HARDWARE FOR MUMPS: IS THERE A CHOICE?

Donna S. Beckman-Cunningham
Science Applications International Corporation
San Diego, California U.S.A.

ABSTRACT

No matter how beautifully it is programmed, every MUMPS software application requires a hardware platform to be useful. Choosing the platform and sizing appropriately can determine the success of the product. Factors such as CPU power, disk sizing, and system configuration (not to mention cost) must be carefully considered. Platforms must be adequate to give performance that meets user expectations and provide for system growth. Hardware technologies change at a blinding pace. No sooner are platforms identified than better, faster, cheaper systems become available. This paper presents a methodology for selecting hardware platforms, and includes considerations specific to MUMPS implementations.

INTRODUCTION

One of the most beneficial features of the MUMPS language is that it was conceived as an "open" system, in the sense that, as long as code conforms to the ANSI standard, an application can be ported to different hardware platforms and operating systems with a minimum amount of change. This flexibility offers the systems engineer the ability to choose the best hardware platform for a MUMPS-based system, and offers the opportunity to maximize system performance and minimize system cost through engineering. The freedom to choose virtually any hardware platform carries with it the responsibility to research all available technologies, often a daunting task given the wide range of possibilities and the speed at which technology changes. This paper will present some techniques for selecting hardware platforms and offer some insight on considerations specific to MUMPS-based applications.

BACKGROUND

Science Applications International Corporation (SAIC) is developing the Composite Health Care System (CHCS), a MUMPS-based hospital information system for the Department of Defense for use in military medical treatment facilities throughout the world. The system integrates data from various areas in the hospital into a single database. These areas include clinical departments, Laboratory, Radiology, Pharmacy, Patient Admission, and Patient Scheduling. The original systems were deployed on Digital's VAX systems, which met the needs of mid-range facilities. In order to reduce costs for smaller sized facilities, the system was ported to a UNIX-based implementation running on PCS. Configurations of five, eight, and thirteen nodes of 486-based PCS are being deployed to these smaller sites. A one node configuration with a subset of CHCS functionality is also being developed.

Because some very large sites were to be deployed, SAIC was asked to perform research into advanced technology platforms that would support implementation of CHCS to a facility with 2500 simultaneous active users that could support in excess of 200 gigabytes of stored data. The methodology employed to identify and evaluate possible platforms is described below. This methodology is utilized in evaluating nearly all new technology to be utilized by CHCS sites.

METHODOLOGY

The methodology presented in Table 1 can be adapted to evaluate technologies as encompassing as entire platforms or as limited as a single product (e.g., a laser printer). Each step is discussed in more detail below.

Table 1. Methodology.

Step 1	Establish requirements
Step 2	Establish selection criteria
Step 3	Research marketplace to determine possible solutions
Step 4	Analyze data to see which solutions meet requirements
Step 5	Apply selection criteria to remaining competitors
Step 6	Rank solutions based on scored
Step 7	Compare results to intuitive conclusions and review process if the winning solution doesn't "feel" right

Establishing Requirements

Requirements are features that a system must have in order to meet the basic need of the application. Sometimes these requirements are dictated by the customer. These requirements can also arise from the application itself. In order to determine requirements, the following questions should be asked:

1.) What is the cost ceiling that would eliminate a solution that is too expensive? The perfect technical solution may not be affordable. Costs should be taken into consideration at the beginning of the exercise to save time by not pursuing solutions that cannot be bought.

2.) Is the operating system specified?

3.) How many devices must be supported? How are they distributed among terminals, printers, and other devices such as laboratory instruments?

4.) How many simultaneous users must be supported? This is an important consideration when evaluating memory and CPU power.

5.) How much storage must be supported? Take into consideration that there is overhead associated with storing data in MUMPS globals (e.g., disk pointer blocks). SAIC assumed a storage efficiency of 75 percent. When estimating disk storage needs, determine what a reasonable rate of data growth is. With 80 GB of storage, SAIC assumed a growth rate of 3.5 GB per month based on observation of growth rates at existing sites.

