Paper 280-28

SAS® Application Performance Monitoring for UNIX

John Hall, Hewlett Packard

Abstract

In many SAS application environments, a strategy for measuring and monitoring system performance is key to maintaining Service Level Objectives (SLOs).

This paper will discuss various aspects of performance monitoring in a SAS environment, both from a dedicated SAS server perspective as well as multiple applications hosted on a single server. The importance of performance baselines, capacity planning, and configuration management will be discussed as it pertains to the SAS application environment. Evaluation of system components such as CPU, memory, disk, and network performance will be discussed individually as well as collectively to develop a complete perspective on both application as well as system level performance. Finally, a case study is reviewed to illustrate how metrics may be used in understanding SAS application performance.

SAS Applications for a single server environment

This is the 'typical' way that many users deploy SAS applications – a dedicated server environment. There may be one, or a few, classes of users:

- Occasional, ad-hoc reporting
- Batch processing
- 'Power' users, often data mining or CPU-intensive operations

This server environment provides an opportunity to optimize the server configuration specifically for SAS – that is, to tailor disk IO, file systems, kernel configurations – to the particular needs of the SAS applications executing on the server. Although there may be multiple SAS users, and thus multiple concurrent SAS sessions, executing on the server, the configuration may be optimized for the efficient execution of SAS applications.

Stacked servers

The stacked server configuration is usually found with the larger, enterprise-class servers such as the HP Superdome and RP8400 servers. These servers provide ample processing and IO capacities to support many different applications. These environments are often loaded with multiples of different types of applications – transaction-oriented RDBMS apps, data warehouse apps, etc. Each of these types of application environments can place different demands on the server, making performance monitoring and tuning more challenging.

Establishing Performance Baselines

“If you don’t know where you’re going, any road will take you there”
-from Alice in Wonderland

This simple phrase describes the situation often found in diagnosing performance problems – there is not a clear definition of ‘normal’. What does the system look like when it is performing as expected? The memory is 100% utilized – is that normal? A SAS program to load a large data warehouse only using 23% of available CPU resources - is that normal? Clearly, establishing a baseline, or set of baselines, is important in understanding and identifying performance issues.

To establish a performance baseline, capture performance data over a period of time. Multiple periods may be used to capture a more complete picture of system performance. For example, if the weekly refresh of the data warehouse is every Monday morning from 2:00 – 6:00 AM, data captured during that time interval becomes a performance baseline for data warehouse loading. Capturing multiple instances of the data warehouse loading will improve the overall performance baseline by providing comparisons over time. This not only provides baseline performance data, but can also serve as a basis for capacity planning or development of service level agreements.

When a suspected performance problem is reported, comparing the baseline statistics to current data can provide valuable insight. Numbers far above or far below the baseline numbers are candidates for further investigation. Many performance tools, such as Perfview from HP, provide a graphical interface, which allows the overlay of multiple graphs. This provides a quick way to compare performance baseline data with the current data and easily detect differences.

Configuration Management

Configuration management is used to describe many areas of tracking and control throughout the computer industry. For the purposes of this paper, we will define configuration management as the tracking and control of a system configuration – that is, the hardware and software components of the server. A few examples will further illustrate: IO card placement, amount of memory installed, values of kernel parameters, and settings of tunable parameters for specific applications (such as MEMSIZE and WORKPATH in the SAS configuration file).

Documenting and tracking changes to the system configuration, and correlating this data with the system baseline data previously discussed can assist greatly in identifying and isolating both actual and potential problems in application performance.

There are many types of configuration tracking tools from various UNIX vendors, some of which have multi-platform (they run on different flavors of UNIX as well as other platforms) capabilities. Links to some of the more popular tools are provided in the 'Links' section at the end of this paper.

Speed isn't everything – or is it?

Performance is measured in different ways. To some, it means being able to complete their SAS job in the shortest time possible. To others, it means having the job complete in a specified timeframe – for example, I need this job to finish before I leave at 5:00. Still others measure performance on a more global scale, looking at the entire server – I need to be able to support 30 concurrent users with a response time of less than 5 seconds.

It's important to establish a definition of performance as early in a project as practical. This definition can often help in determining what attributes a server will need to meet the desired performance goals. Sometimes, these goals are identified in general terms, as above. More often, performance goals are more specifically defined in a Service Level Objective, or SLO. SLOs provide specific and measurable performance targets in diverse areas such as security, capacity, availability and support. These objectives are usually then combined with others into a more formalized document called the Service Level Agreement (SLA).

The importance of the definition of performance cannot be overstated – the clear understanding of this term for your particular environment will determine how performance is measured and monitored, and expectations about how system performance will be managed.

Components of System Performance

System performance is usually described in terms of the components of the system. There is an interdependence of these components on each other to deliver a balanced system – one where the processing power of the CPUs is not compromised by a slow IO system or high memory latency.

