

(NASA-CR-189385) DEVELOPMENT OF A  
DATA REDUCTION EXPERT ASSISTANT  
Final Report, Jul. 1993 - Jun. 1994  
(Space Telescope Science Inst.)  
29 p

N95-17328

Unclas

G3/82 0033948

**Final Report for**  
**Development of a Data Reduction Expert Assistant**  
**USRA Contract 550-74**  
**July 1993 - June 1994**  
**Dr. Glenn E. Miller, Principal Investigator**

100  
 100  
 200  
 33948  
 P-29

This report covers the final year (July 1993-June 1994) for USRA Contract 550-74 "Development of a Data Reduction Expert Assistant".

### Summary

During this year we developed and fielded the final releases of the Data Reduction Expert Assistant (DRACO). The system was successfully applied to two astronomical research projects. The first was the removal of cosmic ray artifacts from Hubble Space Telescope (HST) Wide Field/Planetary Camera data. The second was the reduction and calibration of low-dispersion CCD spectra taken from a ground-based telescope. This has validated our basic approach and demonstrated the applicability of this technology.

This work has been made available to the scientific community in two ways. First, we have published the work in the scientific literature and presented papers at relevant conferences. Secondly, we have made the entire system (including documentation and source code) available to the community via the World Wide Web.

Although funding by NASA has expired, the PI continues to maintain the system and interact with members of the community interested in using DRACO.

### Report

Data reduction is a task often consigned to postdocs, graduate students and technical assistants. It is time consuming and usually requires a substantial amount of knowledge about mundane things such as data analysis software packages, data formats, tape drives, computer systems and the like. Time spent by the scientist on these details is time which is not spent on physical interpretation and modeling. Relieving the scientist of this drudgery is DRACO'S primary goal.

There is much interest in the NASA community on data analysis and reduction problems. The majority of the work tends to focus on facilitating interactive analysis. Although there can be no doubt that interactive tools are very important, we contend that high volumes of data demand less interactive, more automated systems for the bulk of the reduction and analysis process. Traditional "batch" processing systems have been very labor intensive to build and costly to maintain. DRACO provides a new and different way to automate the process. DRACO builds on the foundation of analysis systems such as IDL, IRAF, etc. The scientist "tells" DRACO what to do in general terms by defining a set of file types and operations in a target analysis system. DRACO then examines the actual data at hand and applies these operations to the data. Not only does this automate the reduction process,

but it frees the scientist from much of the data management and dramatically lowers the chance of mistakes (e.g. a typo in a calibration file name). It also enables the scientist to explore the data more easily since the details are handled by DRACO.

Traditionally, analysis procedures have been encoded in a computer programming language such as Fortran, C or "scripts" in operating system or data analysis system languages. The shortcomings to this approach are well known. A more recent alternative was the development of expert system technology. However, expert system technology usually requires an expert who is adept at both the domain knowledge (in this case data analysis) and in computer science. In DRACO, we have deliberately taken an alternative approach. The DRACO system requires no more knowledge about computer science or data analysis than the scientist would need without DRACO. The scientist describes the basic operations in terms which are familiar and natural to this process. DRACO then applies these operations to the data at hand, and performs consistency and monitoring checks.

DRACO has been applied to two separate projects: The first was the removal of cosmic ray artifacts from HST Wide Field/Planetary Camera data for the Medium-Deep Survey Key Project (Griffiths and Ratnatunga). The second was the calibration and reduction of low-dispersion CCD spectra of late-type stars from ground-based telescopes (MacConnell and Roberts).

DRACO'S capabilities include:

- gathers information about the actual data (from header information in the data files)
- develops a plan for data reduction based on the user's goals and actual properties of the data
- produces a command language script to perform the reduction in a target data reduction system (STSDAS in the first project, IRAF in the second)
- performs checks on the data for consistency and quality
- produces an inventory of data

This work has validated the basic principles of the project - that it is possible to provide an intelligent data analysis and reduction assistant with the framework of the current analysis tools used by scientists. It should be noted that data reduction systems such as IRAF have recently begun work on "open" interfaces so that these systems can be driven by external tools (such as DRACO) and used in a more modular fashion.

## **Distribution**

One of the primary goals of this project is to make the DRACO software and documentation available to NASA and scientific communities. After examining a number of alternatives (Cosmic, ftp, Gopher, WWW and others), we selected the World Wide Web (WWW) as the medium for distribution. Three principle factors guided this choice. First, WWW gives the ability to distribute any type of computer product including documentation, graphics, source code and executables. Second, WWW can be easily accessed by a variety of clients including Mosaic, Lynx and Emacs which results coverage of all major computers (Unix, VMS, Macintosh, IBM and more). Finally, use of WWW in the Internet community has grown tremendously in the last year and has become the medium of choice for many.

The Universal Resource Locator (URL) for the DRACO server is:

<http://lor.stsci.edu/DRACO/DRACO.html>

In order to make this information available to the widest possible audience, we have "registered" with two well-known Web servers and they now include references to DRACO. These include the AISR-sponsored Software Support Laboratory at the University of Colorado and the Space Telescope Science Institute's Astronomical Internet Resources.

A copy of some of the Web pages are attached to this report.

## **Publications**

Throughout the project, we communicated the ideas and results via publications and presentations at scientific conferences. Miller gave an invited presentation at the 1992 ESO conference on Astronomy From Large Databases II. Yen was invited to participate in the American Association for Artificial Intelligence's Symposium on Intelligent Scientific Computation. The results of this project were most recently presented at the Third Annual Conference on Astronomical Data Analysis and Software Systems.

(A copy of the papers are attached to this report.)

Miller, G. and Yen, F. (1993). DRACO: An Expert Assistant for Data Reduction and Analysis. Third Annual Conference on Astronomical Data Analysis and Software Systems (ADASS), proceedings published by the Astronomical Society of the Pacific, in press.

Miller, G. (1992). The Data Reduction Expert Assistant. Astronomy from Large Databases II, Hagenau, France, ESO.

Yen, F. (1992). DRACO - A Data Reduction Expert Assistant. AAAI 1992 Fall Symposium on Intelligent Scientific Computation.

ADASS paper

ALDB paper

Yen

also some web pages

# The Data Reduction Expert Assistant

Glenn E. Miller

Space Telescope Science Institute  
3700 San Martin Dr.  
Baltimore, MD 21218 USA  
miller@stsci.edu

## ABSTRACT

Increased access to very large astronomical databases, the use of large format detectors and other developments in observational astronomy have the potential to overwhelm the capacity of most astronomers to analyze data unless new approaches to data reduction are found. This paper reports the initial progress in creating an expert system to assist in the reduction of scientific data. This system, called Draco, takes on much of the mechanics of data reduction, allowing the astronomer to spend more time understanding the physical nature of the data. Draco works in conjunction with existing data analysis systems such as STSDAS/IRAF and is designed to be extensible to new data reduction tasks.

## 1. Introduction

The task of data reduction presents severe obstacles to an astronomer: The volume of data may require much tedious work that is susceptible to errors (e.g., the flat-fielding and bias correction of a few dozen digital images can take several day's time and it is easy to accidentally apply the wrong calibrations to some of the images). Management of the data reduction process may require tracking tens or hundreds of files through many different steps. Limitations of disk space may constrain the order of the reduction (e.g., there may be room for only a few images on disk at any one time). The quality of each reduction step should be evaluated (e.g., stability of internal calibrations, or number of cosmic ray events). Often the entire reduction process must be repeated several times with improved calibration data or improved reduction algorithms. The chosen data analysis system must be mastered sufficiently by the scientist to correctly perform the reduction.

