# Performance
# Resource
# Utilization
# Health Check

**AIS**

System ID: mdel6
From: Friday, 31 July 2009  15:00
To: Saturday, 01 August 2009  01:00

## LEGAL NOTICE

The information contained in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

## SYSTEM MEASURED

|                |                        |
|---------------:|:-----------------------|
| System Name:   | mdel6                  |
| Machine Model: | ia64 hp superdome se   |
| Processors:    | 256 installed, 60 active |
| HP-UX:         | B.11.31                |
| Memory:        | 143 Gigabytes          |
| Disks:         | 190                    |
| LANs:          | 17                     |

## LOG FILE INFORMATION

|                          |                                   |
|-------------------------:|:----------------------------------|
| Data Collection Start:   | Fri, 31 Jul 2009  3:00 PM         |
| Data Collection End:     | Sat, 1 Aug 2009  1:00 AM          |
| Collector:               | SCOPE/UX C.04.60.000(0)           |
| Global 1-hr records:     | 11                                |
| Global 5-min records:    | 116                               |
| Disk 1-hr records:       | 2,090                             |
| LAN 1-hr records:        | 187                               |
| Application 1-hr records:| 65                                |
| Number of Applications:  | 5                                 |
| Number of GSUM Data Gaps:| 2                                 |
| 15 hours                 | starting at 00:00 on Fri, 31 Jul 2009 |
| 21 hours                 | starting at 02:00 on Sat, 1 Aug 2009 |
| Number of NSUM Data Gaps:| 2                                 |
| 15 hours                 | starting at 00:00 on Fri, 31 Jul 2009 |
| 21 hours                 | starting at 02:00 on Sat, 1 Aug 2009 |
| Number of DSUM Data Gaps:| 2                                 |
| 15 hours                 | starting at 00:00 on Fri, 31 Jul 2009 |
| 21 hours                 | starting at 02:00 on Sat, 1 Aug 2009 |

## PURPOSE

The purpose of this system utilization report is to provide you with an understanding of how your system is being utilized and therefore what processing capacity remains available. The information in this report can help to determine if the CPU is powerful enough for the current workload and whether or not the system contains sufficient main memory. It will also provide an understanding of system usage trends.

## METHODOLOGY

System utilization data for this report was obtained with HP OpenView Performance Agent Software. This software collects and summarizes data on a continuous basis to provide a complete record of system utilization. MWA has been carefully crafted to be very efficient; however it does use system resources while collecting data. The CPU used varies from 3-5% depending on the CPU model and the level of detail being collected. About one disk IO is required during each five-minute period.

## QUICK OVERVIEW

The information in this report is based on data collected from system **mdel6** during the two day period from Friday, July 31, 2009 through Saturday, August 1, 2009. There were two gaps in the data collection totaling 36 hours during the 0 hour collection period. data on this page refers to the prime shift hours of 08:00-17:00, Monday through Friday.

## CPU

The CPU was busy 44.3% of the time during the prime shift hours with a peak hourly average of 46.3%. **User mode** processing used 25.4% and **System mode** used 18.8%. The User to System ratio of 1.3:1 is within the guideline that suggests, for best performance, User mode CPU should be at least three times System mode CPU.

User mode processes spent most of their time executing at **Normal** priority (1.3%). Processes running at a **Nice** priority used 23.8% of the CPU.

CONCLUSION: **No obvious CPU bottlenecks** were observed.

## MEMORY

This system has **143 gigabytes** of memory installed. Memory utilization averaged 82.9% during the prime shift. Memory pageout rate was 718.6 pages per second. Memory Swapout rate was 0.0. Average memory queue length was 0.2.

CONCLUSION: **A pageout rate greater than 3.0 indicates a serious shortage of memory.**

## DISK

The average hourly peak disk utilization was 20.7%. The highest average utilization seen on any disk in any hour was 21.2. The average disk queue length was 0.3.

CONCLUSION: **No obvious disk bottlenecks** were observed.

## NETWORK

The network packet rate during the prime shift was 5339 packets per second and the average network queue depth was 0.0.

CONCLUSION: **No obvious network bottlenecks** were observed.

## COMMENTS

While no obvious problems are indicated above keep in mind that hourly averages can smooth over short-term problems. In the Technical Analysis section at the end of this report we will dig deeper into the data collected, including five-minute data, to see if there are problems that haven't appeared above.

# System Utilization Overview
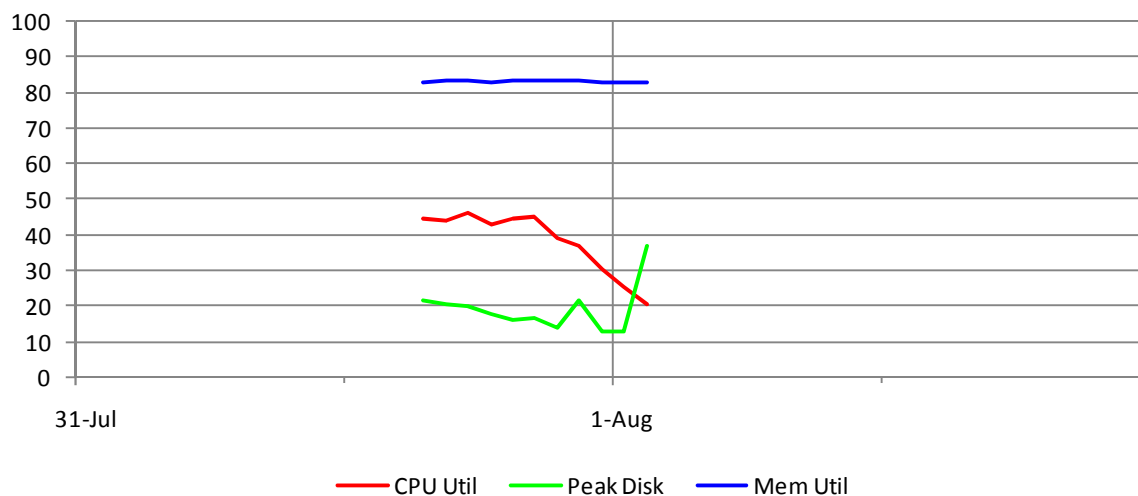## Figure 1 - Percent utilization (hourly averages)



Figure 1 – This graph presents an hour by hour view of the system-wide utilization of these three primary resources: CPU, Disk and Memory. The peak hourly CPU utilization was 46.3% which occurred during the hour beginning at 17:00 on Friday, 31 Jul. CPU utilization of all processors is reported as single number. This means that, at peak utilization, 27.8 of the 60 available processors were effectively 100% busy.

| Hourly Averages | CPU Util | Peak Disk | Mem Util |
|---|---|---|---|
| *Overall* | 38.2% | 19.0% | 82.9% |
| *08:00-17:00* | 44.3% | 20.7% | 82.9% |

**CPU Utilization**          is shown as percent busy executing instructions.

**Peak Disk Utilization**    is the average utilization (percent of the time busy) of the busiest disks during the hour. At the end of each five-minute interval the utilization of the busiest disk is noted. The value plotted here is the hourly average of each of those five-minute values. Note that several disks can be represented in the graphed value. Memory manager IO activity is not included in the Peak Disk Utilization activity.

**Memory Utilization**       is the percentage of physical memory in use during the interval. This includes system memory (occupied by the kernel), buffer cache and user memory. This calculation is done using the byte values for physical memory and used memory, and is therefore more accurate than comparing the reported kilobyte values for physical memory and used memory.

The remaining graphs in this report provide more detailed views of these three primary resources.

**NOTE**: **Prime Shift** in this report refers to the period 08:00-17:00

.

# System Activity

## Figure 2 - Number of Alive and Active Processes (*left scale*), Process Starts per hour and System Calls per second (*right scale*)
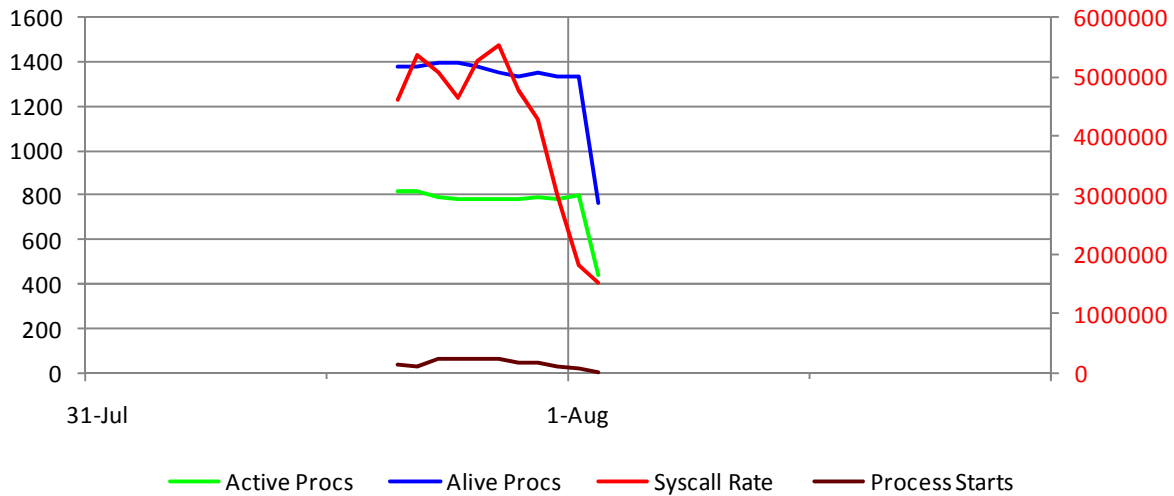


