A First Look at Performance on the HP 3000

Presenter: Sherri Cook with Lund Performance Solutions
428 South East Second Avenue
Albany, OR. 97321
(503)926-3800

Performance on the HP 3000, 9xx systems can be affected by many different factors. This session will provide insight to first year system managers on those factors.

In performance management, we are not so concerned about a few, short bursts of heavy usage, but rather constant heavy usage. In this paper you will see the terms “green zone”, “yellow zone” and “red zone” performance indicators. LPS has found it helpful to divide the various performance indicators into these three stages. Keep in mind that the key to precise performance diagnosis is consistent indicators.

- Green zone - good; no action is needed
- Yellow zone - marginal; deserves close monitoring
- Red zone - critical; bottleneck exists requiring a resolution

First, an important definition: A bottleneck, by definition, is a choking off or a restriction in the flow of a transaction’s progress. When a necessary system resource becomes short in supply, batch and user requests must queue or wait until that resource becomes available. Today we will look at the three potential bottleneck areas system managers should watch:

1) Disk
2) Memory
3) CPU

1) Disk Factors:

There are two areas to consider when managing your disk drives. First, you need to manage your disk space. Managing disk space is very important. Once you have run out of free disk space processing could STOP. To better manage your disk space you should:

- Perform a periodic check on total free space as well as fragmentation levels. Keep in mind that you might have enough total space, but it could be broken into so many pieces it potentially becomes very inefficient.
- Regularly search and destroy files that have not been recently used. A tool such as MPEX (Vesoft) will help you accomplish this. You may want to insure all files are backed up (archived) prior to purging them off of the system. This will give retrieval if someone comes looking for a “destroyed” file.
- Address file fragmentation. You want all of your free space contiguous. To do this you need to either perform a system reload or run a third part disk maintenance tool such as De-Frag/X from LPS.
- When planning to upgrade, remember to plan in tandem with all resources. This means when you need more of one resource, there is a good chance other resources could be getting scarce. When considering a CPU upgrade, for example, be sure to consider more memory and disk space.

Another reason it is desirable to manage disk devices is due to the finite limit of main memory. Permanent data has to be kept somewhere, generally on a disk device. In the perfect world all data would be loaded into main memory and left there, which would allow faster access time than reading from disk. Since this is not possible, the next best thing is to optimize I/O throughput. The following graphs illustrate how to use performance data to determine the balance of data to disks and disk load in conjunction with disk capacity.

- Disk I/O by Drive - This allows you to view the relative activity of each of your physical disk devices. This will help identify disk drives that are doing all of the I/O. Ideally, you want equal I/O on drives of equal speed and size. The I/O rates are fairly evenly balanced in Figure 1.

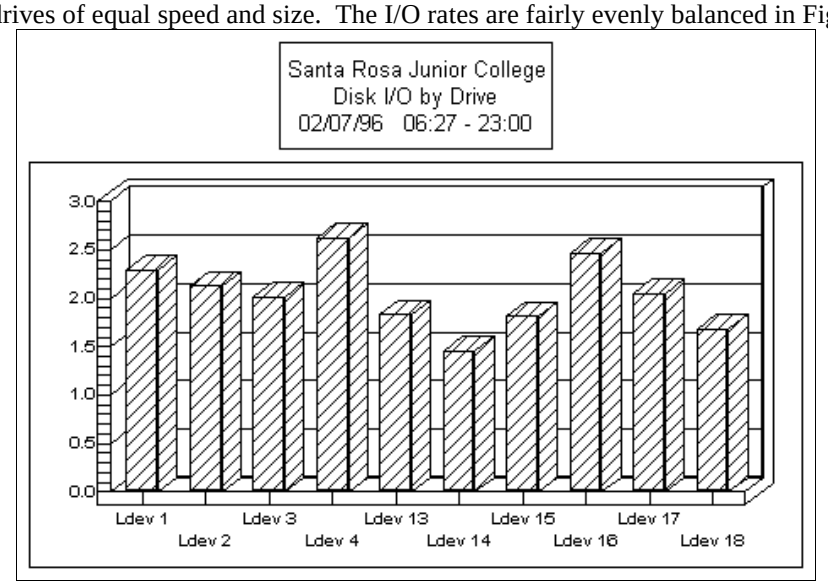

Figure 1

- Disk Service time by drive - This is the time it takes for a disk read or write to complete. The disk service time by drive can be indicative of the type of disk device (SCSI, HPIB, or HPFL). For example if the service time for one drive runs from 12 to 25 milliseconds, this would indicate it is SCSI technology. If the disk service time is from 30 and up, this would indicate HPIB or HPFL in nature, which is older technology.

