



Capacity Planning as a Performance Tuning Tool—Case Study for a Very Large Database Environment

*Gamini Bulumulle and Marcos Bordin, Sun
Professional Services*

Sun BluePrints™ OnLine—July 2003



<http://www.sun.com/blueprints>

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95045 U.S.A.
650 960-1300

Part No. 817-3176-10
Revision A, July 2003

Copyright 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of the product or of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, docs.sun.com, BluePrints, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and in other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and in other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc. ORACLE is a registered trademark of Oracle Corporation.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. a les droits de propriété intellectuels relatants à la technologie qui est décrit dans ce document. En particulier, et sans la limitation, ces droits de propriété intellectuels peuvent inclure un ou plus des brevets américains énumérés à <http://www.sun.com/patents> et un ou les brevets plus supplémentaires ou les applications de brevet en attente dans les Etats-Unis et dans les autres pays.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, docs.sun.com, BluePrints et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc. ORACLE est une marque déposée registre de Oracle Corporation.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciées de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.



Please



Adobe PostScript

Capacity Planning as a Performance Tuning Tool—Case Study for a Very Large Database Environment

This article discusses the performance and scalability impact due to severe CPU and I/O bottlenecks in a very large database (over 20 terabytes). It describes the methodologies used to collect performance data in a production environment, and explains how to evaluate and analyze the memory, CPU, network, I/O, and Oracle database in a production server by using the following tools:

- Solaris™ Operating Environment (Solaris OE)—Standard UNIX® tools
- Oracle STATSPACK performance evaluation software from ORACLE® Corporation.
- Trace Normal Form (TNF)
- TeamQuest Model software from Team Quest Corporation
- VERITAS Tool VxBench from VERITAS Corporation

The article is intended for use by intermediate to advanced performance tuning experts, database administrators, and TeamQuest specialists. It assumes that the reader has a basic understanding of performance analysis tools and capacity planning.

The article addresses the following topics:

- Analysis and High-Level Observations
- Resolving CPU and I/O Bottlenecks Through Modeling and Capacity Planning
- Conclusions
- Recommendations
- I/O Infrastructure Performance Improvement Methodology
- Data Tables

The article discusses the chronological events of “what-if” analysis using the TeamQuest modeling software to resolve CPU and I/O bottlenecks, for projections of the database server performance and scalability, and to simulate effects of performance tuning. It also provides a detailed analysis of the findings, and discusses the theoretical analyses and solutions.

Finally, it provides an in-depth discussion and analysis of the solution, that is, how to resolve the I/O and CPU bottlenecks by balancing the I/O on the existing controllers and adding new controllers.

The first part of the article presents the result of performance analysis with respect to the CPU, I/O, and Oracle database using the tools previously stated. The second part, describes the CPU, I/O tuning, and capacity planning methodology, and its results. Finally, the article provides conclusions, recommendations, and the methodology for improving I/O infrastructure performance.

The performance analysis, tuning, and capacity planning methods described in this article can be applied to any servers in a production environment. Performance analysis and capacity planning is a continuous effort. When the application or the environment change, as result of a performance optimization for instance, the performance analysis has to be revisited and the capacity planning model recalibrated. For a system that is operating on the upper limit of its capacity, performance optimization is a continuous search for the next resource constraint.

The performance analysis methodology starts with an analysis of the top five system resources being utilized during the peak-period and the percentage of utilization associated to each one. In this case study, the I/O controllers and CPUs topped out at roughly 80 percent utilization and the disk drives reached their peak at 70-to-80 percent utilization. Once these thresholds were reached, response times degraded rapidly (depending the workloads, more than one workload may be depicted).

Teamquest performance tools were used to provide the performance analysis and capacity planning results. The Teamquest Framework component was installed on the systems to be monitored. This component implements the system workload definitions and collects detailed performance data. The Teamquest View component allows real time and historical analysis of the performance data being collected on any number of systems on the network.

Analysis and High-Level Observations

In this case study, the primary focus was on I/O and CPU utilization as it was observed that server was neither memory nor network bound. CPU utilization was high during the peak period, sometimes above the 80 percent threshold considered acceptable to avoid impacting overall system performance. The

critical component of CPU utilization was I/O Wait, which accounts for about 50 percent of CPU utilization during the peak period. This corresponds to the time the system uses (wastes) managing the I/O queue and waiting for I/O. CPU wait in user mode reached 6-to-8 percent on some peak periods.

The following paragraphs analyze and show the performance data obtained with VERITAS, TeamQuest, and the Solaris OE standard tools and the TNF and Oracle STATSPACK data for the top five system resources and preliminary data for Oracle.

VERITAS Analysis and Observation

TeamQuest reported 1,300 I/O operations per second (IOPS) on the system during the peak period. Tests with the VERITAS VxBench indicate a total capacity of the I/O subsystem to sustain more than 11,000 IOPS. These are logical IOPS, what the operating system sees as the controllers get hit. Although there are differences in these two measurements, in both scenarios the data was collected at the operating system level.

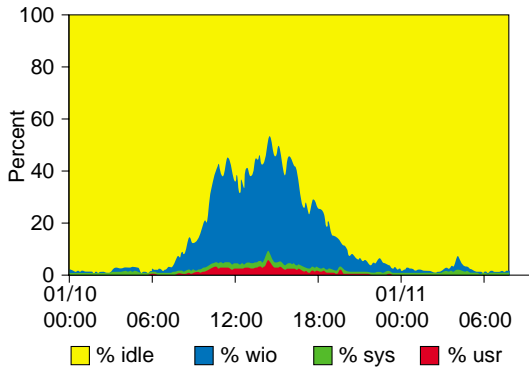
TeamQuest Analysis and Observations

The TeamQuest analysis and observations concern the CPU and disk utilization.

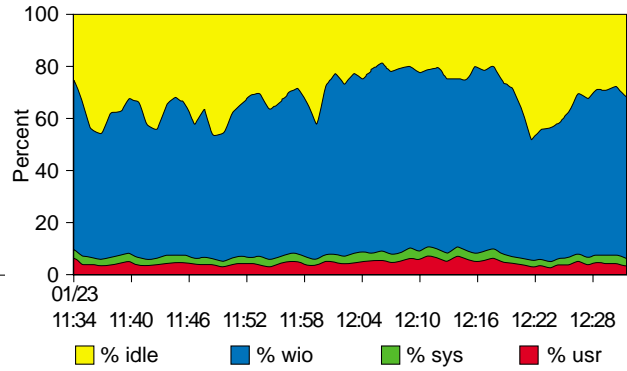
CPU

The following charts show a CPU utilization of about 50-to-60 percent during the working hours (10:00 a.m. to 4:00 p.m). A small peak occurs during the night, possibly related to some batch processing.

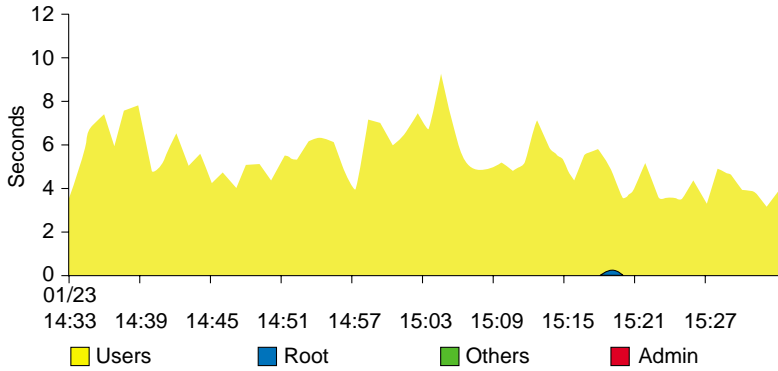
CPU Utilization For 24 Hours



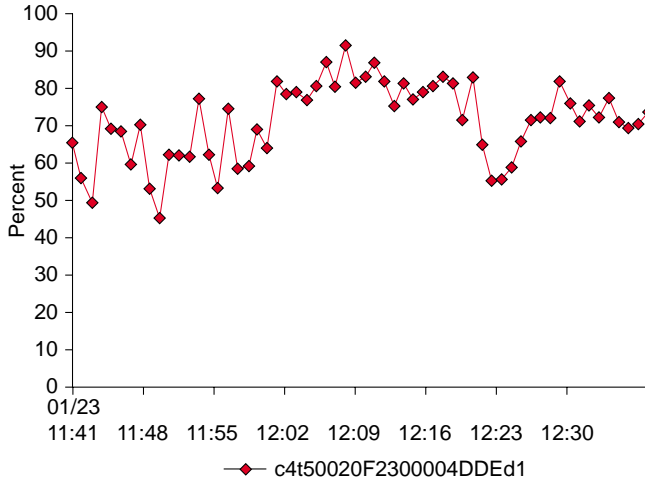
CPU Utilization During Peak Period



Wait CPU Time By Workload



Disk Utilization During Peak Period



The graphs and the tables in “Data Tables” on page 36 indicate a significant proportion of wait I/O time. Typically, this indicates poor disk responsiveness in single processor systems. The disks were unable to service all of the transaction requests and an I/O requests queue formed. While this operation fully utilizes the disks, it forces the operating system to spend a lot of time managing the queue and waiting for I/O to complete before additional processing can take place.