6.) What are the security needs of the system?

7.) What percentage of time must the system be up and available? Availability in this context can be defined as MTBF (MTBF+MTTR) where MTBF is Mean Time Between Failures and MTTR is Mean Time to Recover. CHCS is on-line 24 hours per day, 7 days per week. These availability demands carry a price tag; redundancy is planned into the system and platforms are built so that backups are performed with the system on-line. Systems that can afford downtime for maintenance may have less stringent availability requirements, which could reduce the cost of the platform.

8.) What growth is predicted for the overall system? In addition to disk storage projections, the future need for memory and CPU power should be anticipated so that needed expansion can occur without having to replace the system completely. This is especially important when considering CPU power and memory.

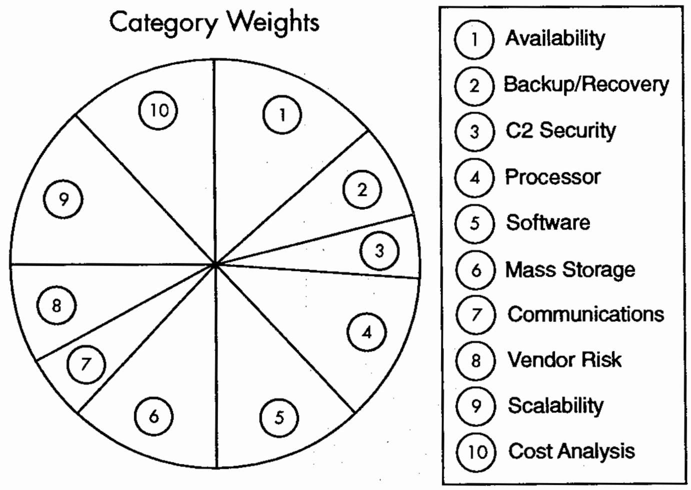
Establishing Selection Criteria

After candidates that do not meet the basic requirements are eliminated, selection criteria can be used to distinguish among the remaining candidates. After establishing the criteria, assign a weight to each one. The higher the number assigned, the more important the category. The evaluation task is easier if the weights add up to an even number, e.g., 100. These weights should be assigned based on the relative importance of the category. Remember: these selection criteria are applied after it has been ascertained that the candidate meets the basic requirements.
The categories presented in Table 2 were used by SAIC in evaluating high-end platforms for CHCS.

Table 2. Selection Categories.

Category	Considerations
Availability	MTBF, availability factors above minimum required
Backup/Recovery	Ease of use, speed of recovery
C2 Security	C2 (Government) certification status, security features
Processor	Power, speed
Software	Operating system, MUMPS, ease of conversion from present system
Scalability	Changes needed to scale platform to mid-size and small sites
Cost Analysis	Total cost, including maintenance
Vendor Risk	Viability of vendor, financial soundness , size of customer base
Communications	Compatibility with deployed communications architecture
Mass Storage	Disk subsystem, capacity, I/O

The category of "Security", for example received the fewest number of points, not because it was the least important, but because once it had been ascertained that the system met the requirements for security, additional security features were not judged to be as important as other features (for example, the flexibility of scaling the proposed solution downward or upward as new requirements were defined). Athough cost was an important consideration, the customer for CHCS will not compromise system availability to save cost, and therefore cost did not carry as much weight as availability.



Scores are assigned by giving the best candidate the highest score. The range for scoring can be established in a variety of ways. The candidates can be ranked from best to worst using the scores 1-n, where "n" equals the number of candidates evaluated, or features can be rated on the basis of their desirability. Table 3 presents an example of how ratings can be determined:

Table 3. Rating Chart.