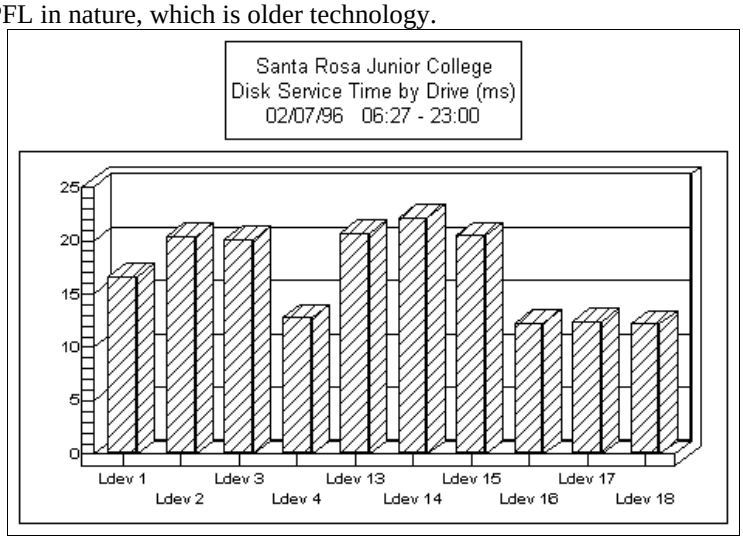

figure 2

Therefore, we are able to deduce from figure 2, that these devices are probably SCSI.

- Disk I/O Queue Length - This is the average number of disk I/O requests waiting for a specific disk drive. The high end of the green zone threshold is .5 I/O's per second.

"Green Zone"	"Yellow Zone"	"Red Zone"
<.5	.5 - 1.0	>1.0

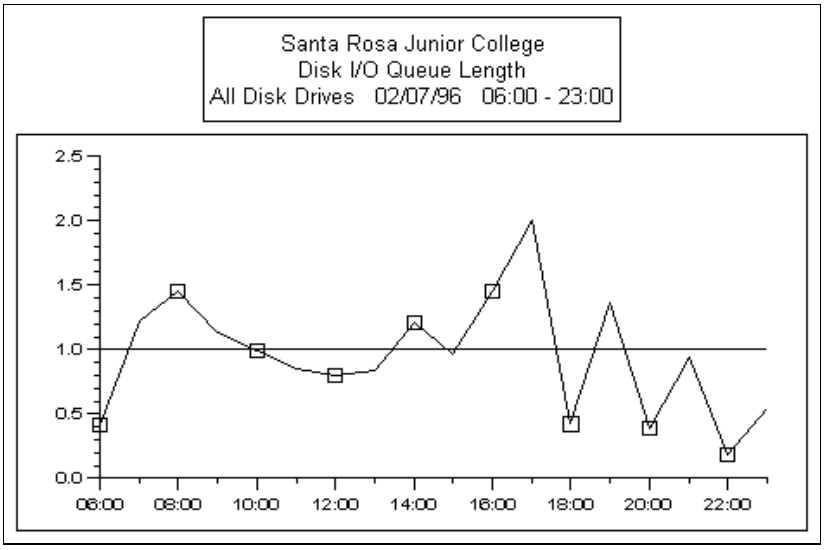

Figure 3

A high average Disk Queue Length could indicate the following:

1. Poor data locality - Data being retrieved is scattered all over a disk or within a particular file.
2. Not enough memory exists to perform pre-fretching.