Figure 2 - This is a good graph for establishing a baseline of system usage. Over time, you should be aware of the number of active processes on your system when resources are performing well as well as the number of alive processes.

| Hourly Averages | Alive Processes | Active Processes | Syscall Rate | Process Starts |
|---|---|---|---|---|
| **Overall** | 1306.7 | 762.9 | 4,154,828 | 162,526 |
| **08:00-17:00** | 1379.5 | 819.5 | 4,964,613 | 137,418 |

The system call rate is also good for monitoring system changes. As your system develops bottlenecks, you may see this figure increase. The changes may also be due to application changes or additions. As a guideline, a system call rate of 3,000 per processor is a common starting point, although this is an older heuristic value and may not be good for today's larger and faster systems. Your site may differ and that is the purpose of establishing a baseline. Active processes may typically appear as 28%-38% of alive processes and again, your site may differ due to the type of workload experienced.

| | |
|---|---|
| **Alive Processes** | (left scale) The number of processes alive on the system during the interval. Processes that were alive for only a portion of the interval are counted according to the percent of the interval that they were alive. |
| **Active Processes** | (left scale) The number of processes that consumed CPU during the interval. |
| **System Call Rate** | (right scale) The average number of system calls per second during the interval. Systems with IO intensive applications will show higher rates. Compare this to the logical IO rate (Figure 18) to see if the high rates correspond to higher disk IO requests. As this increases, you should see higher System CPU in Figure 4. Rapid spawning of processes will also increase this and will show up as increased Context Switch CPU in Figure 4. |
| **Process Starts** | (right scale) The number of processes that started during each hour of the interval. |

# Average Number of Processes Queued
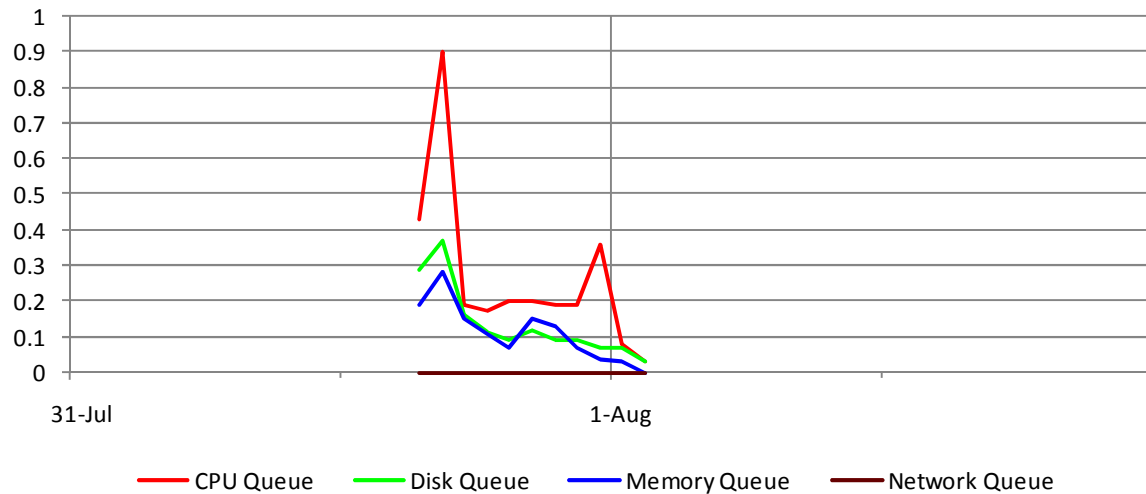## Figure 3 - Number of processes queued (hourly averages)



Figure 3 – The data in this graph shows how smoothly processes are moving through the system. The **CPU Queue** line shows the average number of processes waiting on the CPU but not in a sleep state. This is an indication of the CPU load. The disk load can be judged by studying the **Disk Queue** line. The line shows the average number of processes waiting on a disk resource. The **Mem Queue** line shows the average number of processes waiting on memory. The **Network Queue** line shows the average number of processes blocked on the network subsystem during the interval. These will be processes blocked on wait states such as lan, nfs, rpc or mbuf.

Units on the vertical scale (Y-axis) represent the average number of processes waiting on a resource.

| Hourly Averages | CPU Queue | Disk Queue | Mem Queue | Network Queue |
|---|---|---|---|---|
| *Overall* | 0.27 | 0.14 | 0.11 | 0.00 |
| *08:00-17:00* | 0.67 | 0.33 | 0.24 | 0.00 |

# System CPU Utilization
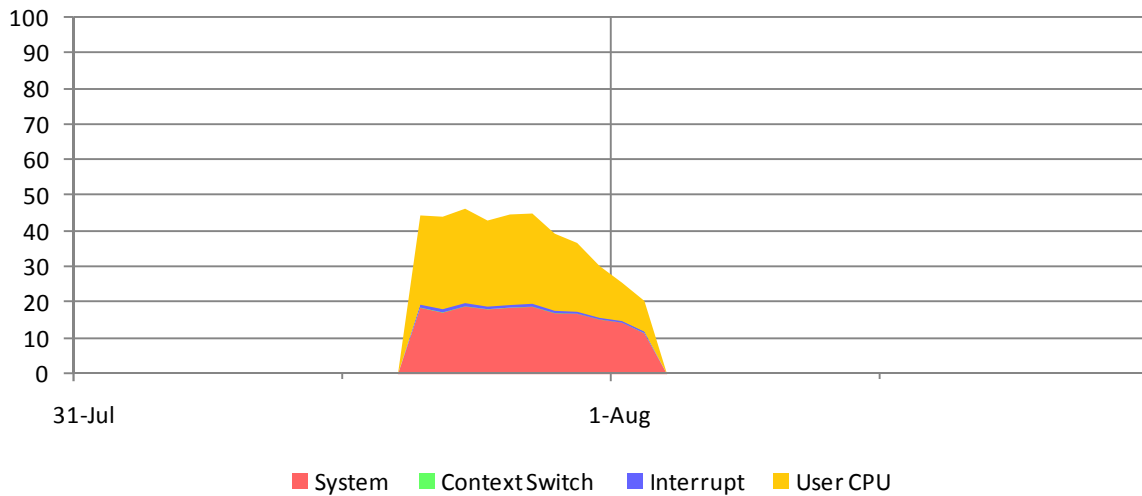## Figure 4 - Percent utilization (hourly averages)



Figure 4 - Once total CPU utilization is known the next question to ask is how much of the CPU is being consumed by users and how much is being used to manage the system. The amount of CPU used for system calls is characteristic of the workload. Typically, the ratio for User CPU vs System CPU is 3:1. A lower ratio may indicate misuse of system calls within an application.

| Hourly Averages | System | Context Switch | Interrupt | Other |
|---|---|---|---|---|
| *Overall* | 16.5% | 0.1% | 0.9% | 20.6% |
| *08:00-17:00* | 17.6% | 0.2% | 1.1% | 25.4% |

| | |
|---|---|
| **System** | CPU time used by system processes and system calls made by "normal" processes. |
| **Context Switch** | CPU time used to switch from one executing process to one of a higher priority process that became ready to run. Typically 1-3%. |
| **Interrupt** | CPU Used by the kernel's interrupt handler to process anomalies found during instruction processing. Interrupts occur as a result of faults, traps, check and device interrupts. Typically 1-5%. |
| **Other CPU** | CPU used by the User CPU processes. This includes the real-time, time share and Nice processes. |
| **Idle** | is the absence of any CPU activity. This includes those periods when a process is waiting for a disk IO to complete and no other process is ready to use the CPU. Idle time is not explicitly shown on the graph. It is the area between "Other" and the 100% line. |