These are significant problems that inhibit progress by forcing the scientist to expend time and effort on the mechanics of reduction rather than understanding the physical nature of the data. The growing availability of large astronomical databases and increased use of large format detectors threatens to magnify these problems to an overwhelming degree. Other scientific disciplines share this concern, e.g., NASA's Earth Observing System (EOS) will collect many hundreds of megabytes of data each day.

---

Invited paper to appear in the Proceedings of Astronomy from Large Databases II, 14-16 September, 1992, Haguenu, France.

We are developing Draco<sup>19</sup>, which is an expert system tool for the management and reduction of data.<sup>†</sup> Draco builds on the foundation of existing data analysis systems such as STSDAS/IRAF. Draco gathers information about the available data (typically from header information in the data files), develops a plan for data reduction based on a template supplied by the astronomer and translates the plan into explicit reduction commands. An important feature of Draco is its generality and extensibility - new types of data analysis tasks or additional data analysis systems can easily be added without modifying existing software. This work is an extension of a successful prototype system for the calibration of CCD images developed by Johnston<sup>10</sup>.

Draco's role in the data reduction process is modeled after a human assistant (at the level of an advanced undergraduate or beginning graduate student). With a human assistant, the astronomer describes the reduction process, demonstrates it on some data and notes various steps to be checked during the reduction (e.g., typical number of cosmic ray hits per pixel per second, average variation in flat fields, etc.). Once trained, the human assistant will reliably perform the reductions on new data sets and call attention to any unusual situations (e.g., missing calibration files, abnormally large number of bad pixels, etc.). A human assistant is (usually) able to adapt to simple changes in the reduction process with little or no additional training (e.g., using new calibration data or adjusting parameters within an algorithm).

Our goals are for Draco to accurately perform the reductions according to the description provided by the user, to alert the user to potential problems in the reduction, and to be readily extensible to new types of data reduction and data analysis systems. (The analogy of Draco to a human assistant should not be carried to an extreme. Unlike a human assistant, Draco will not learn from its mistakes nor will it discover new information. Section 2 mentions some programs that have some of these capabilities.) This automation frees the scientist from much of the drudgery in data reduction and should allow more time for the exploration and modelling of data.

This paper is a progress report on our initial work on Draco. Section 2 discusses related work, while Section 3 presents the design and implementation of Draco. Section 4 describes the use of the first version of the software. A final section summarizes our investigation and outlines our future work.

## **2. Previous Work on Expert Scientific Assistants**

Our current investigation ensues from the work of Johnston<sup>10</sup> who developed the Data Analysis Assistant (DAA), a prototype system for data reduction. The problem addressed by this work was Charge Coupled Device (CCD) calibration since it is a common data

---

<sup>†</sup> For the purposes of this paper, there is no need to distinguish between the terms data "reduction", "calibration" and "analysis" since Draco can provide assistance for all.

reduction task and it provided a suitable test case for the concept. CCD calibration consisted of four steps:

1. Extraction of a subimage representing valid data
2. Bias subtraction
3. Dark current subtraction
4. Flat fielding

The first two steps depend only on the characteristics of the detector and instrument mode. The last two steps are more involved since dark and flat calibration images are usually performed periodically through an observing run and therefore must be identified, matched and averaged to the appropriate science images.

Information about data and data reduction was organized in three knowledge bases: data, instrument modes and tasks. The data knowledge base described the astronomer's actual data (e.g., darks, flats, science). The instrument knowledge base held information such as the bias value or location of bad pixels. The task knowledge base recorded information about the data reduction process. Tasks were divided into two types: primitive and compound. Primitive tasks were those which could be implemented with a single command (or simple series of commands) in a specific data analysis system. Compound tasks represented the higher level operations.

To use the system, the astronomer first supplied the DAA with a description of all relevant images, i.e., dark, flat-field and science. (This information was available in the image header information, but a means to read headers was not implemented in the DAA.) The DAA generated a plan by using a set of forward-chaining production rules to associate flat fields and dark images with the proper science image, check for missing calibration files and expand compound tasks into primitive operations. Once a plan was complete, the user selected one of the two target languages, STSDAS or MIDAS and the general plan was converted to an explicit script of image processing commands in the chosen language. The user then executed the script file on an image analysis workstation.

The DAA was implemented in the Lisp-based Knowledge Engineering Environment (KEE) expert system shell (a product of Intellicorp, Inc.) on a Symbolics Lisp workstation. The DAA used KEE's rule and object systems as well as KEE's graphical user interface. Portions of the DAA were written in Lisp.

The DAA demonstrated two important concepts. First, it was possible to separate the system's knowledge of data reduction from the control strategy information. This allowed the system to accommodate new types of data or new data analysis functions without massive changes to the control software, as might be the case in a system written in a procedural language such as Fortran or C, or operating system command languages such as Unix shell scripts. For example, the format of individual reduction system commands was attached to the primitive task objects. This facilitates using existing information in new reduction procedures and provides a straightforward way to add new reduction systems. Second, it proved the feasibility of constructing a data reduction plan



on the basis of a generalized knowledge of data reduction, specific knowledge of commands in a particular data analysis system and knowledge of the actual data. This yielded a very general framework which could readily accommodate new types of data reduction.

To conclude this section, I briefly mention some other work on expert scientific assistants. An expert assistant for the preparation of Hubble Space Telescope (HST) observing proposals was developed by Adorf and di Serego Alighieri<sup>2</sup>. A system which planned experiments in molecular genetics was originally developed by Stefik<sup>17</sup> and recent work is reported in Noordewier and Travis<sup>15</sup>. Buchanan, et al.<sup>3</sup> developed a system which controlled particle accelerator experimental parameters.

Abelson, et al.<sup>1</sup> review their work on tools which prepare numerical experiments from high-level specifications of physical models. For example, the Bifurcation Interpreter and KAM programs investigate problems in dynamics by identifying interesting features, performing additional calculations of such features and reporting the results to the scientist. Keller and Rimon<sup>11</sup> are developing a knowledge-based software environment to support the development of scientific models. Fabiano, Bettini and Chin<sup>5</sup> describe a program which assists users in choosing parameters for complex quantum chemistry programs. Lucks and Gladwell<sup>12</sup> describe a framework for representing and reasoning with expert knowledge that has been used to advise in the selection of differential equation software from numerical subroutine libraries and for the identification of parallel science observations for the Hubble Space Telescope.

Artificial intelligence technology has often been applied to classification problems. Fayyad, et al.<sup>6</sup> are applying machine learning techniques to identify objects in the second Palomar Sky Survey. Cheeseman, et al.<sup>4</sup> used a program to discover new classes of objects in IRAS spectra. Thonnat and Clement<sup>18</sup> developed an expert system to control the processing of galactic images and the extraction of parameters such as size, ellipticity and luminosity profile. The results from this system are used by another expert system which classifies the galaxies.

These are just a few of the hundreds of scientific applications of expert systems and artificial intelligence (see Murtagh and Heck<sup>14</sup> for more references). For the most recent developments, the reader should consult the proceedings of conferences such as the *Conference on Artificial Intelligence Applications* and *Innovative Applications of Artificial Intelligence*, *National Conference on Artificial Intelligence*.

