Customisation of the CS6400

Dave Haworth, Manchester Computing,
University of Manchester, England

ABSTRACT: *The University of Manchester has been providing a National Datasets Service on a CS6400 for over eighteen months. The predecessor system had been based upon traditional mainframe hardware, with an operating system offering the range of facilities we came to expect from such equipment. The transition to the CS6400, and an operating system environment more usually associated with workstations, necessitated the development and customisation of the system in order to provide our users with the system environment required. This presentation will describe the areas we particularly felt deficient and how we have addressed them.*

Introduction

The University of Manchester is one of the largest higher educational establishments in the United Kingdom with some 20,000 students and 6,000 staff. The University has a particularly strong scientific background, and indeed many of us here today can perhaps thank (or otherwise) the University for the development of the worlds first stored program computer nearly 50 years ago.

Computing provision ranges from 6000 plus PCs; over 1000 workstations; a variety of intermediate equipment including an EL98 and a KSR; a Fujitsu VPX 240/10 providing a National Vector Supercomputer service; and the Cray CS6400 providing a National Datasets service. A range of ancilliary services are also offered including the ubiquitous computer shop, a full range of general campus training, and Manchester leads a consortium of Universities in the region with provision of a High Performance Computing Training and Education Centre.

Configuration

Our CS6400 has: 12 processors; 768 Mbytes memory; 170 Gbytes of disc storage; a variety of tape equipment including 1/2" open-reel decks, QIC, Exabyte and DAT; and a CNT Channelink SCSI Interface Unit to connect a Memorex ATL. The recent award of two further datasets contracts will push the disc storage over 200 Gbytes, and purchase of a further system card primarily for the Sbus ports and additional memory.

The system was purchased to provide a National Datasets Service and this we call MIDAS - **M**anchester **I**nformation **D**atasets and **A**ssociated **S**ervices.

MIDAS Aims

The aims of the MIDAS service can be summarised as:

- Provide the UK academic community with the widest possible access to strategic research and teaching datasets;
- Provide computing facilities for the storage, access, manipulation, and analysis of large and complex datasets;
- Provide a range of high quality support services, including documentation, training, statistical and software advice;
- Work in close collaboration with other organisations, such as the Economic and Social Science Research Council (ESRC) Data Archive, to promote and extend the secondary analysis of strategic research datasets.

Further information on MIDAS can be found via the WWW at URL:

http://midas.ac.uk/

MIDAS Datasets

The range of datasets supported by MIDAS can be nicely summarised by the following categorisations (and examples):

- UK Census of Population Statistics (1991 Local Base and Small Area Statistics)
- Government and other large continuous surveys (General Household Survey)
- National and International macro-economic time series databanks (IMF International Financial Statistics)
- Scientific datasets (Mossbauer Effect Reference Database)
- Bibliographic datasets (CURL Database/OPAC)
- Digital Map datasets (Bartholomews digital map data)

In total, there are now in excess of 135 distinct datasets.

A range of Applications Software is provided to assist in the processing and analysis of the data, in particular:

SAS; SPSS; NAG; Ingres; BRS/Search; ARC/INFO; SIR; BMDP; GAUSS; S+; RATS; STATA

The user community is now approaching the 3000 mark, these users being scattered across the whole of the United Kingdom. Getting information, either of a permanent or a transient nature, to these users is a constant problem. All forms of communication are used as appropriate, including a regular newsletter, the 'news' facility, message-of -the-day, e-mail and the friendly voice on the phone.

What did the mainframe offer

The predecessor system was an Amdahl 5890/300E running the IBM operating system VM/CMS. The facilities offered by this system included:

- a comprehensive batch service
- resource allocation and control system
- a controllable interactive environment
- filestore backup and archiving to an automatic tape library
- comprehensive Filestore Manager that included Access Control Facilities
- operating system development in flight, i.e an OS under OS
- cost effective provision of hardware resilience and expansion

Does the CS6400, particularly the vendor supplied Solaris operating system environment accomplish all these things? Not as yet (!). Consequently the University has developed software in many areas to provide the platform necessary for our MIDAS service.

What have we done

Batch Service

Prior to the acquisition of the CS6400, the University had evaluated the various NQS systems available with a view to standardise on one version to facilitate support . Proprietary ones were very soon dismissed since even at low cost per unit, the overall cost for the campus was going to prove excessive. Evaluation of the NQS software freely available across the 'net' had resulted in the decision to standardise on the Cosmic NQS as modified by Cern.

