

Configuration and Tuning of Oracle7 for Novell UnixWare 2.0 on Compaq Servers

White Paper

**Prepared By
Database Engineering
Compaq Computer Corporation**

March 1995

COMPAQ

NOTICE

The information in this publication is subject to change without notice.

COMPAQ COMPUTER CORPORATION SHALL NOT BE LIABLE FOR TECHNICAL OR EDITORIAL ERRORS OR OMISSIONS CONTAINED HEREIN, NOR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL.

This publication contains information protected by copyright. Except for internal use distribution, no part of this publication may be photocopied or reproduced in any form without prior written consent from Compaq Computer Corporation.

This publication does not constitute an endorsement of the product or products that were tested. The configuration or configurations tested or described may or may not be the only available solution. This test is not a determination of product quality or correctness, nor does it ensure compliance with any federal, state or local requirements. Compaq does not warrant products other than its own strictly as stated in COMPAQ product warranties.

Product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

© 1994 Compaq Computer Corporation

All rights reserved. Printed in the USA

COMPAQ, SYSTEMPRO, PROLIANT, PROSIGNIA

Registered US Patent and Trademark Office.

*Configuration and Tuning of Oracle7 for Novell UnixWare 2.0 on
Compaq Servers*

First Edition (March 1995)

Document Number 404A/0395

Compaq Computer Corporation



Table of Contents

Introduction1

Oracle Overview2

 Architecture2

 New Features in Oracle7 Version 7.13

Tuning Goals4

Basic tuning5

 Installation Issues5

 Asynchronous I/O5

 Post-Wait5

UNIX Scheduling6

I/O Tuning6

 Separate Sequential and Random I/O's6

 Layout of Tables and Files6

 Don't exceed I/O limits7

CPU Scalability7

Memory Tuning8

 Oracle DB Block Buffers8

 SGA Size8

PSE Pages10

User Capacity11

Network Tuning11

 SQL*Net11

 TCP/IP11

Management Considerations11

 Compaq Insight Manager (CIM)11

 Oracle Server Manager11

 DB General12

Conclusion12

References12

Configuration and Tuning of Oracle7 for Novell UnixWare 2.0 on Compaq Servers

Introduction

The purpose of this document is to share the knowledge acquired by Compaq Systems Engineers in the area of configuration and performance tuning of Oracle7 on UnixWare 2.0 on the Compaq ProLiant family of Servers. It is our desire to deliver the best technical information possible on a specific topic in a timely manner and in a highly useable format. Any comments, suggestions and feedback are always appreciated.

The information presented in this document is based on both Oracle7 version 7.0 and version 7.1 for UnixWare 2.0. Most information provided is valid for both versions. Information that is version specific will be noted as such. Since there is an abundance of information available concerning general tuning of Oracle7 on the UNIX platform, much of the discussions will be focused on Compaq specific tuning. Wherever possible, references will be made to other useful tuning documentation.

Other publications covering these and related topics listed below:

“**Configuring Compaq RAID Technology for Database Servers**”, Document Number 267A/0294

“**Installation and Configuration Guide for Oracle7 on UnixWare**”, Oracle Part Number

“**ORACLE7 Server Administrator’s Guide**”, Oracle Part Number 6694-70-1292

“**Oracle7 Server Application Developer’s Guide**”, Oracle Part Number 6695-70-1292

Oracle Overview

Architecture

An Oracle instance consists of the System Global Area (SGA), the Oracle background processes, the control and configuration files, the database files, and the redo log files. The SGA is a group of shared memory structures that contains the following:

Database Buffers	Contains most recently used database blocks
Shared Pool	Contains shared SQL areas and data dictionary cache.
Redo Log Buffers	Logs changes made to the database.
Cursors	Handles for memory associated with specific statements.

The background processes consist of the following:

Database Writer	DBWR	Writes database blocks to disk. There may be more than one.
Log Writer	LGWR	Writes the redo log entries to disk.
Process Monitor	PMON	Responsible for the management of Oracle Processes.
System Monitor	SMON	Performs instance recovery and cleanup.
Recovery Process	RECO	Resolves failures with the distributed option.
Archival Process	ARCH	Copies full on-line redo log files to the archive device (if in ARCHIVELOG mode).
Dedicated Checkpoint Process	CKPT	Signals DBWR to perform updates on all data and control files of the database. If not present LGWR does this.