### **3. Design and Implementation**

#### **3.1 Scientific Data Analysis Systems**

In the last decade, scientific data analysis systems have grown in number and functionality. Widely used astronomical data analysis systems include the Interactive Reduction and Analysis Facility (IRAF) developed at NOAO, the Space Telescope Science Data Analysis System (STSDAS) developed at the STScI, the Astronomical

Image Processing System (AIPS) developed at NRAO and the Munich Interactive Data Analysis System (MIDAS) developed at ESO. The Interactive Data Language (IDL) is used in astronomy and other disciplines such as climate research. (See Hanisch<sup>9</sup> for a review).

The philosophy of these systems is usually similar to the philosophy of most computer operating systems (e.g., Unix, VMS): there is a command language (CL) which serves as the user interface in a "command/prompt" mode. The CL executes either single commands interactively, or scripts (procedures) of commands (generally with a choice of interactive or batch execution). CL commands reduce to the execution of modular operators which work on standardized types of data files. Two major strengths of this philosophy are:

- Flexibility for the user - individual commands can be chained (or "pipelined") to construct powerful, customized procedures.
- Facilitate development - there is a well-defined (though not usually simple) process for adding new modules to a system. Thus many programmers and scientists may independently contribute to the growth of a system.

The success of this approach is shown by their growth. Data analysis systems developed at one institution have been adopted as the standard at many universities and research institutes (e.g., IRAF) and systems developed for a particular wavelength range have been adapted to serve multiple spectral domains (e.g., AIPS). Packages developed independently have been incorporated into larger systems (e.g., the incorporation of DAOPHOT and Feigelson's Survival Analysis into IRAF/STSDAS).

However, this approach has some serious drawbacks (which are well known to users). Learning a system is not easy and even experts cannot be familiar with all parts of the system. Within a system, programs authored by different people may have different definitions or naming conventions which adds to the confusion of a novice user. To compound the problem, some users have to learn more than one system depending on where or how they obtain their data. This is especially true for multi-spectral observations which are often taken at several different observatories.

Although many commands are conceptually simple (e.g., subtract two images), some commands are quite sophisticated and require the specification of many parameters (some of which are interdependent) to get the correct results. If a complex procedure does not perform the desired tasks, the user is faced with the daunting possibilities of either modifying a large, complex program or writing a new program. Either choice can take many weeks or months.

It has also proven very difficult to capture and make available expert knowledge. Users can obtain assistance from manuals (often quite large, hard to use and harder to keep up to date), on-line help (often of little use to the non-expert and surprisingly hard to maintain), or by befriending the local expert on a particular topic.

Some of these problems will be lessened by efforts within scientific disciplines to adopt one analysis system as a standard (e.g., IRAF as a standard for astronomical data analysis), yet these effort cannot solve all the difficulties listed above. In particular, standardization will not lessen the data management problem nor the time needed to learn the system (including new modules as they are added). Researchers working in multi-spectral or interdisciplinary domains are likely to be faced with an amalgam of analysis systems for years to come.

Several groups are investigating solutions to these problems. A graphical user interface based on X-windows is being added to IRAF and the development of a hypertext help system has been proposed. A number of groups are exploring visualization systems which allow a scientist to interact with data in a more intuitive way in order to facilitate communication of results, browsing and discovery of new features<sup>7,16</sup>. However, as visualization systems are (by design) highly interactive, they will not lessen the data management problems addressed by Draco.

### *3.2 Draco - Data Reduction Expert Assistant*

The present work demonstrates one approach to solving some of the above problems in scientific data analysis. Draco is an expert assistant which does the following:

- gathers information about the actual data (from header information in the data files).
- develops a plan for data reduction based on the user's goals and actual properties of the data
- produces a command language script to perform the reduction in a specific data analysis system
- performs checks on the data for consistency and quality

By producing a command script in the language of a data analysis system, Draco builds on the foundation of these systems, rather than creating yet another analysis system.

For Draco we have adopted a different design from the rule-based approach of Johnston's DAA. Draco can be likened to an algebra or very simple programming language. The user defines a set of primitive operations and combines them to perform reduction procedures. Draco does not know anything about the semantics of the primitives, but it does know which combinations are syntactically valid. This design was motivated by the realities of cutting-edge scientific work: In our discussions with colleagues it became clear that an important characteristic of scientific data analysis is that experts often disagree on how to best perform reductions. For example, there are several algorithms for removing cosmic ray artifacts from HST Wide Field/Planetary Camera (WF/PC) data<sup>8,13</sup> and the proper choice of algorithm depends on the type of science to be extracted from the data. The variety of techniques available to correct for the HST's spherical aberration provides another example. Since the "rules" for data reduction vary from user to user (and even week to week for a single user), it did not appear feasible to us to collect this information as a set of expert system production rules. The alternative provided by Draco allows the user to specify the reduction steps at an abstract level.

Figure 1 illustrates the organization of information in Draco, using the removal of cosmic ray noise from WF/PC images as an example. *Primitives* represent basic data analysis operations and each primitive has one or more *implementations* which are commands or programs which accomplish the primitive. A *procedure* is a template for data reduction built from primitives.

The command

```
(make-script remove-CR-noise :input "mydir")
```

causes Draco to generate a *script* for removing cosmic ray artifacts using the procedure `remove-CR-noise`. Science image files are taken from the directory `mydir`. The script is then executed and produces a log file which records the reduction steps.

A procedure for cosmic ray removal is defined by:

```
(define-procedure
  :name          remove-CR-noise
  :documentation "procedure for removing CR noise"
  :primitives    (find-like-images CR-removal))
```

The operations defined by the primitives are executed in the order specified, that is, Draco currently implements a pipeline for reductions.

The primitive `CR-removal` is defined as follows:

```
(define-primitive
  :name          CR-removal
  :documentation "remove cosmic ray noise"
  :input         image
  :output        image
  :reconcile     :conjunctive
  :concrete      (STSDAS-CR-removal))
```

The `:input` and `:output` parameters specify the data types that this primitive reads and writes. (Data types are an abstraction which are realized in terms of file types, e.g., in SDAS, images are stored in Generic Edited Information Set files.) The `:reconcile` parameter defines the action when multiple inputs are encountered. The value of `:conjunctive` in the example indicates that the primitive should treat the data as a single input (i.e., multiple images are processed as a unit to determine the cosmic ray hits). An alternative value for this parameter is `:distributive` which causes the primitive to process each input separately, i.e., to iterate over its inputs. The `:concrete` keyword names the implementation which is defined as:

```

(define-implementation
:name                STSDAS-CR-removal
:documentation      "STSDAS cosmic ray removal function"
:package            IRAF
:initialize-once    ("stsdas" "wfpc")
:syntax             "combine -in -out option=\"crreject\" usedqf=\"yes\""

```

The syntax parameter records the format of specific procedures and commands such as the STSDAS "combine" procedure in our example. The ~in and ~out tokens are placeholders for the input and output file lists, respectively. Many analysis commands have initializations which must be invoked prior to execution. In this example, the "stsdas" and "wfpc" commands must be issued to IRAF to select the proper packages which contain the combine procedure. The :initialize-once keyword's actions will be performed before the first instance of this command. The optional parameter :initialize (not used in this example) is used when commands must be invoked with each use.