Value	Description
5	Significantly exceeds need
4	Exceeds basic need
3	Meets basic need
2	Barely meets basic need
1	Questionable

Researching the Marketplace

The research into possible solutions is usually the easiest step since vendors are quite willing to cooperate in providing information. Other sources such as trade publications can be used. In San Diego, SAIC has consulted information databases in our local university libraries as well as scanned the Internet for relevant articles. Some publications such as PC Week perform product comparisons and their results can be useful in determining which technologies should be considered. In the high-end architecture study, after narrowing the possibilities to six vendors, SAIC issued a "Request for Information" (RFI) asking these vendors to propose a specific platform. The vendors were given both the requirements and the selection criteria, although they were not given the weights of the criteria.

Eliminating Noncompliant Solutions

Utilizing a checklist, each solution should be evaluated to ascertain that it meets minimum requirements. This is a boolean operation; all questions must be answered "yes" for the solution to survive this step.

Applying Selection Criteria

The application of the selection criteria is the most subjective step of the process and it is helpful if several people participate. Assigning scores to how well the criteria are met objectifies the process, and the engineering knowledge and experience of the evaluators is a key element in the success of the enterprise. By evaluating and scoring each criteria, a mathematical basis for choice is established.

The following example is derived from the comparison of I/O throughputs of the six systems SAIC evaluated for the high-end computing platform. Out of a total of 600 points, this category represented 24 points. I/O throughput was given a weight of 4. The maximum possible score was 6 (6x4=24).

Vendor 1 was judged to provide more than adequate I/O bandwidth. As disks were added, it was not anticipated that they would have any direct effect on individual system performance. Vendors 2 and 3 seemed to have more than adequate I/O bandwidth but it appeared more likely that the processors would be affected as more Small Computer System Interfae (SCSI) controllers and disks were added to the systems. Vendor 4's solution seemed undersized to meet performance needs. Vendor 5's solution presented the possibility that the proposed disk architecture would overwhelm the system bus. Vendor 6's solution seemed bound by the bandwidth of the Ethernet, which presented a significant bottleneck. Table 3 is an example of how scoring was done.

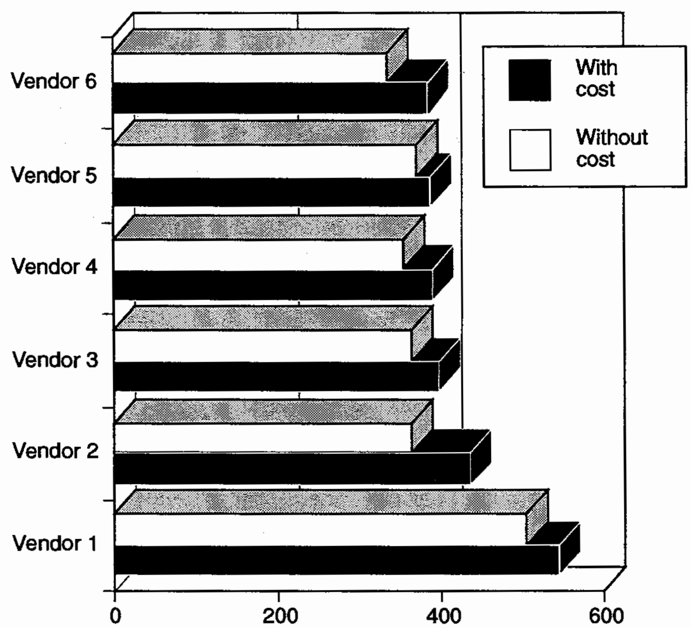
Table 4. Scoring of selection criteria.

VENDOR	WEIGHT	SCORE	TOTAL
#1	4	6	24
#2	4	5	20
#3	4	5	20
#4	4	4	16
#5	4	3	12
#6	4	1	4

Ranking the Competitors

By adding up all of the scores for each category of a candidate solution, the competitors can be ranked. Results can be depicted graphically. It is also useful to compile two rankings, one including cost and the other excluding cost. This gives a picture of the best technical solution as well as the best overall solution.

