Figure 1: Exposure Time Calculator.

estimate the required exposure to obtain a spectrum with the given S/N.

For example, input parameters might include: a 15th V-magnitude star with an effective temperature of 5000K. The observations are to be taken during bright time at 1.5 airmasses and with 1.0 arcsec seeing. The wavelength of interest is 6000 Å employing the Red cross-disperser and a 1.2 arcsec slit (see Fig. 1).

With these input parameters S/N of 15 per pixel (in the summed spectra) can be achieved in an exposure of approximately 2600 seconds.

The simulator can be used to plan an observing run but should not be substituted for actual observing!

S. Randich; e-mail: srandich@eso.org

The Eclipse Software

N. DEVILLARD, ESO

A Brief History

Adonis, the Adaptive Optics instrument on use on the 3.6-m telescope on La Silla, is available for the astronomical community since April 1993. Experience has been acquired by many users since then, and what Adonis needed most was to gather this experience, and then offer a set of data-reduction facilities and guidelines for observation and calibration procedures. To get all the expertise, a working group was created, dealing with High-Resolution Infra-Red Data Reduction. Its output was directly fed into software development to produce the *eclipse* software.

When we took the decision to develop *eclipse*, we had to set up how far it should go when reducing the data, and which language/platform should be host. By questioning Adonis users, we found out that writing the Data Reduction Software within a standard reduction package such as MIDAS, IRAF, or IDL, would certainly reduce the number of potential users. Furthermore, high-resolution data reduction techniques are constantly evolving, and it is not always easy or even possible to add up brand new algorithms to most standard software packages. The result had to work simultaneously under most other data-processing software packages, and modifications should be easy to quickly add new methods as they come out of the Working Group.

The decision was taken to develop in ANSI C, which acts as a *lingua franca* for Unix-based workstations. The set of supported machines is: any machine running a Unix Operating System. In this way, we ensure portability, code reusability, and integration into all major data-reduction software packages.

Overview

eclipse offers an open environment for data-reduction algorithm development and simple pipeline processings. At the highest level, the user will find a manual describing how to observe and calibrate data using Adonis in its different modes, with associated example reduction scripts ready to be called from within standard software or directly from the Unix command line. So far, covered reduction aspects are:

- pixel gain map creation
- dead pixel detection and correction
- simple data classification from FITS information
- sky-subtraction and flat-field division
- shift-and-add
- data extraction and merging
- statistics computation
- cube arithmetic
- spatial filtering
- Fourier transform
- image resampling

All of these high-level procedures are running without user interaction, which is typical of a pipeline, number-crunching approach.

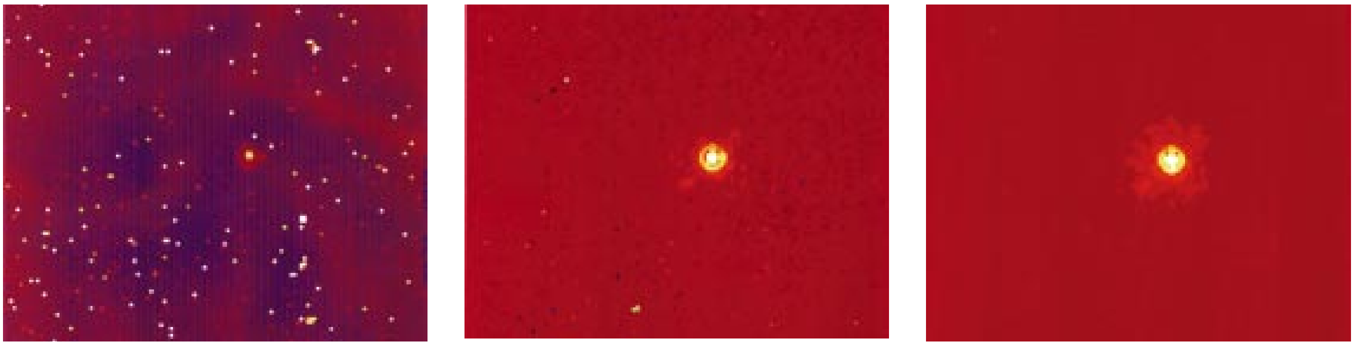

Figure 1: From left to right: Figure (a) is a raw frame, Figure (b) is sky-substracted, Figure (c) is bad pixel corrected and flat-fielded. The whole process used 1.26 seconds CPU time on the La Silla main server, including flat-field creation from a set of sky images and bad pixel detection. The total amount of involved data is about 5 megabytes.

