with the lunar eclipse and the partial solar eclipse – to dedicated, often highly successful, efforts to accelerate school plans for getting an Internet connection.

We at ESO have been pleased to be involved in this project from the beginning and thereby to contribute to the three goals mentioned above. We are also confident that Astronomy On-Line will provide a stimulus to young people to become more interested in natural sciences, especially at a time when worries have been expressed at many European Universities about the dwindling number of young people who are considering to pursue a career in these fields.

The European Week for Scientific and Technological Culture

The objective for the European Week for Scientific and Technological Culture is to improve European citizens' knowledge and understanding of science and technology, particularly in their European dimension: it addresses both pan-European scientific and technological co-operation, as well as science and technology in each European country.

"The Week" is organised every autumn by the European Commission in collaboration with national and international research organisations, universities, museums, TV, etc.

EUROPEAN WEEK FOR SCIENTIFIC AND TECHNOLOGICAL CULTURE

Acknowledgements

Astronomy On-Line was made possible by the active personal involvement of a large number of dedicated people all over Europe and, indeed, on several continents. Thanks to their unremitting efforts, it became possible to implement in an extremely short time, a project which will undoubtedly serve as a very useful example for similar events within other subject areas.

Richard West, e-mail: rwest@eso.org

Availability of IDL at ESO/ST-ECF

R. ALBRECHT, W. FREUDLING, R. THOMAS,
Space Telescope – European Coordinating Facility

Introduction

The range of astronomical computing has become so broad that there is no one single data-processing system which covers all possible aspects. Thus, in addition to the indigenous data analysis system MIDAS (Munich Image Data Analysis System) ESO supports a number of other analysis systems. The two most important ones are, at this time, IRAF (Interactive Reduction and Analysis Facility), and IDL (Interactive Data Language).

For many years now, IDL has played a significant role in astronomical data analysis. Originally developed by Dave Stern of Research Systems Inc. in the late seventies, IDL has survived many transitions of hardware platforms and operating systems, all the time improving the functionality while retaining full backward compatibility. This is a truly remarkable achievement.

For Space Telescope data analysis, IDL has been an important tool since the very early days. Since it had been used for the analysis of IUE data, and staff of the IUE centre at GSFC was heavily involved in the development of the Goddard High Resolution Spectrograph (GHRS), a large amount of spectroscopic analysis software was available in IDL. In addition, IDL provided the possibility to read the HST native data format, GEIS (Generic Edited Information Set).

Because of its importance for HST data analysis, ST-ECF has been supporting IDL since 1984. More recently, IDL has been made available for the full ESO community.

An often heard objection to the use of IDL is its commercial nature, i.e. license fees have to be paid. However, there are several aspects to consider: the costs are quite moderate in comparison with the other computer system related costs (hardware, operating system, utility software), and buying an IDL license opens up an enormous amount of astronomical application software which has been developed over the years, representing an effort of hundreds of person-years; and, IDL has the potential of considerable cost savings by increasing the efficiency and productivity of software developers.

IDL is available on the ESO local area network to staff and visiting astronomers. Copies of the most important astronomical libraries are available locally, or they can be downloaded through the Internet.

Application Programme Development

Experience shows that it is very difficult to incorporate contributed software written by astronomers for their own research into systems like IRAF and MIDS. The reason for this is that the requirements for adherence to standards, level of documentation, testing, quality control and configuration management have to be so strict that researchers are unwilling and/or incapable of following them. On the other hand, such requirements are absolutely necessary in order to keep the system transportable, maintainable, and reliable. As a consequence, we can hardly ever incorporate "contributed" software into MIDAS or IRAF as is. Instead, we have to take the research software (which may or may not run under one of those systems), re-design and re-code it in the proper manner, and then incorporate it into the target system.

We have found this process to be very smooth when IDL is used. IDL allows quick and easy, step-by-step prototyping by the researcher or by the s/w developer. A large function library encourages modularity. The code is eminently readable, making the process of re-casting it into a different target language very easy. In fact, IDL can be considered a powerful detailed level design language with the advantage that it actually executes.