- Read Hit Percentage - The percentage of time that disk read requests were satisfied in main memory. High values are desirable. The Read Hit percentage is a function primarily of data locality (locality is how likely future data retrieved with current data will be needed).

"Green Zone"	"Yellow Zone"	"Red Zone"
>95	85 - 95	<85

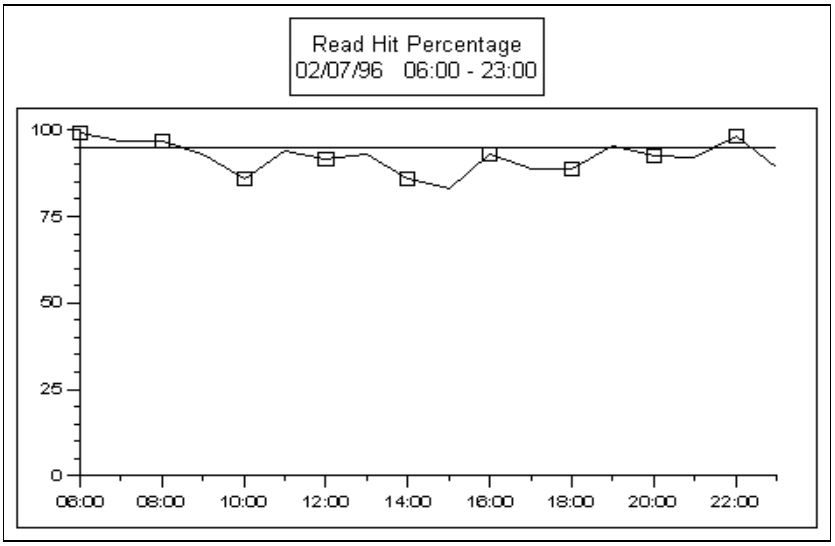

Figure 4

Figure 3 and 4 reveals a general "yellow" to "red" zone for data locality.

- CPU Pause for Disk - The time that the CPU was waiting (paused) for disk I/O's to complete. CPU paused for disk can be caused by poor data locality and means that the CPU had to wait instead of work. When pause for disk ranges above the 5% it can also indicate a memory bottleneck. Figure 4 shows a good pause for disk indication for the day.

"Green Zone"	"Yellow Zone"	"Red Zone"
<5	5 - 15	>15

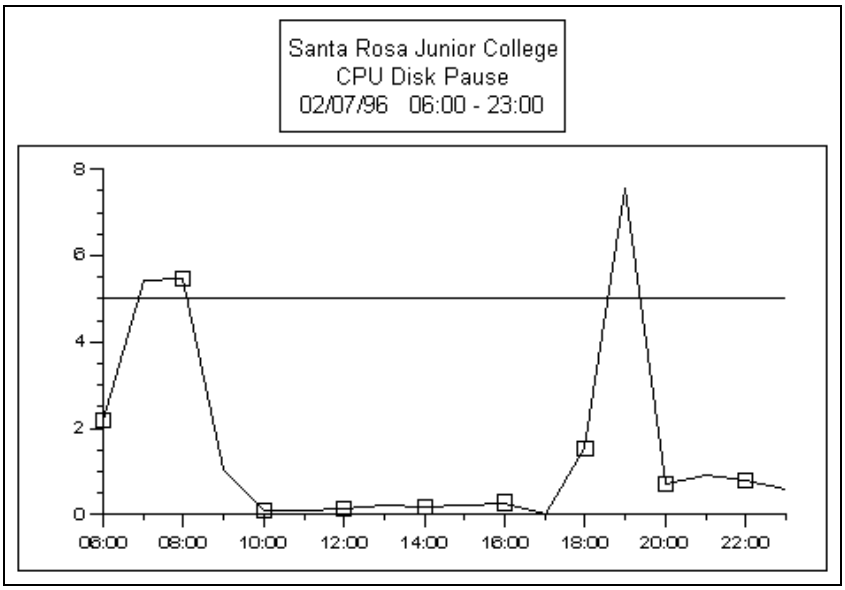

Figure 5

Helpful hints on maintaining your disk throughput:

- Check file structures for inefficiencies. For example make sure your detail and master sets are on separate disk drives.
- Check for I/O balancing across disk devices. You can use FILERPT for this. You will find FILERPT in the contributed library and also LPS distributes it on the LPS tapes.
- Data locality - Bullet number 1 will improve data locality within databases. Other ideas are:
  1. Archive history, the more data you have, the more your application may have to search.
  2. Utilize chain re-packing features in tools such as Adager, DBGeneral, and Flexibase
  3. Examine users' data entry/inquiry habits and see if from an application standpoint you can help minimize any data retrieval randomness.
  4. File Fragmentation: File Fragmentation is a phenomena in which a file's set of extents become physically disassociated. Fragmentation can be addressed by doing a system reload or by running "De-Frag/X" from LPS.

2) Memory factor:

Main memory is used by the CPU as a scratch pad or workbench. In order for any work to be performed, components of that work program code and data must be resident in main memory. The bottom line with memory, you have to have enough. Better to have too much than not enough. The fabulous 9xx systems perform very well with ample memory, but are virtually crippled by a memory shortage.

- Rule of thumb when deciding on HOW MUCH MEMORY:

30 to 50 megabytes for the MPE/iX operating system (if on 5.0 lean 50)
1 to 2 megabytes per user
2 to 4 megabytes per batch job

Good user response times depend on an adequate amount of main memory being available, otherwise elevated disk activity could result. The result can increased user response time and decreased batch throughput. HPPA systems were designed with large amounts of memory in mind. A shortage will incur disk accesses for transient paging (moving memory pages from memory to virtual storage to make room for processes which need memory space). Keep in mind that when a secondary storage I/O occurs, your transaction response time is negatively impacted.

An adequate amount of memory is very important. The following are the main indicators for a memory shortage:

- Memory Manager - This includes the time spent on memory allocation for users processes which cannot be given the CPU's attention (launched) until necessary segments are made present in memory. The CPU should never spend more then 8% of its time managing memory. Ideally you want it to be below 4% but the high end systems levels should be less then 8%. Figure 6 points to adequate amount of main memory.

"Green Zone"	"Yellow Zone"	"Red Zone"
<4	4 - 8	>8

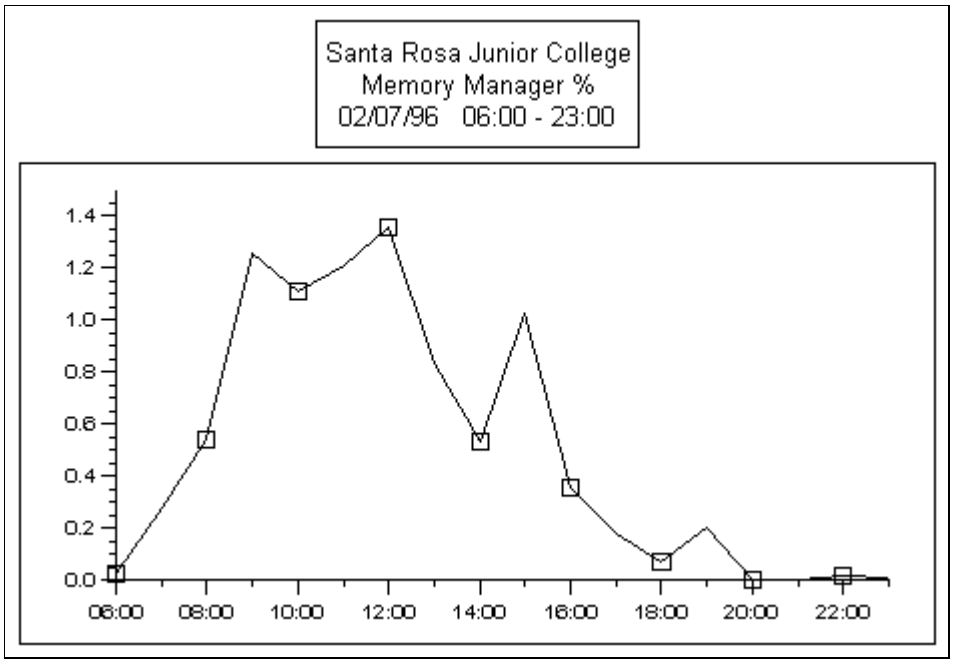

Figure 6

- Page Fault Rate - A page fault is requested whenever an object is not found in main memory. The following chart will help decide how many page faults are acceptable on your system. Figure 7 shows a slight pressure on memory. This indicator deserves close monitoring.