Disks

Disk drives reach their peak at 70-to-80 percent utilization. The values shown in the graph indicate very high device utilization.

Solaris OE Tools Analysis and Observations

The “%busy” column in TABLE 4 in the “Data Tables” on page 36” shows the disk utilization. Disks that are utilized over 65 percent should be considered a concern and any disk utilized over 90 percent represents a serious performance bottleneck.

However, disk utilization alone (%busy) is not an indicator of bad disk performance. You should always correlate (%busy) with (service time in milliseconds (ms)). The rule of thumb is that %busy is high and disk service time is high too (over 50 ms) for a long period of time.

Oracle Analysis and Observation

The following table lists the data collected for the top five wait events by using the Oracle STATSPACK tool.

Event	Waits	Wait Time (cs)	%Total Wait Time
db file sequential read	1,501,970	2,913,058	99.71
db file scattered read	2,346	5,449	.19
db file parallel read	400	1,286	.04
control file parallel write	198	471	.02
latch free	364	380	.01

It is important to understand the significance of the wait event and the average wait time (ms) per wait. The focus should be on the wait event `db_file_sequential_read` which constitutes the majority of the waits (greater than 99 percent)—in this case, it is the top wait event. The average wait time per wait is 1940 ms.

Oracle uses cs for measuring time. One cs is 1/10 of a second so 2,913,058 cs for total of 1,501,970 wait events gives an average of 1940 ms.

Also, the accumulated results do not show that some wait events are really bad during peak time because the calculated average is good.

TNF Analysis and Observation

TNF analysis further confirmed the preceding results. In addition, very high queue depth and response times were observed. For TNF output data, refer to TABLE 4 in “Data Tables” on page 36”.

Resolving CPU and I/O Bottlenecks Through Modeling and Capacity Planning

The following paragraphs discuss the activities that comprise the modeling and capacity planning process:

- Establish a Baseline System Performance Database
- Define the Peak Workload Period
- Build a Baseline Model of the System
- Perform “What If” Analysis

Establish a Baseline System Performance Database

The capacity planning process starts by collecting detailed performance data for the workloads on the systems under study. Workloads are logical groups of processes running on the system that represent different applications and/or groups of users. By defining workloads, you can view system performance data in terms that make

sense from a business or functional perspective. Collect performance data for a period of time considered adequate to include the peak workload period of the system.

Define the Peak Workload Period

After the performance data has been collected, you can analyze it to define the peak workload period. The criteria used to identify the peak period varies from system to system, depending on the types of workloads being run.

For many workload types, CPU utilization is the metric used to establish the peak period. However, for database servers, disk I/O may be the most important system activity to measure. For network file servers, network load may be the defining value. CPU utilization was used to define the peak workload period for the systems in this case study.

Build a Baseline Model of the System

Once the peak period is known, you can extract the performance data and system hardware configuration for that time interval to be used as input to the system model. The model is based on each resource in the system (treat each CPU, I/O controller, disk drive, and so forth as a service queue, and view the system as a network of these queues). Use the measured peak period performance data to calibrate the model, that is, use the measured data to find the coefficients of the queuing equations and thus define the system in mathematical terms.

When calibrated, the model represents the baseline system as it existed when you collected the performance data. The baseline model shows system resource utilization, throughput, response time, and queue delay (stretch) factors for each workload defined.

Perform “What If” Analysis

With the baseline system model in hand, you can perform “what if” analysis. This analysis projects the workload growth and system configuration changes.

You can increase the amount of work the system is doing to see what the performance impact might be. You can also modify the system hardware configuration to determine the effect. Modification can include adding CPUs, increasing CPU speed, adding I/O controllers or changing their performance characteristics, adding disk drives, changing RAID characteristics, and so forth. In addition, you can combine workloads from different systems on a single system to

see how they might perform together. System modeling is a powerful tool for capacity and performance planning, allowing any number of workload intensities and system configurations to be explored in advance of system utilization growth and equipment deployment.

A set of four chart types provide the results of the “what if” analysis:

- Stretch factor
- Response time
- CPU utilization by workload
- Active resource

The “stretch factor” chart shows the amount of queuing/contention in the system for each workload. The minimum value of one means there is no queuing; that is the work is being performed as fast as it is being presented to the system. A value of two or above means that the workload is spending as much time or more waiting in a queue than it is being serviced by the system; this situation should be avoided.

$$\text{Stretch Factor} = (\text{Queue_Time} + \text{Service_Time}) / (\text{Service_Time})$$

According to the preceding formula, if the Queue_Time is zero, the stretch factor is one. This should be the case in an ideally tuned system.

The “response time” chart shows “time to completion” requests, which can be theoretically associated and are directly proportional to the actual response time on characterizing the system load.

The “CPU utilization by workload” chart shows which workloads are associated with CPU utilization (user mode).

The “active resource” chart shows utilization percentages for I/O resources.

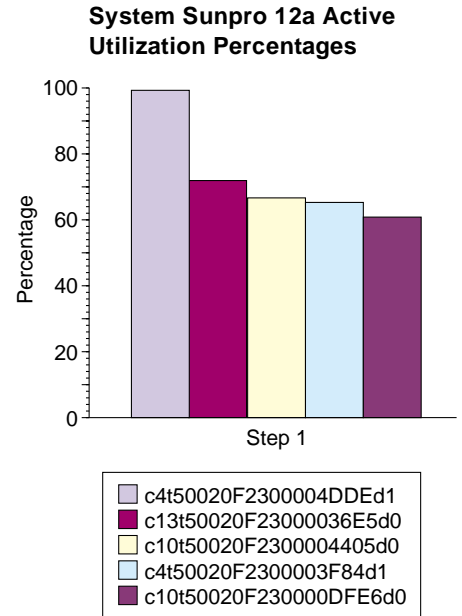
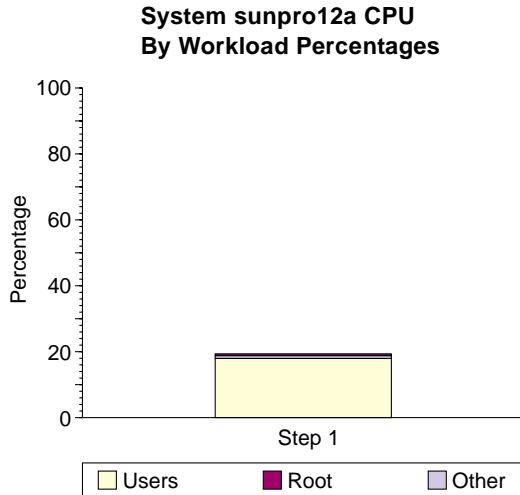
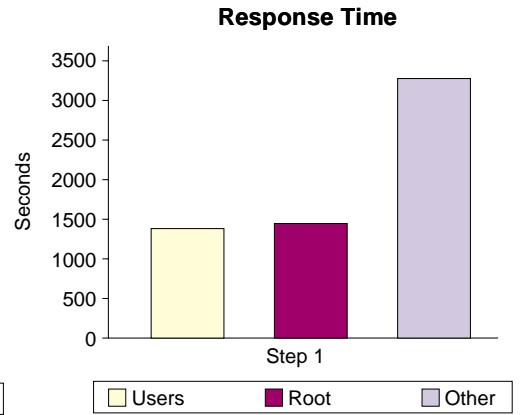
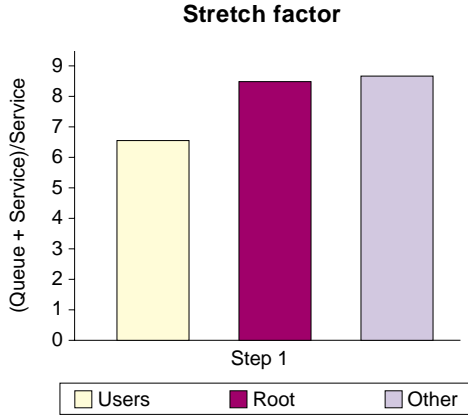
Each set of charts shows the system configuration running the baseline production workloads during the peak hour. The charts also show multiples of the baseline workload up to the point where the configuration fails to provide adequate capacity. Additional sets of charts for each system show improved hardware configurations that address the deficiencies found in the current production systems.