Here are the major components commonly found in a system, with a short explanation and some of the more popular UNIX tools used to evaluate their performance:

<u>CPU</u> – Central Processing Unit, although this term is used more commonly to describe the processor as well as its surrounding components such as cache memory.

Tools:

glance (multi-purpose)
sar (multi-purpose)
ps
xload
top

<u>Memory</u> – used for short-term storage of instructions and data. There are generally two types of memory: physical memory (RAM), which is the memory installed in the server on physical cards, and virtual memory, which includes physical memory as well as the backing store (the overflow area used when physical memory is fully utilized).

Tools:

glance (multi-purpose)
sar (multi-purpose)
size
vmstat
ipcs

<u>Disk</u> – includes all of the hard disk types that are available to the server. These may be directly attached, over a network, or part of a Storage Area Network (SAN). For purposes of performance, the disk resource involves the movement of data between the physical disk and the server rather than the capacity of the disk. For SAS processing, access to the disk is through the file system and IO management subsystems. This is often the most important system resource in a SAS application environment, and tuning of file systems and matching of workloads to the available IO configuration is important to SAS performance.

Tools:

glance (multi-purpose)
sar (multi-purpose)
bdf
df
iostat

Network – an often-overlooked component of server performance, poor network performance can severely degrade application performance if the application is dependent on data accessed over the network. There are network performance dependencies for many SAS installations. For example, consider SAS database engine servers, SAS Intrnet, and SAS Integration Technologies. These, as well as additional services, are dependent on a reliable, robust network. In addition, for SAS Version 9 evaluation of the new SAS metadata and OLAP server technology will require consideration of network performance. NFS-mounted file systems are also a popular method of file sharing which can affect the overall performance of SAS applications.

Tools:

glance (multi-purpose)
netstat
nettune (HP-UX 10.x) and ndd (HP-UX 11.0 and later)
nfsstat

A practical example

In order to demonstrate some of the fundamental concepts discussed earlier in this paper, analysis of the execution of a SAS application is in order. A SAS MP CONNECT job was chosen to illustrate the ability to analyze multiple processors throughout the job execution.

The SAS job chosen for this demonstration loops through a standard set of Base SAS procedures, creating a SAS MP CONNECT session for each iteration of the loop. For this test, we chose to create 4 concurrent sessions.

For illustrative purposes, we chose to use GlancePlus/UX from Hewlett Packard to track performance of the SAS job. GlancePlus/UX is an interactive performance-monitoring tool that displays various system and application metrics on a series of screens. These screens can be text or graphical (Motif) based. Glance is available for HP-UX, Solaris, and AIX.

GlancePlus/UX displays both process (specific application) and global (system-wide) information, with tunable thresholds to allow for filtering output to items of specific interest. More detailed information on GlancePlus/UX and its capabilities can be found in the Links section of this paper.

Figure 1 illustrates the global view of the server. This main screen shows the four graphs of CPU, disk, memory, and network over a specified time interval. Update intervals, as well as the appearance of the main screen, (color selection as well as pie chart/bar chart, etc.) can be modified. This screen is also called the control screen, because of the many functions that can be selected from the pull-down menus.

The ‘ALARM’ functionality of GlancePlus allows the definition of control parameters (e.g., when System CPU utilization exceeds 65% for more than 20 minutes). When an ALARM condition is detected, that section of the main screen (CPU, in this case) changes color. It can also be configured to raise a certain Glance window (CPU Utilization, in this case) and/or generate a mail message.

Each of these four areas can be 'drilled down' by selecting the icon below the screen.