Following our own further customisation of course, installation on the Fujitsu and several campus clusters had already commenced prior to purchase of the CS6400 and availability of NQE. The need to standardise on one NQS, and the lack of one particular feature in NQE in respect of queue ordering on priority, resulted in our NQS being installed on the CS6400.

The queue ordering feature on priority is required by the scheduling mechanisms we utilise at Manchester, which is often fondly referred to as a Larmouth style scheduler. Normal schedulers if we use that phrase usually work on the principle of first in/first out queuing. With a Larmouth style scheme, the position at which jobs are placed in the queue is dependent upon a rate of working factor for a project, site or department for example as required.

The CS6400 resources are essentially allocated to "sites" not individual users. So the scheme takes a sites allocation and its duration to determine a "target rate of working" (TRW) for the site i.e essentially an average. An NQS scheduling factor is then determined when a job is submitted by comparing the "Actual work rate" (AWR) with the TWR, in fact simply dividing the AWR by the TWR.

From this scheduling factor, a queue priority is determined. Priority 32 equates to a scheduling factor of 1, thus projects computing at a higher rate than their TWR will have a queue priority smaller than 32, and vice versa.

Additionally, to simplify life for the users, they do not specify the target queue on their job. They specify the resources required by the job, we then allocate the job to the most appropriate queue.

Resource Allocation and Control

Standard Unix accounting systems as supplied by vendors are usually limited, there being one or two exceptions such as Unicos. The Unix detailed process level accounting information is not suitable for our needs, our particular requirements being:

- “job” accounting: users submit a job, that is what they want information on;
- “true cost” information: charging formula;
- “account” information: usage reports, scheduling priority;
- “report” data: management information, reports for circulation;
- “control” data: interface to the batch system.

When we commenced the transition from our IBM environment to Unix just over two ago, we looked for a commercial Unix resource accounting package. At that time, there appeared to be three potential suppliers, however: one did not reply to requests for information; one upon evaluation appeared to be selling a product that was still in the design phase; and the one remaining supplied us with an evaluation system but it lacked a number of our desired facilities, in particular lack of support for NQS.

We consequently developed our own package and this was subsequently installed on the CS6400. Currently we do not offer job accounting, and have to settle for site accounting. Provision of an accounting system such as that offered by Cray for Unicos would do very nicely to satisfy our requirements.

Controllable interactive environment

Solaris, like many other Unix based operating systems originally designed for a desktop, allows the user (who may indeed be the owner and only user) to basically run anything they want. It is "their" machine so why should they not run 10 concurrent processes each requiring 10 hours cpu, and why is my interactive response poor!

As previously described, we utilise NQS to process and control such large work, but how do we make people use it so that we can protect the interactive response times?. To encourage the use of NQS we run a program developed locally called the **governor**. This monitors each users use of the system and limits their interactive work to at most 15 minutes cpu per

process. As with everything else, there are exceptions and a configuration file is available to handle these.

Filestore backup and archiving to an automatic tape library

For some years now, we have been utilising an automated operations environment, indeed a dark room environment for much of the week, as part of a continual service provision throughout the year. The use of robotic tape equipment for filestore backup, file archiving, and user tape storage has underpinned this strategy. The Memorex robotic tape equipment we have installed was originally built for an IBM environment and significant developments were necessary both by ourselves and Memorex to support a Unix/SCSI channel environment.

This development was described in a full session at the Denver CUG meeting, and the paper can be found in the proceedings.

Comprehensive Filestore Manager that included Access Control Facilities

We needed a mechanism to enable us to restrict access to a large number of the on-line datasets to authorised users. This was a mandatory requirement imposed upon us by the ESRC Data Archive, the Office of Population Censuses and Surveys, and other data suppliers. We have to demonstrate that this mechanism is both secure and foolproof.

In the absence of Access Control lists in present versions of Solaris, we have to resort to the use of Unix groups as the mechanism of control. However, the maximum number of groups to which a user can belong is only 16 in Solaris. This was insufficient for a simple implementation using static assignment of users to groups, because we have a large number of datasets, which can each have their own list of authorised users, and each user may may be authorised to access a large number of datasets.

We discussed solutions with Cray, in particular the possibility of an Access Control List facility within Solaris. This was projected for Solaris 2.5 (and indeed now committed), but in early 1994, this was seen as some 2 years from implementation and so an interim scheme was deemed necessary.