Size of system	Green	Yellow	Red
Small 920/922/925/932/935	less then 4	5-7	8 or above
Medium 917-947/949/918-928	less then 8	9-11	12 or above
Large 950/955/967/968/978/977	less then 13	14-18	19 or above
Very Large 987/987-xxx/988/980-xxx/990	less then 20	20-40	40 or above
Extra Large 992/995-xxx	40	40-60	60 or above
General Rule	less then 10	10-25	25 or above

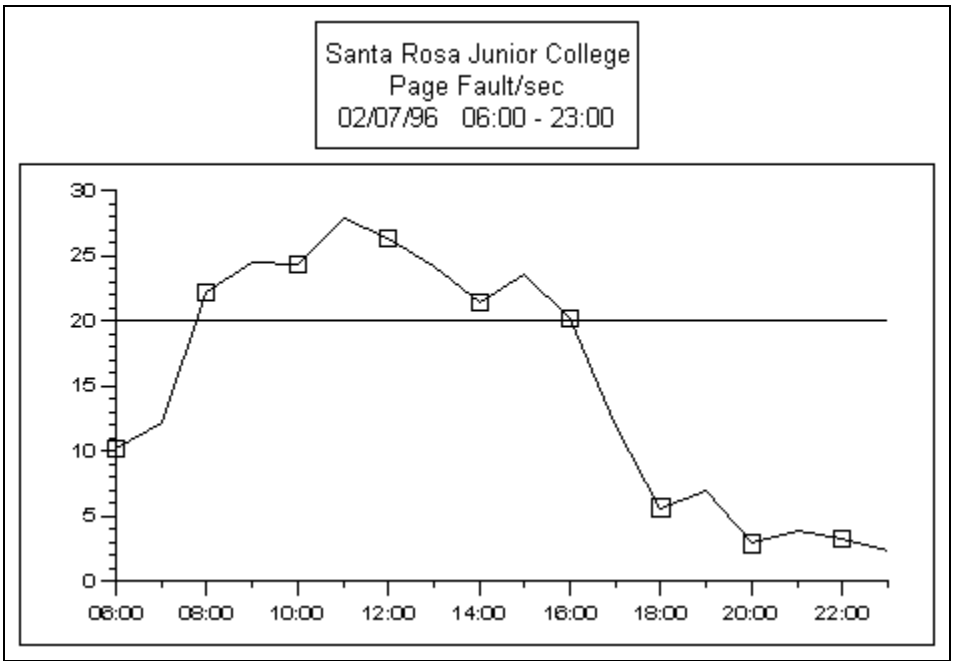

Figure 7

The bottom line for memory is, better have to much than not enough!

3) CPU Factor:

The CPU is considered the engine within a computer system. The following indicators will assist you in determining if you have a CPU bottleneck.

- CPU total busy - The total time the CPU was expending "horsepower" rather than waiting for events to take place. The total amount of work done includes both useful and overhead tasks. The red zone threshold is 85%. The total busy consists of the following:

"Green Zone"	"Yellow Zone"	"Red Zone"
<50	50 - 85	>85

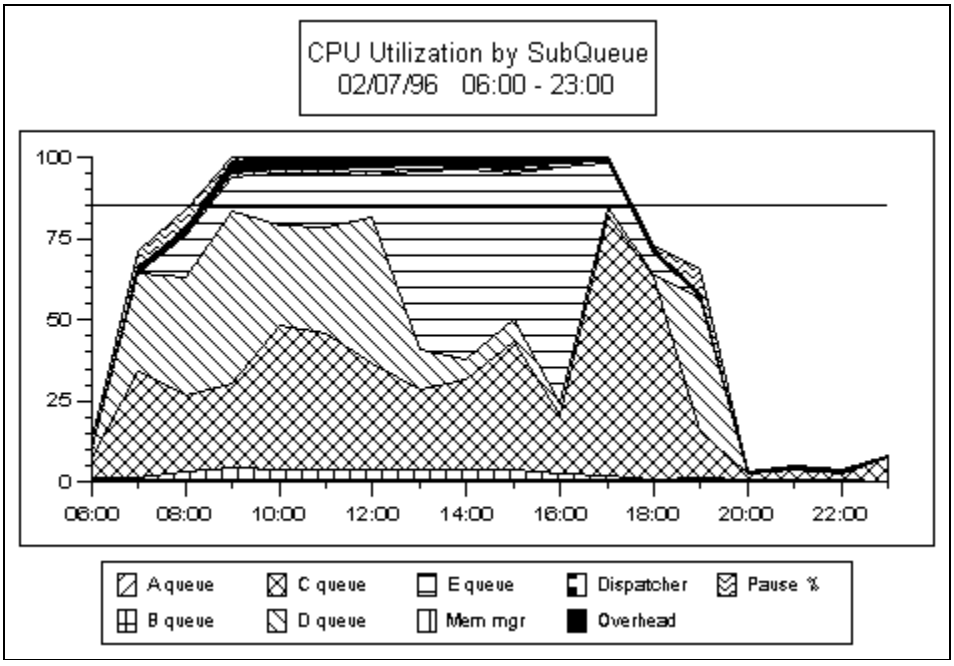

Figure 8

1. Busy on user processing. This category is considered to be useful CPU time spent on behalf of user activity.
2. Busy on allocating memory. It is the time the cpu spent allocating and moving pages of memory. See the above notes on the memory indicators for more on memory.
3. Busy on overhead tasks. (Dispatcher activity terminal I/O, disk requests, etc....)
4. Pause for disk. The time the CPU could have been busy performing productive tasks but was hindered in doing so because necessary pieces of data were missing.
5. CPU Idle. The CPU had nothing to do and was not waiting on any event.

- CPU Compatibly Mode - CPU compatibility mode is how often the CPU has to switch code from native mode to compatibility mode. This could be very bad for your system performance. Most systems will see slight amount of CPU compatibility mode due to the fact the operating system uses compatibility mode for some of its operations. The red zone threshold is 50%. Recompile your programs with MPE/iX native mode compilers to take full advantage of the Hewlett-Packard precision Architecture, if at all possible. Figure 9 shows a large amount of compatibility mode. SRJC should investigate the cause of this.