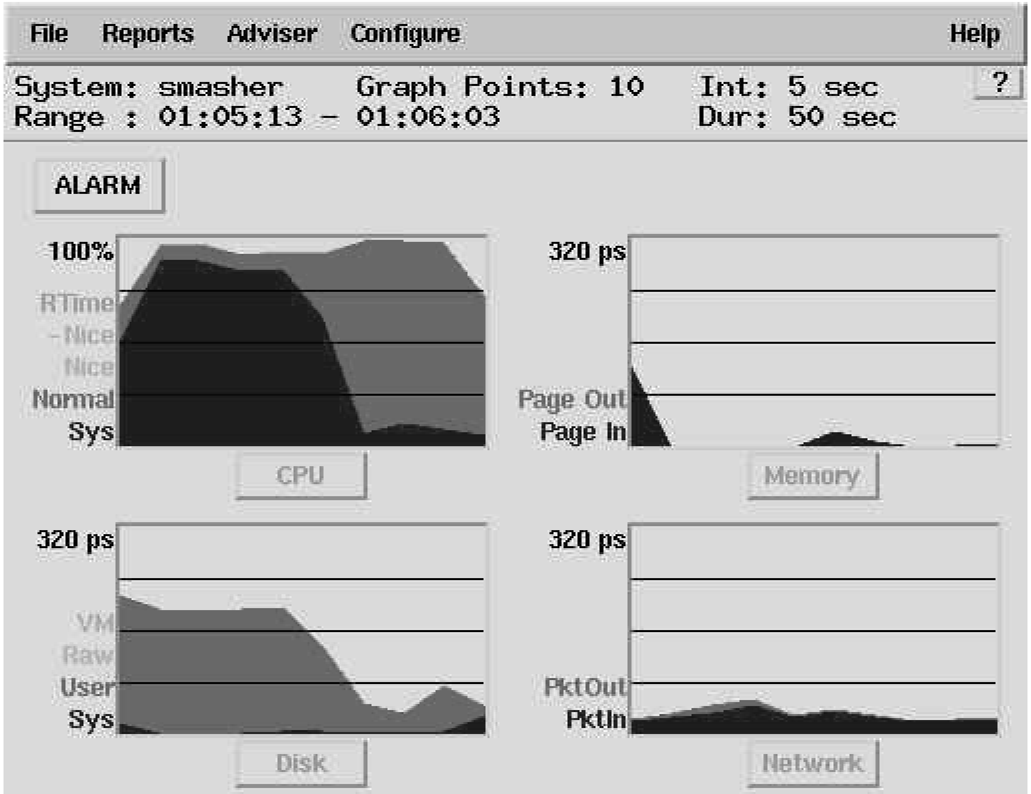


Figure 1 - Global Chart

The CPU chart, in Figure 2, shows two interesting graphs. The top chart portrays a history of the one-minute Load Average metric, which can be used as one indication of a CPU bottleneck. On the bottom is the CPU Utilization by Processor chart, which shows a stacked bar chart for each CPU in the system. This chart is most useful in Symmetric Multi-Processor (SMP) environments. More detailed information on a per-processor basis is available in a tabular chart as seen in Figure 3.

In this illustration (Figure 2), all 4 CPUs are almost fully utilized. (This is not to infer that all processors being fully utilized is a bad thing). By what process(es)? We will use other screens to isolate the specific process(es) that are consuming the CPU resources.

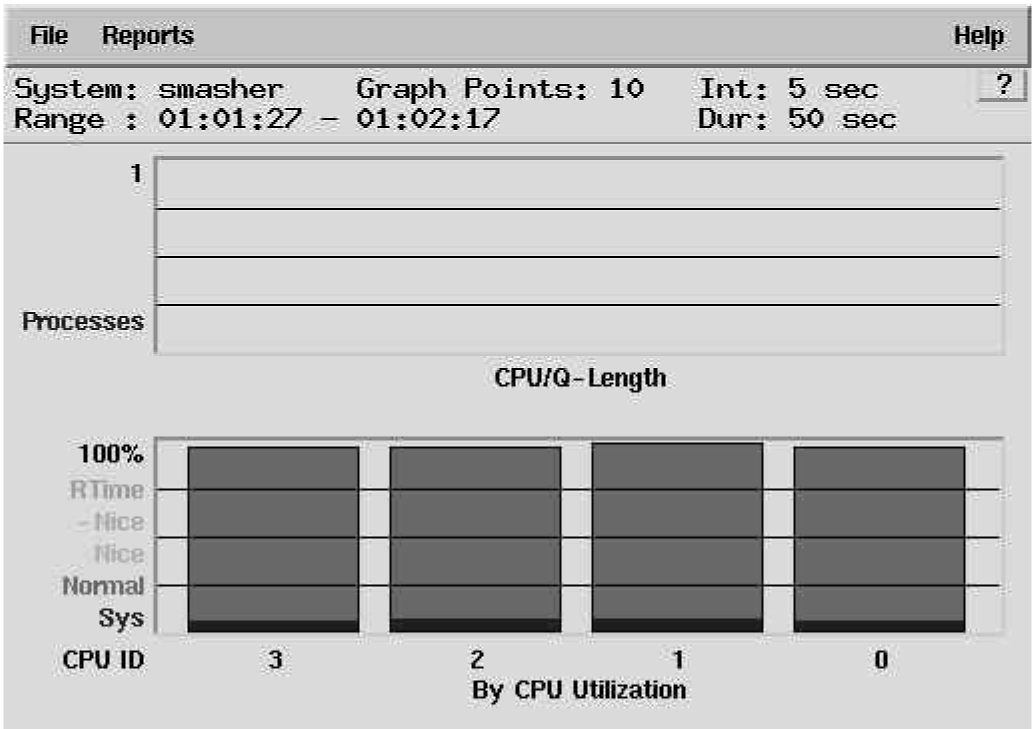
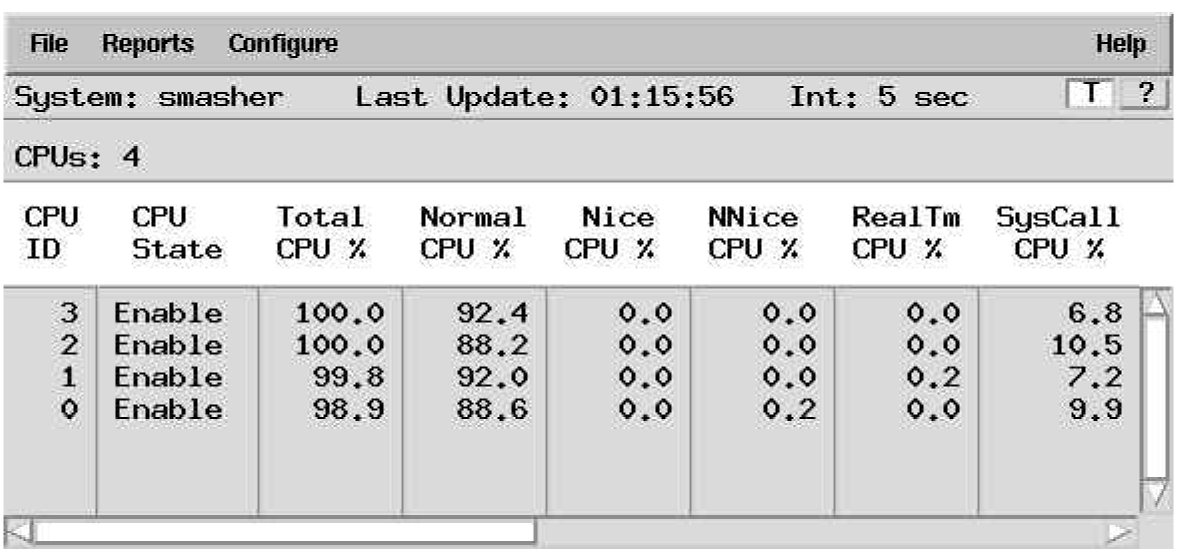