Of course, there is a penalty for all this: although IDL is very fast as far as interactive languages go, and it can actually be pre-compiled to speed up the execution, it cannot compete with Fortran or C when it comes to pure number crunching involving a large number of
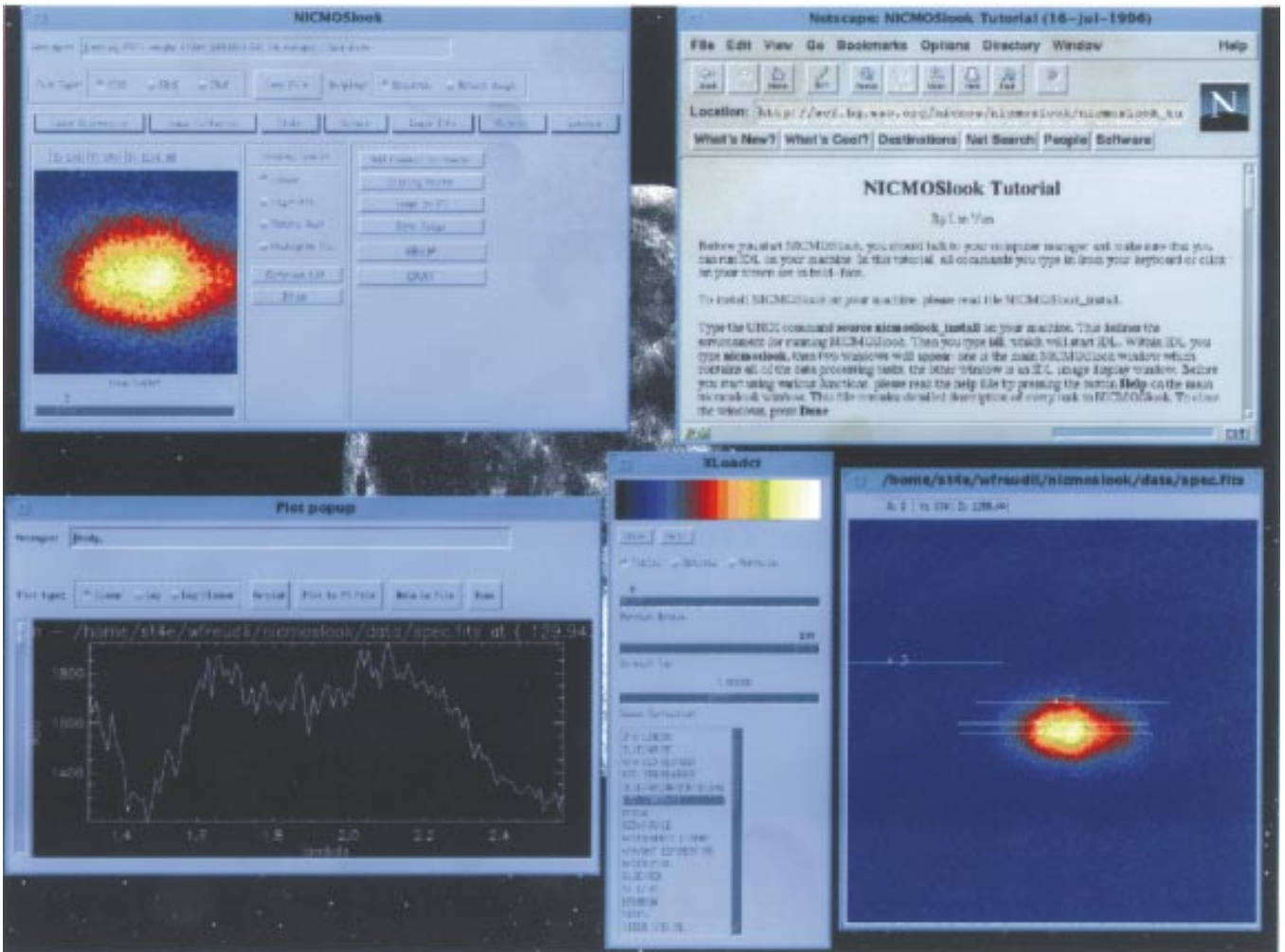
Figure 1: NICMOSLook screen shot, showing the graphical user interface, display panels, and on-line documentation implemented in html.

large data sets. This is one of the reasons why it is necessary to re-code application programmes. Another reason is the fact that sophisticated data analysis systems offer enormous functionality which, however, can only be used if the application programme is compatible within the system.