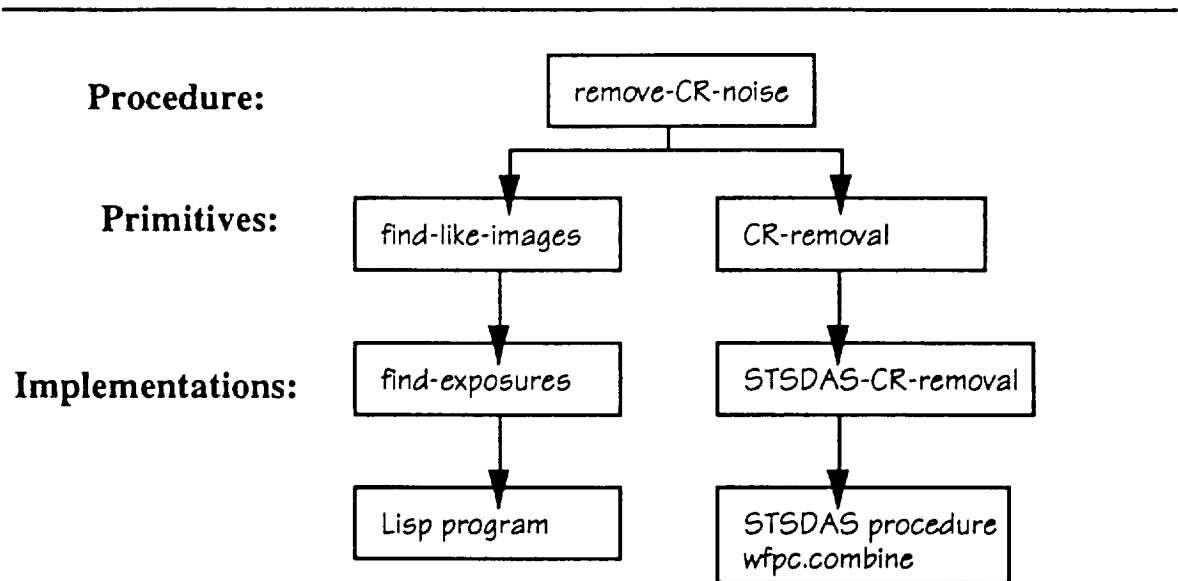


Figure 1 - Data structures for the removal of cosmic ray noise from WF/PC images.

The definition of Draco's structures such as primitives, implementations and data types creates a base of knowledge which can be applied to new data reduction problems. A new analysis system can be added to Draco by defining the appropriate implementations and file types. It is common for astronomers to write operating system command language scripts (e.g., Unix Shell or VMS DCL) to reduce data. The advantages of the Draco-generated scripts are clear: Draco provides a higher level of abstraction and handles many lower level details for the user. It is usually very difficult to modify custom command language scripts to different reduction tasks whereas Draco facilitates reuse of its component data structures.

In addition to the ability to create scripts, Draco provides a tool to inventory the files within a directory. Rather than relying solely on file extension conventions, the inventory uses file recognizers which (generally) open and read files to determine their format and contents (e.g., particular FITS keywords). This tool is useful for the management of the large number of files involved in data reduction and in providing quality checks on the data.

Draco is written in mostly in Common Lisp with some of the file recognition functions in C and Bourne shell code. Object-oriented programming is used to represent and operate the data structures (using the Common Lisp Object System). The DAA was prototyped on a special-purpose workstation (Lisp machine) and used a costly expert system shell. Since then, both workstation and software technology has evolved to the point that Draco is implemented on the same class of workstation commonly used for data analysis (e.g., Sun Sparcstation) and only requires an inexpensive Lisp environment. This makes possible the distribution of Draco to the scientific community.

#### **4. Experience with Draco**

In discussions with research groups at the STScI (along with our own experience in astronomical research) we found a number of common problems:

- The data management problem is severe. Many astronomers today have more data than can quickly be reduced and analyzed. Some data may wait months or years before the scientist can hire a postdoc or graduate assistant to reduce and analyze it.
- Despite best efforts to calibrate data only once, there is a continuing need to recalibrate data. This is true even if an observatory provides calibrated data (as does STScI). Often this is because the best calibration data are not available until well after the observations are taken.
- The removal of instrumental signatures from data is seldom routine, especially when state-of-the-art detectors are involved or when striving for quantitative results or high accuracy. The scientist therefore needs to be able to experiment with different parameters in the reduction algorithms as well as different algorithms.
- There is an inertia to remain within the analysis systems, computer operating systems and programming languages with which the astronomer is comfortable, despite serious shortcomings with these systems. Part of this is due to a justifiable skepticism that a new system will actually be better. Another major factor is that a scientist must usually concentrate on research and may have little time to provide tools which are useful to others.

An important part of our project plan is the early involvement of scientists in the use and evaluation of Draco. We sought astronomers who were faced with a large amount of data to reduce and whose projects were such that even early versions of Draco would reward them for their investment of time in the project. Detailed discussions were held with three research groups at the STScI. Our first users are R. Griffiths and K. Ratnatunga of the HST Medium-Deep Survey (MDS) Key Project. HST Key Projects are those which were identified by the astronomical community as having high scientific importance and involving a large amount of HST observing time. Data is shared by many astronomers

with different interests. The scientific objectives of the Medium-Deep Survey include serendipitous discoveries, observations of rare objects, morphology and distribution of faint galaxies, active nuclei of distant galaxies, galactic structure, and distant solar-system objects. The observing program involves obtaining image data with WF/PC and Faint Object Camera in parallel with other HST observing programs. Since removal of cosmic ray artifacts from WF/PC data is an important step in the reduction process, we adopted it as the first sample problem for Draco. A small set of MDS files has been processed with Draco. Further use of Draco by the MDS is on hold at this time as they are revising their basic data reduction procedures in order to quantitatively account for measurement and reduction errors. This revision may cause them to write new software or to modify existing packages.

Earlier work (c.f. Section 2) has shown that it is possible to build software which provides expert assistance for scientific tasks. In our project we are trying to bring the expert assistant out of the prototype stage and into the hands of researchers. An open question is whether there is a sufficient audience for this type of tool. Some users either do not have much data or have less stringent analysis requirements and can be satisfied with existing analysis tools. Other users prefer to write their own software for analysis in order to be sure the reductions are done correctly (or must do so because no suitable software exists). Draco is aimed primarily at users between these two types.

## 5. Summary

This paper reports our initial efforts in developing an expert assistant for the reduction of scientific data. The first version of the software has been used to manage the removal of cosmic ray artifacts from HST Medium-Deep survey WF/PC data using an STSDAS procedure. Our approach holds promise for addressing several critical problems in dealing with large amounts of data. Although astronomical data reduction has been the focus of our initial work, the Draco system is directly applicable to many other fields of science including space physics and earth sciences.

We plan to implement several more versions of Draco over the coming year, each with increasing capability and addressing more involved reduction tasks. It will then be made available to the community. Possibilities for future work include: Adding a graphical user interface for defining and editing Draco entities would make its use more intuitive. The ability for Draco to monitor the execution of procedures and provide status and diagnostic information would be helpful. Adding a means for procedures (scripts) to branch or iterate in a general way might be useful. A script would need to examine the output of a reduction program in order to determine the next implementation to invoke, and possibly, some of the parameters with which to invoke it. We have deferred implementing such a feature since it is not clear if such a step can currently be automated for most reductions. Consider for example an iterative deconvolution algorithm. The number of iterations is usually determined by the astronomer's visual inspection since there exists no image analysis program which can determine the "best" number of iterations for an image (or at least no program which is generally accepted by astronomers). Even the current version of Draco can provide substantial assistance for such a task. Draco could create a number

of deconvolved images (e.g., 25, 30, ... iterations) and run some statistical analysis modules on the images. The astronomer would examine this output to select the desired images.