Figure 2 - CPU Chart



File Reports Configure Help

System: smasher Last Update: 01:15:56 Int: 5 sec

CPUs: 4

CPU ID	CPU State	Total CPU %	Normal CPU %	Nice CPU %	NNice CPU %	RealTm CPU %	SysCall CPU %
3	Enable	100.0	92.4	0.0	0.0	0.0	6.8
2	Enable	100.0	88.2	0.0	0.0	0.0	10.5
1	Enable	99.8	92.0	0.0	0.0	0.2	7.2
0	Enable	98.9	88.6	0.0	0.2	0.0	9.9

Figure 3 - CPU by Processor

By examining the list of processes in Figure 4 executing on the server, we can see that there are four SAS processes executing, each one almost fully utilizing the available resources on each processor. Again, this is not a bad thing – given that we see from Figure 1 there are no Queue Lengths, meaning there is nothing else waiting on processors – so having a fully utilized system is, in this case, a good thing!

File Reports Admin Configure Help

System: smasher Last Update: 23:20:17 Int: 5 sec

Processes: 5 of 118 Selected Users: 5

Process Name	PID	CPU %	Phys IO Rt	Stop Reason	Pri	User Name
sas	4059	100.0	11.7	PRI	241	john
sas	4057	99.8	19.5	PRI	241	john
sas	4058	98.9	20.4	PRI	241	john
sas	4060	98.2	12.6	PRI	241	john
vxfsd	36	0.0	2.2	SYSTM	138	root

Figure 4 - Process List

By selecting one of the processes from the process list, we can determine more detail about a specific process (Figure 5). This report screen shows per-process CPU, memory, and disk performance metrics for the selected process over the specified interval. If desired, we could collect cumulative statistics for the process over the entire duration of the process.

File Reports Admin Configure Help

System: smasher Last Update: 23:20:07 Int: 5 sec

sas PID: 4060 PPID: 4056 User: john State: active

Interval for collection: 5 sec
Process Start Time : Tue May 7 23:18:14 2002

Total CPU	99.4%	Logical Reads	1558	Total RSS (kb)	4256
		Logical Writes	0	Total VSS (kb)	7784
User CPU (Nice)	96.8%	Rem Log Reads	0		
System Call CPU	2.5%	Rem Log Writes	0	Traps	310
Interrupt CPU	0.0%	Phys. Reads	0	Vir Faults	0
Cont Switch CPU	0.0%	Phys. Writes	0	Mem Faults	0
		Rem Phys Reads	0	Disk Faults	0
Scheduler	HPUX	Rem Phys Writes	0	Deactivations	0
Priority	241	FS Reads	0	Forks & VForks	0
Nice Value	20	FS Writes	0		
Dispatches	317	VM Reads	0	Signals Received	0
Forced CSwitch	285	VM Writes	0	Messages Sent	0
Voluntary CSwitch	4	System Reads	0	Messages Received	0
Running CPU	2	System Writes	0		
CPU Switches	0	Raw Reads	0	NonDisk Logl Reads	0
Wait Reason	PRI	Raw Writes	0	NonDisk Logl Writes	0
Effective UserID	105	Phys. IO Rate	0.0	NonDisk Phys Reads	0
Real User ID	105	Total IO Bytes	0kb	NonDisk Phys Writes	0

Figure 5 - Process Resources (SAS)

Similar reports are available for disk, memory, and network statistics. For example, Figure 6 shows a Disk Graph, showing the ten disks that are experiencing the heaviest usage. This chart shows a striped file system utilization profile.