Example: NICMOSLook and NICMOS grism extraction pipeline

A recent example of the use of IDL at ESO/ST-ECF is the development of a pipeline for the extraction and calibration of NICMOS grism spectra. NICMOS (Near Infrared Camera and Multi-Object Spectrometer) was installed in the Hubble Space Telescope (HST) during the 2nd Servicing Mission in February 1997. Software is being developed at the Space Telescope Science Institute (STScI) to ensure the proper calibration of data obtained with NICMOS. Software is also being developed within the NICMOS Investigation Definition Team (IDT, Principal Investigator Roger Thompson) to support the IDT science programme.

Among the operating modes of NICMOS is a grism spectrum mode. While this mode is not being considered one of

the primary operating modes, it nonetheless provides a very important scientific capability and has the highest potential for serendipitous discovery.

NicmosLook is an interactive tool to extract spectra for individual sources on a NICMOS grism image. A matching direct image of the same field is used to determine the location of sources and therefore the zero point for the wavelength scale for each individual spectrum.

The tool is implemented as an IDL widget. After loading both the grism image and the direct image, both images can individually be displayed and manipulated. Objects can be located by the user using the cursor, by supplying object co-ordinates or by supplying a threshold for an automatic object search.

Spectra can be extracted for any user-selected object or for all objects at once. The dispersion and distortion spectra are read from a database. There are several options for the spatial weighting of the spectra. The output of NICMOSLook is wavelength calibrated spectra, which are plotted on the screen and can be saved as postscript files or ASCII data files (Fig. 1).

NICMOSLook is a convenient tool to extract spectra from small numbers of grism images. However, the routine extraction of spectra from large numbers of NICMOS grism images requires a tool which extracts spectra without human intervention. Establishing the requirements for such a software package, we quickly found that we did not have the resources to write it from scratch. On the other hand, it soon became evident that most of the individual steps needed for the reduction have been coded before in one way or the other. The challenge was to harvest as much as possible of this available functionality and to combine it into a package which would meet the requirements of minimum human intervention and operational resilience.

The resulting IDL programme, calnicc, uses the SExtractor programme to identify objects and classifies them as stars or galaxies using a neural network approach (Bertin & Arnouts, 1996). The wavelength calibration of the extracted spectra is performed using the position of the objects as determined by the SExtractor programme as the zero point, and using parameterised dispersion relations. After background subtraction and extractions of the spectra, they
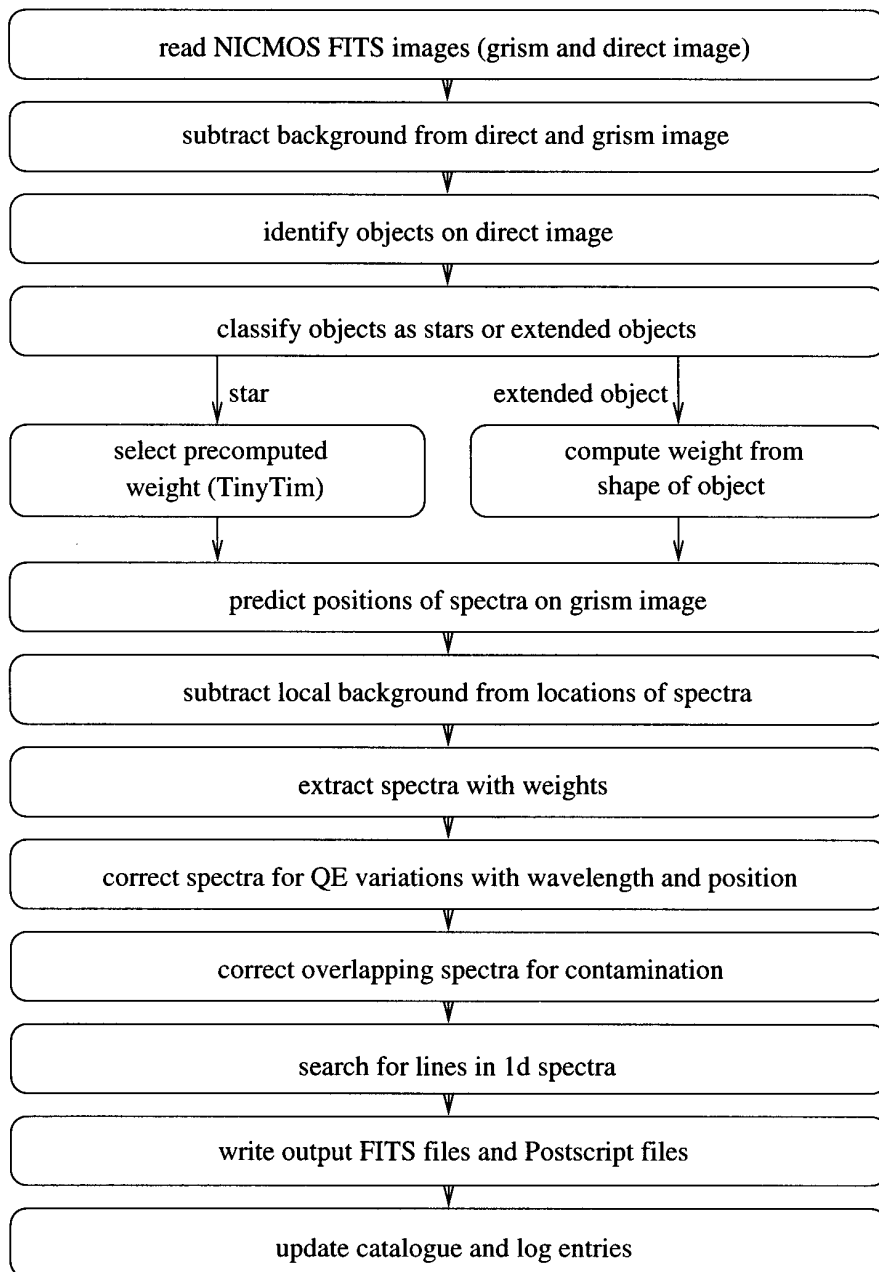
```
read NICMOS FITS images (grism and direct image)
                        ↓
subtract background from direct and grism image
                        ↓
identify objects on direct image
                        ↓
classify objects as stars or extended objects
          ↓ star              extended object ↓
select precomputed          compute weight from
weight (TinyTim)            shape of object
          ↓                              ↓
predict positions of spectra on grism image
                        ↓
subtract local background from locations of spectra
                        ↓
extract spectra with weights
                        ↓
correct spectra for QE variations with wavelength and position
                        ↓
correct overlapping spectra for contamination
                        ↓
search for lines in 1d spectra
                        ↓
write output FITS files and Postscript files
                        ↓
update catalogue and log entries
```