For the systems in the case study, physical memory size was not an issue. This statement ignores any application memory leak problems. It does, however, take into account the memory upgrades already scheduled for the production systems.

For the systems in the study, network bandwidth should not be an issue if the additional network interfaces are provided on each system, as recommended in the following paragraphs.

Baseline Charts

The following charts establish the baseline for the “what if” analysis in this case study.



Observations

The chart in the preceding figure shows that the stretch factor for the *user workload* is approximately 6.56, which is about six times the acceptable level. Further analysis verified that, primarily, the following I/O devices contribute to the queue delay:

Controller 4: 4DDEd1, 03F84d1

Controller 13: 36E5d0, 54A8d0, 5762d1

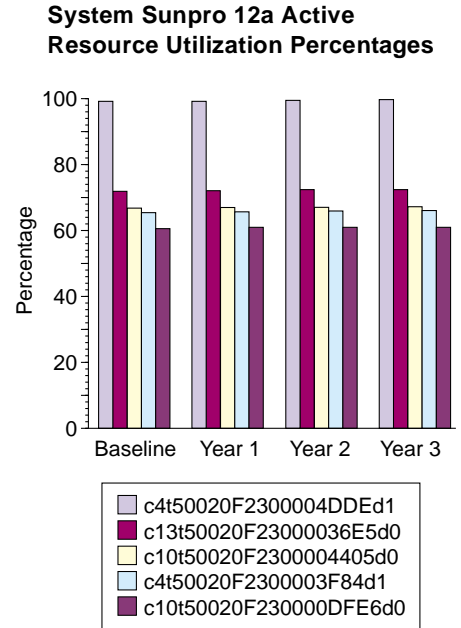
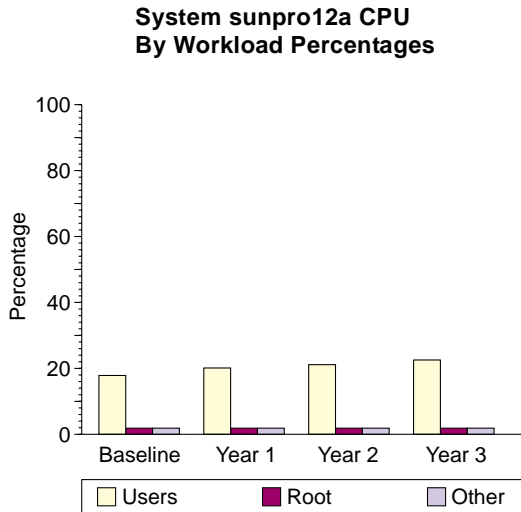
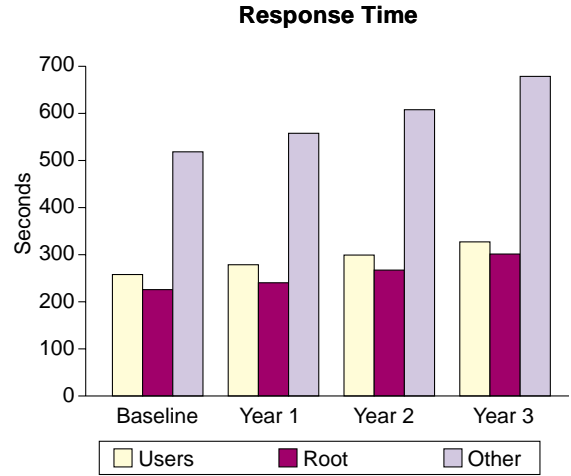
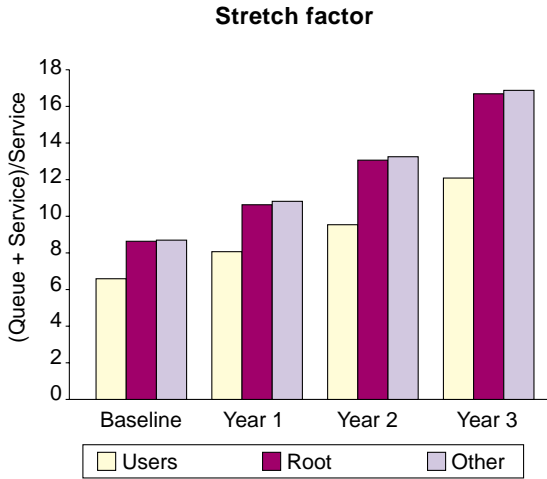
Controller 10: 4405d0, DFE6d0

Controller 7: 3E36d1, 5A19d1, AA3d0

CPU utilization in the user mode is low, under 20 percent, reflecting the stretch factor described previously.

Exercising the Model

The following charts project a 20 percent growth for three periods.



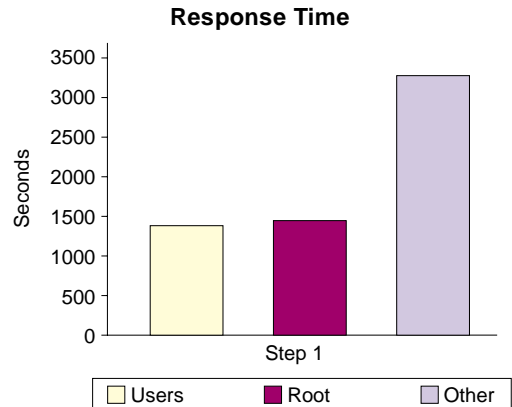
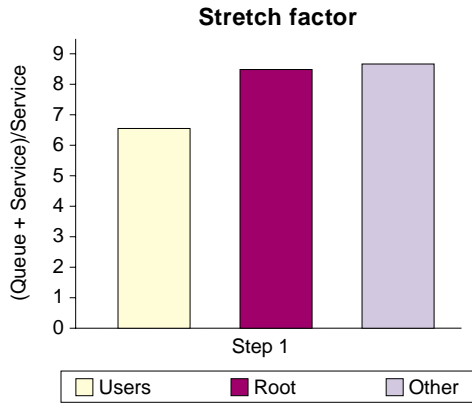
Observations

Simulating 20 percent compound increases in user workload for three periods increased the stretch factor and corresponding response time progressively for the next three periods.

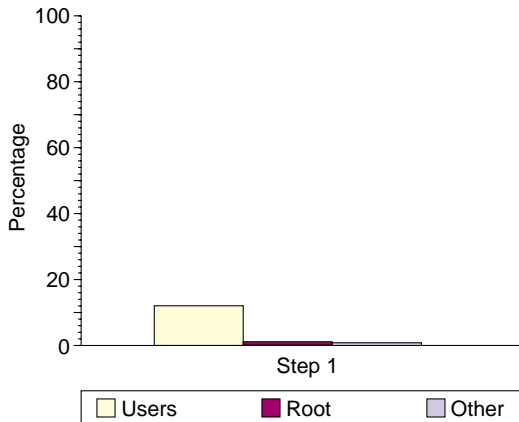
	Baseline (BL)	BL + 20%	BL + 44%	BL + 72.8%
Stretch factor	6.56	8.01	9.96	12.34
Response time	1382	1697 (22%)	2097 (52%)	2598 (88%)

Adding CPUs to the Current Configuration

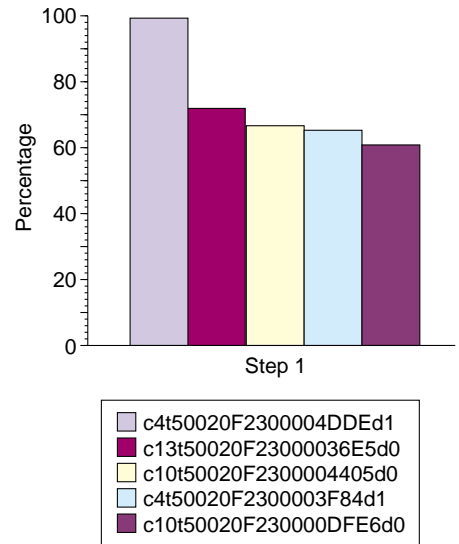
The following charts show the results of adding CPUs to the current configuration.