"Green Zone"	"Yellow Zone"	"Red Zone"
<10	10 - 50	>50

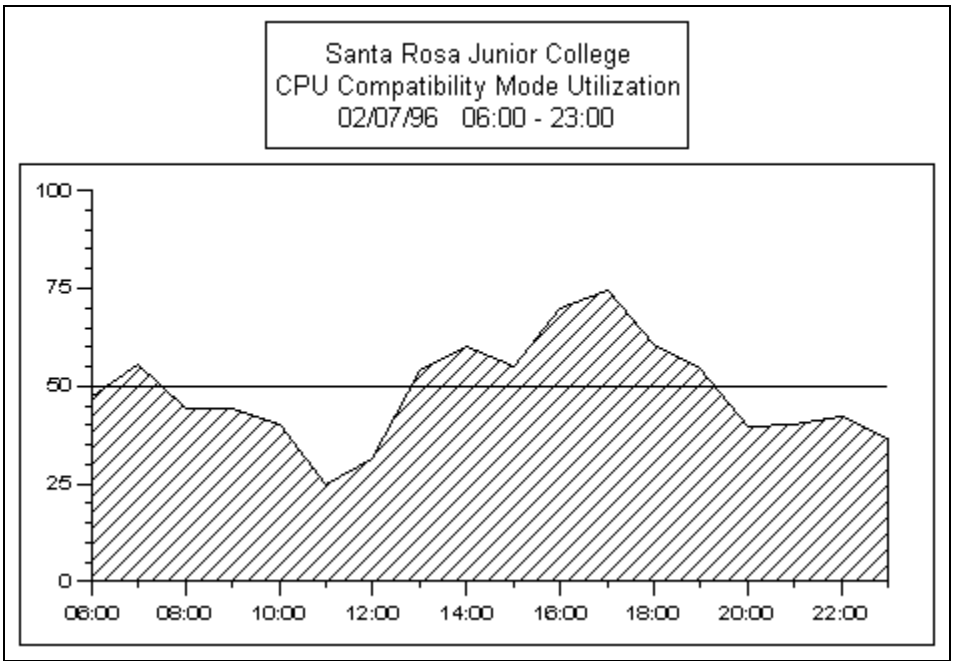

Figure 9

- CPU Queue Length. This is the number of processes which require use of the CPU in order to continue processing. The higher the CPU queue length, the less the CPU is able to satisfy current demands. The "red zone' threshold is 15. You should never have more then fifteen process waiting for the CPU. Above this number could indicate a CPU bottleneck.

"Green Zone"	"Yellow Zone"	"Red Zone"
<5	5 - 15	>15

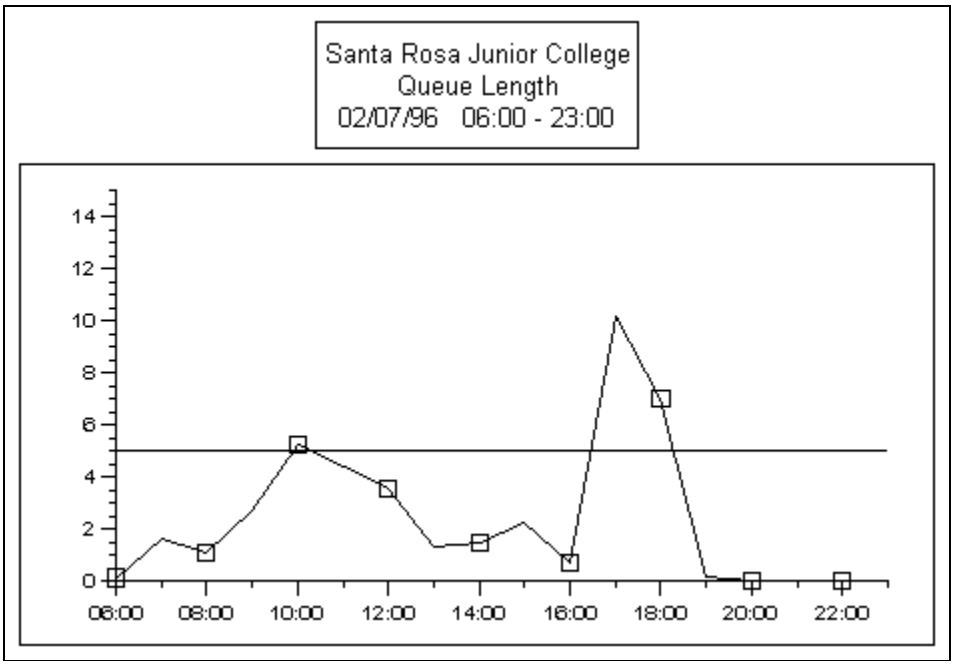

Figure 10

CPU saturation has two remedies:
1. Reduce demand for the CPU
2. Increase the supply by buying bigger and better hardware.

Here's some final performance tips:

- Be ***Proactive***! address indicators that are in the "yellow zone" performance
- Get a Performance Tool
- When addressing multiple bottlenecks start with disk bottlenecks, then memory bottlenecks and then CPU bottlenecks. Disk is the cheapest factor to address and then memory and then CPU.
- If you have found two bottlenecks, such as disk and memory, first address the disk bottleneck (the cheapest) and then re-evaluate the memory bottleneck. When an indicator is bad enough it will put pressure on other indicators to make it seem bad also. For example: figure 7 shows a slight memory pressure but once SRJE addresses the data locality problem most likely their memory pressure will go away.
- Always keep in mind too much memory is better than not enough. You can never have to much memory.

Good luck on your performance tuning!