OPERATIONS MANAGEMENT IN A MULTI-HP3000 ENVIRONMENT

Roberto Drassinower
Teresa Brzozowski
Carolian Systems International Inc.
3397 American Drive, #5
Mississauga, Ontario
L4V 1T8 CANADA

This paper discusses centralization and automation in a multi-HP3000 environment. It offers suggestions on how to plan for and introduce network-wide operations control, thus making large shops more manageable and efficient. This paper deals with centralization and automation as two distinct issues: where centralization is required before automation can begin.

Introduction: The Operations Optimization Project

In recent years, multi-system installations have become more common in the HP3000 community. While the advantages of both distributed and local network systems are often obvious, the challenges associated with their management as one coherent unit are all too often underestimated. Since each machine is typically monitored and controlled through its independent console, an overall perspective of network wide operations is difficult, if not impossible to obtain. Yet sensitivity to global issues and centralized decision-making are essential to the efficient management of large scale integrated systems. The problem is vastly complicated when a central site has to support remote sites. Often a lack of adequate operations staff limits control over production and severely impairs the flow of system status information back to head office. Similarly, the sheer size and complexity of local multi-machine installations can also render traditional operations management techniques ineffective.

The possibilities for streamlining networked operations do exist in an HP3000 environment. Careful planning can offer existing operations staff greater control over distributed systems from one location making them more efficient. At the same time, the resulting centralization of information provides management with a base upon which to build sensible network-wide policies and procedures for production. Such policies inevitably lead to improvements in overall system throughput and higher system utilization. Moreover, continually reviewing operations management systems guarantees the optimal use of existing resources as installations grow. What is needed, and is sometimes missing, is a firm commitment to an on-going, long-term operations optimization project. Although this will divert resources away from satisfying some demands of the data processing department, failure to review procedures may significantly erode its operating efficiency and ability to service the company in the long term. Since the results of your labour will be a long term solution, you

can implement distinct phases without incurring prohibitive expenses, while at the same time remaining confident of a proper design and direction. In the long run, it will be well worth the effort.

In tackling the issues of centralizing and automating your DP shop, it is important to understand fully how your operations work and more importantly, how you would like your shop to work. Therefore, I would like to review some of the basic concepts about the operations environment before continuing.

Today, HP3000's are controlled by an operator through local console. This system console is the lifeline to the HP3000, since all operations status messages pass through it. It is a hub where all problems can be defined into two categories: those that require human intervention, and those that report an error condition to which someone should be alerted. In this scenario, it is assumed that trained operators are constantly monitoring the console screen so that they can respond to the requirements of the message immediately. This also assumes that the operators are trained in a variety of areas, so that they can handle job management issues, system security policies, internal system management, data comm, and so on. In short, good operators have to specialize in many different areas - a jack of all trades. Traditionally, the operators must also contend with a variety of messages which are irrelevant to their primary tasks. For example, if the operator is mainly concerned with executing and monitoring an installation's production schedule, then that person will only want to view job completion and error messages from among all activity passing through the console: logon messages, HPDesk messages, data comm messages and so on. The result is that the operator has to act as a human filter. And, as the site becomes more active or complex, the operator's job becomes increasingly difficult. The result is that often critical information will merely scroll off of a busy console and disappear, at which point the operator is faced with paging through hard-copy STDLISTs.

Now imagine this task in a multi-machine environment. Busy shops might need an operator per system in order to keep operations running smoothly. Keep in mind, however, that this setup makes centralization difficult by the fact that each console is physically separate from the next. Messages coming from multiple machines cannot be grouped logically. And an operator always has to be at the console to respond to messages. This can mean a lot of wasted time for a number of operators, since they are only concerned with a small percentage of the total messages.

The First Step to Improvement

Throughout this discussion I will be making general recommendations which, when grouped together, provide the basis for a long-term operations review plan. The first of these is as

fundamental as it is self-evident: commit resources to examining existing operations practises through a long-term operations optimization project. It needs to be an on-going project so that your operations optimization can accommodate changes in demands.

As I alluded to earlier, the most substantial obstacle this kind of project must contend with is the difficult task of providing operations staff with both the information and control required to efficiently oversee multiple systems. Crucial to overcoming this problem is to determine the nature of these needs precisely, such that they can be satisfied as efficiently as possible.