System sunpro12a CPU Utilization By Workload Percentages



System sunpro12a Active Resource Utilization Percentages



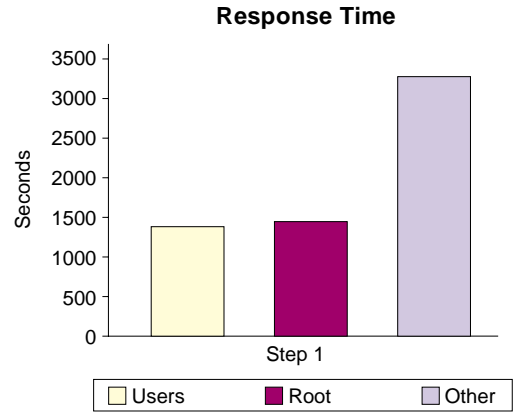
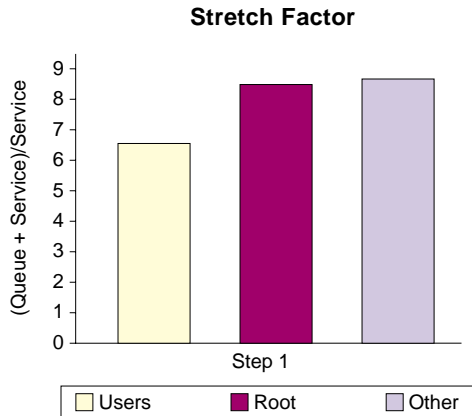
Observations

Simulating the addition of four CPUs to the current environment showed that adding CPUs has no significant impact on the response time.

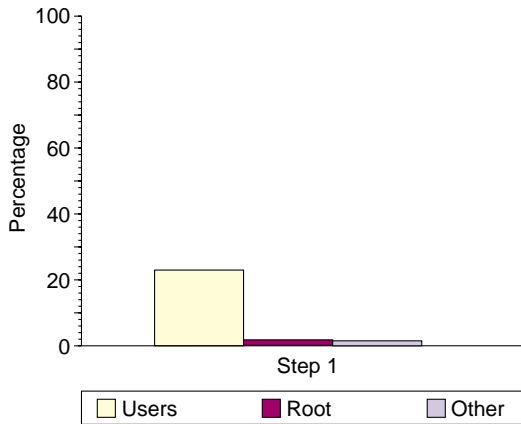
	Baseline	Adding four CPUs
Stretch Factor	6.56	6.51
Response Time	1382	1383

Removing CPUs From the Current Configuration

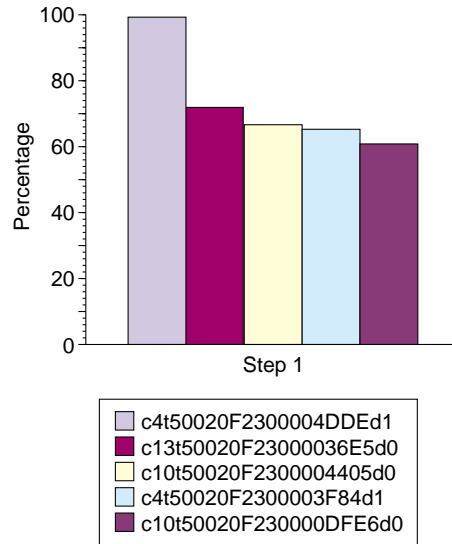
The following charts show the results of removing CPUs from the current configuration.



System sunpro12a CPU Utilization By Workload Percentages



System sunpro12a Active Resource Utilization Percentages



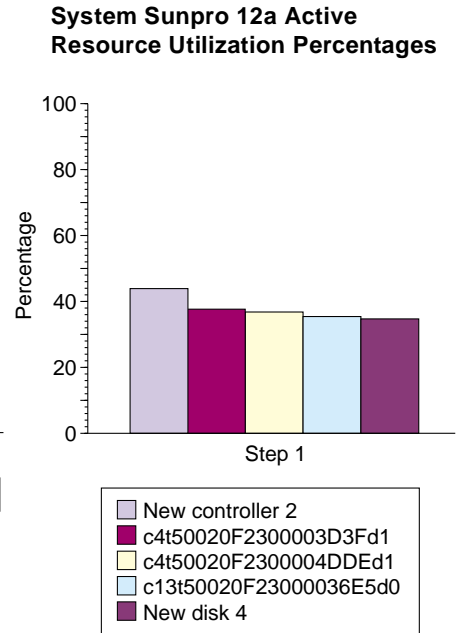
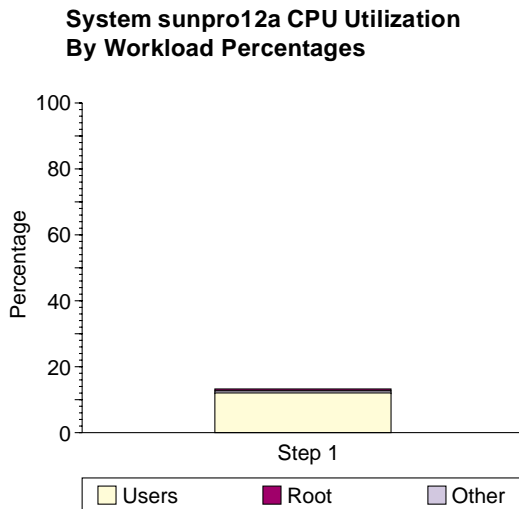
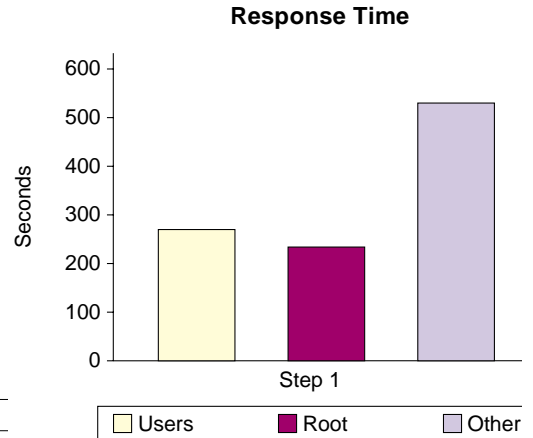
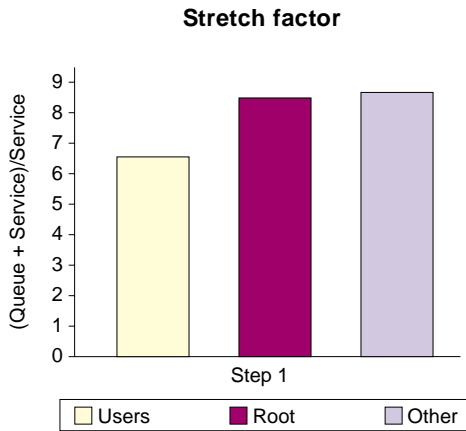
Observations

Simulating the removal of four CPUs from the current environment showed that removing CPUs has no significant impact on the stretch factor or response time.

	BaseLine	Removing four CPUs
Stretch Factor	6.56	6.59
Response Time	1382	1382

Balancing the I/O

The following charts show the result of balancing the I/O.



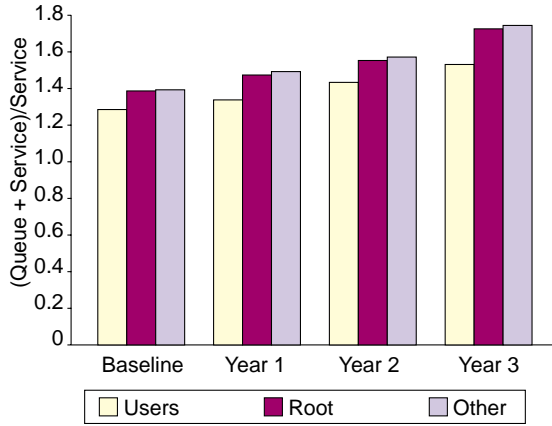
Observations

Balancing the I/O among the existing devices reduced the stretch factor and response time by one half. Still, the stretch factor is about three times the acceptable level. The next test evaluated the behavior of the model after adding controllers and disks drives and further distributing the I/O. First, one additional Fibre Channel controller with seven 73-gigabyte 10,000-RPM disk drives was configured and the I/O was balanced among the new controller and drives. The stretch factor and response time went down further, but was still twice the acceptable level. Finally, a second additional controller with the same specifications as the first one was configured. This configuration put the stretch factor and response time within the acceptable level.

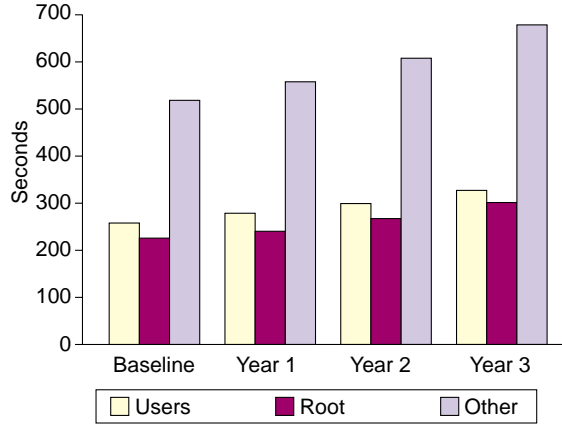
Projecting 20 Percent Growth for Three Periods After Balancing the I/O

The following charts show the result of projecting 20 percent growth after balancing the I/O.