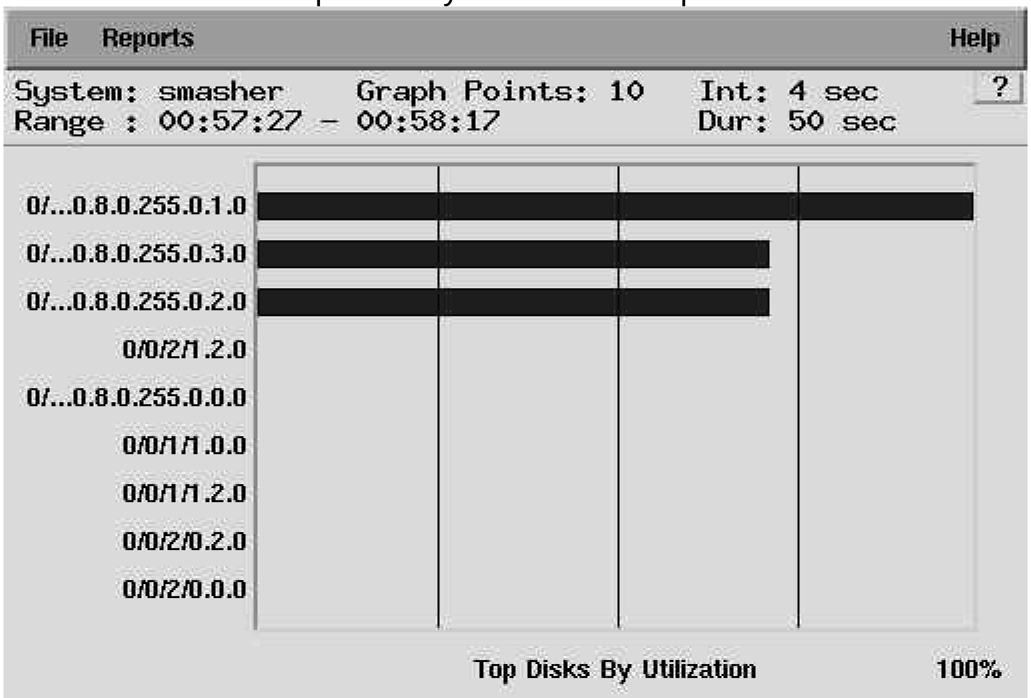
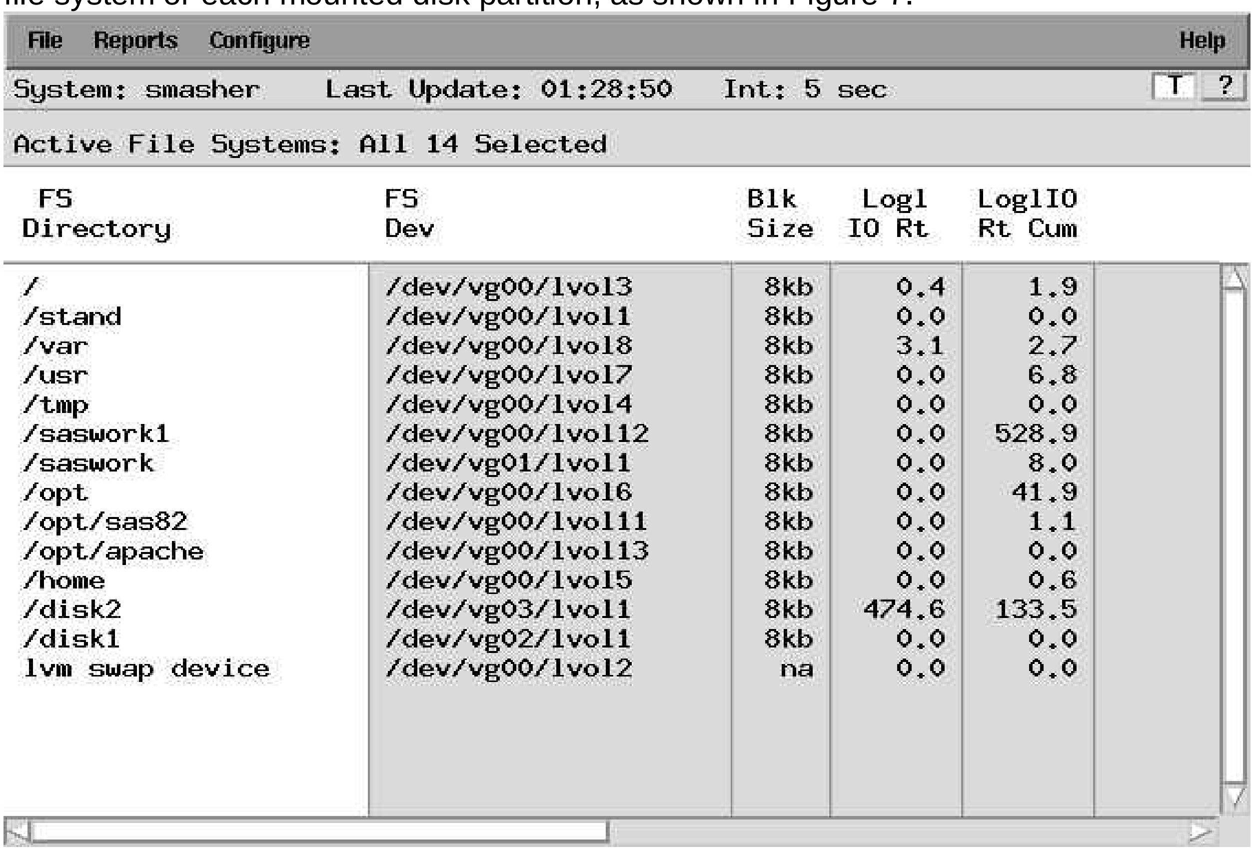