Going one layer down reveals that *eclipse* is made of 32 Unix commands, each of them delivered with a manual page containing explanations about algorithms, input parameters and files, and several examples. By combining these commands into scripts, it is possible to create simple pipelines handling basic calibration and cosmetic procedures in a fast and efficient way. Notice that commands are scriptable from any high-level programming language, e.g. *MIDAS*, *IRAF*, *IDL*, *perl*, *tcl*, or standard Unix shells. Most of all, feedback from the working group allows a fast development of the latest algorithms, specific to Adonis.

Going to the very heart of *eclipse* means getting into 30,000 lines of documented ANSI C code. Written in an object-oriented way, it provides the programmer with a set of typical objects (images, cubes, pixel maps . . .) to develop with. From this level, it is possible to embed validated algorithms, which allows strong optimisations both in terms of speed and disk space, and encapsulation of these algorithms for higher-level processes. Notice that optimisation in the case of Adonis is a major concern: with data cubes reaching sizes of several hundred megabytes, it is easy to overflow disks, saturate memory, or in the best case monopolise computing power for several hours, preventing any other computer work! And the situation is likely to get worse on these aspects, with ever increasing detector sizes… Nearly all algorithms included in the *eclipse* code have been especially designed to handle these aspects in the best possible way, making use of automatic disk swapping and delicate memory handling routines.

Notice that a link to off-line data-processing software is essential. *eclipse* does not provide any complex post-processing algorithm such as deconvolution, nor does it contain any image or data displayer. Global data analysers such as MIDAS provide the full range of functionality needed for evolved post-processing and analyses; *eclipse* is to be used as a pre-processor, a signal-processing engine.

The Adonis Pipeline

A set of scripts and Unix commands has been especially designed to take care of most basic data reductions for Adonis. This has been set up during August 1996 on La Silla, with help from observers, telescope and Adonis teams.

The following set of operations has been automated:

- flat-field creation
- bad pixel detection
- sky extraction from data cube, averaging, and subtraction from object frames
- average of the result

There is a preparation phase, during which the observers have to identify their files according to their logbook, and sort them out, preferably in separate directories. It is then possible to design quickly a Unix script to launch a unique reduction command on all directories, and get cleaned data in a very short time.

Let us browse through an example:

Figure 1(a) shows a raw image of an object taken in thermal infrared. Noise is dominant, the object is barely visible among the noisy background. Figure (b) shows the same image after subtraction of the closest (in time) averaged sky. On Figure (c), bad pixels have been cleaned, and a flat-field division has been applied. The star is now cleaned, ready for deeper analysis.

The whole process applied on the La Silla main server (kila) took 1.26 seconds CPU time, and 12.73 seconds user time with 70 active users, for a total amount of more than 5 megabytes of data. The given figure includes processing time for flat-field creation and bad pixel detection. Such processing speed implies that data cleaning takes place in less time than needed for acquisition, which makes such a pipeline a good candidate for on-line display.

Future Developments

For the moment, *eclipse* handles directly all imaging modes in all wavelengths for Adonis. By combining its low-level functionalities, it is certainly possible to develop new modules, specific to other instrumental modes (coronography, spectro-imaging with Fabry-Perot, polarimetry, astrometry, . . .). Christian Drouet D'Aubigny (ESO-Garching) is right now working on the Fabry-Perot extension for Adonis, together with Patrice Corporon (ESO-Chile), both coopérants.

Julian Christou's deconvolution software: *idac*, has been recently linked to *eclipse*. In this way, the user also has an iterative blind deconvolution algorithm readily available in La Silla's machines. Though it would probably not be used as on-line software, *idac* may need some of *eclipse* functionalities to provide default start-up estimation images, create bad pixel maps, or apply cube arithmetics.

eclipse is covered by the MIDAS GNU public general license, which allows us to distribute it on the World Wide Web. It should be included in the future in MIDAS distribution as a contribution for Adonis. MIDAS has been enhanced to support data cube manipulations, viewing, and direct FITS access, which ensures that a complete toolbox is available to the Adonis user for data reduction.

Finally, portability made it possible to install *eclipse* in many other observatories and institutes, for data reduction of eventually other instruments on other telescopes.

I would like to thank all people participating in the High-Resolution Data Reduction Working Group for their kind help and patience in explaining to me the secrets of infrared data processing. Note that for 16 months the development team consisted of a single person (N.D.), but, in fact, many people worked on the algorithms, and it would be hard to acknowledge them all here. The goal has been reached: Adonis is now provided to the astronomical community together with a documented and dedicated data reduction package.

Most eclipse documentation is available on-line on the ESO server at the following address: http://www.eso.org/~ndevilla/adonis/eclipse.

Nicolas Devillard
e-mail: ndevil@eso.org