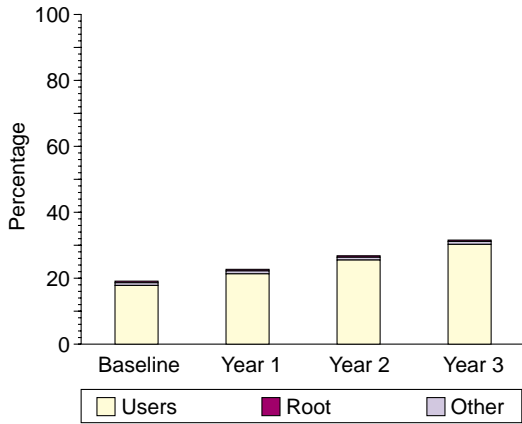
Stretch factor



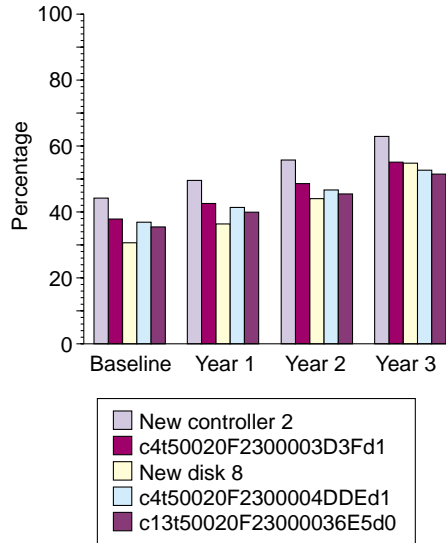
Response Time



System sunpro12a CPU Utilization By Workload Percentages



System Sunpro 12a Active Resource Utilization Percentages



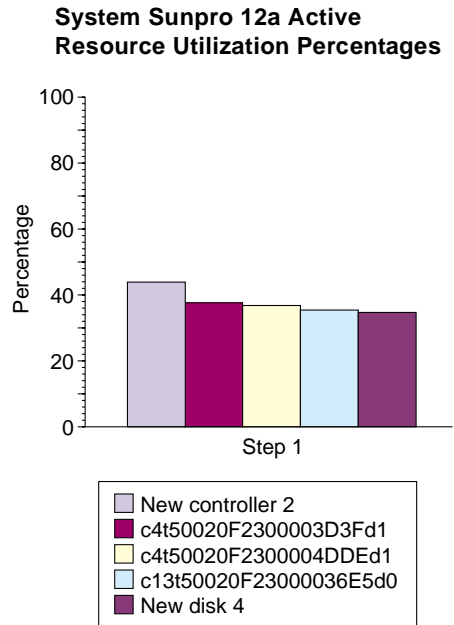
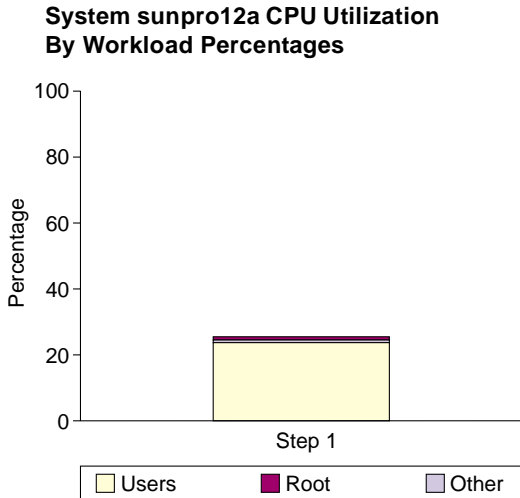
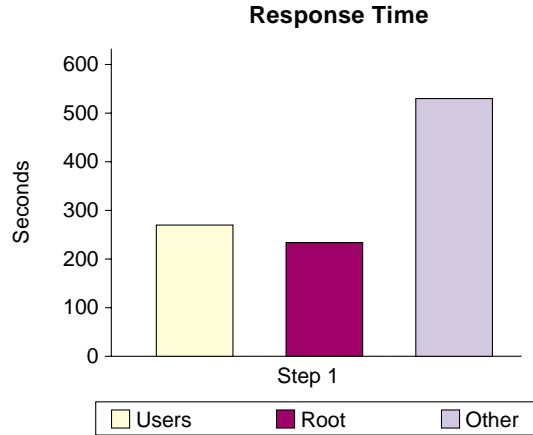
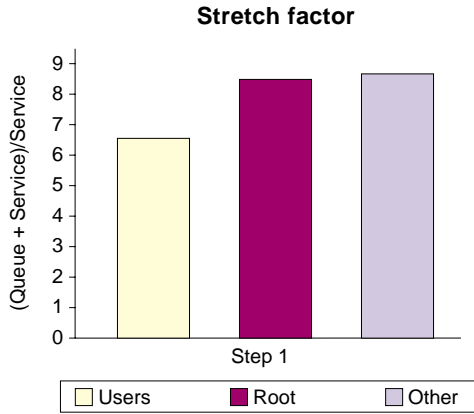
Observations

Simulating 20 percent compound increases in user workload for three periods increased the stretch factor and corresponding response time progressively for the next three periods. The response time figures were within the acceptable level.

	Baseline (BL)	BL + 20%	BL + 44%	BL + 72.8%
Stretch factor	1.28	1.34	1.43	1.55
Response time	273	286 (5%)	303 (11%)	330 (21%)

Adding CPUs After Balancing the I/O

The following charts show the results of adding CPUs after balancing the I/O.



Observations

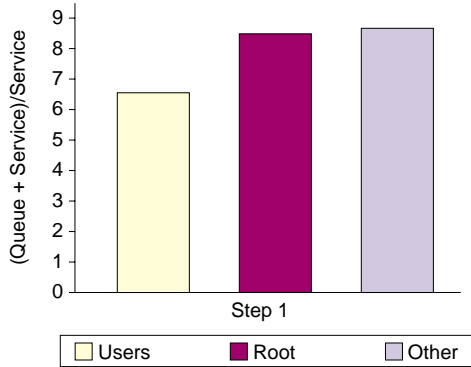
Adding CPUs after balancing the I/O has absolutely has no impact on the stretch factor and response time. This confirms the accuracy of the model, reflecting correctly a situation where a CPU is added to balanced system.

	Baseline	Adding four CPUs
Stretch factor	1.28	1.28
Response Time	273	275

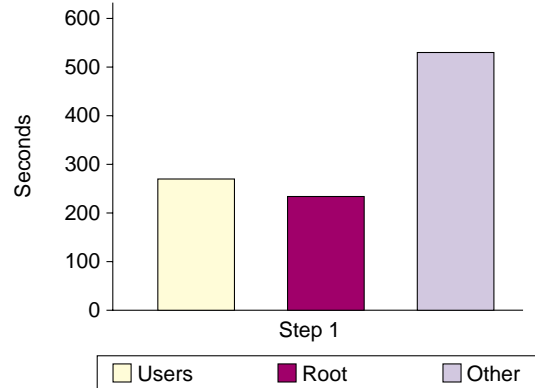
Removing CPUs After Balancing the I/O

The following charts show the results of removing CPUs from the current configuration after balancing the I/O.

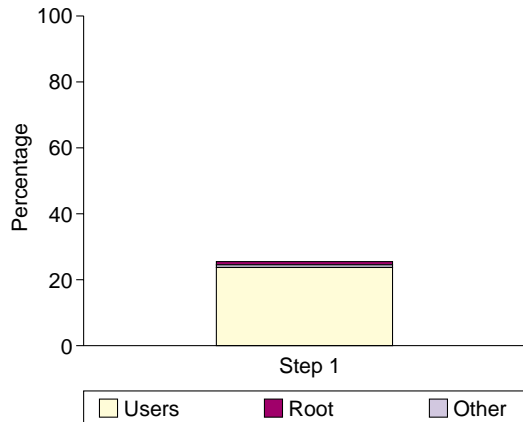
Stretch factor



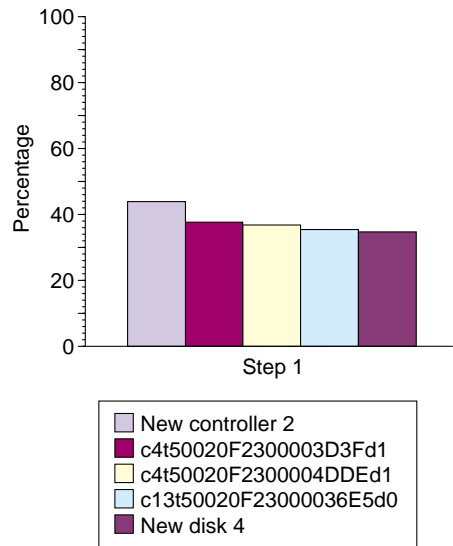
Response Time



System sunpro12a CPU Utilization By Workload Percentages



System Sunpro 12a Active Resource Utilization Percentages



Removing CPUs after balancing the I/O has no impact on the stretch factor or response time. The user mode utilization increases substantially, suggesting the approach of the lower limit for the number of CPUs for the current load.

