Support Functions Influencing the Experience of T3D Users at ARSC

Mike Ess, Arctic Region Supercomputing Center

ABSTRACT: *The T3D users experience different problems than the typical YMP users. Some of these problems are due to the T3D being a new product and others are specific to parallel processing. The support of T3D users recognizes these two distinct problems. The first difference can be handled with an aggressive effort to provide T3D users with T3D information similar to what is commonly available to the YMP users. The second difference is much deeper and it requires an educational effort from User Services and a commitment from the users. The potential users of the T3D can be classified by their commitment, MPP experience and their access to T3D information. User Services can address the needs of each user according to such classifications.*

Current situation

At a very superficial level, it has been easy to sell Massively Parallel Processing. Almost all users could expand their problem so that lack of computer resources is the bottleneck to better results. For example, making a model with finer granularity is sometimes no more than a change in a parameter statement. Similarly, available datasets, as in molecular biology, are growing exponentially and satellites keep sending us data that we can't process fast enough. MPPs are touted as the solution for these 'big' problems and there are certainly enough 'big' problems that we'd like to solve.

Even though workstations have become wide spread and offer tremendous price performance, they have not solved these problems, especially when their cache, memory and I/O capabilities are exhausted. The YMP line of computers can handle these problems, but I believe most YMP sites are like our YMP site and we are running at 800% utilization 7 days a week. Beyond this immediate situation, a human can always describe a problem that no current computer can solve.

Into this world come the MPPs. The 'big' problems exist and the proposed hardware solutions, like the T3D, are here. The difficulty is that the software solutions are missing. Each user comes with their own problem and beyond the provided compilers and uniprocessor libraries they are on their own. User Services' role is to help the user program the problem onto the machine. In this paper we describe how ARSC tries to help our users program the T3D.

ARSC's configuration

At ARSC, we have a 8 processor YMP with 1 gigaword of MOS memory. The 1 gigaword of memory is our distinction as a YMP site and it alleviates many T3D problems that other sites may experience:

1. mppexec size is usually not a problem
2. ldcache comes out of the 1GW and speeds up I/O

With 8 CPU's, we use 3 IOCs for our I/O devices and we have 5 more IOCs to connect to T3D IOGs. Because we have sufficient memory and IOCs we are not a candidate for Phase II or Phase III I/O on the T3D. Our /tmp file system is 29 gigawords of DD-62s and is ldcached. The rest of our files systems are on a collection of DD-60s and DD-62s.

ARSC was one of the first T3D sites becoming operational in February, 1994. Our T3D has 128 processors and we recently upgraded the memory to 8MW per node. Before that upgrade we had four times as much memory on the YMP as on the T3D and there was little incentive to move to the T3D.

We are currently running 1.2 MAX and are now internally testing the 1.2 Programming Environment. After that upgrade we will install cf90 and C++. Because we have few T3D production users we want to be as up to date on T3D software releases as possible, but we are worried about disruptions to our YMP production users.

Initially we limit all interactive users of the T3D to 8 PEs with a one hour limit on connect time. We are very liberal in allowing 64 and 128 PEs in interactive mode upon request to User Services. This is one way we keep in contact with our users. For production runs our NQS batch queues are set up as:

4 8 PEs queues for 24 hours

2 16 PEs queues for 24 hours

1 32 PEs queues for 24 hours

T3D users are charged at a rate of:

1 Service Unit = 1 YMP Processor hours
= 20 T3D Processor hours

Disk Storage, Silo Storage, YMP memory and Tape are charged to T3D users as for YMP users.

ARSC users

Half of ARSC's users are affiliated with the DoD and network in from the Lower 48. These users are mainly interested in solving large problems and the big memory YMP is a particular draw for them. ARSC's T3D is their access to large memory via MPPs. The other half of ARSC's users are university researchers from around the U.S. but we also have users in Korea and Norway. Like most T3D sites, the YMP is the production machine and the T3D is the research machine. The goal of our site is to move more of this research into production mode on the T3D and to move production from the YMP into production on the T3D. This is particularly true for the large memory DoD codes that currently run on the YMP.

