Astronomical Data Analysis Software and Systems IV
ASP Conference Series, Vol. 77, 1995
R. A. Shaw, H. E. Payne, and J. J. E. Hayes, eds.

Porting CGS4DR to Unix

P. N. Daly

Joint Astronomy Centre, 660 N. A'ohōkū Place, Hilo HI 96720.

Abstract. This paper discusses the port of the CGS4 Data Reduction system to Unix, and describes a novel solution to some non-portable code.

1. Introduction

The CGS4 Data Reduction system, CGS4DR, is a suite of tasks for automatically reducing data from the long-slit spectrometer on the 3.8m United Kingdom Infra-Red Telescope (UKIRT) on Mauna Kea, Hawai'i (Daly et al. 1994 and references therein). Delivered with the instrument in 1991, the core software consists of four ADAM tasks written in FORTRAN under VMS. Since that time, British astronomy has moved away from VMS and towards Unix. The ADAM Support Group was charged with porting the ADAM system, and a considerable fraction of that project is now complete. Although not required by UKIRT in the foreseeable future, since the data acquisition system will remain VAX-based for some time, CGS4DR is currently being ported to UNIX—it will provide a strict test of the final release of the ported ADAM system.

2. Portability Issues

During the period 1991–1994, the code was modified substantially to enhance the facilities available to the astronomer in reducing spectroscopic data in real-time at the telescope, while remaining true to the original design paradigm. The more obvious features of non-portable code, VMS run time library calls and the like, were removed, and the code was altered to conform to ADAM V2 instrumentation software standards (Kelly & Chipperfield 1992). Two minor, but annoying, problems were discovered: mixed data types within a common block, and common blocks with variable lists exceeding the maximum line length, even when compiled in "–e" mode. Reformatting the common block cured both problems. Code porting problems were few, and error-free compilation has been achieved. Local extensions to Figaro (Shortridge 1993), used by CGS4DR for accessing data structures and so forth, have also been ported.

3. Design Issues

A major design flaw of CGS4DR was the use of indexed sequential access method (ISAM) files. Such files are DEC FORTRAN compiler extensions not supported under many other operating systems. They were used to allow the data acquisition (DA) system and the data reduction (DR) system should communicate only via a time stamped queue. A second use of an ISAM file—maintaining an index of reduced observations—was added later.

3.1. QMAN The Generic Queue Manager

The (ISAM disk file) data reduction queue accepted time stamped commands and was periodically read by the DR system by task re-scheduling. The following data reduction sequence could have been generated by the DA system automatically adding observation numbers 2–3 to the "tail" of the queue and the observer inserting observation number 1 at the "head" of the queue:

```
000913:11:12:49.81  REDUCE 0940913_1
000913:11:12:49.82  END 0940913_1
940913:11:12:44.38  REDUCE 0940913_2
940913:11:12:44.47  END 0940913_2
940913:11:12:44.50  REDUCE 0940913_3
940913:11:12:44.51  END 0940913_3
```

CGS4DR *always* read the *oldest* command from the queue and the use of an ISAM file with a time stamp as the primary key was, at the time, an efficient solution to these requirements. Since this file was also on disk, it was easy to recover in the event of a software failure.

In porting this application, a new task was created to replace the ISAM file: QMAN, the generic queue manager (Daly 1994). QMAN is a standard ADAM I-task and maintains a list of 1000 character string entries *in memory*. Incoming strings are time stamped using an accurate modified Julian date (MJD) and the queue position may be specified as either "OLDEST" or "NEWEST." To QMAN, the above queue might look like this:

```
 49608.47223391203614 REDUCE 0940913_1
 49608.47223390046202 END 0940913_1
-49608.47206021990678 REDUCE 0940913_2
-49608.47206023148101 END 0940913_2
-49608.47206024305524 REDUCE 0940913_3
-49608.47206025462947 END 0940913_3
```

In this scheme, the "NEWEST" position has the maximum MJD whereas the "OLDEST" position has the latest MJD *negated* and so has the minimum value. The MJD is a monotonically increasing function, and provides a powerful and flexible means of maintaining a time stamped queue. With a suitable combination of reads and writes, QMAN may mimic either a FIFO or a LIFO buffer.

An important distinction now is that the Unix–CGS4DR software always reads the *newest* command from QMAN. Further, after each write by the DA system the queue is saved to disk and each read by the DR system is a destructive process that releases that slot in memory for subsequent commands. QMAN also features an optional (global) password protection scheme and a (local) lock manager to provide exclusive access to the database.

3.2. The Index of Reduced Observations

There is second use of an ISAM file in CGS4DR: to maintain an index of already reduced observations that may be used as calibration frames in later reduction sequences. When such a calibration frame is required, a search is made according to some defined criteria—data size, wavelength, oversampling parameters and so forth—and the frame closest in time to the present observation being reduced is selected. There are several alternatives for re-coding this functionality.