Figure 6 - Disk Graph

This information is also available in table form, displaying IO metrics for each mounted file system or each mounted disk partition, as shown in Figure 7.



File Reports Configure Help

System: smasher Last Update: 01:28:50 Int: 5 sec

Active File Systems: All 14 Selected

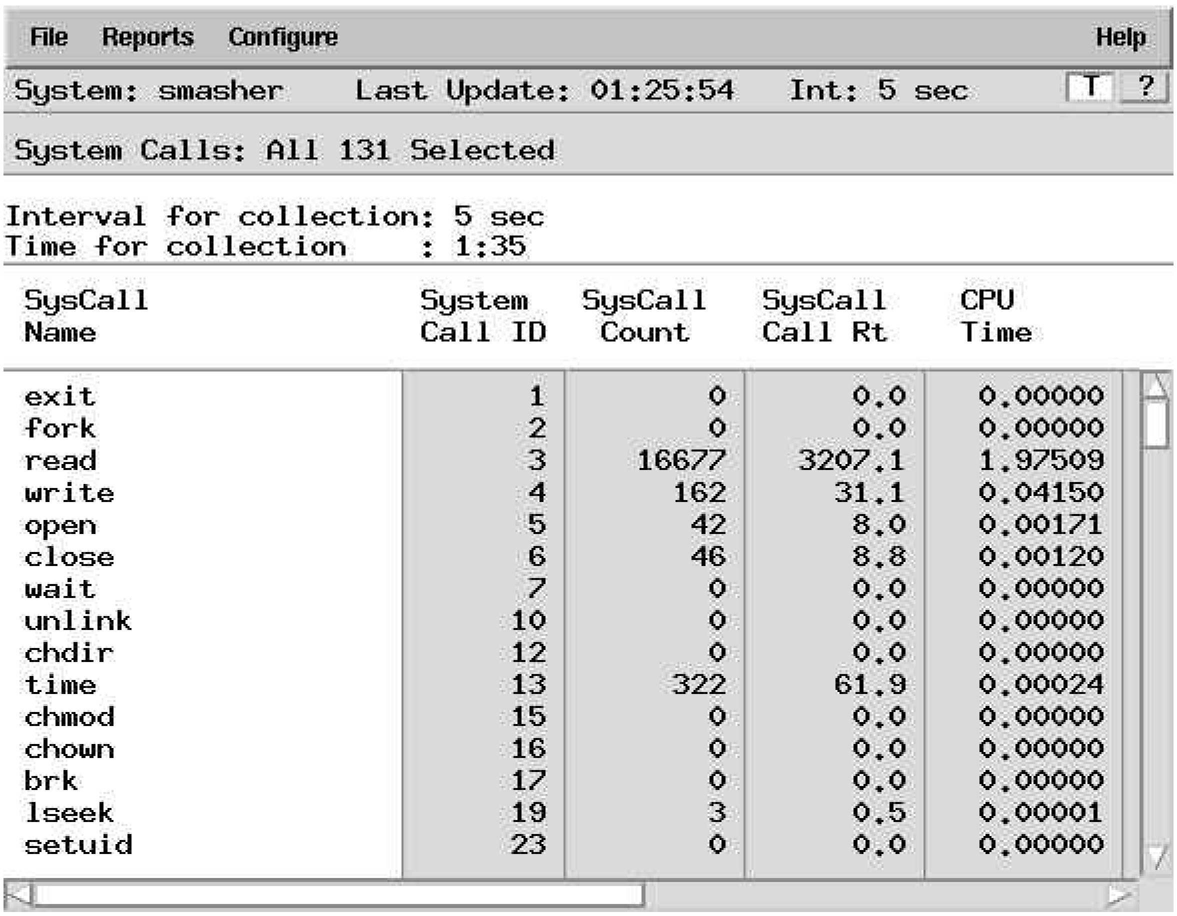
FS Directory	FS Dev	Blk Size	Logl IO Rt	LoglIO Rt Cum
/	/dev/vg00/lvol3	8kb	0.4	1.9
/stand	/dev/vg00/lvol1	8kb	0.0	0.0
/var	/dev/vg00/lvol8	8kb	3.1	2.7
/usr	/dev/vg00/lvol7	8kb	0.0	6.8
/tmp	/dev/vg00/lvol4	8kb	0.0	0.0
/saswork1	/dev/vg00/lvol12	8kb	0.0	528.9
/saswork	/dev/vg01/lvol1	8kb	0.0	8.0
/opt	/dev/vg00/lvol6	8kb	0.0	41.9
/opt/sas82	/dev/vg00/lvol11	8kb	0.0	1.1
/opt/apache	/dev/vg00/lvol13	8kb	0.0	0.0
/home	/dev/vg00/lvol5	8kb	0.0	0.6
/disk2	/dev/vg03/lvol1	8kb	474.6	133.5
/disk1	/dev/vg02/lvol1	8kb	0.0	0.0
lvm swap device	/dev/vg00/lvol2	na	0.0	0.0