Classifying T3D Users

Some time ago, when I worked for an oil company in their seismic processing division, all geophysicists/programmers/processors were divided into four classes bases on their abilities:

1. develops a new seismic process
2. implements new process
3. modifies an existing process
4. runs process with different data sets

Managers used this division of abilities to allocate resources and plan projects. Employees quickly understood their position in the company and with respect to their coworkers. Such divisions, however distasteful, are necessary and inevitable. I have developed a similar set of characteristics to describe potential T3D users.

The problem with getting users running on MPPs is not the lack of problems or a lack of hardware, it is that the problem has not been programmed for MPP. The success on the T3D will be determined by the success of the users' programming on the T3D. To estimate this success, I classify potential T3D users and I use this classification to decide what level of support should be applied. Potential T3D users have three components that will determine their success on the T3D and each component is the general answer to a corresponding question:

Commitment	How much time are they willing to investing getting their code running?
MPP Experience	Have they run on other MPPs or is this their first attempt?
T3D Information	Do they have access to T3D information?

Commitment

As anyone who has programmed on the T3D will attest, it takes time. And many other activities are competing for this time. Users' commitment can be measured in the amount of time they are willing to invest. On the T3D, their time will be spent:

1. Doing a conversion from
   a. a YMP code
   b. or a workstation code
   c. or a existing MPP code
2. Measuring the performance
   a. of different algorithms
   b. for different configurations
3. Tracking down differences and bugs
   a. between serial and parallel
   b. between the YMP and T3D
4. Experimenting with
   a. data partitions
   b. library routines
   c. file systems and formats

Certainly programming on the T3D will take more time than learning a new editor or even a graphics package; Success will need time.

Two examples will show how I have used this classification of commitment. Many inquiries received by ARSC come from businessmen or entrepreneurs, who are unaware of the big programming effort to be made on the T3D. For these potential users, a problem exists and they think only the access to the machine inhibits their success. If they have considered the programming problem at all, they assume that it is someone else's problem or can be solved with the tools that came with the machine. Stressing the programming problem and determining who will do the programming sometimes puts an end to these potential users or correctly sends them looking for programming resources.

Another situation, common at the University of Alaska, is a professor that has both the 'big' problem and the experience to parallelize it but does not have the time to make it work. This is what graduate students are for. In my recent T3D class at UAF, there were three such pairs of a interested researcher and the grad student who would implement the project. The common idea of these two examples is that without someone willing to invest the time, a T3D project will not happen and User Services can not provide programming resources for all users' problems.

Sometimes a user's interest might lapse and User Services can encourage the user to try again or try with a new idea. To this end, User Services should be proactive in monitoring T3D usage and contact a user when the usage stops. To get some idea of T3D usage, I run a script built on mppmon that records usage. Commitment from the user is a valuable asset and User Services should try to build upon it.

MPP Experience

MPPs did not start with CRI's T3D. Going back to the '70s there have been multiprocessors like the Illiac IV, the Burroughs Scientific Processor, and the ICL DAP. Even in the commercial world there has been ten years of distributed memory machines, starting with the nCUBE, iPSC and Thinking Machines products offered in 1984. CRI and T3D

users have all benefited from these past experiences. The users with the best chance of success on the T3D are those users who have learned the hard lessons on earlier machines and are still trying.

Many users who have converted their application to run on any MPP have the experience to port that code to the T3D. The programming environment on the T3D is richer and more stable than many of the preceding MPP programming environments. There are underlying commonalties between all MPPs. For example, all MPPs had a "send" and "receive" of a message and the difference between MPPs is only a matter of syntax. Similarly a user who has programmed on MPPs before has an understanding of the general principles of

1. local access .vs. remote access
2. the expense of communication
3. load balancing

For these experienced users, Users Services' job is to provide them with up-to-date technical information and access to the machine.

As an example, at ARSC we have several users whose code is in PVM even though they know SHMEMs are faster. Because they run on several different MPPs the portability of their code is more important to them than the extra speed of being optimal on the T3D. If MPP users maintain their code in a portable way there is a variety of platforms available. This is the one of the few advantages in being a MPP user.