# User CPU Utilization
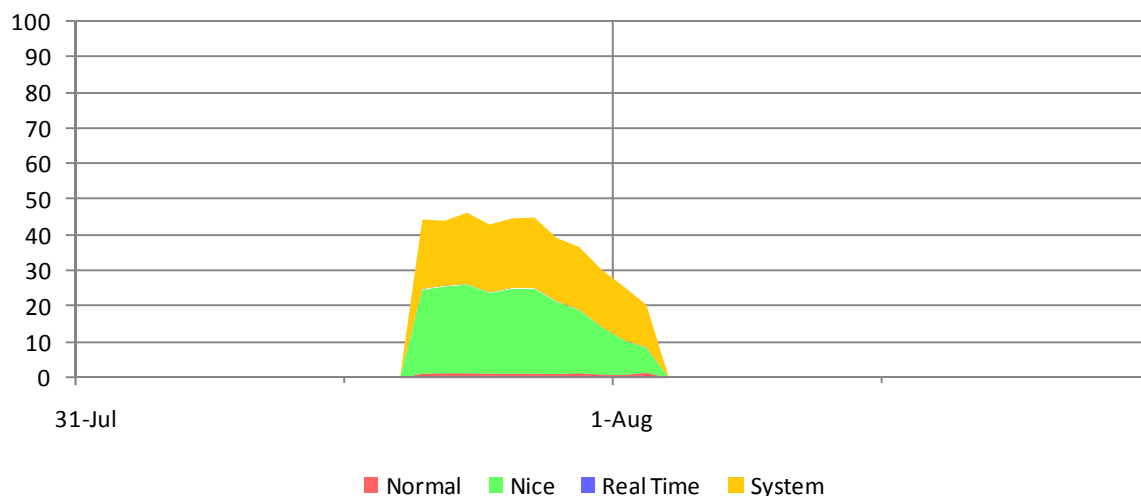## Figure 5 – Hourly averages



Figure 5 - The amount of CPU time available to process user tasks is the total amount of CPU time available minus the amount of CPU time needed to manage the system. The available user processing time can then be analyzed further to see what type of user activity is being performed on the system.

| Hourly Averages | Normal | Nice | Real-time | Other |
|---|---|---|---|---|
| *Overall* | 1.3% | 19.1% | 0.2% | 17.6% |
| *08:00-17:00* | 1.3% | 23.8% | 0.3% | 18.8% |

| | |
|---|---|
| **Normal** | CPU time spent while executing user code with timeshare priority and the default "nice" value of 20. This typically includes all interactive and foreground processes. |
| **Nice** | CPU time spent while executing user code with timeshare priority and non-default "niceness." Processes that fall in this category are typically background processes with a slightly better priority than other timeshare processes, for example HPGlance. |
| **Real Time** | CPU time spent while executing user code with real time priority. Real time priority is usually given to very time-critical processes that should have priority over all timesharing processes. |
| **Other** | CPU time spent while executing system calls. This includes system calls (such as read, write) made by user processes and pure system processes such as vhand and swapper. |
| **Idle** | is the absence of any CPU activity. Idle time is not explicitly shown on the graph. It is the area between "Other" and the 100% line. |

# CPU Utilization by Application
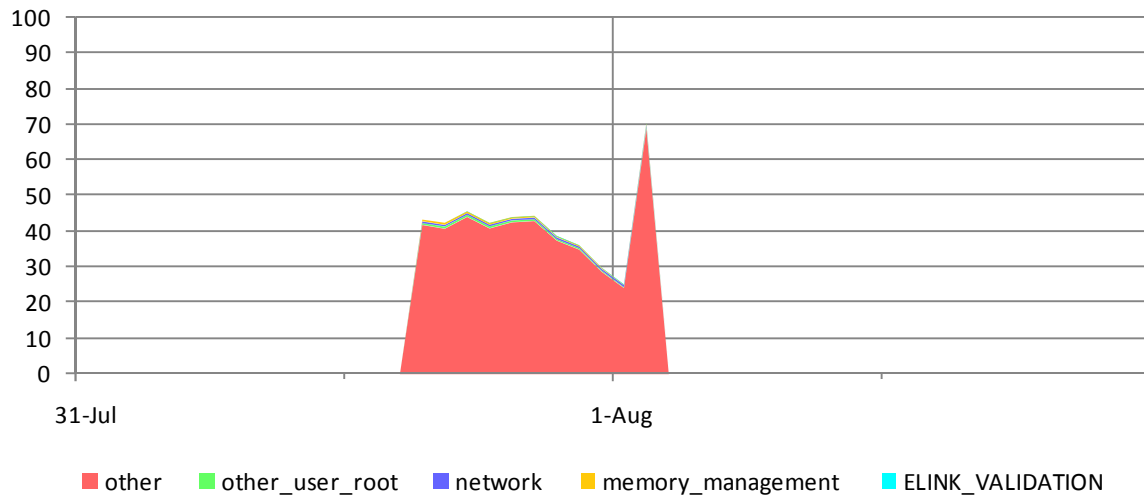## Figure 6 – Hourly averages



**Figure 6** - A total of 5 applications were seen during the period monitored. To keep the graph readable only the top five (by CPU utilization) are shown here. On many systems a single application accounts for most of the CPU used. This graph makes it easy to identify the top application. You can also determine when it is active.

Applications are defined in a "parm" file before data collection begins. Each application must be properly defined to insure that the application information in this report is accurate. The table below lists all applications that used the CPU during the measurement period. The number of hours during which the application showed some activity is shown in the **Records** column.

| Application | Records | %CPU | CPU Secs | IOs/Sec | PhysIOs |
|-------------|--------:|-----:|---------:|--------:|--------:|
| other | 11 | 40.66 | 12,905 | 3864.38 | 126,527,085 |
| other_user_root | 11 | 0.42 | 142 | 34.45 | 1,114,477 |
| network | 11 | 0.43 | 141 | 828.95 | 28,149,672 |
| memory_management | 11 | 0.41 | 138 | 568.31 | 18,563,253 |
| ELINK_VALIDATION | 10 | 0.10 | 14 | 4.51 | 61,316 |

# CPU Utilization Pattern By Hour of Day
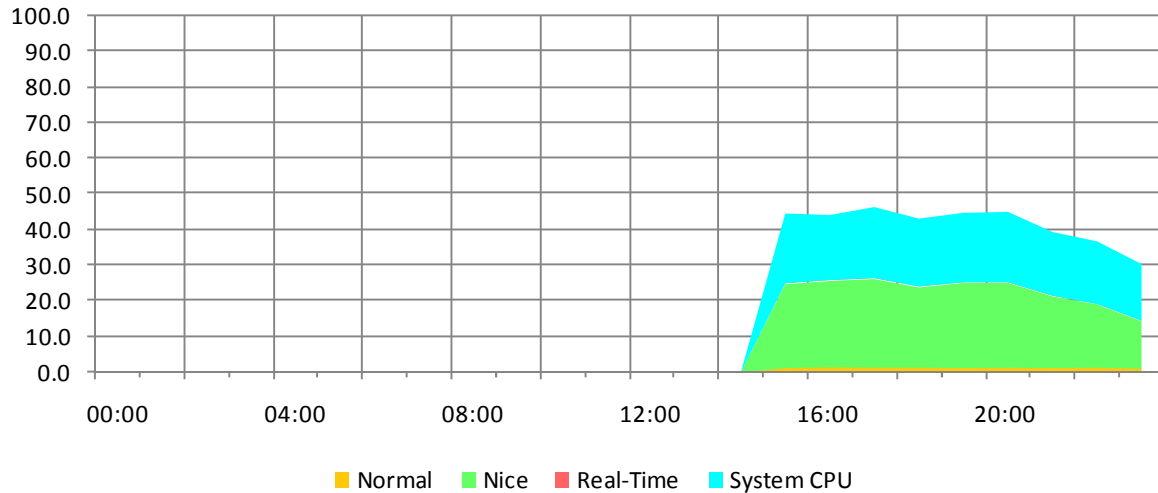## Figure 7 Average by hour of day



Figure 7 - This graph shows the average utilization of the CPU by time of day with weekends omitted. For example, the amount of CPU used by Normal processes for the 10am period is the average of Normal CPU used between 10-11 a.m. during the collection period except for weekend days (Saturday and Sunday).

Some questions that can be answered by looking at the graph are:

At what hour of the day is the system most heavily used?

How much of the CPU is used by Nice priority applications during a given hour?

Are any Normal User processes active between midnight and 8 a.m.?

When can additional User processes be scheduled without impacting current online Normal users?

What is the best time to add a new Real Time process to insure Minimum conflict with other Real Time processes?