## Acknowledgements

Along with the author, Felix Yen, Mark Johnston and Robert Hanisch are investigators on the Draco project. We thank Ron Gilliland, Richard Griffiths, Keith Horne, Phil Martel and Kavan Ratnatunga for discussions about data reduction and their thoughtful comments on the design and development of Draco. This work is supported by NASA's Astrophysics Information Systems Research Program through the Center of Excellence in Space Data and Information Sciences by a contract with the Space Telescope Science Institute which is operated by AURA for NASA.

## References

1. Abelson, H., Esienberg, M., Halfant, M., Katzenelson, J., Sacks, E., Sussman, G., Wisdom, J., Yip, K., 1989, "Intelligence in Scientific Computing", *Communications of the ACM*, **32**, 546.
2. Adorf, H.-M., and di Serego Alighieri, S., 1989, "An Expert Assistant Supporting Hubble Space Telescope Proposal Preparation," in *Data Analysis in Astronomy III*, ed. V. DiGesù, et al., (New York: Plenum Press), 225.
3. Buchanan, B., Sullivan, J., Cheng, T. and Clearwater, S., 1988, "Simulation-Assisted Inductive Learning", in *Proceedings of the Seventh National Conference on Artificial Intelligence* (San Mateo, CA: Morgan Kaufmann), 552.
4. Cheeseman, P., Stutz, J., Self, M., Taylor, W., Goebel, J., Volk, K., and Walker, H., 1989, *Automatic Classification of Spectra from the Infrared Astronomical Satellite (IRAS)*, NASA Reference Publication 1217.
5. Fabiano, A., Bettini, C. and Chin, S., 1991, "A Model Based Assistant for Quantum Chemistry Programs", *Seventh Conference on AI Applications*, (Miami: IEEE), 114.
6. Fayyad, U., Doyle, R., Weir, N., and Djorgovski, S., 1992, "Applying Machine Learning Classification Techniques to Automate Sky object Cataloguing", *International Space Year Conference on Earth and Space Science Information Systems*, in press.
7. Foley, J., 1990, "Scientific Data Visualization Software: Trends and Directions.", *International Journal of Supercomputer Applications*, **4**, 154.
8. Groth, E., 1992, "An Algorithm for Removing Cosmic Rays from Two or More Cosmic Ray Split Exposures", in preparation.
9. Hanisch, R., 1992, "Image Processing, Data Analysis Software and Computer Systems for CCD Data Reduction and Analysis" in *Astronomical CCD Observing and Reduction*, ed. S. Howell (San Francisco: Astronomical Society of the Pacific), 285.
10. Johnston, M., 1987, "An Expert System Approach to Astronomical Data Analysis," *Proceedings of the 1987 Goddard Conference on Space Applications of Artificial Intelligence*.
11. Keller, R. and Rimon, M., 1992, "A Knowledge-based Software Development Environment for Scientific Model-building", *Proceedings of the Seventh Knowledge-Based Software Engineering Conference*, in press.
12. Lucks, M. and Gladwell, I., 1992, "Functional Knowledge Representation in AI Applications for Scientific Computing", *AAAI 1992 Fall Symposium on Intelligent Scientific Computation*, in press.



13. Murtagh, F. and Adorf, H.-M., 1991, "Detecting Cosmic Ray Hits on HST WF/PC Images using Neural Networks and Other Discriminant Analysis Approaches", in *Data Analysis in Astronomy IV*, ed. V. DiGesù, et al., (New York: Plenum Press), 103.
14. Murtagh, F. and Heck, A., 1989, *Knowledge Based Systems in Astronomy*, Lecture Notes in Physics #329, (Berlin: Springer Verlag).
15. Noordewier, M., and Travis, L., 1990, "Case Study of a Knowledge-Based System Which Plans Molecular Genetics Experiments", *1990 Conference on Artificial Intelligence Applications*, (San Mateo, CA: IEEE), 257.
16. Senay, H. and Ignatius, E., 1992, "A Knowledge Based System for Scientific Data Visualization" *Center for Excellence in Space and Data Information Systems Technical Report 92-79*.
17. Stefik, M., 1981, "Planning with Constraints (MOLGEN)", *Artificial Intelligence*, 16, 111.
18. Thonnat, M., and Bijaoui, A., 1989, "Knowledge Based Classification of Galaxies," *Knowledge-Based Systems in Astronomy*, ed. A. Heck and F. Murtagh, (Berlin: Springer-Verlag), 121.
19. Yen, F., 1992, "Draco - A Data Reduction Expert Assistant", *AAAI 1992 Fall Symposium on Intelligent Scientific Computation*, in press.

# DRACO: An Expert Assistant for Data Reduction and Analysis

Glenn Miller and Felix Yen

*Space Telescope Science Institute, 3700 San Martin Dr., Baltimore MD 21218*

## **Abstract.**

The use of large format detectors, increased access to very large astronomical databases, and other developments in observational astronomy have led to the situation where many astronomers are overwhelmed by the reduction and analysis process. This paper reports a novel approach to data reduction and analysis which works in conjunction with existing analysis systems such as STSDAS/IRAF. This system, called DRACO, takes on much of the mechanics of the process, allowing the astronomer to spend more time understanding the physical nature of the data. In developing DRACO we encountered a number of shortcomings of current data analysis systems which hinder the ability to effectively automate routine tasks. We maintain that these difficulties are fundamental (not specific to our approach) and must be addressed by conventional and next-generation analysis systems.

## **1. Introduction**

The task of data reduction presents severe obstacles to an astronomer: The volume of data may require tedious work that is susceptible to errors (e.g., the flat-fielding and bias correction of a few dozen digital images can take several day's time and it is easy to accidentally apply the wrong calibrations to some of the images). Management of the data reduction process may require tracking tens or hundreds of files through many different steps. The quality of each reduction step should be evaluated (e.g., stability of internal calibrations, or abnormally large number of bad pixels). Often the entire reduction process must be repeated several times with improved calibration data or improved reduction algorithms.

These are significant problems that inhibit progress by forcing the scientist to expend time and effort on the mechanics of reduction rather than understanding the physical nature of the data. We have developed DRACO, which is a tool for the management of data reduction and analysis. DRACO builds on the foundation of existing data analysis systems such as STSDAS/IRAF, IDL, MIDAS, etc. DRACO gathers information about the available data, develops a plan for data reduction based on a template supplied by the astronomer, and translates the plan into explicit reduction commands. An important feature of DRACO is its generality and extensibility - new types of data analysis tasks or additional data analysis systems can easily be added without modifying existing software.

This work is an extension of a successful prototype system for the calibration of CCD images developed by Johnston (1987).

## 2. The Draco System

To use DRACO, the astronomer first describes the reduction process by defining three key entities:

- *Procedure* - This is an abstract user program for the reduction, e.g. bias correction, dark removal, field flattening, and extraction of a spectrum
- *Primitive* - These are the abstract data analysis operations used to build a procedure (e.g. bias correction or flat fielding).
- *Implementation* - These implement primitives, usually by invoking an underlying analysis system, e.g. IRAF, STSDAS, etc.

Primitives are the basic building blocks which are used to construct the reduction procedures and insulate the astronomer from many of the details of the underlying analysis systems. Primitives form a library of routines from which new reduction and analyses can be adapted from existing ones. Adding a new analysis package to DRACO consists of creating the appropriate implementations for a set of primitives. Refer to Miller (1992), Yen (1993) for more details on Draco.

Once these entities are defined, the astronomer invokes DRACO's `start` function, specifying the directory containing the data. DRACO gathers information about the data at hand (usually by reading header information in the files), expands the procedure into a reduction plan based on the actual data, creates a command language script in the target analysis system and finally executes the script. The time-consuming reduction takes place without further attention from the astronomer. DRACO logs all steps for later review and calls attention to problems such as missing data or calibration files.