	Baseline	Removing four CPUs
Stretch Factor	1.28	1.29
Response Time	273	272

Conclusions

TeamQuest reports 1300 IOPS on the system during the peak period. Tests with VERITAS VxBench indicate a total capacity of the I/O subsystem to sustain more than 11,000 IOPS. Based on this information, the underlying architecture is neither the limiting nor the contributing factor for high I/O CPU utilization and, therefore, about 12 percent of the total I/O subsystem capacity is utilized during the peak period. This corresponds with the observation that a few controllers have a very high utilization rate during the peak period.

The capacity planning model indicates that the database server is running above the acceptable limits of resources contention. The infrastructure could deliver, according to the model, about four times better response time if the load on the I/O subsystem can be alleviated and/or distributed over more I/O controllers and devices.

The following table shows that increasing the workload with the current infrastructure and application environment further degrades the response time.

	Baseline (BL)	BL + 20%	BL + 44%	BL + 72.8%
Stretch factor	6.56	8.01	9.96	12.34
Response time	1382	1697 (22%)	2097 (52%)	2598 (88%)

In fact, response time degradation is directly proportional to the workload and grows faster as the workload increases. Several simulations were performed on the capacity planning model to balance the I/O subsystem. Balancing the I/O among the *existing* devices reduces the stretch factor and response time by one half, which is considered a very good achievement.

To further reduce the stretch factor, the behavior of the model after adding new controllers with disks drives was evaluated. First, one additional Fibre Channel controller with seven 73-gigabyte 10,000-RPM disk drives was configured and the I/O was balanced among the new controller and drives. The stretch factor and response time went down further, but was still twice the acceptable level. Finally, a second additional controller with the same specifications as the first one was configured. This configuration put the stretch factor and response time within the acceptable level.

The following table shows, after balancing the I/O, the system stretch factor drops below the recommended thresholds and the response time grows moderately with workload increases.

	Baseline (BL)	BL + 20%	BL + 44%	BL + 72.8%
Stretch factor	1.28	1.34	1.43	1.55
Response time	273	286 (5%)	303 (11%)	330 (21%)

In summary, although substantial performance improvement can be achieved by balancing the I/O among existing devices, complete optimization based on stretch factor analysis was only achieved by adding and then balancing the I/O on devices. Compared to the response times observed, after balancing the I/O and optimizing stretch time, the infrastructure could support the 20 percent workload increase for the next three growth periods.

According to the model projections, adding or removing some CPUs would have minimal impact on performance.

Adding other resources to the environment was simulated but, according to the model, application performance improvement could only be achieved by first addressing the I/O resource contentions.

One final conclusion—after implementing the simulated results and suggestions in this report, the customer was able to increase IOPS from 1300 to 4800 and enhance the response time and throughput considerably.

Recommendations

The following paragraphs provide recommendations for changes in:

- I/O
- CPU
- Oracle

I/O

Changes in the I/O infrastructure, such as a technology change, would not map to a substantial improvement on system performance. Most performance gain is expected to be obtained by fine tuning the application layer and improving the efficiency of its usage of the I/O subsystem.

- Reduce the “cost of I/O” by optimizing the application layer. Sometimes by just reducing the amount of I/O, each I/O can become cheaper because less I/O means less contention on I/O resources. This could only be achieved by an in-depth analysis of the I/O demand by the application and database design. This path would lead to further data path multithreading, improving efficiency on the I/O infrastructure. Fine tuning of parallel queries and Oracle partitioning, with its associated mapping to the I/O infrastructure, for example, can improve performance substantially. See more details on the Oracle-related recommendations in the “Oracle” paragraph.
- Reduce the “cost of I/O” by optimizing the infrastructure layer. Some performance improvement can be obtained by increasing the number of spindles and/or controllers for the “hot spot” devices. This is associated with the distribution of database hot spots (tablespaces, datafiles and indexes) over multiple disk arrays. Further, volume-level stripping can also be considered an alternative to improve I/O response time.

CPU

- Alleviate the CPU share spent in I/O wait mode in order to improve overall system performance. Note, however, that adding CPU power to such an out-of-balance system, without changing its I/O utilization pattern, may do nothing to improve performance. In fact, in some cases it makes the situation worse, because more CPU power will drive more I/O requests. Since the disks cannot handle the present load, the queues just get longer, requiring more CPU time for I/O queue management and on I/O wait.

- Improve the effectiveness of I/O subsystem utilization first and then verify the impact on CPU utilization. Moderate CPU expansion may reduce CPU wait in user mode, potentially improving performance by that factor (6 to 8 percent)—see associated risk reported on the item previously. Simulations on the effect of increasing CPU power are presented in the “Resolving CPU and I/O Bottlenecks Through Modeling and Capacity Planning” section of this article and indicate its convenience.

Oracle

- Oracle parallel queries and table partitioning have a major impact on specific queries response time. Many instances of the same query, however, may be touching a specific I/O path or media at the same time. It would better to explore the possibility of duplicating “hot spots” from the application perspective, such as indexes, files, or partitions, and implement some kind of load balancing at the application or Oracle level. This can greatly improve I/O performance by improving I/O multithreading efficiency and manageability. This approach may be particularly feasible on a read-only database.
- Oracle server processes read from database files when a requested data item cannot be found within the Oracle buffer cache. Waiting for data from disk is one of the most common reasons for delays experienced by server processes, so any reduction in these delays helps to improve performance. You can optimize data file I/O by trying to improve the hit rate in the buffer cache and striping data files across a sufficient number of disks. The alternative is to reduce the number of physical reads per executions at queries and the Oracle partitioning level.
- Review system parameters for large Oracle databases. Changing some system parameters can have a positive impact on performance. For example, the `/etc/system` parameter, `shmmx`, can be changed to `0xffffffff`.

I/O Infrastructure Performance Improvement Methodology

This section describes a methodology for performance improvement when there is a serious I/O bottleneck. Effective utilization of the I/O infrastructure capacity is the key to environment performance improvement. In the “Resolving CPU and I/O Bottlenecks Through Modeling and Capacity Planning” section, a capacity planning model was used to verify the positive impact on performance of a balanced I/O load distribution. In the simulation, new controllers and disks were added to the model

and I/O load distribution was forced across the new devices. The simulations confirmed the benefits associated with a better I/O distribution (though not to be attributed to the addition of the new hardware).

Large database servers are a dynamic, evolving environment. You must be aware that most of the tuning efforts for optimization of I/O performance may become obsolete when the application I/O infrastructure utilization pattern changes. Therefore, performance tuning of such environments should be considered a continuous process rather than a specific one-time, effort and action plan. The suggested action plan is presented as a **process** for I/O optimization, to be used every time there is a change in the application environment that impacts its I/O utilization pattern.

Infrastructure Optimization Plan

TABLE 1 contains the details of the infrastructure optimization process..

TABLE 1 Infrastructure Optimization Process Details

Action	Expected Result	Mitigation Effort	Effort
1. Evaluate and implement all possible database server optimization for improved I/O distribution.	See Oracle and Application Optimization Suggestions	See Oracle and Application Optimization Suggestions	See Oracle and Application Optimization Suggestions
2. Identify the high utilization devices (hot spots) on the I/O infrastructure. Using standard Solaris OE tools or TeamQuest Viewer, identify those devices, (LUNs) with a utilization greater than 80%.	Medium to high impact. Five to 15% performance improvement can be expected in association with each device for which excessive utilization can be resolved. There should be four-to-six devices currently eligible for optimization in the environment.	Low risk. Application support and database administration staff is very familiar with operations involving relocation of logical volumes relocation.	Low for evaluation and medium for implementation. Two-to-four hours for device identification and mapping of database structures. Implementation time depends on ability to execute operation in a test and build environment or during maintenance windows on the production environment.
3. Identify the logical volumes associated with the high-utilization devices.			
4. Identify the database structures associated with those logical volumes.			