As you may have guessed, the issues that must be confronted during such a study are not unlike those a systems analyst must wrestle with while designing an order entry system. For example, this individual would be concerned about the kind of system access data entry personnel require and the content of management reports. Quite clearly, different individuals often will require highly specific information which is of little consequence to others. On the other hand, recognizing common needs is just as important to the design of an integrated system.

In designing operations management systems much the same is true. Operators will need access to the consoles of all systems. In addition, particular operators may only be interested in a subset of all console messages. An operator charged with servicing tape drives will have keen interest in tape mount requests but will not be so concerned about interactive users logging on and off. A security specialist, on the other hand, should be aware of remote users logging on through dial-up modems. A tape librarian will profit from reports regarding the frequency in which particular tapes are used, and from knowing when these tapes are needed during the production schedule.

To design a proper operations management system you will need to identify the flow of information within your company: from users to operations staff and technical support, so that you are sure to have accounted for all types of messages. This will help you to identify the information which is critical to your operations and who should receive these messages. It is only when you have a good understanding of your environment that you can move towards centralization, the separation of tasks and automation.

Why Centralize?

For the same reasons that you want to divide console messages into logical categories for your operators, you should aim to structure your DP department in a manner that reflects the logical structure of your installation. In multi-machine environments, machines are rarely isolated from one another with respect to your company's production. Most shops network their machines in order to ease the transfer of information. There are usually many dependencies among machines for production, so the independent management of

each HP3000 is an inefficient way to control your DP environment. It would be better to have control and management of multiple systems centralized so that all information can be easily related together. The result is that DP managers can make better management decisions and increase overall network uptime because they are working in a more controlled environment.

Some of you probably have, or are already taking steps towards centralization today. Batch scheduling software is a good example. It allows you to do sophisticated batch job scheduling and gives you the ability to have cross-machine job dependencies. Such utilities are available from third party vendors such as OCS and Unison. By knowing the status of jobs on all of your systems at one central site, you can truly run a distributed network efficiently. When provided with up-to-date information about the status of all your systems and by having all production scheduling information routed back to a central site, you can schedule jobs throughout a network and load balance across a number of systems. You can maximize the use of your resources by having all systems available from a central location.

Centralization also has implications on system security. By having central control of all consoles, you are reducing the number of access paths to your systems; that is, fewer people need the high level capabilities for operating your machines. By minimizing the paths for access to your systems, you increase security. Secondly, since all console messages are centralized, it is easier to watch for unauthorized access. Logon violations can be grouped together for easier identification.

Centralization of your consoles will also allow you to centralize your MIS staff: both operators and specialists. If all machines can be accessed and full console control is available from one location, then you don't need to have duplicate staff at remote sites. Because commands can be issued centrally, your resources and expertise can be at one site.

Towards Automation

Once centralization is achieved, you can start to consider automation. It is, however, important to note that automation is a relative term. It should be viewed as an effort to reduce human intervention without necessarily eliminating it. So, the real issue is how much to automate. Significant gains in reduced intervention can be attained without substantial costs associated with automation and, in particular, artificial intelligence applications. Also, automation comes with time: until you know exactly how and what to automate, you shouldn't try to do it.

Before implementing an automation strategy, you need to know exactly what to automate. The key ingredient for such an analysis is information: log all your operations activity, that is, the console traffic over a period of time. Next, analyze it to see what can be automated, design a plan and only then start to

implement procedures. Undoubtedly, you would have gone through a similar process in implementing a job scheduling package.

In the instance of a job scheduling package, it's a tool that most shops "grow in to". In many cases, system managers start out by using the MPE :STREAM JOBxxx;AT=time command to run jobs at night unattended. Running jobs overnight is often the solution to completing processing without effecting performance for on-line users. However, where job dependencies exist and jobs have to be run sequentially, using the STREAM command may not be enough. With STREAM, you would have to guess when job A will complete so that you can stream job B to run once A has finished. If your predictions are off, jobs could be colliding or alternatively, you could have a lot of idle time on your system. At this point, a shop considers a job scheduling package to automate all overnight procedures. Once the scheduling package is installed, it is up to you to carefully plan your production so that processing continues properly. Typically, when people learn that the package is up and running, they will find more and more jobs that can be scheduled into it. Anything that can be automated, will be automated over time. Most shops find that automating job scheduling makes life easier for operators as there is less of a need for human intervention. The result is an increase in overall productivity of your machine since available processing time has been optimized.

