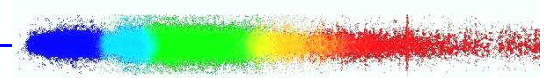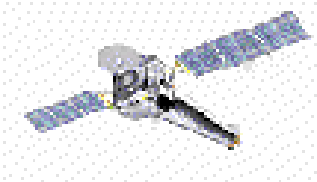
Example Scripts, Documentation

Some ISIS and Sherpa scripts are printed in a separate handout. Many examples are available on the web:

- **ISIS examples:** `http://space.mit.edu/CXC/ISIS/examples.html` — **single-ion spectrum, fake data with line broadening, monte-carlo line profile analysis, continuum modeling.**

- **Sherpa command-line threads:**
  `http://cxc.harvard.edu/ciao/threads/sherpa_grating/`

- **ISIS manual:** `http://space.mit.edu/CXC/isis/manual.html`

- **Sherpa/GUIDE manual:**
  `http://cxc.harvard.edu/ciao/download/doc/guide_doc.ps`

- **WebGUIDE:** `http://obsvis.harvard.edu/WebGUIDE`

- **Sherpa on-line help ("ahelp"):** `http://cxc.harvard.edu/ciao/ahelp`

- **Sherpa manual:**
  `http://cxc.harvard.edu/ciao/download/doc/sherpa_html_manual/`
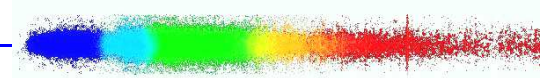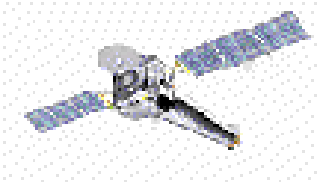
Other Uses of ISIS as a prototyping & development platform

Pileup models via ISIS module; (imaging-mode now in sherpa, Xspec)

`cfitsio` S-Lang interface (now in ciao)

`contamarf`: rapid implementation of new calibration effect - high-resolution contamination function.
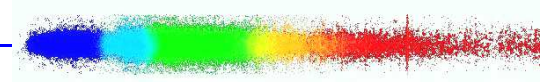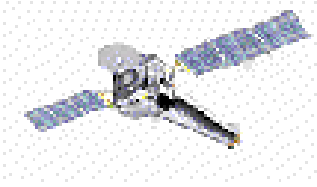
Bayesian blocks light curves

kT map modeling - dynamic binning, individually fitting a spectra in a thousand regions in extended sources.

"PI-on-the-fly" for ACIS RMF regridding.

ACIS time-dependent gain prototype

Distributed processing.

Emission measure modeling

These "Other Uses" (mostly non-grating specific) demonstrate the power of a programmable scripting language combined with function-based data and model access.

Some have been possible in CIAO 2.3, and CIAO 3.0 S-Lang infrastructure provides greatly enhanced scripted modeling support for such:

X-ray Dust halos: custom sherpa-S-Lang models for Chandra PSF-fitting of extended sources.

ChAMPs Spectra, perl-driven-sherpa for hundreds of spectral fits (to be converted to CIAO 3 perl-S-Lang interface).
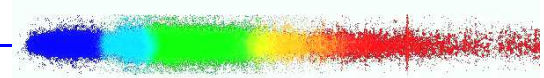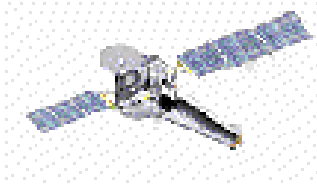
Plans

With the CIAO 3 release we are in a position to look at integrated approaches for sharing and coordinating technologies and lessons-learned from Sherpa and ISIS. We recognize the value in a global approach, and also the value in limited-scope applications. We are beginning to explore in detail divergence/convergence pros and cons, as well as common bases for future enhancements.

- ISIS
    - No major development plans for the core system.
    - Enhance, bugfix as per user request.
    - Apply to research; automate repetitive tasks.
    - Prototype new applications (e.g., emission measure modeling support; interactive event browsing)
    - Develop/test new slang modules (such as GSL, PVM, GTK).
    - Improve documentation.
    - Improve/add measurement tabulation/output — e.g, use Sherpa's "MDL".
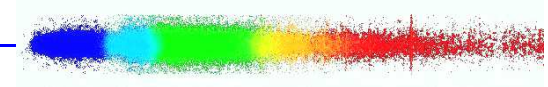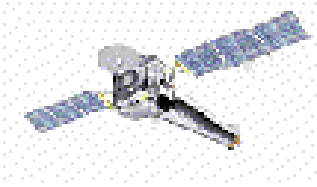
- **Sherpa**

  - **Support a wide range of users.**

  - **Continue adding S-Lang functions useful for library access, common tasks, or automation.**

  - **Implement sophisticated plasma modelling codes based on AtomDB.**

  - **Improve interactive plotting, GUIs, scatter data, "MDL" (see other talks).**