A Language for Gigabyte Data Processing, and its Use at Sunspot Observatory

German Altgelt

I. Processing Huge Amounts of Data: a Specific Feature of Data Processing in Optical Astronomy

¿From the viewpoint of data processing, there are two main differences between processing astronomical observations and processing other physical data:

- First, in physics we usually have the results of *experiments*; we can start an experiment when we are ready, and often we can change the rate of the experiment at our will. In astronomy, we have *observations*. We have no control over when a process that is currently observed starts, we cannot slow it down or speed it up.
- Second, physical experiments are usually *repeatable* (this is one of the major ideas of science); therefore, if we delete the results, we can always reproduce them again. In astronomy, all the processes we observe are unique. Therefore, we need to store all the information that we have, so that in addition to the data processing for which this particular experiment was designed, we will be able to use this information later. Astronomers often use the information produced centuries ago; the possibility to compare the current photos with a century-old ones is one of the main reasons behind the maintenance of the near-city observatories (Greenwich near London, Pulkovo near St. Petersburg, Russia) in which the optical conditions (due to city's light pollution) are no longer good.

As a result, in astronomy, we must *keep* all the information that we get. And this means lots of information: e.g., a good quality photo contains approximately a Gigabyte of information.

This problem occurs in different areas of astronomy, e.g., in radioastronomy. The higher the frequency, the more information we get per second, and hence, the more information we need to store. Therefore, this problem is most urgent in optical astronomy, where we get up to gigabytes of data from each experiment.

II. In Particular, This Means that Error Estimation Should Be Done Off-Line

Modern computers cannot process data that fast. This means that we have to store the data and process it off-line, doing only the most basic data processing on-line. In particular, this means that *error estimation* (one of the goals of interval data processing) has to be done off-line.

Algorithms are still more or less the same; the main computer problem is now that we need to store and retrieve gigabytes of memory. This problem is handled in the described project.

III. Case Study: Liquid Crystal Polarimeter

Larry November and Olivier Brouchard from the Sunspot Observatory are designing the instrument called LCP (*Liquid Crystal Polarimeter*; see, e.g., [?] and references therein).

The LCP computer is a 586 PC with four DSP's (digital signal processors) that runs electro-optical elements via an I/O board and programmable signal generator. The DSP's receive data at 10 M pixels per second from two 1000×1000 CCD sensors. The computer will be equipped with a large disk (probably 9 GB) and an gigabyte high density 8mm tape (5 GB per tape recording at 1.5 MB per second). The LCP computer will run with Solaris operating system to have accessibility over our sun network which is currently used to control other hardware at the observatory.

IV. The Project

The project was to design a program to run in the background; its purpose is to copy files from the disk to a gigabyte tape for archiving. There can be several users calling this program at one time, this part has been developed by Scott Ovelmen by using a Message Queue and a Semaphore. Many options are available for this program, from controlling the blocksize of the gigabyte to deleting a file after it

Department of Computer Science, University of Texas at El Paso, El Paso, TX 79968, email galtgelt@cs.utep.edu. The work was done during the Summer of 1995 at the Sacramento Peak National Solar Observatory, in Sunspot, New Mexico, under the supervision of Larry November and Olivier Brouchard from the Observatory and Vladik Kreinovich from the Department.

has been archived.

The part that I designed was to decode and execute the instructions sent to the message queue. A consideration was to how the options to each call were going to look, and it was decided to add a dash (`-`) and a single letter after it, followed by a colon `:`. So, for example, to initialize a tape you would write

```
tapeset T -n:NameOfTape
```

or simply

```
tapeset T
```

in which case the computer would give the name of the current tape. There are four basic commands that are described by letters `T`, `W`, `E`, and `S`.

The `T` command is given by the user every time the user inserts a new tape in the drive. Also, it can be used to fix a given tape if there is an error in the disk file containing what is in that particular tape. The possible options for `T` are:

- `-n:` to name a tape,
- `-f:` to fix a tape,
- `-e:` to go to the current end of master and start from there, and
- `-t:` to specify a path for the current tape drive (device).

The `W` command allows the user to write data from the disk to the tape drive. It can do several things; by specifying:

- `-b:blocksize` you control the blocksize of the current file of the tape;
- `-c:path` you change the current working directory;
- `-d:` you delete the files after sending them to tape;
- `-e:(tapename)` after executing this `W` command, the tape will be rewound and ejected. If (tapename) is specified, then a new tape will be made as if a `T` command was executed; continues after user changes tapes;
- `-f:path` you just write the specified files to tape without deleting them;
- `-t:path` specifies the current device for the tape drive.

The `E` command means an End of tape and a rewind.

The `S` command gives you a status report of the tape, and also gives a list of what is still left in the message queue.

Acknowledgments. The author was partially supported by NSF Grants No. EEC-9322370 and No. CDA 9522903.

References

[1] L. J. November and L. M. Wilkins, "The Liquid Crystal Polarimeter, a solide state imager for solar vector magnetic fields", *Optical Engineering*, 1995 (to appear).