First, clearly, a separate invocation of the QMAN task could fulfill this role. Take the simplest example of a BIAS frame. CGS4DR requires only that the detector size should match and so a BIAS could be filed using the code fragment:

```
      CALL TASK_OBEY( 'QMAN', 'WRITE',
    :   'STRING="OBSN=ODIR:0940913_1 TYPE=BIAS QLTY=GOOD '/
    :   /'ROWS=256 COLS=256" QPOSITION="NEWEST"',
    :   OUTVAL, QMAN_PATH, QMAN_MESSID, STATUS )
```

When a BIAS is required, therefore, QMAN searches for the "NEWEST" observation (i.e., closest in time) using the READ command in SEARCH mode:

```
      CALL TASK_OBEY( 'QMAN', 'READ',
    :   'READ_MODE="SEARCH" SEARCH_MODE="NEWEST" '/
    :   /'DESTRUCTIVE=FALSE STRING="TYPE=BIAS QLTY=GOOD '/
    :   /'ROWS=256 COLS=256" QPOSITION="NEWEST"',
    :   OUTVAL, QMAN_PATH, QMAN_MESSID, STATUS )
```

This search is made *non-destructively* so that the same index is maintained throughout. At the end of the data reduction session, the complete index may be saved to an ASCII text file. To file an observation as BAD would simply require searching the database in *destructive* mode and ignoring the result of the read (which is always passed back). The BAD observation would, therefore, be erased from the database (and could be re-filed with "QLTY=BAD" so that all observations appear in the database as they do at present). The problem with invoking a second incarnation of QMAN in this way is that it takes more memory, uses up a task slot and the reduction task—an A-task in ADAM parlance—would then have to talk directly to another task which is bad form since that is the domain of I-tasks. A-tasks should simply "do something and return."

Using simple ASCII text files would overcome these problems. A separate file for each calibration type would be used; e.g., cgs4_940913.bias and so forth. Continuing with the BIAS example, the string written to the file would be similar to that above but with the addition of an encoded MJD as used by QMAN. Thus the difference in time between the observation currently being reduced and the calibration frame could be quickly evaluated. Such a text file for BIAS frames might contain the following lines:

```
RODIR:R0940913_1 BIAS GOOD 256x256 MJD=49608.47223390046202
RODIR:R0940913_2 BIAS GOOD 256x256 MJD=49608.47223391203614
```

The drawback here is the relative slowness of (sequential) disk file access although the size of these files would be small (typically, less than 50 frames of calibration data are taken per night). A further complication—not insurmountable—is the ability to (re-)file an observation as BAD when desired. The optimum solution to this problem has yet to be determined.

4. De-coupling the Software Tasks

VMS–CGS4DR consists of four tightly coupled tasks: a control task, a reduction task, a plotting task, and an engineering task. QMAN has essentially de-coupled the management of the data reduction queue from the control task. Might de-coupling benefit the other tasks? The plotting task is in the process of being de-coupled from the control task, making it standalone, and potentially useful to other applications, since it can access both Figaro DST and Starlink NDF format data files.

Under the VMS system, the control task, when requesting a sub-plot of an image, for example, would send a message such as:

```
IMAGE DATA="RODIR:R0940912_1" PORT=0      -
PLANE=DATA WHOLE=FALSE XSTART=20 XEND=40 -
YSTART=20 YEND=40 AUTOSCALE=FALSE HIGH=1000.0 LOW=0.0
```

Under Unix–CGS4DR, the plotting task now maintains its own noticeboard so the control task sends a far simpler message:

```
DISPLAY DATA="RODIR:R0940809_1" PORT=0
```

The plotting task then reads the display characteristics of port 0 from the noticeboard global section. This global section may be manipulated directly from the command line or user interface to tailor the display as desired.

5. User Interface and GUIs

At present, there is no clear way to interface Unix–ADAM tasks with a GUI although groups are investigating ICLmenus (the Starlink approach) and Tcl/Tk. The first release of Unix–CGS4DR, intended for delivery before the close of 1994, might well have a command line interface alone.

References

Daly, P. N., Bridger, A., & Krisciunas, K. 1994, in Astronomical Data Analysis Software and Systems III, ASP Conf. Ser., Vol. 61, eds. D. R. Crabtree, R. J. Hanisch, & J. Barnes (San Francisco, ASP), p. 457dalyp

Kelly, B. D., & Chipperfield A. J. 1992, "ADAM—Guide to Writing Instrumentation Tasks", SUN/134, Starlink Project (DRAL/Rutherford Appleton Laboratory)

Shortridge, K. 1993, in Astronomical Data Analysis Software and Systems II, ASP Conf. Ser., Vol. 52, eds. R. J. Hanisch, R. J. V. Brissenden, & J. Barnes (San Francisco, ASP), p. 219

Daly, P. N. 1994, "QMAN—A Generic Queue Manager", UON/16, UKIRT, (Hilo, Joint Astronomy Centre).