By producing a command script in the language of a data analysis system, DRACO builds on the foundation of these systems, rather than creating yet another analysis system. It is common for astronomers to write operating system command language scripts (e.g., Unix Shell or VMS DCL) to reduce data. The advantages of the DRACO-generated scripts are clear: DRACO provides a higher level of abstraction and handles many lower level details for the user. It is usually very difficult to modify custom command language scripts for different reduction tasks whereas DRACO facilitates reuse of its component data structures.

### 2.1. Experience

DRACO has been applied to two separate astronomical projects with successful results. The first version of DRACO was used to manage the removal of cosmic ray artifacts from a sample of HST WF/PC data for the Medium-Deep Survey Key Project (provided by Griffiths and Ratnatunga). A revised version of DRACO was used to extract spectroscopic data from ground-based telescope data (provided by MacConnell and Roberts), including bias and dark removal, flattening and extraction of the spectra. In this latter case, DRACO managed a significant amount of data: 325Mb in 650 separate files.

## 2.2. Availability

DRACO is an initial system which addresses the issues of automating data reduction and analysis. In order to encourage development of similar ideas and systems, DRACO (including source code and documentation) is available to the community from the Space Telescope Science Institute server ([stsci.edu](http://stsci.edu)) via anonymous ftp, Gopher and World Wide Web.

## 3. Discussion

In the course of this project we had extensive discussions with many research groups. Contrary to the prevailing view that the lack of visualization tools or graphical user interfaces is a major impediment to research, we found that a more serious problem for all groups was the difficulty of managing the data reduction process. DRACO demonstrates a simple and effective way to perform data reduction with less human interaction and fewer errors. Other approaches are being explored such as the Khoros system (Rasure 1991) and commercial systems such as Silicon Graphics Explorer or BBN Cornerstone. We encourage developers of astronomical data analysis systems to incorporate these ideas into future work. From our experience, we can identify a number of general capabilities which would greatly enable astronomical researchers. Users need tools to:

- Describe the data at hand - Directory listings of files are usually the only tool available to describe the data. Users need more powerful tools (e.g. graphical) which understand and display the types of data (flat, bias, comparison spectrum, etc.) and the relationships between the data (e.g. identifying multiple images of the same target).
- Describe reduction process and algorithms at a high level - Users generally have to work at quite a low level, dealing with the specifics of the analysis system, operating system, file types, etc.
- Facilitate experimentation with reduction parameters and different algorithms - Data reduction is an integral part of the scientific process and it is therefore vital to experiment with relevant parameters of the reduction (number of iterations in a restoration algorithm, cosmic ray removal parameters, etc.). If reducing the data just once requires too much time and effort then experimentation becomes impractical and an important aspect of the scientific method is sacrificed.
- Make it possible to resume data reduction/analysis after interruptions.
- Perform data quality checks.
- Provide traceability: Too often data analysis systems do not adequately document what operations were performed on the data. As a minimum, systems must document in the headers what steps were performed, including input data, algorithm, parameters and software version.

A change in the basic philosophy of scientific data analysis systems is needed. Rather than being designed as monolithic systems which are the complete environment for all data reduction and analysis, systems should begin a migration towards "interoperability" where they can be invoked by other systems and software. A client-server technology (communicating via Unix sockets or some other protocol) seen in many other computer applications seems a likely architecture to fulfill this goal.

#### 4. Summary

Management of the data reduction process is an important problem facing astronomers who deal with observational data. The lack of effective data management tools can often be overwhelming. DRACO demonstrates one approach to providing automated assistance in order to free the astronomer to concentrate on scientific issues. DRACO works in concert with existing data analysis systems. We assert that current and future data analysis systems must provide tools for effective automation of procedures and data management.

**Acknowledgments.** We thank Mark Johnston, Bob Hanisch, Ron Gilliland, Richard Griffiths, Keith Horne, Jack MacConnell, Jim Roberts, Kavan Ratnatunga and Phil Martel for discussions about data reduction and their thoughtful comments on the design and development of DRACO. This work is supported by NASA's Astrophysics Information Systems Research Program through CES-DIS by a contract with the Space Telescope Science Institute which is operated by AURA for NASA.

#### References

- Johnston, M. 1987, "An Expert System Approach to Astronomical Data Analysis", Proceedings of the Goddard Conference on Space Applications of Artificial Intelligence, NASA.
- Miller, G. 1992, "The Data Reduction Expert Assistant", Astronomy from Large Databases II, Hagenau, France, ed. A. Heck and F. Murtagh, ESO.
- Rasure, J. and M. Williams 1991, "An Integrated Visual Language and Software Development Environment", Journal of Visual Languages and Computing, 2: 217-246.
- Yen, F. 1993 "Draco Design Document", STScI AP SB Technical Report 1992-07, available from [stsci.edu](http://stsci.edu).

# Draco — A Data Reduction Expert Assistant

Felix Yen

Advance Planning Systems Branch  
Space Telescope Science Institute  
3700 San Martin Drive  
Baltimore, MD 21218  
yen@stsci.edu

## Abstract

Draco is an attempt at partially automate data reduction and analysis. These notes describe our experiences with the project. We discuss the obstacles we have had to face and show how they influenced Draco's design.

Draco is a framework for existing software tools. It uses a minimalist representation system that focuses on the syntax of initializing and invoking programs. There is a working prototype which translates user-defined procedures and primitives into executable scripts appropriate for the user's data. A great deal of emphasis is placed on making sure the user understands what is happening at all times.

## Introduction

### Motivation

The Space Telescope Science Institute<sup>1</sup> is primarily concerned with operating the Hubble Space Telescope (HST). Like data obtained by other telescopes or other measuring devices, raw HST data contains *instrument signatures*. A signature is an instrument-specific artifact that can sometimes be measured, e.g. by turning on an instrument without opening its shutter, and subsequently removed from the raw data in a process known as *calibration*.

Calibrated data may require additional processing before it can be analyzed. For example, one might need to remove cosmic ray noise from the calibrated data before analyzing it. The process which renders raw data useful for analysis is known as *reduction*. This process encompasses calibration, but the dividing line between reduction and analysis is not well-defined partly because there is no best reduction technique for any given data set. Two scientists could conceivably share

---

<sup>1</sup>Operated by the Association of Universities for Research in Astronomy, Inc. for the National Aeronautics and Space Administration (NASA). This work has been supported by NASA under grant number NAS5-31338.

the same set of raw data yet reduce it differently because they are interested in different phenomena, e.g. bright stars versus faint galaxies.

In any case, data reduction is a task often assigned to postdoctoral employees, graduate students, and such. It is time-consuming and it often requires a substantial amount of knowledge about mundane things like tape drives, data analysis software packages, and data formats. Relieving researchers of this drudgery is one of Draco's primary goals.

The sheer volume of data being collected today introduces a second, related goal of equal importance. Some projects may not be able to afford to pay people to manually reduce its data in a timely fashion.<sup>2</sup>

Note that the data reduction process is not endemic to the HST or to astronomy. Furthermore, the technology we are about to describe is applicable not only to those domains in which raw data is reduced and analyzed, but to any domain rife with repetitive, menial computations. Please keep in mind that many of the references to reduction and analysis in these notes could just as well be references to arbitrary computations.

### Developmental History

Design and development of Draco began in October, 1991. The first prototype, about 3700 lines of Common Lisp, C, and Bourne shell code, was completed and released in August, 1992. Development will continue through most of 1993.