In automating your entire operations, the process for implementation is much the same: review what you are currently doing, analyze what can be automated, plan for implementation, and implement. The job doesn't stop there either. Just as jobs continue to be added to a scheduling package, you should continue to evaluate tasks as candidates for automation. After the initial implementation, your focus becomes one of review and analysis of tasks to be automated.

Where Should Analysis Start?

Listing from your console will give you an accurate picture of what happens on a system from an operational point of view. From a log file, you need to identify which mechanical tasks it is possible to automate. In many cases, either HP or a third party offers a way of automating a task to varying degrees (as in the example of batch job scheduling). Currently, there is a wide variety of software packages that can be purchased to help you automate: automatic error detection, high-speed backup, database manipulation, spooling, performance analysis, and so on. An in-depth study of console activity and your overall operational goals will help you to decide which products may be of benefit to your shop. The key to these tools when moving towards automation is that they are programmable, where expert systems and artificial intelligence may be closest to complete automation.

As I discussed earlier, the most critical console messages are those that require human intervention. These could range anywhere

from a reply to a tape request to correcting an error in a job to restarting a failed system. In each case, someone has to be made aware of the situation, otherwise, processing will not continue as planned. What is needed is some sort of alert mechanism. Depending on the criticality of the message, the appropriate reporting mechanism could require a console command to be issued, a message to be sent to a specialist, or a programmer on call to be contacted. Through an alert mechanism, you will know how much human interaction is necessary and where. This is invaluable information for your operations automation plan to help determine your requirements for staffing and software tools.

Simply stated, the rationale for analyzing your console activity, is so that you will know exactly what to automate and how. This helps to make sure that all bases are covered when moving to automation. The more information you have about your console activity, the more situations you can account for. Again, going back to the job scheduling example, if you know what the various types of possible errors are, then you can set up a number of contingency plans to compensate for those errors. Another example is system failures and automation. Through analysis of your console activity, you will know what are the most common reasons for system downtime, and you can then set up procedures to restart your system, tailoring the restart to the type of failure.

Also remember that automation is something that you can grow in to. In the early stages, the mechanism you use to alert personnel to critical situations may simply be a beeping message at the console. Later, it may grow to include a voice messaging system that automatically dials a programmer on call. To handle the problem of loading tapes and replying to requests, some companies are going so far as to get robots to mount tapes and then a reply can be programmed to execute automatically.

Even for less critical activity, it is a good idea to also log information so that you can monitor the demand for certain types of tasks. Again, this will also help your specialists to complete their jobs more efficiently. Take the example of system performance. Both Hewlett-Packard and Carolian Systems are taking steps towards making it easier for users to centralize and automate performance analysis through the creation of an interface between a performance tool and the console. HP has chosen to tie OPT/3000 to their recently announced OpenView product which will allow performance relation information to come back to a central site. Carolian has added a module to SYSVIEW, so that it can run in a "monitor" mode, which is designed to pass information about unusual performance situations back to a central console.

In summary, in order to manage multi-HP3000 sites effectively, centralization and automation of console activity are necessary. Careful analysis and logging of information is required to develop an effective operations optimization strategy. Once your objectives have been defined, you can start to implement your plans. Centralization of control of all HP3000's will increase

your control, allow for more standardization and optimize the use of both hardware and human resources.

Automation should be implemented over time, so that you can automate as many different tasks as possible. Any tools that you choose to use to help you automate should be flexible, programmable and allow for growth. Automation frees up time for your operators so that they can concentrate on critical tasks. As utilities become more and more sophisticated, it may even be possible to program these tasks.

From reading through any of the HP publications, you will note that centralization and automation are the directions of many vendors and users alike. Many companies are realizing the need for standardization and efficient use of both hardware and human resources, and hence the interest in centralization and automation.