# Application CPU Utilization by Hour of Day
## Figure 8 - Averaged over weekdays (Mon-Fri)



- ■ other
- ■ other_user_root
- ■ network
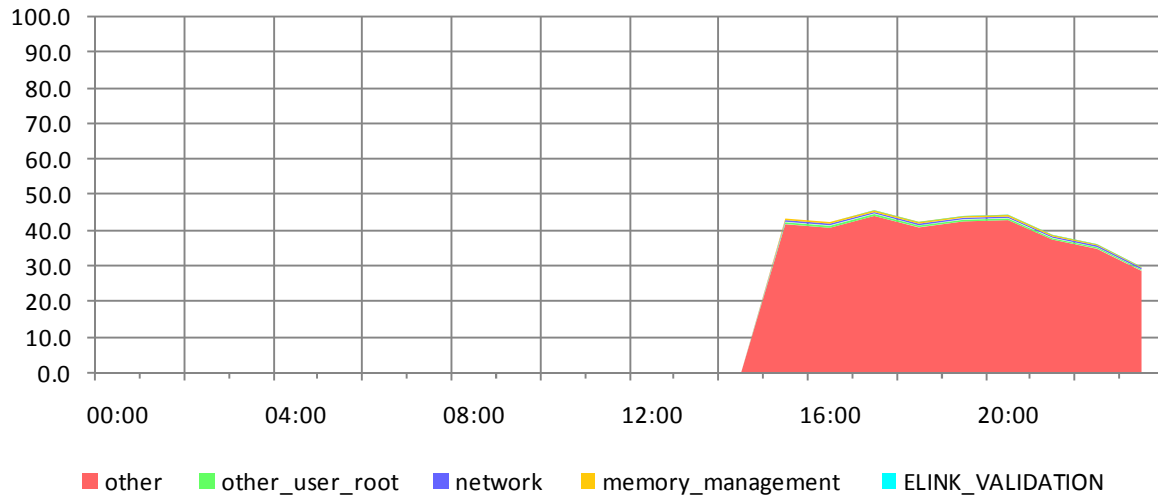- ■ memory_management
- ■ ELINK_VALIDATION

Figure 8 - This graph provides application detail of the average CPU usage for each hour, by the user processes during week days. Saturdays and Sundays have been omitted.

## Application Activity by Hour of Day

Which applications are active on weekdays from 8-10 a.m.?

Which applications are active during the "peak" processing times?

During what hours are the different applications busy?

# Daytime CPU Distribution by Activity
## Figure 9 - Data is for the Prime Shift period (08:00-17:00)



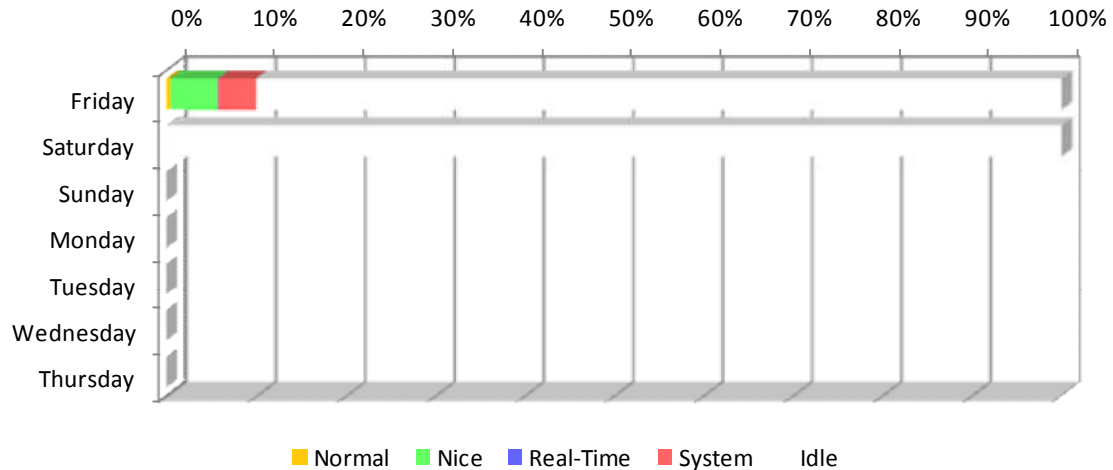**Normal**   **Nice**   **Real-Time**   **System**   Idle

Figure 9 – This chart shows CPU activity broken down by activity, averaged over the prime shift for each day of the week. This can be used to see how the system workload varies each day during the week.

You should keep in mind that idle time may include some time when the CPU was waiting for disk IO completion. A non-interactive User process can be responsible for a considerable number of disk IO requests that could translate into CPU Idle time. On the other hand a CPU intensive program can hide the fact that other programs are hampered by disk IO bottlenecks.

## Activity Pattern By Day of Week

How busy was the CPU during the prime shift on Wednesday?

How much User processing was done during the day on Friday?

Of the User processing done during the day on Friday, how much was at a Normal priority? How much was Niced or at a lower priority?

# Daytime CPU Distribution by Application
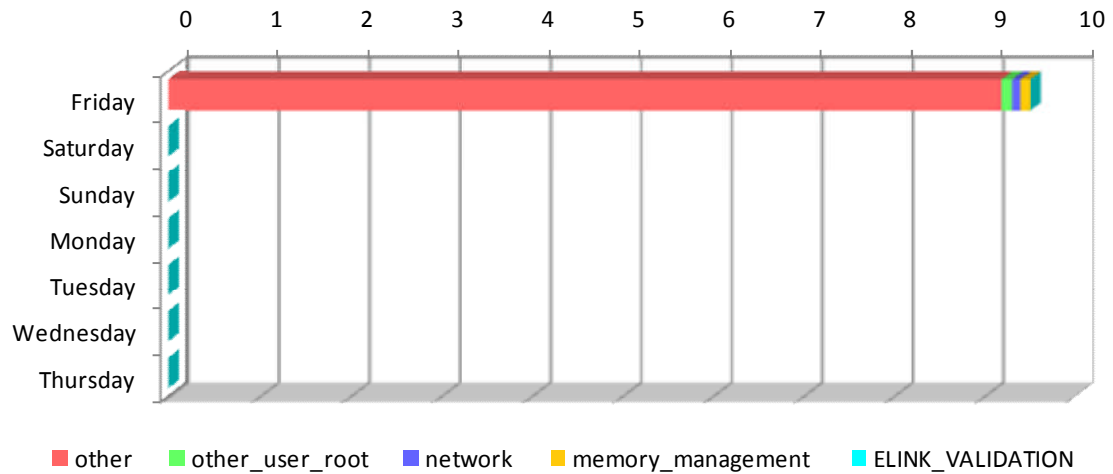### Figure 10 - Data is for Prime Shift period (08:00-17:00)



Figure 10 - Shows CPU activity by application, averaged over the prime shift for each day of the week. This can be used to see how the application workloads vary by day during the week.

## Application CPU Pattern by Day of Week

Which application used the most CPU on Thursday during the prime shift?

Is the CPU utilization during the day equally distributed between the applications?

Which days have the most batch processing during the prime shift?

# Cumulative Distribution of CPU Utilization
## Figure 11 - Percent of Time at or Above Noted Utilization
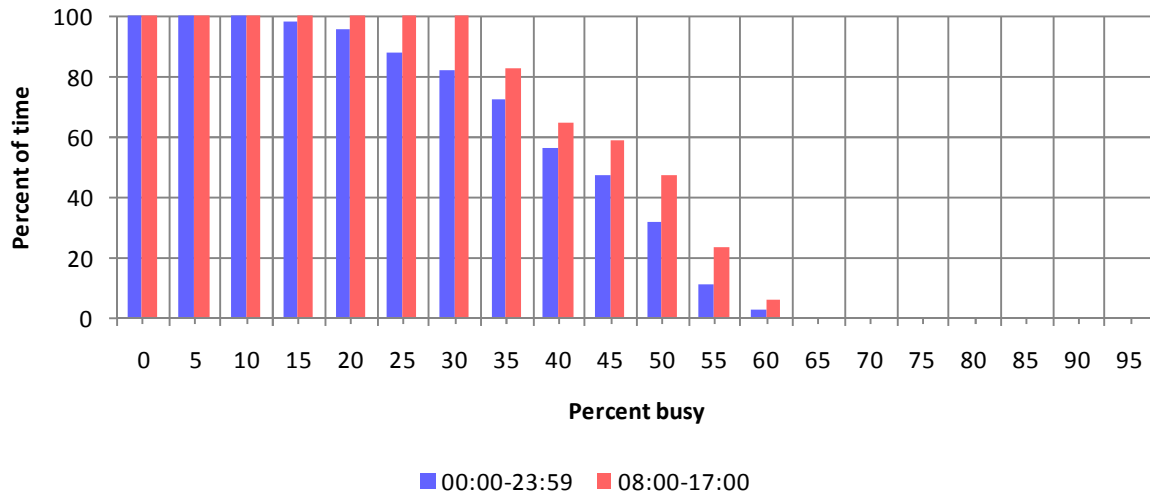


■ 00:00-23:59   ■ 08:00-17:00

Figure 11 – This chart shows the cumulative percentage of time at each level of CPU utilization for the entire collection period. The data is based on **five-minute averages** excluding Saturday and Sunday. One line is plotted based on 24 hours a day and the other is for the prime shift period only (08:00-17:00).

A quick glance at the relative position of these two lines shows how the prime shift utilization compares with the 24-hour period. Normally one would expect the prime shift to plot higher than the entire 24-hour period.

This graph provides data that allows one to conclude "the CPU was at least X% busy Y% of the time." The horizontal axis represents the X value and the vertical axis represents the Y value.