## Problems

### No Consensus

Our main problem is the astronomy experts' tendency to disagree. There are at least five algorithms for removing cosmic ray noise and some suspect that the proper choice of algorithm hinges on the type of analysis that will ultimately be performed. There are also at least five image restoration techniques that attempt

---

<sup>2</sup>Observations taken with the HST become public after a one-year grace period so scientists have some incentive to analyze their data quickly.

to compensate for the HST primary mirror's spherical aberration. Scientists often wish to experiment with different algorithms in order to discover which ones are best suited for their data.

Neither the older paradigm of procedurally encoding domain knowledge or the newer expert systems paradigm are well suited to supporting this sort of experimentation. Furthermore, both paradigms suffer from other problems. For example, procedurally encoded systems can be extremely difficult to maintain because a single fact about the domain may be encoded in several places. On the other hand, rule-based expert systems can also be difficult to maintain because one finds that the individual facts, when placed in a domain-independent framework, often interact in unexpected ways. There is also the well-publicized knowledge acquisition bottleneck, i.e. determining what the facts are and when one has accumulated enough facts.

The problem could be summed up as follows: the ground-breaking scientist is often unsure about what the facts really are. The conventional approaches, including the expert systems approach, do not work well in such a domain. We feel that AI technology can still provide a solution.

### Inertia

A scientist has good reason to be skeptical of new software claiming to replicate the functionality provided by his current analysis software package, especially since his package has probably earned not only his respect but that of his peers. Astronomers are particularly concerned with understanding how their data is being manipulated. This inertia, along with our limited project resources, convinced us that we needed to design a system that can use existing reduction software, and the astronomer's concern for his data convinced us that our system must be able to assure the user that the proper reduction steps have been carried out.

### Heterogeneity & Antiquity

There are *many* software tools including several widely-used analysis packages and several data file formats. Integrating all these tools in a single framework will not be easy, especially since few of these tools were designed with this sort of integration in mind. These tools must *eventually* be replaced by a single, coherent, extensible analysis package, but integrating existing packages seems to be the most appropriate solution today.

## Solutions

### Overview

We make two observations. First, our goal is to optimize a man-machine system of scientists, software developers, and software. The system's purpose is to collect, reduce, and subsequently analyze data. The

traditional software engineer attempts to optimize the software part of the system, i.e. make the software as powerful as possible. This does not necessarily optimize the system as a whole (but it is a good way to stay employed). Given the flexibility required by our domain, we feel that our efforts would be better spent developing software which is *not* knowledgeable in the sense that it does not anticipate user demands. What it does do is allow the user to easily adapt his software to the situation at hand. In short, Draco is not an expert system, but an environment that facilitates the integration of existing software tools.

Our second observation: purely syntactic manipulations are often quite useful. Draco can be likened to an algebra or a programming language. One defines a set of *primitives* and is allowed to combine them to form *procedures*. Draco does not know anything about the semantics of the primitives, but it does know which combinations of primitives are syntactically valid, and therefore potentially useful. The user is responsible for insuring that the procedures are truly useful.

Procedures and primitives are abstract entities. The user provides additional information about concrete entities such as primitive *implementations* and the data formats corresponding to these implementations and Draco uses this information to translate a procedure into an executable script tailored to the user's data.

### Keeping the User Informed

In order to keep the user at ease, we have made every effort to generate an audit trail as the reduction proceeds. The trail begins with an inventory of the user's data files. This file inventory is generated by a collection of file type *recognizers* and *reporters*. In practice, these recognizers are very thorough; they often read the file in order to determine its type. A sample file inventory is given in Figure 1.

An executable script should produce a log file recording the data reduction or analysis steps performed. Once again, Draco does not know how to produce such a log file; it merely provides a little syntax enabling the user to define his entities so that appropriate entries will be entered in the log.

### Integrating Existing Packages

Depending on the packages, i.e. if they all have fairly reasonable command-line interfaces, this can be a surprisingly easy task. Draco makes do simply by storing a character string template for each of the user's tools. For example, a user might define a primitive RCRN (Remove Cosmic Ray Noise) as in Figure 2.

RCRN is defined in terms of its input data type *image-set*, its output data type *image*, and the implementations list following the *:concrete* keyword. The *:reconcile* keyword determines how multiple inputs are to be handled. The default value *nil* specifies that multiple inputs should signal an error condition, a value of

Figure 1: Sample file inventory

Inventory for /marian/data2/mds  
Generated on 02-Aug-1992, 16:32:16  
Draco version: 1

GEIS-CALIB-IMAGE files -

```
w0u11d03t.c0h =
  FILETYPE: 'SCI'
  IMAGETYP: 'EXTERNAL'
  INSTRUME: 'WFPC'
  FILTNAM1: 'F555W'
  FILTNAM2: ' '
```

```
w0u11a04t.c0h =
  FILETYPE: 'SCI'
  IMAGETYP: 'EXTERNAL'
  INSTRUME: 'WFPC'
  FILTNAM1: 'F785LP'
  FILTNAM2: ' '
```

DIRECTORY files -

uparm: directory

UNIX files -

Draco.inv: Draco document  
mbox: mail folder

*conjunctive* specifies that the corresponding implementation should process all inputs at once, whereas the value in the example, *:distributive*, specifies that the implementation should be invoked once for each input.

One popular way of removing cosmic ray noise is to take several exposures and then average their pixel values ignoring those which are unusually bright. The STSDAS<sup>3</sup> program *wfpc.combine*<sup>4</sup> can be used to perform this reduction step.

The implementation STSDAS-RCRN (Figure 3) is defined in terms of its analysis package IRAF<sup>5</sup>, its input file type *IRAF-image-list*, its output file type *GEIS-calib-image*<sup>6</sup>, its initialization command templates, and a template specifying its invocation syntax. In these templates, “-in” represents the implementation’s input, “-out” represents its output, and “-log” represents the log file to be generated.

<sup>3</sup>STSDAS = Space Telescope Science Data Analysis System

<sup>4</sup>WFPC = Wide Field/Planetary Camera

<sup>5</sup>STSDAS is layered on the Image Reduction and Analysis Facility.

<sup>6</sup>The STSDAS data file format is known as the Generic Edited Information Set.

Figure 2: Sample primitive specification

```
(define-primitive
  :name RCRN
  :documentation "remove cosmic ray noise"
  :input image-set
  :reconcile :distributive
  :output image
  :concrete (STSDAS-RCRN)
)
```

Figure 3: Sample implementation specification

```
(define-implementation
  :name STSDAS-RCRN
  :documentation "a CR removal program"
  :draco-package IRAF
  :input IRAF-image-list
  :output GEIS-calib-image
  :initialize-once ("stsdas"
                  "wfpc"
                  "combine logfile=\ -log\ ")
  :initialize ("combine usedqf=yes"
              "combine outtype=r"
              "combine option=\ crreject\ ")
  :syntax "combine @-in -out"
)
```

An implementation need not be part of a software package. Stand-alone Unix<sup>TM</sup> programs and Common Lisp functions may also be used to implement primitives.

## Conclusions

At this point, it should be clear that our initialization and invocation templates have trivialized the problem of automatically generating calls to existing reduction programs. The moral of our story appears to be that one need not accumulate vast amounts of declarative knowledge in order to create a useful “expert system,” even (especially?) if one’s domain experts frequently disagree with each other.

The first Draco prototype demonstrates that one only needs a minimalist representation system to harness existing data reduction tools. Instead of undertaking the costly enterprise of declaratively encoding algorithms and other techniques, e.g. iterative image deconvolution methods, we have made use of the vast amount of procedurally-encoded domain knowledge that already exists.