TABLE 1 Infrastructure Optimization Process Details (Continued)

Action	Expected Result	Mitigation Effort	Effort
<p>5. Verify that the contents of each high-utilization logical volume can be distributed across less utilized and/or spare logical volumes. Execute I/O distribution and verify new utilization numbers for the affected device*.</p>	<p>Depends on the number of high utilization devices that can be optimized through relocation of the database files.</p>		
<p>6. If utilization rates for all devices are under 80%, document new volume configuration to be reproduced in the next database build and re-run[†] the TeamQuest capacity planning model, observing the model stretch factor and load growth projections for the new configuration.</p>			
<p>7. If devices utilization is still high, verify if those devices are sharing the same controller with other high utilization devices.</p>	<p>Low impact.</p> <p>Zero to 10% performance improvement, depending on impact of controller distribution on the I/O devices queue.</p>	<p>Low risk.</p> <p>Same as the above (execution involves the same identification and logical volume relocation process).</p>	<p>Low effort for evaluation and medium for implementation.</p>

TABLE 1 Infrastructure Optimization Process Details (Continued)

Action	Expected Result	Mitigation Effort	Effort
<p>8. Distribute the high utilization devices evenly across the available controllers by relocating logical volumes contents, for example.</p>	<p>Controllers are not perceived today as bottlenecks on I/O performance. Note, however, that this step is recommended because the controllers <i>may</i> become a limiting resource as data starts moving faster, due to improved I/O distribution.</p>		<p>Same as above (execution involves the same identification and logical volume relocation process).</p>
<p>9. If utilization rates for all devices are under 80%, document new volume contents mapping to controllers, to be reproduced in the next database build. Re-run the TeamQuest capacity planning model, observing model the stretch factor and load growth projections for the new configuration.</p>			
<p>10. If neither logical volume contents can be relocated for better I/O distribution or the controller where the high-utilization device is located is overloaded, consider stripping the logical volume across more than one LUN.</p>	<p>Medium impact.</p> <p>For each stripped logical volume replacing a current highly utilized volume/device, 5 to 15% performance improvement can be expected.</p>	<p>Medium risk.</p> <p>Logical volumes and file systems will be removed and the re-created logical volumes striped.</p> <p>Operation can possibly be executed on the building environment and then on a database about to promoted to production to minimize impact on the production environment.</p>	<p>Medium effort.</p> <p>About 4-to-8 hours per logical volume for research on stripping configuration and deployment.</p>

TABLE 1 Infrastructure Optimization Process Details *(Continued)*

Action	Expected Result	Mitigation Effort	Effort
<p>11. Evaluate Operations impact on volume stripping. Operations permitting, create “high performance” volumes by stripping a logical volume across two LUNs. Two non-stripped logical volumes can be converted in two stripped logical volumes on the respective two LUNs. Addition of new devices to the environment may facilitate execution and make it viable.</p>			
<p>12. Relocate the identified high utilization devices/database structures to the newly created stripped volumes. Evaluate utilization of new devices.</p>			
<p>13. If utilization rates for all devices are under 80%, document new volume organization to be reproduced on the next database build. Re-run the TeamQuest capacity planning model, observing the model stretch factor and load growth projections for the new configuration.</p>			

TABLE 1 Infrastructure Optimization Process Details (Continued)

Action	Expected Result	Mitigation Effort	Effort
14. Depending on the impact of the stripping on performance, consider further logical volume stripping over a higher number of LUNs.			
15. If a given controller remains with more than three high-utilization devices after the operations listed above, consider adding a new HBA to the I/O infrastructure for better I/O distribution.	<p>Low impact.</p> <p>Zero to 10% performance improvement, depending on impact of controller distribution on the I/O devices queue.</p>	<p>Low risk.</p> <p>Support staff is very familiar with HBA-related configurations.</p>	<p>Medium effort.</p> <p>About 4 hours.</p> <p>Hardware upgrade on the production to be executed during a maintenance window.</p>
16. Verify devices utilization after the new controller is added. If improvement is verified, re-run the TeamQuest capacity planning model, observing the model stretch factor and load growth projections for the new configuration.			

*If there are more possible I/O distributions than there are under utilized devices, add new devices/LUNs to the environment and further implement application level I/O distribution.

†Collect system utilization information, re-calibrate the capacity planning model and re-run load growth simulations for the new environment.

Oracle and Application Optimization Suggestions

Oracle-related optimization is beyond the scope of this analysis. However, TABLE 2 lists some ideas for performance enhancement, based on the I/O distribution improvement principle.

TABLE 2 performance enhancement Ideas

<p>1. Implement the planned Oracle database segmentation changes for improved I/O distribution.</p>	<p>High performance improvement.</p>	<p>Low risk</p>	<p>Low risk.</p>
	<p>A well balanced I/O infrastructure utilization has the potential to improve performance two to three times according to simulations on the capacity planning model.</p>		
<p>2. Explore the possibility of implementing load balance to the I/O infrastructure at the application level. If there are a few queries that are hot spots from the database standpoint, use a specific index file, for example. Potentially that index file can be duplicated on a read-only database and I/O load balancing implemented at the application/query level. Viability to be determined by the application development and database management teams.</p>	<p>High performance improvement.</p>	<p>Medium risk, due to application code change.</p>	<p>High, due to application code change.</p>
	<p>Due to the same reasons listed above. Even higher potential for performance improvement, actually now under application direct control.</p>		
<p>3. Review application architecture</p>	<p>High performance improvement</p>	<p>Low</p>	<p>High, due to application re-architecting</p>

Data Tables

TABLE 3, TABLE 4, and TABLE 5 list the data collected in this case study.

TABLE 3 Disk and CPU Utilization Data

rps	ssd1183		ssd1187			ssd1197			ssd311			cpu			
	wps	util	rps	wps	util	rps	wps	util	rps	wps	util	us	sy	wt	ld
299	0	100.0	161	0	85.0	133	14	79.1	137	0	89.0	11	13	75	1
263	0	100.0	139	1	88.4	140	25	81.1	140	0	87.7	9	11	79	1
278	1	99.0	153	0	87.8	116	27	76.1	136	1	88.1	9	8	82	0
271	0	98.9	169	1	90.8	154	31	83.0	149	0	90.6	9	8	83	0
272	0	100.0	151	0	87.6	127	49	77.0	148	0	84.6	8	7	85	0

TABLE 4 Disk Busy Percentage and Average Service Time Data

device	%busy	avque	r+w/s	blks/s	await	avserv
ssd92,e	69	1.2	139	4457	0.0	8.6
ssd129	94	2.8	201	6434	0.0	14.0
ssd129,e	94	2.8	201	6434	0.0	14.0
ssd156	81	1.7	147	3703	0.0	11.6
ssd156,e	81	1.7	147	3703	0.0	11.6
ssd166	46	0.9	92	2950	0.0	9.2
ssd166,e	46	0.9	92	2950	0.0	9.2
ssd167	94	4.0	217	6936	0.0	18.5
ssd167,e	94	4.0	217	6936	0.0	18.5
ssd178,e	67	1.2	125	3986	0.0	9.4
ssd183	100	15.4	263	14829	0.0	58.4
ssd183,e	100	15.4	263	14829	0.0	58.4
ssd184	23	0.3	25	785	0.0	11.6
ssd187	91	2.8	138	4425	0.0	19.9

TABLE 4 Disk Busy Percentage and Average Service Time Data (*Continued*)

device	%busy	avque	r+w/s	blks/s	await	avserv
ssd187,e	91	2.8	138	4425	0.0	19.9
ssd189	31	0.4	27	879	0.0	14.1
ssd189,e	31	0.4	27	879	0.0	14.1
ssd191	92	3.8	191	6120	0.0	19.8
ssd191,e	92	3.8	191	6120	0.0	19.8

where:

us is the percentage of time the CPU spends running the user application.

sy is the percentage of time the CPU spends controlling the system.

wt is the percentage of time the CPU spends waiting for the disks to complete I/O.

Id is the percentage of time the CPU is idle. Subtracting this number from 100 will give the percent utilization.

%busy is how busy the disk drives are.

avque is the average number of I/Os that are waiting in queue to be serviced.

r+w/s is the total number of reads and writes (I/Os) done by the disks in the time interval chosen.

blks/s is the number of blocks per second transferred; the block length is usually 512 bytes or a multiple of 512 bytes.