# Peak Hourly CPU Utilization
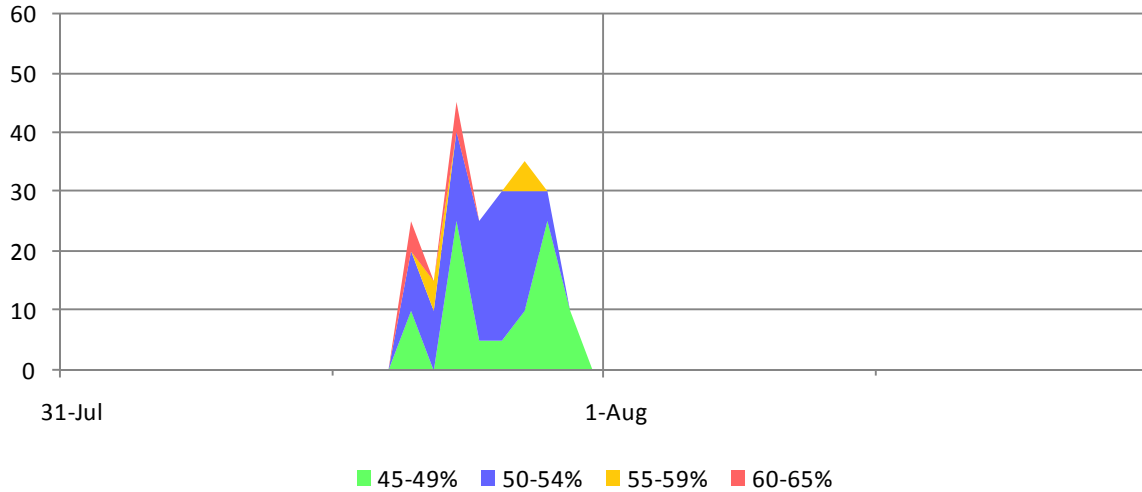## Figure 12 - Minutes Busy Per Hour at Peak Utilization



Figure 12 – The data plotted here is based on **five-minute averages** to provide a better picture of the utilization pattern during each hour. Other CPU graphs in this report are based on hourly averages. When they show that the CPU utilization was 50% for a certain hour we don't know whether it was 100% busy for 30 minutes and idle the other 30 minutes, stayed close to 50% busy for the entire hour, or some other pattern that averaged 50% busy for the hour.

To develop the data for this graph we first search for the highest CPU utilization in any five-minute period. For this system the highest average utilization during any five-minute period was **62.0%** while the highest average utilization during any one hour was **46.3%.** Starting with the five-minute number we determine the four ranges in which to accumulate total minutes. Once the ranges have been determined we examine the CPU utilization for each five-minute period and add 5 to the range it fits in, if any.

The vertical axis represents minutes busy per hour while the horizontal axis represents the time of day. The minor tick marks on the horizontal axis are placed at 12 noon.

# CPU Utilization during Prime Shift
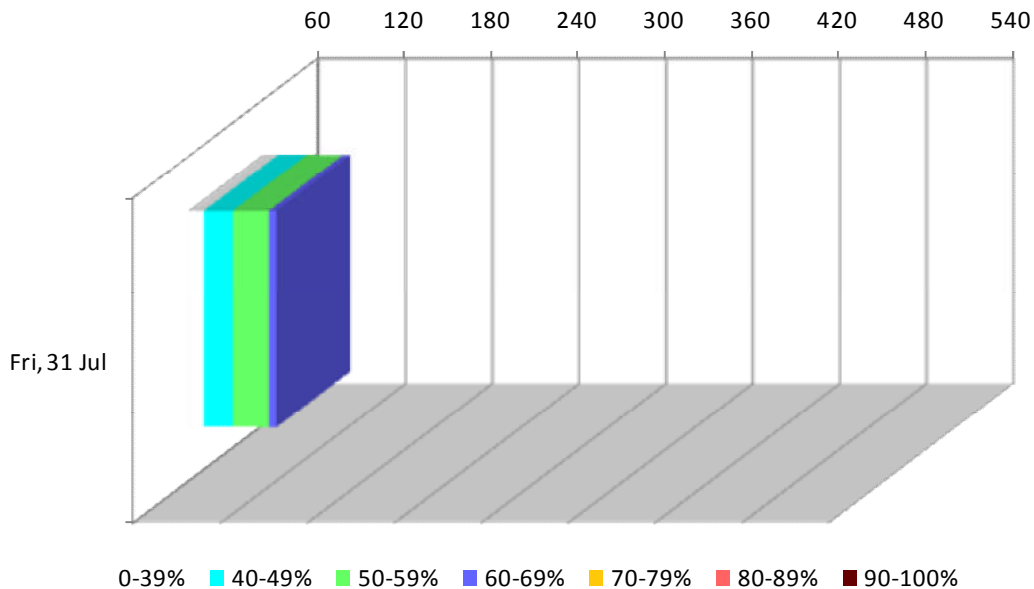## Figure 13 - Minutes Busy at Noted Bands



Figure 13 - This graph, based on **five-minute data**, gives a picture of CPU utilization each day between the hours of **08:00-17:00**. The utilization level is divided into seven bands.

To develop the data for this graph we examine the CPU utilization for each five-minute period and add 5 to the range it fits in. Any value less than 40% falls into the first band, any value 40% or greater and less than 50% falls into the second band, etc.

Weekdays are shown on the vertical axis. Minutes of CPU busy is shown on the horizontal axis. There are 540 minutes in each prime shift so that is the maximum value on the x-axis. The x-axis minimum value will be zero only if the CPU was 0-39% busy for less than one-hour during any day. The minimum value on the x-axis is moved up 60 minutes at a time to shorten the lowest band. This leaves more room for the more interesting upper bands.

If a bar does not extend all the way to the right side then data was not available for the entire period of time. Normally this will happen only on the first and/or last day.

# Virtual Memory Disk I/Os per Second
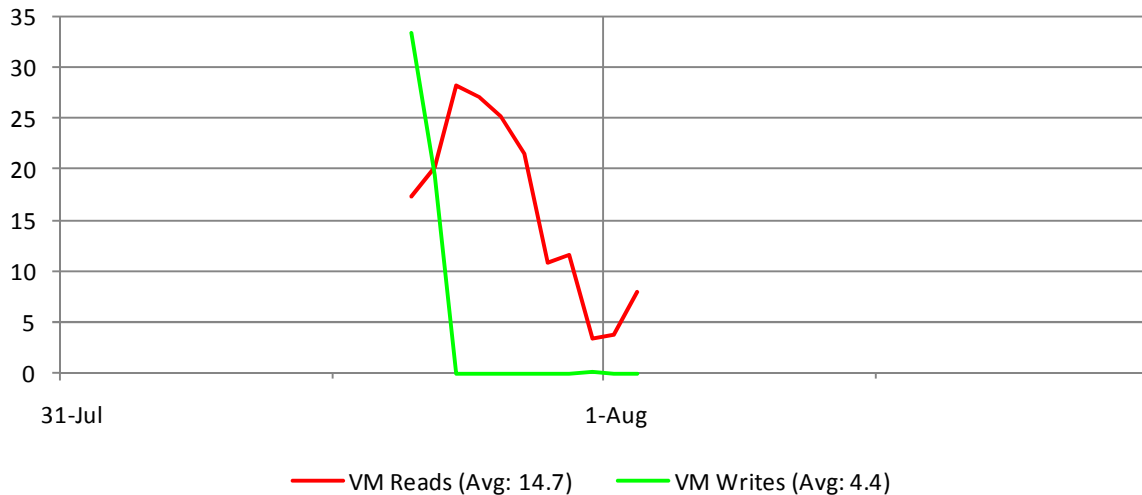## Figure 14 - Hourly averages



Figure 14 - The average hourly rate of virtual memory disk reads and writes are graphed here. These metrics help determine whether or not the system has sufficient main memory installed. When memory utilization is close to 100% and the VM write rate is higher than 10-15 IOs per second additional memory may be called for.

Virtual Memory disk I/Os occur when the memory manager must transfer the data area of a process to or from a swap area on disk, or must transfer text (code) from disk as a result of having to make room in memory for use by other processes. Keep in mind that IO for memory mapped files may be included here. Also a high rate of process creations may show up as higher values here without indicating a memory problem.

When memory utilization is close to 100% and the virtual memory write rate exceeds 10-15 per second, the system may have a memory bottleneck. If memory mapped IO is occurring on the system a high rate may be observed even when memory utilization is below 90%.

The virtual memory read rate may be high due to executable files being loaded into memory (i.e. a high process create rate) or reads from the swap area. Memory mapped reads also affect this metric. The rate can appear high when memory is not an issue.

Memory utilization is shown in Figure 1 and process creation rates may be inferred from Figure 2.