Therefore to control access to our datasets, we have written and implemented a setuid/setgid privileged program which is called by the appropriate dataset access script. Each dataset is set up with read permission given only to its own group; the control program checks the user and dataset combination against an authorisation list, and if found, dynamically changes the users current group to the appropriate group for that dataset; it then resets the userid back to that of the user and continues to run the normal dataset access program. A note of caution: because it is a setuid program, care has to be taken to ensure that there are no unwanted side-effects, for example ensuring that the user's ulimit values are not reset.

Resilience improvements

To minimise downtime on the CS6400 in the event of a catastrophic disc failure with the boot disc, we have established an alternate boot disc from which to make a fast recovery.

In addition, we have added a second disc to the SSP to similarly provide a degree of resilience to a disc failure (with the cost of discs these days being measured in a few hundred pounds, and the reliance of the system on availability of the SSP, the SSP should be supplied with two discs as standard).

But without further significant injection of capital, that is about as far as we can go with the CS6400 due to its architectural design. These limitations will be discussed later.

What is outstanding (or what we do not like!)

Operating system development in flight, i.e an OS under OS

Despite being an academic site, and therefore assumed by some that computing facilities are provided to give us a toy to play with, we live in a competitive environment. We have to fight for contracts, our services are measured not only by the users but also by the funding bodies, so it is extremely important that we provide a professional, high quality service. We aim to provide 100% availability 52 weeks a year, so those major Solaris releases (which we can at least plan for) and those horrendous jumbo patches (which we can't) when they arrive give us major problems.

Our predecessor system offered the capability of an OS under OS environment, and this was extrememly successful in minimising the amount of dedicated time and service interruptions needed for any OS upgrades. This option cost us nothing to provide, and was available to the system developers throughout the day.

Cray now offer a domains facility on the CS6400 with one of the benefits being the ability to run a second version of Solaris in parallel with a production system. This however, is not a low cost option, with further complexities introduced by the hardware design of the system. Let us look at the two options:

a. we could purchase a system board with a minimum configuration, together with the required network connection and a disc, primarily for testing: cost too prohibitive.

b. we could transfer a system board from the production domain to the test domain as and when required: All well and good, but that for us equates to around one third of the processing capacity and memory, and of even more significance, the loss of the Sbus ports.

An OS under OS facility would appear to be a more suitable and cost effective solution, at least to the customer.

Cost effective provision of hardware resilience and expansion

Cray make extensive claims about the resilience features available with the CS6400. Features such as dual pathing to discs, dynamic reconfiguration of discs, mirroring, hot swap capability with system boards for example. Well, if you have a pretty large system, and do not use meta devices, there is a finite chance that you could take advantage of these facilities.

A hardware architecture that links cpu modules, memory, and Sbus controller ports and calls it a system board is very restrictive. We originally bought a three system card system, which in terms of cpu power was more than adequate. However, this only

gives us 12 Sbus ports of which we devoted 9 to disc equipment. With some 170 Gbytes of disc, we had to utilise daisy chaining of disc trays to accommodate this on the available Sbus ports. To mirror these, or provide dual pathing, would have required three more system cards!

It is our particular limitation with Sbus ports that also restricts utilisation of the hot swap capability for system cards.

Future design directions need to take into account a requirement to provide cost effective support and enhancement of I/O subsystems. Disc controllers on our mainframes provided dual pathing.

Selective operating system upgrades

Most of the time, I am happy to say that we have a working system with no apparent problems. The system runs for weeks without incident, and then the kernal jumbo patch arrives. Whoever christened these jumbo was not exaggerating - they often contain well over a megabyte of data. It is often clear that Sun have had problems with the patch, the version number revealing that a number of re-issues have had to be made. On occasions we have been advised not to install because whilst it fixes one of our problems, it will create others. Lets have a bit of sense please with the patch process - we do have services to run.

Documentation for new releases

The University fully endorses any moves to facilitate a paperless environment, but the expectation that CS6400 customers can prepare for, and install, major new releases of Solaris by reliance on Answerbook need to rethink this. Cray staff who were on-site at the time of initial installation of the system realised this and supplied us a copy. Since then, we have had to obtain a copy ourselves. This is not difficult, nor expensive, but something we feel the supplier should do.

Summary

The CS6400 provides an excellent response time capability and impressive performance with our applications. Therefore in respect of CPU performance, we are offering mainframe capability;

The hardware design in respect of I/O controller support and capability is, except for sites with substantial budgets, not as yet in the mainframe class;

Raw Solaris 2.3 was definitely still in the desktop world. The developments we have undertaken, together with some of the more recent features of Solaris 2.4 and 2.5 are gradually shifting this position, but we believe there is still much that SUN and Cray could do to before we would categorically state that Solaris was mainframe class.