Architecture Evaluation



Engineer's Intuition

If after evaluating the results, the final ranking does not seem right, the scoring should be reassessed. In some cases, the weight given each category may need to be changed. Objectifying the selection process does not mean that intuition or "gut feeling" should be ignored.

CONSIDERATIONS FOR MUMPS-BASED SYSTEMS

The following considerations may influence choosing platforms for MUMPS based systems:

Read/Write Ratios

When evaluating technologies such as RAID, be aware that a MUMPS-based system may read from cache but write to disk. This means that you may be sizing a disk subsystem based on the assumption that your application is "read" intensive when it may actually be more write intensive than you realize. From a programming perspective, CHCS activity consists of 80 percent reads and 20 percent writes. Because of caching, from an I/O subsystem perspective, CHCS actually is a 65/45 application. For this reason, disk caching is not as advantageous to a MUMPS-based system as to other types of applications.

Using TPC Benchmark A to Evaluate Performance

The Transaction Processing Performance Council (TPC) is a standards group that has developed a benchmark test to compare multiuser database systems. This benchmark simulates a banking application consisting of debit and credit transactions and is considered applicable to MUMPS-based systems. The resulting measure is the "tps" or Transactions Per Second.

After evaluating installed systems and extrapolating what the requirements of a high-end architecture would be, SAIC determined that a TPC-A rating of 1,300 tps would be required.[1]

[1] SAIC. "High-End Computing Platform for the Composite Health Care System"

This was extrapolated from installed systems with the following workload:

1.) 825 users at peak periods.
2.) 240 physical read/write I/Os per second (55% reads, 45% writes).
3.) 382 tpsA.

Vendors publish their TPC-A ratings. Scoring should be done on the credibility of the vendor's tps claims.

Memory and Symbol Table Size

Making the Symbol Table larger has a tendency to improve performance for a single task, however, on a system with many users, the maximum amount of memory that can be used for symbol table space is the amount needed for fixed operating system tasks (OSMEM) subtracted from the total amount of memory (MEM), divided by the maximum expected number of users (MAXUSERS):

$$SYMBOL\ TABLE\ SIZE = \frac{MEM - OSMEM}{MAXUSERS}$$

This number ended up being smaller than expected! This is because the effects are cumulative; in a 1000 user system, for every kilobyte of memory you give a user, you have to add a megabyte of memory to the machine. For this reason, memory is an important concern when sizing systems for MUMPS implementations.

Distribution of the Database over Physical Disks

The net access time experienced by a user improves when data is spread out over multiple spindles. Keep in mind that even if a global variable is so large that it spans six complete disk drives, the main activity related to that global variable will take place in the physical location where the most current data is being stored. The trick is to find the perfect compromise between spreading a global variable over as many disks as possible versus keeping the data on as few physical resources as possible. Putting a number of small volume files on one disk saves money. Spreading these files across twelve disks costs more but results in better performance. If one needs to buy twelve disks anyway to hold a database, it is better to spread the data across the disks, making sure to leave adequate growth area on each disk.

CONCLUSION

The choices faced when choosing a hardware platform are much the same as any type of consumer decision, for example, buying an automobile. Excess memory and disk space are always good to have (as are leather seats and fuel injection). Through careful analysis, however, the systems engineer can determine which platform can meet the need of the application, provide the customer with satisfaction, and optimize the price/performance ratio.

Note: An earlier version of this paper was presented at MTA-Europe in Luxembourg, 1994, and was published in M Professional Magazine.

Bibliography

B. Herring, "TPC Benchmark A - An Industry Standard Performance Test", *MUG Quarterly*, 10:1.

T. Hock, "Proposal Evaluation, Trade Study Best Practices." , (SAIC, 1994).

D. Cunningham, "Systems Engineering Analysis and Recommendation Report: High-End Computing Platform for the Composite Health Care System", (CHCS) (SAIC, 1994).