# Memory Utilization
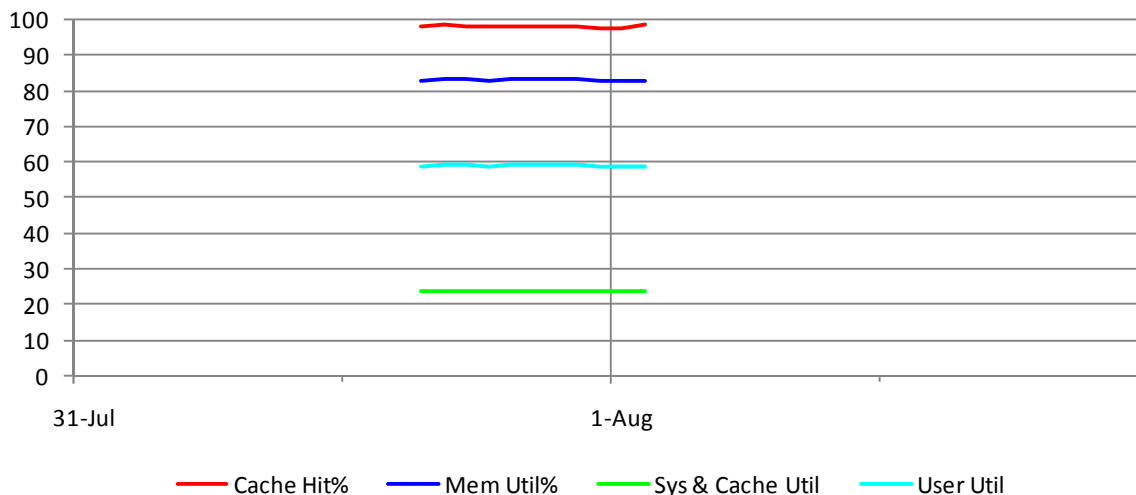## Figure 15 - Hourly averages



Figure 15 - In the process of establishing a baseline for your system, you need to determine the average amount of resident memory used by a typical user. This will help in analyzing current and future growth in the user community. Along with this, you need to establish a baseline as to how your memory is being utilized. Do you have enough buffer-cache? Do you have too much buffer-cache configured? Do you have a good mix of User and System usage? Your Cache Hit % should be at 90% or higher as a guideline on most systems. This is not an absolute, so lower cache hit rates may be normal for your environment.

| Hourly Average | Cache Hit% | Mem Util% | Sys & Cache Util | User Util |
|---|---|---|---|---|
| *Overall* | 98.3% | 82.9% | 23.8% | 59.1% |
| *08:00-17:00* | 98.6% | 82.9% | 23.9% | 59.1% |

| | |
|---|---|
| **Cache Hit Percent** | The percentage of physical reads satisfied in the file system buffer cache. A low hit rate indicates low efficiency of the buffer cache, either because applications have poor data locality, the buffer cache is too small, or the type of data access is very random. |
| **Mem Utilization Percent** | The percentage of physical memory in use during the interval. This includes system memory (occupied by the kernel), buffer cache and User memory. When memory utilization is 100% other metrics must be checked to verify that a memory bottleneck actually exists. |
| **Sys & Cache Utilization** | The percentage of physical memory used by the system and the buffer cache. |
| **User Utilization** | The percentage of physical memory allocated to user code and data at the end of the interval. A continual increase in this metric may indicate a program with a memory leak. |

# Memory Paging
## Figure 16 - Hourly averages
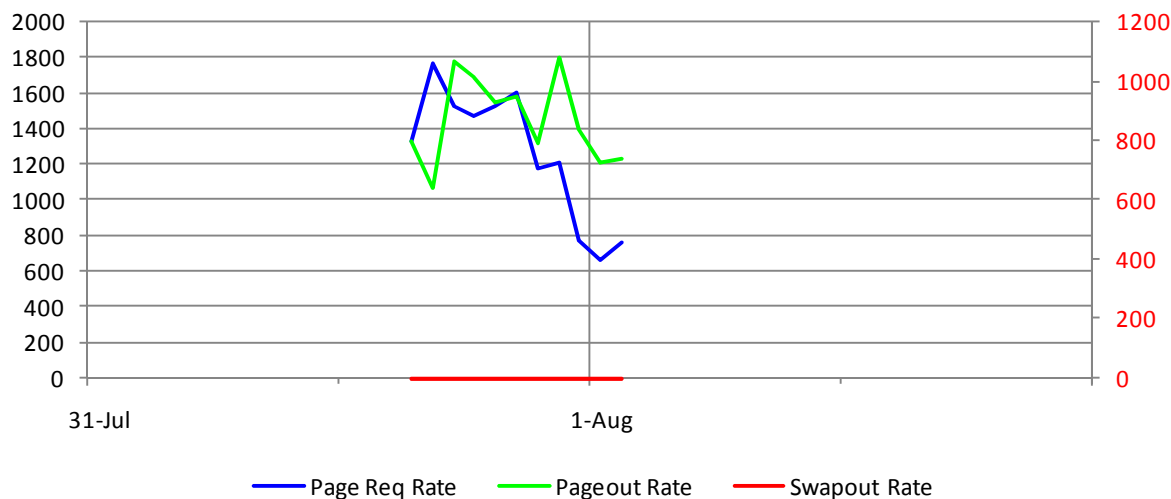


Figure 16 – Use left scale for **Page Req Rate** and right scale for **Pageout Rate** and **Swapout Rate**. Paging is a process by which memory pages are brought into memory and removed from memory. Page-ins occur when a process starts up, when a process requests dynamic memory, and during page faults after a page-out, a swap-in or a reactivation. Page-outs and deactivation occur when memory is scarce. With deactivation the process stops executing for a while so it does not soon cause its pages to be brought in again. A system under severe memory pressure will see numbers other then zero for deactivation and also an increase in page-out rate. Before adding more memory Glance should be used to monitor deactivation rates during a busy interval.

| Hourly Averages | Page Req Rate | Pageout Rate | Swapout Rate |
|---|---|---|---|
| *Overall* | 1258.6 | 866.98 | 0.00 |
| *08:00-17:00* | 1549.0 | 718.65 | 0.00 |

| | |
|---|---|
| **Page Request Rate** | **(Left Scale)** The number of page requests to and from the disk during the last interval. |
| **Pageout Rate** | **(Right Scale)** The total number of page outs to the disk per second during the interval. As a general rule of thumb a number greater than five may be an indicator of memory pressure. However, this is only one indicator which should be validated with other metrics. Other than deactivation, no single metric by itself is indicative of memory pressure. |
| **Swapout Rate** | **(Right Scale)** The number of **deactivations** per second during the interval. Deactivated processes are removed from the run queue. |

# Virtual Memory
Figure 17 - Hourly averages



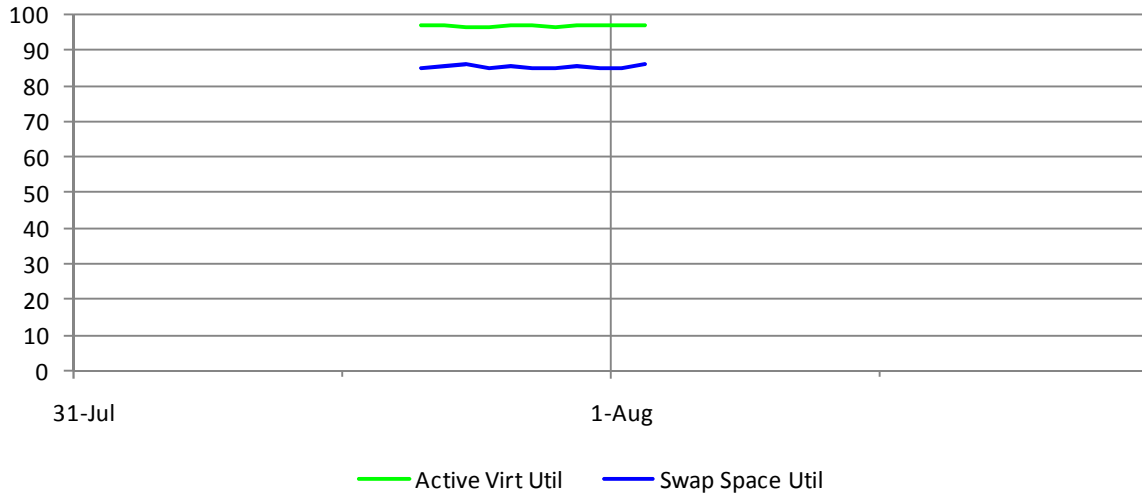Active Virt Util        Swap Space Util

Figure 17 - Virtual memory may be located in physical memory, the swap area (raw or file system swap), executable files or memory mapped file. If a system becomes low on available swap space, it will reach a point at which no new processes or virtual memory can be created.

| Hourly Averages | Active Virt Util | Swap Space Util |
|---|---|---|
| Overall | 96.8% | 85.4% |
| 08:00-17:00 | 96.8% | 85.2% |

**Active Virtual Utilization** The percentage of total virtual memory active at the end of the interval. Active virtual memory is virtual memory associated with the processes that are currently on the run queue or processes that have executed recently. This is the sum of the virtual memory sizes of the Data and Stack regions for these processes.