TABLE 5 TNF Output Data

	i/o	read	maximum	average	std_dev	maximum	average	std_dev	xfer	maxq	avg random	busy	
	rate	pct	iint_arr	iint_arr	iint_arr	resp_tme	resp_tme	resp_tme	size	depth	qdepth	%	%
total	3291.5	97.9	5.40	0.30	0.39	250.77	22.11	22.50	22152	124	72.80	75.56	100.00
c12t50020F2300006A96d0s4	170.9	99.9	202.58	5.83	13.02	187.35	34.57	28.79	38816	28	5.91	23.62	83.25
c12t50020F2300006AFDd0s4	168.6	83.3	58.82	5.92	6.56	42.07	10.88	5.27	13071	22	1.83	85.50	77.40
c12t50020F2300006BFCd0s4	176.1	99.9	125.78	5.67	8.14	129.38	21.84	14.67	28399	27	3.84	57.58	86.27
c12t50020F23000079F7d0s4	179.8	99.9	63.79	5.55	6.41	66.91	19.33	7.74	16539	21	3.47	99.72	91.77
c12t50020F23000084C9d0s4	24.3	99.4	428.91	41.05	52.55	39.96	14.12	4.44	16336	4	0.34	99.71	28.65
c12t50020F23000084D9d0s4	151.5	99.9	86.82	6.59	7.28	65.63	20.07	7.53	16421	17	3.04	99.60	90.41
c13t50020F2300003503d1s4	29.9	99.3	353.22	33.33	39.00	31.19	9.71	4.00	16333	5	0.29	99.34	24.50
c13t50020F2300003E49d0s4	270.2	99.9	79.76	3.69	6.46	250.77	58.03	39.62	28656	42	15.68	40.48	99.27
c13t50020F23000059EEd0s4	141.6	99.9	84.13	7.05	7.83	83.78	19.15	7.02	16454	15	2.71	99.44	88.62
c13t50020F2300006C3Bd1s4	84.4	99.7	101.14	11.82	12.28	46.46	9.70	3.76	16607	11	0.81	99.19	52.11

TABLE 5 TNF Output Data (Continued)

	i/o	read	maximum	average	std_dev	maximum	average	std_dev	xfer	maxq	avg random	busy	
	rate	pct	iint_arr	iint_arr	iint_arr	resp_tme	resp_tme	resp_tme	size	depth	qdepth	%	%
c14i50020F230000BF7Ed0s4	17.9	95.0	528.77	55.24	66.18	49.55	10.58	5.07	16121	5	0.19	99.13	16.72
c14i50020F230000C18Dd0s4	19.6	99.5	534.99	50.81	59.42	42.35	11.93	4.11	16348	4	0.23	99.85	20.64
c14i50020F230000DDD0d0s4	67.1	99.7	187.87	14.86	16.31	39.74	9.22	3.40	16607	10	0.61	99.20	42.32
c14i50020F230000E17Cd0s4	13.9	99.0	1155.20	71.57	93.39	36.89	9.99	4.45	16475	4	0.13	98.98	12.56
c15i50020F2300004442d0s4	22.6	99.6	448.67	43.96	52.30	38.18	13.47	4.41	16334	4	0.30	99.93	25.89
c15i50020F23000097ACd0s4	199.3	99.9	54.94	5.01	5.94	66.57	17.57	7.73	16412	23	3.50	99.39	90.17
c15i50020F23000097ACd1s4	87.8	99.9	106.51	11.37	12.43	39.06	8.52	3.21	16376	11	0.74	99.71	49.53
c16i50020F230000528Cd0s4	3.8	97.0	1876.53	260.92	392.30	35.89	5.67	5.30	16748	2	0.02	97.77	2.13
c16i50020F23000069B0d0s4	55.7	99.5	240.65	17.93	23.01	32.98	12.72	4.00	16322	7	0.70	99.59	49.21
c16i50020F230000BDF5d0s4	30.2	99.5	346.84	33.04	40.45	53.23	12.27	4.65	16345	4	0.37	99.76	30.00
c17i50020F2300005720d1s4	174.3	82.6	82.12	5.73	6.32	42.94	10.69	5.22	12997	32	1.86	84.44	78.04
c17i50020F230000629Bd1s4	5.8	43.9	3817.82	169.20	366.20	35.45	8.31	5.29	34241	11	0.04	56.79	3.40
c17i50020F2300006ADE1s4	221.9	99.9	79.71	4.50	6.76	158.83	28.91	20.97	28627	36	6.41	51.05	93.78
c17i50020F2300006C1Bd1s4	210.6	99.9	190.50	4.73	10.56	243.25	36.53	30.79	38451	34	7.69	22.94	89.09
c17i50020F2300007C4Ad1s4	14.9	98.9	759.66	66.88	92.67	43.80	12.66	5.19	16433	4	0.18	97.54	16.51
c17i50020F23000086B7d1s4	26.4	99.6	483.71	37.76	46.38	45.23	14.97	4.37	16340	5	0.39	99.84	32.35
c18i50020F2300004359d1s4	74.5	99.8	130.16	13.40	13.99	40.73	11.42	4.23	16491	12	0.85	98.99	55.09
c19i50020F230000A6B7d1s4	4.0	97.5	2534.77	245.34	390.26	31.79	5.91	5.53	16185	2	0.02	97.92	2.39
c19i50020F230000B56Cd1s4	88.0	99.7	121.12	11.34	11.76	58.36	15.74	6.51	16379	9	1.38	99.47	71.46
c19i50020F230000C01Cd1s4	18.9	98.8	677.92	52.63	66.50	37.31	11.54	4.09	16286	3	0.21	99.62	19.23
c7i50020F2300003BDEd1s4	123.5	99.8	805.88	8.07	23.61	195.14	31.30	25.04	42903	24	3.86	17.35	66.57
c7i50020F230000514Fd1s4	30.9	99.4	815.84	32.26	45.45	47.66	15.33	4.27	16339	4	0.47	99.63	37.45
c7i50020F230000647Dd0s4	189.4	99.9	51.30	5.27	5.65	54.09	14.25	5.71	16489	17	2.70	99.19	89.07
c8i50020F230000A6BED1s4	2.0	96.5	4325.08	481.07	745.10	40.51	8.45	4.55	16099	2	0.01	99.30	1.70
c8i50020F230000BCB0d1s4	110.5	99.8	71.93	9.04	9.28	35.24	9.57	3.76	16603	13	1.05	99.18	61.34
c8i50020F230000C0F4d1s4	16.9	99.1	477.62	58.80	70.83	47.08	11.78	4.08	16316	5	0.20	99.58	17.75
c8i50020F230000DF81d1s4	17.1	99.4	781.56	58.19	90.76	34.54	8.60	4.37	16336	3	0.14	99.50	13.74
c8i50020F230000E37Ed1s4	44.1	99.7	197.25	22.61	24.94	38.05	9.67	3.80	16619	10	0.42	99.04	31.20
Controller No.													
c_7	344.0	99.8	49.38	2.90	3.76	195.14	20.47	17.62	25963	34	7.04	75.56	97.81
c_8	190.8	99.7	57.60	5.23	5.59	47.08	9.69	3.93	16552	14	1.85	75.56	81.10
c_12	871.5	96.6	17.90	1.14	1.51	187.35	21.17	17.10	22608	51	18.45	75.56	99.98
c_13	526.3	99.8	30.31	1.89	2.57	250.77	37.06	36.02	22737	46	19.50	75.56	99.96
c_14	118.7	98.9	100.69	8.40	9.41	49.55	9.96	4.07	16472	13	1.18	75.56	66.78
c_15	310.0	99.8	40.19	3.22	3.65	66.57	14.70	7.69	16393	26	4.55	75.56	96.23
c_16	89.8	99.2	149.04	11.11	12.89	53.23	12.25	4.53	16332	8	1.10	75.56	65.09
c_17	654.2	94.7	24.75	1.52	2.16	243.25	25.39	23.99	26898	55	16.61	75.56	99.88
c_18	74.6	99.6	130.16	13.38	13.98	40.73	11.41	4.25	16477	12	0.85	75.56	55.10
c_19	111.1	99.4	121.12	8.98	9.66	58.36	14.65	6.57	16344	9	1.62	75.56	77.42

About the Author

Gamini Bulumulle is a Senior IT Architect with Sun Professional Services. He has been with Sun for six years. Previously he worked with AT&T Bell Labs as a Senior Network Engineer and Oracle Corporation as a Senior Consulting Technical Specialist. He has a Bachelor of Engineering with a Minor in Mathematics and Major in Computer Engineering, a Master of Science in Computer Engineering, and is presently a Ph.D. candidate and working on his dissertation in Computer Engineering.

Marcos Bordin is a PS Technical Manager with Sun Professional Services. He has been with Sun for six years. Before joining Sun Professional Services in 2000 he worked as a presales system engineer for three years in the Sun Sales organization. His focus areas are Data Center Operations, IT Infrastructure, and Project Management.

Ordering Sun Documents

The SunDocsSM program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals through this program.

Accessing Sun Documentation Online

The `docs.sun.com` web site enables you to access Sun technical documentation online. You can browse the `docs.sun.com` archive or search for a specific book title or subject. The URL is `http://docs.sun.com/`.