For first-time MPP users on the T3D, the situation is different. These users usually come from the YMP world and at the very least should have an understanding of a two speed computer and Amdahl's law. In the YMP world, the two speeds are scalar and vector and in the T3D world they are serial and parallel. This background is stressed in the first day of the ARSC T3D class where the students are given the history and characteristics of MPPs before they are shown the specifics of programming on the T3D.

Another advantage from the YMP world is that on the YMP there is an extensive set of tools to estimate the parallelism in their code. Flowview, perfview and atexpert can all be used to get an estimate of the parallel/serial distribution that will determine whether their code will run well on a MPP. With this estimate and an understanding of Amdahl's law the potential user can predict the level of success before any coding. With expectations at a reasonable level, potential users can go on to other issues such as using the large memory on the T3D and planning their conversion.

Without some experience with these programming and performance issues, it is almost inevitable that using the T3D will be a failure. The T3D hardware and software is probably the best in the MPP world today and if the user can't make it work on the T3D there is little hope that it will work somewhere else. The user must have a high level of programming ability and an appreciation of the performance issues. These can be taught in a training class but they usually take years to learn. There is no shortcut to success on MPPs.

At ARSC the YMP is heavily used, and some users have tried to used the T3D without realizing the programming effort involved. A Users Services consultant can go over the YMP flowview results from their code and maybe give them some suggestions on their YMP code before they jump into the T3D. This focuses the users onto the internals of their own code and away from the details of the T3D. Once the users have a good understanding of the performance issues on the YMP and why that performance is limited, then the conversion to the T3D is well motivated.

T3D Information

The one area where User Services can make a big difference is in providing information. Information about available libraries, reports, techniques, software changes can all determine success on the T3D. Without this information the user can become frustrated and discouraged. Their commitment can be squandered on problems that could be avoided with adequate information. There are certainly enough conceptual problems with making MPPs work, and to waste time on avoidable problems is a real loss.

User Services is in the best position for collecting and delivering this information. Sitting between CRI and the user, User Services can collect from both sides and transmit to the other. The sources of information are not just CRI but also other users. At ARSC, we have concentrated on a weekly newsletter as the vehicle to disseminate this information. Feedback from the users on the newsletter keeps us in contact with our active users. CRI, as a provider of a product, benefits from the user feedback carried to them by Users Services.

At ARSC we collect T3D information from:

Cray Research Incorporated	
CRIinform	known problems, software releases
CRSL	topics in depth
Onsite analysts	communication to and from Eagan
Regional analyst	future plans
NeWS net.groups	comp.cray,comp.parallel
Our users	problem reports, tricks
Other Cray Sites	available reports, papers manuals
T3D user's group mailing list	questions for study

The information we distribute is:

- Software and hardware status
  - releases
  - bug reports
- CRI manuals
- Publicly available articles
- Locally developed subjects

We distribute information with:

- ARSC's T3D newsletter
- T3D classes at UAF
- ARSC Mosaic server
- News on denali
  - MotD
  - /usr/local/example/mpp
- One to one communication
  - phone calls
  - email
  - U.S. mail

It is a full time job to monitor, collect and distribute this information. That is exactly why no user can be expected to have all this information and still have time to do their own programming. If a user thinks they can do it without access to this available information then they haven't yet started their effort. Having all this information in one place, like User Services, focuses this information. This maximizes every one's effort, Users Services collects and distributing information and users concentrate on programming their own codes.

Conclusion

No matter how the MPP hardware changes in the future, making MPPs useful will always be a software problem. Software tools from the vendor can help but these tools have not been standardized nor are they applicable to each user. CRI has done a exemplary job in providing tools like Craft Fortran, Apprentice and Totalview that make the T3D users' life easier than on most other systems. The user's software problem can be solved with time and effort but experience and access to information will smooth the path to success. User Services has a primary goal of providing comprehensive, current information to the user, this makes the best use of everyone's experience and effort.