**Swap Space Utilization** The percent of swap space available that was reserved by running processes in the interval.

# Average Disk IOs per Second
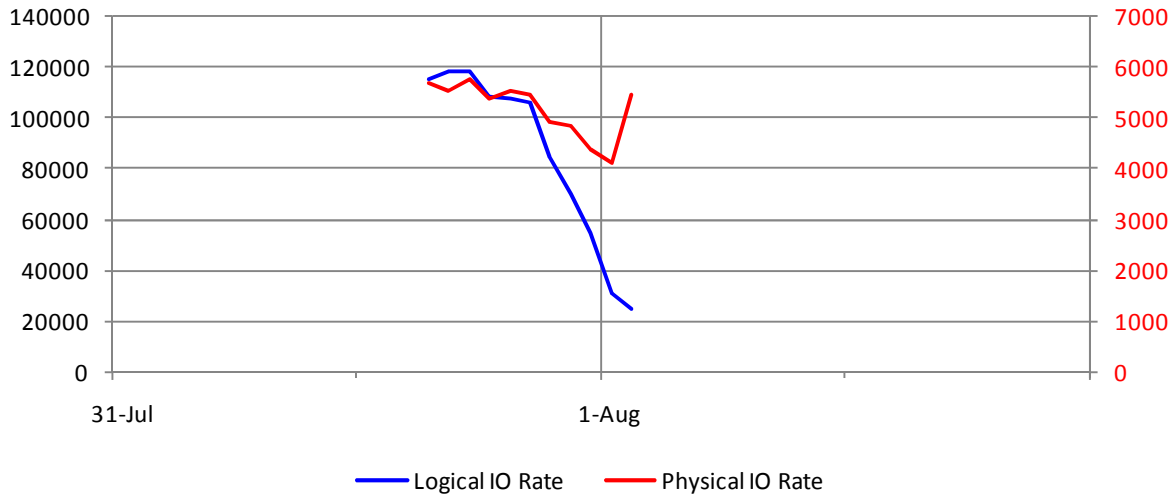## Figure 18 – Logical IO (*left scale*), Physical IO (*right scale*)



Figure 18 – This graph shows the hourly average physical and logical disk IOs that occurred during the monitored period. These values represent the sum of all disk IOs on the system and is inclusive of both system and user requests.

The average service times for disk arrays is typically less than 6 ms, whereas an independent disk might show service times of 10-14 ms. Service times greater than these can be indicative of an IO bottleneck if they persist over long intervals.

| Hourly Averages | Logical IO Rate | Physical IO Rate | Log/Phys Ratio | Percent Physical |
|:---:|:---:|:---:|:---:|:---:|
| *Overall* | 84,933 | 5,186 | 16.4:1 | 6.1% |
| *08:00-17:00* | 116,346 | 5,616 | 20.7:1 | 4.8% |

**Logical IO Transfers**    The number of transfers per second to or from all disks. These transfers may or may not have been satisfied by the file system buffer cache. Memory management transfers are not included here. High logical IO rates can be indicative of insufficient data buffering at the database level.

**Physical IO Transfers**    (Right scale) Actual requests by all the file system or a raw IO activity that could not be satisfied by the buffering system. Memory management transfers are not included here. A large amount of physical IO in relation to logical IO can be due to additional read-ahead activity when on logical request has been made. This can be typical in a "pseudo random" access environment or one that is typically random but with interspersed sequential access causing read-ahead activity without a subsequent logical request.

## Disk IO Rate by Application
### Figure 19 – Disk IOs per second (hourly averages)
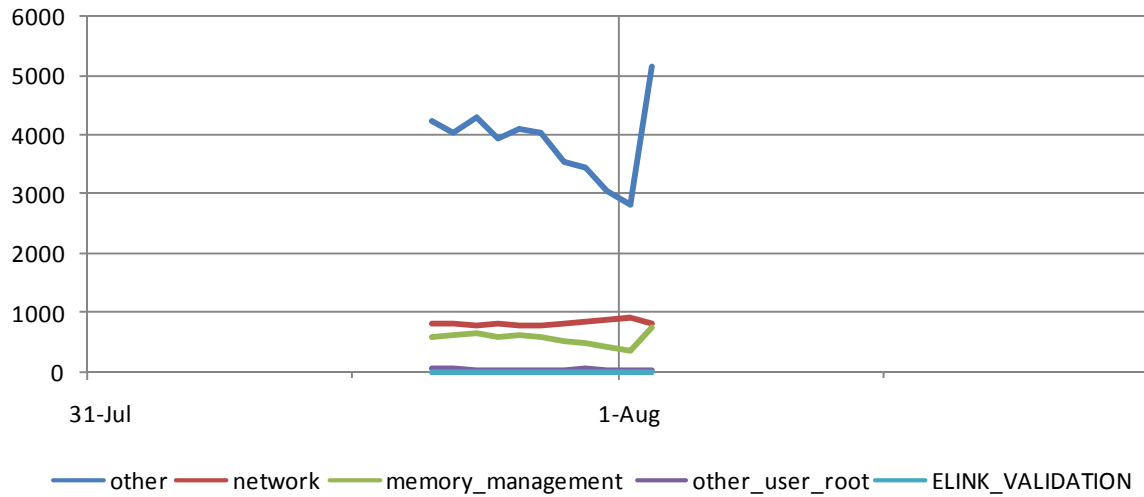


Figure 19 – This graph shows the <u>hourly average disk request</u> rate for each of the top five applications. The top five applications are chosen by total physical IO count. This is a detailed breakout by application of the disk IO activity shown in Figure 18.

| Application | Records | %CPU | CPU Secs | IOs/Sec | PhysIOs |
|---|---|---|---|---|---|
| other | 11 | 40.66 | 12,905 | 3864.38 | 126,527,085 |
| network | 11 | 0.43 | 141 | 828.95 | 28,149,672 |
| memory_management | 11 | 0.41 | 138 | 568.31 | 18,563,253 |
| other_user_root | 11 | 0.42 | 142 | 34.45 | 1,114,477 |
| ELINK_VALIDATION | 10 | 0.10 | 14 | 4.51 | 61,316 |

# Disk Activity on the Five Busiest Drives
## Figure 20 - Hourly Average Disk Activity
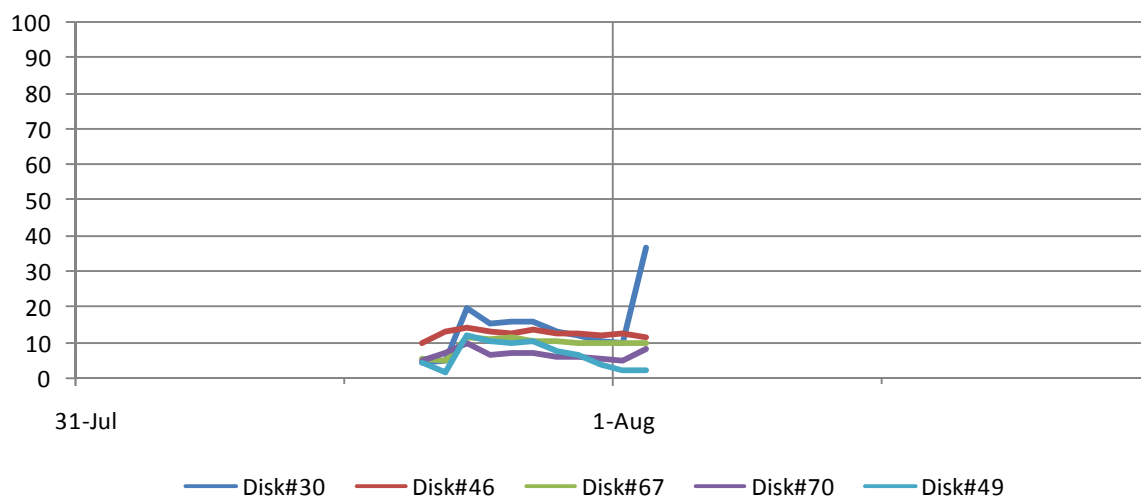


Figure 20 - This graph shows the hour-by-hour percent utilization of the five busiest disks on the system. The disks plotted were chosen <u>based on total utilization</u> (sum of hourly utilization values) for the entire measurement period. Ideally one disk will not be substantially busier than the others.