Figure 7 - IO by File System

Note that GlancePlus/UX is a real-time tool – it displays what is happening as it occurs. There are some capabilities to capture some of the information from GlancePlus/UX, although it is highly recommended to use a post-processing tool such as HP Openview Performance to provide more comprehensive analysis and forecasting capabilities.

There are *many* other screens and charts which are available in GlancePlus/UX. Some of these are useful for establishing a performance baseline, or just getting a 'feel' for what's happening with the server at a specific point in time.

Figure 8 shows system calls for the entire system. This displays global information about the count and elapsed time of each system call. Unlike other Glance windows, the System Call window does not begin collecting data until the window is opened. Once opened, it continues to collect data until GlancePlus/UX is exited, even if the System Calls window is closed.



SysCall Name	System Call ID	SysCall Count	SysCall Call Rt	CPU Time
exit	1	0	0.0	0.00000
fork	2	0	0.0	0.00000
read	3	16677	3207.1	1.97509
write	4	162	31.1	0.04150
open	5	42	8.0	0.00171
close	6	46	8.8	0.00120
wait	7	0	0.0	0.00000
unlink	10	0	0.0	0.00000
chdir	12	0	0.0	0.00000
time	13	322	61.9	0.00024
chmod	15	0	0.0	0.00000
chown	16	0	0.0	0.00000
brk	17	0	0.0	0.00000
lseek	19	3	0.5	0.00001
setuid	23	0	0.0	0.00000

Figure 8 - System Calls

Figure 9 shows current and maximum utilization for system tables. This graph presents a way to identify potential system table capacity problems by showing the maximum usage in relation to capacity. Figure 10 shows similar information in a table form, displaying the use and configured sizes of system tables and selected buffer pools. This information can be used to determine how well the system kernel parameters are tuned to the average workload being monitored.