## Future Directions

Our code generation problem would be much more difficult if our procedures had more structure. Data reduction procedures tend to be pipelines of programs that do little more than read data and write transformed data. More complex procedures might invoke predicates to control branching or looping. Draco would have to undergo substantial changes if it is to support additional procedural complexity, but these changes seem to be fairly straightforward, i.e. not challenging from a representational point of view.

One bit of complexity that might be implemented shortly is the ability to automatically invoke data format conversion programs in order to integrate software packages that do not share a common data format. Once again, this modification does not appear to be challenging from an AI point of view.

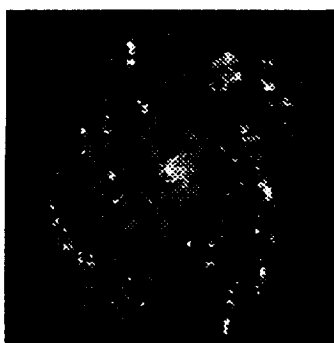
What would be challenging is giving the user the ability to specify a set of analysis goals and then selecting the procedure most suited to these goals. Such ambition would require a much richer representation language, and more time than we can spare.

## Acknowledgements

I would like to thank Glenn Miller for his excellent review of earlier drafts of these notes and for his help in designing Draco. Mark Johnston also plays a critical role in this project.

## **Draco – A Data Reduction Expert Assistant**

The use of large format detectors, increased access to very large astronomical databases, and other developments in observational astronomy have led to the situation where many astronomers are overwhelmed by the reduction and analysis process. Draco is a novel approach to data reduction and analysis which works in conjunction with existing analysis systems such as STSDAS/IRAF, IDL, etc. Draco takes on much of the mechanics of the process, allowing the astronomer to spend more time understanding the physical nature of the data.



[More About Draco \(Publications\)](#)



[Status of the Project](#)



[Draco Team](#)

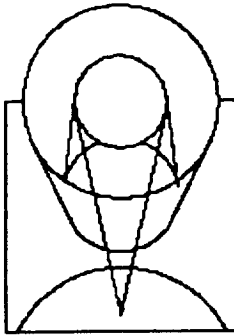


[How to Get Draco](#)

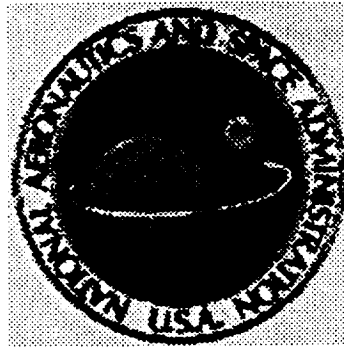


[Related Places to Explore](#)

---



**SPACE  
TELESCOPE  
SCIENCE  
INSTITUTE**



Operated by the Association of Universities for Research in Astronomy for the National Aeronautics and Space Administration.

This work was supported by a NASA Applied Systems Information Research Program (AISR) contract to the STScI, which we gratefully acknowledge.

---

*Last updated 16 January 1994, Glenn Miller (miller@stsci.edu)*

The following papers on Draco are available:

**1. The Data Reduction Expert Assistant, Glenn Miller.**

This is the paper to read first. It gives an introduction to the project, including its origins and relation to other work, as well as a discussion of how Draco works. You can view the [HTML file](#), [Postscript file](#) or [Rich Text Format \(MS Word\) file](#).

**2. Draco: An Expert Assistant for Data Reduction and Analysis, Glenn Miller and Felix Yen.**

This is the most recent paper on Draco, and includes a discussion of future directions. You can view the [Postscript file](#) or [LaTeX source file](#), or the [ADASS III Conference Proceedings](#)

**3. Draco Design Document, Felix Yen.**

This is the reference and user's manual for Draco. You can view the [Postscript file](#) or [FrameMaker file](#).

**4. Draco – A Data Reduction Expert Assistant, Felix Yen.**

A discussion of Draco for the Artificial Intelligence community. You can view the [Postscript file](#) or [Latex source file](#).

**5. An Expert Assistant for Astronomical Data Reduction, Mark Johnston.**

This describes Johnston's 1987 prototype study which inspired Draco. This paper is not available on-line at this time. Send mail to [miller@stsci.edu](mailto:miller@stsci.edu) for a reprint. Be sure to include your mailing address.

## **Bibliography**

1. Invited talk presented at "Astronomy From Large Databases II", Haguenu, France, Sep, 1992. Published in ESO Conference and Workshop Proceedings No. 43, Ed. A. Heck and F. Murtagh, pp. 99–110.
2. Presented at the 1993 Conference on Astronomical Data Analysis and Software Systems ([ADASS](#)), Victoria, BC, Oct, 1993.

Conference proceedings published by the Astronomical Society of the Pacific (in press).

3. Space Telescope Science Institute Advance Planning Systems Branch Technical Report APSB 1992-07.
4. Presented at the 1992 American Association for Artificial Intelligence Fall Symposium on Intelligent Scientific Computation, Cambridge, MA, Oct, 1992.
5. Presented at the 1987 Goddard Conference on Space Applications of Artificial Intelligence and Robotics, Greenbelt, MD, May 1987. Proceedings published by NASA Goddard.

**REPORT DOCUMENTATION PAGE**Form Approved  
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

<b>1. AGENCY USE ONLY (Leave blank)</b>		<b>2. REPORT DATE</b> July 5, 1994	<b>3. REPORT TYPE AND DATES COVERED</b> Contractor Report	
<b>4. TITLE AND SUBTITLE</b> Development of a Data Reduction Expert Assistant - Final Report			<b>5. FUNDING NUMBERS</b>  930	
<b>6. AUTHOR(S)</b> Dr. Glenn E. Miller				
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Association of Universities for Research in Astronomy Space Telescope Science Institute 3700 San Martin Drive Baltimore, MD 21218			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>  NAS5-32337 5555-02	
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> National Aeronautics and Space Administration - OSSA Washington, D.C. 20546-0001  Universities Space Research Association 10227 Wincopin Circle, Suite 212 Columbia, MD 21044			<b>10. SPONSORING/MONITORING AGENCY REPORT NUMBER</b>  CR-189385	
<b>11. SUPPLEMENTARY NOTES</b>  Technical Monitor: J. Hollis, Code 930				
<b>12a. DISTRIBUTION/AVAILABILITY STATEMENT</b> Unclassified-Unlimited Subject Category 82 Report is available from the NASA Center for AeroSpace Information, 800 Elkridge Landing Road, Linthicum Heights, MD 21090; (301) 621-0390.			<b>12b. DISTRIBUTION CODE</b>	
<b>13. ABSTRACT (Maximum 200 words)</b> This report documents the development and deployment of the Data Reduction Expert Assistant (DRACO). The system was successfully applied to two astronomical research projects. The first was the removal of cosmic ray artifacts from Hubble Space Telescope (HST) Wide Field Planetary Camera data. The second was the reduction and calibration of low-dispersion CCD spectra taken from a ground-based telescope. This has validated our basic approach and demonstrated the applicability of this technology.  This work has been made available to the scientific community in two ways. First, we have published the work in the scientific literature and presented papers at relevant conferences. Secondly, we have made the entire system (including documentation and source code) available to the community via the World Wide Web.				
<b>14. SUBJECT TERMS</b> Astronomy, Data Management, Expert Assistant, Data Reduction			<b>15. NUMBER OF PAGES</b> 28	
			<b>16. PRICE CODE</b>	
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> Unlimited	