| Disk Address | Disk Num | Num Recs | Avg QLen | Max Util | Avg SrvcMs | Peak SrvcMs | PkSrvc DateTime | Physical IO Count |
|---|---|---|---|---|---|---|---|---|
| 64000/0xfa00/0x1f | 30 | 11 | 0.16 | 36.66 | 1.36 | 3.51 | 07/31 16:00 | 6,931,357 |
| 64000/0xfa00/0x38 | 46 | 11 | 0.20 | 14.03 | 1.73 | 2.09 | 07/31 15:00 | 4,875,710 |
| 64000/0xfa00/0x37 | 67 | 11 | 0.28 | 11.23 | 2.05 | 3.10 | 07/31 16:00 | 3,261,991 |
| 64000/0xfa00/0x1e | 70 | 11 | 0.57 | 10.11 | 2.36 | 2.97 | 07/31 18:00 | 2,005,897 |
| 64000/0xfa00/0x19 | 49 | 11 | 0.01 | 11.93 | 1.59 | 2.59 | 07/31 15:00 | 4,093,019 |
| 64000/0xfa00/0x2b | 129 | 11 | 3.41 | 9.32 | 3.57 | 3.92 | 07/31 17:00 | 4,219,223 |
| 64000/0xfa00/0x28 | 141 | 11 | 3.30 | 9.18 | 3.63 | 3.94 | 07/31 20:00 | 4,208,614 |
| 64000/0xfa00/0x27 | 127 | 11 | 3.10 | 8.88 | 3.46 | 3.91 | 07/31 17:00 | 4,091,120 |
| 64000/0xfa00/0x17 | 60 | 11 | 0.61 | 9.19 | 1.94 | 3.64 | 07/31 16:00 | 4,088,502 |
| 64000/0xfa00/0x29 | 111 | 11 | 3.32 | 9.01 | 3.52 | 3.92 | 07/31 17:00 | 3,882,313 |
| 64000/0xfa00/0x2c | 122 | 11 | 3.20 | 9.34 | 3.46 | 3.86 | 07/31 17:00 | 3,855,883 |
| 64000/0xfa00/0x2a | 79 | 11 | 3.26 | 9.24 | 3.61 | 3.91 | 07/31 17:00 | 3,831,908 |
| 64000/0xfa00/0x18 | 54 | 11 | 0.03 | 9.18 | 2.00 | 2.26 | 07/31 16:00 | 3,723,399 |
| 64000/0xfa00/0x1d | 80 | 11 | 0.04 | 8.66 | 2.07 | 2.53 | 07/31 15:00 | 3,648,459 |
| 64000/0xfa00/0xb3 | 35 | 11 | 0.03 | 9.31 | 1.93 | 2.72 | 07/31 16:00 | 3,151,748 |

The table above lists 15 of the 190 disks on the system, accounting for 34% of all system-wide physical IOs.The first five entries in the table correspond to the five disks in the graph. The remaining entries are sorted by Physical IO Count. The **Num Recs** column indicates in how many hours the drive showed some activity. The **Max Util** column shows the percent utilization of the drive during its busiest hour. **Avg QLen** shows the average number of physical IO requests that were queued for the disk averaged over all hours. The **Peak SrvcMs** column shows the maximum service time, in milliseconds, for a disk IO request during the collection period. The service peak occurred during the hour noted in the **PkSrvc DateTime** column.

## LAN Activity by Interface
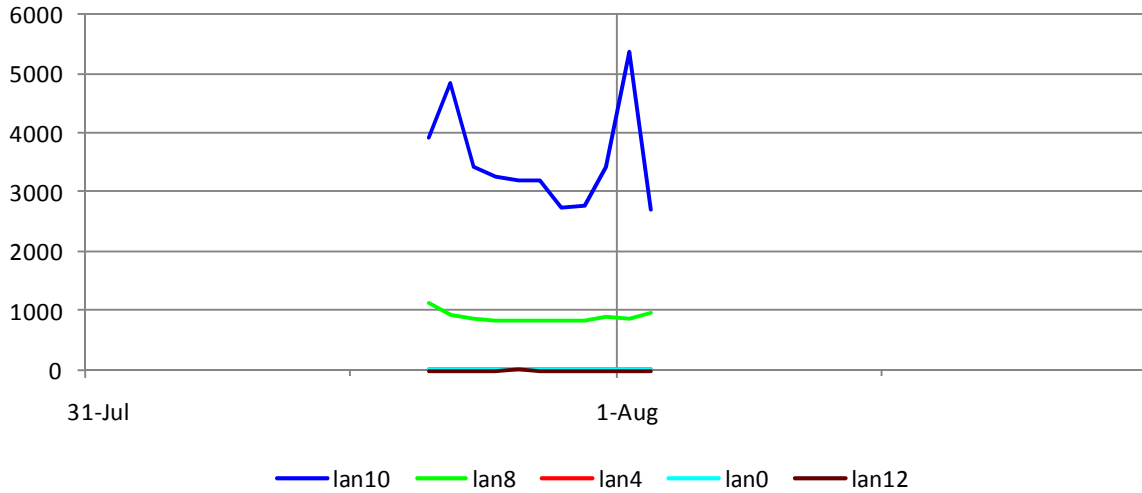### Figure 21 – Packets per second (hourly averages)



Figure 21 – This graph gives a visual indication of the hourly average number of data packets that were transmitted and received on the indicated LAN. Units on the vertical scale (Y-axis) represent the number of internet packets per second sent and received.

On systems with more than five LAN interfaces only the top five are graphed. The LANs included in the graph above account for 100.0% of all LAN activity that occurred during the monitored period.

A LAN packet can be as small as 64 bytes or as large as 1,500 bytes of data.

| LAN Name | Recs | Packets In | In Rate | Packets Out | Out Rate | Bytes In | Bytes Out |
|---|---|---|---|---|---|---|---|
| lan0 | 11 | 24,190 | 0.73 | 0 | 0.00 | 0.0 KB | 0.0 KB |
| lan10 | 11 | 69,731,237 | 2009.48 | 51,256,779 | 1529.06 | 61.0 GB | 39.9 GB |
| lan12 | 3 | 3,247 | 0.37 | 0 | 0.00 | 0.0 KB | 0.0 KB |
| lan14 | 3 | 3,247 | 0.37 | 0 | 0.00 | 0.0 KB | 0.0 KB |
| lan4 | 11 | 24,190 | 0.73 | 0 | 0.00 | 0.0 KB | 0.0 KB |
| lan8 | 11 | 14,916,783 | 441.53 | 15,548,593 | 459.37 | 3.0 GB | 5.7 GB |

## SYSTEM ENVIRONMENT

The information in this report is based on data collected during the two day period from Friday, July 31, 2009 through Saturday, August 1, 2009. Data was collected during each of the 11 hours during this period.

The system being analyzed is a **ia64 hp superdome se** with **143 gigabytes** of memory. The system processing workload was predominately "Normal" usage with a User to System ratio of 1.3:1. In OLTP environments a User-to-System ratio of about 3:1 may be considered good. Normal processes accounted for about 1% of the activity with real-time processing accounting for less than 1%. Nice CPU amounted to 23.8%. System CPU averaged about 19% during the prime shift hours. Processing occurred on all days, 24 hours a day, including weekends. Prime shift for this report is **08:00-17:00**.

**The remainder of this page is only a suggestion on how you might structure the rest of the report.**

## THE CPU

The CPU was 100% busy during most of the monitored period. The majority of this "busy time" (80% or above) was for direct user activity, versus system overhead.

CPU utilization exceeding 80% is typically an indication of a potential bottleneck. However, with background processing active, it is common to have CPU utilization exceed the 80% mark, since think time for user data entry, etc., is not required. Interactive users should not be experiencing performance problems.

Another indicator of a CPU bottleneck is the CPU Queue depth. A CPU queue depth consistently greater than five is an indicator that processes are continually waiting to use the CPU. Your CPU queue averaged a little over one and peaked at three (See Figure 3). So, although the CPU was continually being used at 100%, the CPU is not a bottleneck.

## MEMORY

The Memory Management CPU utilization, a component of the category "Other" was typically below the 5% mark where potential pressure is indicated. With the exception of a short duration peak on Friday at 11:00 AM, the Memory Queue was one or less. This is an indication that not much swapping activity was necessary (See Figure 3). Figure 9 shows that the Memory Management IO rate was well below the 5 IOs per second mark which indicates a memory shortage. All indications of a memory shortage are negative.

## DISK IO

The Physical Disk IO utilization was well below maximum disk capacities. The Disk Queues were virtually non existent. However, disk placement strategy should be reviewed to spread out the IOs among the three drives. Drive c1d0s is rarely accessed. Almost all disk accesses are spread out fairly evenly between drives c3d0s and cld0s. Load management is very important in planning for future growth.

## RECOMMENDATIONS

Apex is making full use of the CPU on this system. Continued monitoring of this resource and the transaction throughput is recommended in order to keep up with pro-active growth planning. Evening out the Disk IOs between the three drives will help insure the best utilization of your current system configuration.

### Parm File

# DO NOT RETAIN THE FOLLOWING IN THE REPORT TO THE CUSTOMER.

Insert parm file here or delete this page. To delete the last page select all paragraphs on the page (except the header and footer) and press the delete key. With the cursor at the left margin, just under the header press the backspace key. This may leave the page break marker on what is now the last page. If the paragraph button on the standard toolbar isn't depressed click it so the page break becomes visible if it is there. If it is there select it and delete it. Finally right click in the table of contents, select "Update Field," then select "Update Entire table."

To add another page to the appendices section insert a page break and copy the second "tc" field above to the first line of the new page. Don't forget to change the Appendix number and description in the "tc" field. If you don't see the "tc" fields immediately under the header you can make them visible by pressing control-shift asterisk.