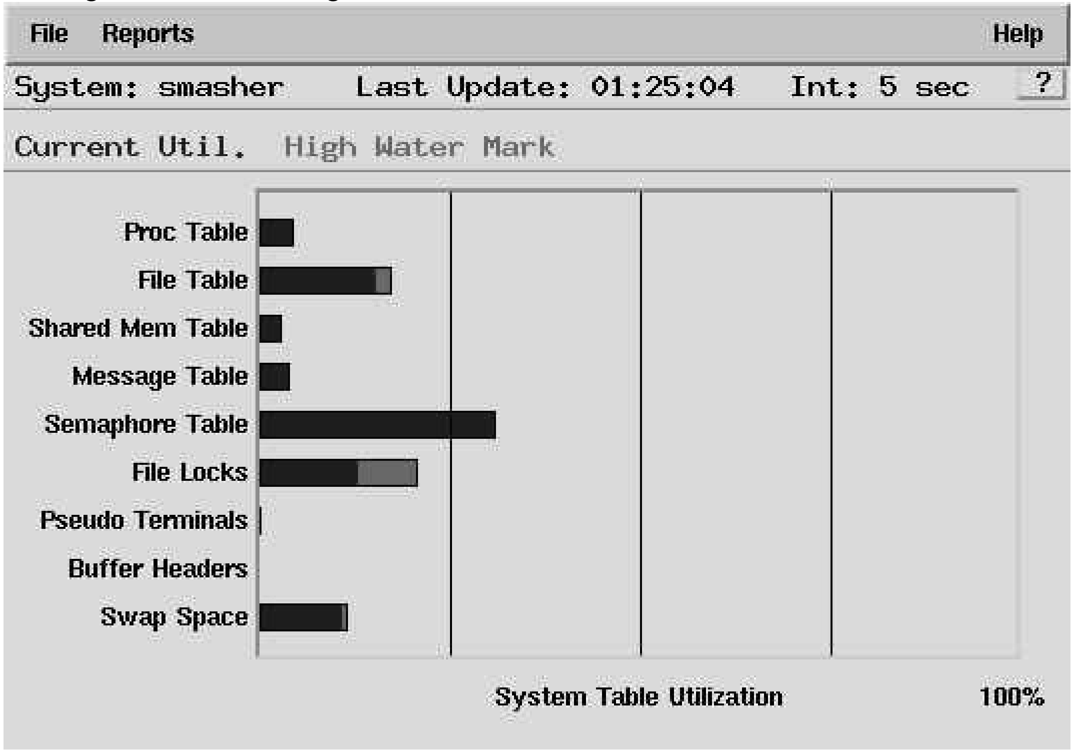


Figure 9 - System Table Utilization

File Reports Help

System: smasher Last Update: 01:27:29 Int: 5 sec ?

System Table	Avail	Used	Util%	High%
Proc Table (nproc)	2048	98	5%	5%
File Table (nfile)	3010	536	18%	18%
Shared Mem Table (shmmni)	200	6	3%	3%
Message Table (msgmni)	50	2	4%	4%
Semaphore Table (semmni)	64	20	31%	31%
File Locks (nflocks)	200	42	21%	21%
Pseudo Terminals (npty)	200	1	1%	1%
Buffer Headers (nbuf)	na	80409	na	na

System Table	Avail	Reqs	Used	High
Shared Memory	200.0gb	13.1mb		
Message Buffers	16kb		0kb	0kb
Inode Cache (ninode)	3008		985	1589
DNLC Cache	4032			

	Min	Max	Avail	Used	High
Buffer Cache	204.8mb	614.4mb	614.4mb	614.4mb	614.4mb

	Avail Size	Used Size	Reserved Size	Util%
Swap Space	7.1gb	506mb	871mb	12%

Figure 10 - System Table Chart

Finally, Figure 11 shows a System Information table that provides a summary of information, including the hostname, system type, operating system information, and current system parameter values.

File Reports Help

System: smasher Last Update: 01:27:55 Int: 5 sec ?

Nodename	hpweb4	OS Name	HP-UX
System Type	HP9000	OS Release	B.11.00
Model	9000/800/L3000-5x	OS Kernel Type	64 bits

Number of CPUs	4
Number of Disks	2
Number of Swap Areas	2
Number of Net Interfaces	2
Number of Available Volume Groups	2
Physical Memory	4.00gb
Available Swap Area	7.1gb
Memory Region Max Page Size	64.0mb

Figure 11 - System Information

Conclusion

Understanding and managing server performance can be a complicated task. Establishing performance baselines can help set and manage user's expectations as well as serve as a reference for performance comparision. The use of tools to establish performance baselines, as well as monitor and manage system and application performance, is crucial in developing and maintaining acceptable system performance.

Links

Excellent paper on UNIX performance tools and their usage:

http://www.circle4.com/jaqui/papers/webunuk.html

HP-UX Kernel Tuning and Performance Guide

http://www.ussg.iu.edu/hp/hpux-tune.html

"Making you GlancePlus Pak Perform" – an excellent tutorial on HP GlancePlus

http://www.interex.org/pubcontent/enterprise/jul00/14grum.html

"Using Glance in Advisor mode to collect and log diagnostic metrics"

http://www.openview.hp.com/library/papers/Papers_HTML-6.asp

SarCheck™ - formats and analyzes sar logs

http://www.sarcheck.com/schp.htm

HP-UX Performance Cookbook

http://www.interex.org/conference/iworks2001/proceedings/1034/1034.pdf

References

Robert F. Sauers and Peter S. Weygant, “HP-UX Tuning and Performance: Concepts, Tools, and Methods”, Prentice Hall, July 1999, ISBN 0-13-102716-6.

Tom Madell, “Disk and File Management Tasks on HP-UX”, Prentice Hall, October 1996, ISBN 0-13-518861-X.

Author Contact Information

John Hall
Hewlett Packard
15501 Weston Parkway, Suite 150
Cary, NC 27513
(919) 677-3606
j.hall@hp.com