Figure 2: NICMOS grism spectrum extraction pipeline processing steps.

are corrected for the wavelength dependence of the quantum efficiency of the detector. The flux scale is then computed using the standard NICMOS flux calibration data. The extracted spectra are automatically checked for artifacts from bad data and contamination from nearby objects. An attempt is made to correct spectra for the contamination. All spectra are automatically searched for emission and absorption lines. In addition, the continuum emission is determined. The final data product consists of binary FITS tables with the spectra, error estimates, object parameters derived from the direct imaging and details of the spectrum extraction process (Fig. 2).

IDL allowed the quick merging of components written by different authors, including components written in different languages. This was an important aspect of our project as the object extraction software is only available in C.

The software will be used for the extraction and calibration of NICMOS grism spectra during the orbital verification and initial science programme of NICMOS. We expect to have to do considerable changes in response to in-orbit performance. Once the software is stable and the data volume increases, we are in a position to speed up the processing by re-casting the software as an IRAF task and executing it in the HST calibration pipeline system, using the existing software as the detailed design.

Conclusions

IDL in many ways combines the advantages of an interactive and a compiled language. The speed penalty mentioned above is not significant considering the speeds afforded today even by low-cost hardware.

IDL is a traditional language as far as syntax and usage mode is concerned. It is thus usable by astronomers, which is to say by users with a minimum computer science background, who do not easily adapt to state-of-the-art object oriented paradigms. However, recent add-ons to IDL make it possible for the professional programmer to use object-oriented concepts.

By using IDL, it was possible to develop a software package consisting of about 5000 lines of IDL and C code in about 3 person-months. The total effort, including requirements definition, reviewing, testing, documentation, and integration was about twice that. This fast-track development, which was made possible by the use of IDL, enabled us to produce the required software on time. The portability of the code made delivering it to the NICMOS IDT and the STScI very easy. It was mainly due to the demonstrated availability and capability of this software package that a major NICMOS grism survey proposal was approved by the HST Time Allocation Committee.

Reference

Bertin, E., Arnouts, S., 1995, *A&A Suppl.*, **117**, 393.

Rudi Albrecht
e-mail: ralbrech@eso.org