When a shadow process wants to read from the database, it checks to see if the data exists in the SGA, if it does it reads it from the memory, else it goes directly to the datafiles and reads the data into the SGA. When a process wants to write to the database, the shadow process writes into the SGA only, and at a later time DBWR writes this "dirty" data out to disk. The redo log is used to keep a history of all transactions in order for the database to perform an instance recovery. A minimum of two redo logs is required and more may be used. When a redo log fills, a log switch occurs. At log switch time all new redo information goes to the next redo log file in line. If the system is running in ARCHIVELOG mode, which is recommended, the previous log file is copied out to an archive log file. With a recent backup, the redo log files, and the redo log archive files, the database can be recovered if needed. The control and configuration files are used to store information of the state layout of the database as well as system tunables. A more in-depth discussion of the Oracle7 architecture can be found in the *Oracle7 Server Concepts Manual*.

New Features in Oracle7 Version 7.1

Although Oracle7 version 7.1 is primarily a maintenance release, there are several new features, briefly outlined below. For more details on these new features, refer to the *Oracle7 Server Documentation Addendum Release 7.1*.

Server Manager

Server manager is a new administration tool for Oracle7. Server manager has a Graphical User Interface (GUI) using Motif and X-Windows for UnixWare 2.0. Server manager can also be run remotely from UNIX clients running Motif or from DOS clients running Microsoft Windows. Server Manager provides all of the features previously available in SQLDBA plus additional configuration and performance monitoring information.

Procedural Option

Features previously limited to users of PL/SQL, such as stored procedures and triggers, are now available to all users. Also, multiple triggers of the same type are allowed on a single table.

Symmetric Replication

Symmetric replication allows multiple, updatable copies of data at different sites in a distributed environment. Oracle's symmetric replication facility will be available on a different schedule from version 7.1. More information regarding replicated servers will be provided as it becomes available.

Consistent Snapshot Refresh

Consistent Snapshot Refresh feature allows multiple related snapshots to be kept consistent to any point in time.

Enhancements to SQL and PL/SQL

Oracle7 version 7.1 conforms to entry level ANSI SQL92 standard requirements. SQL has been enhanced to allow PL/SQL user defined functions to be called from within SQL statements.

Read Only Tablespaces

Read only tablespaces can now be created. This provides for a reduction in backup and recovery since these tablespaces don't require backup as well as allowing read only media such as CD-ROM and WORM drives to store data.

Parallel Recovery

Parallel Recovery allows multiple processes to perform the recovery process thus decreasing the time it takes to recover the database. On multiprocessor machines the performance improvement should be significant. During a typical recovery, the processors spend most of the time waiting on I/O from the disk. Parallel recovery will allow more threads of operation thus allowing better throughput to the disks as well as allowing more CPUs to be utilized.

Improved Security when Connecting to a Remote Database

Passwords are now encrypted when they are sent over the network.

Parallel Query

Performance of bulk operations such as data loading, queries, and index creation is greatly improved by taking advantage of multiple CPUs. During complex queries where parallelism can be used the time it takes to do each query should be reduced.

SQL*Net Oracle Names

This allows you to do a database link without providing a connect string to the database. SQL*Net Oracle Names uses the Oracle Network Manager to provide connection information to all of the machines on the network.

Tuning Goals

In order to achieve the best performing system possible there are several factors which must be reviewed. These include optimization of the hardware, the Oracle Server, the operating system and the application software. This paper will focus on the hardware, Oracle Server and the OS. It is important to tune the application to take advantage of the system. Due to the diversity of applications they are beyond the scope of this paper.

The tuning process and parameters in this paper should be used as a starting point. Tuning is an iterative process that evolves as user and work loads change on your system.

An optimally tuned Oracle7 on UnixWare 2.0 system should have the following characteristics:

- There will be little or no waiting on I/O. This is verified by running *sar* or *rtpm*. This indicates that the CPUs always have some work to do while there are outstanding I/Os.
- Most of the CPU utilization is in user mode. Again, this is verified by running *sar* or *rtpm* and looking at the percentage of CPU time spent in kernel and in user time. System time can be thought of as operating system overhead such as time spent in the I/O subsystem or in system calls. The higher the percentage of user to system time that you have, the better.

```
sar -u 5 5
```

```
UNIX_SV oracle1 4.2MP 2.0 i386 01/29/95
```

09:00:10	%usr	%sys	%wio	%idle
09:00:15	69	23	6	1
09:00:20	69	24	8	0
09:00:25	68	23	7	2
09:00:30	67	24	8	1
09:00:35	69	24	7	0
Average	68	24	7	1

- Users should see good response times. A system that appears to be tuned well and is experiencing poor response times could have inefficient statements in the application or could have excess latencies in the I/O subsystem or network. Since well tuned database applications are CPU bound an additional system processor could improve response times.

Basic tuning

Installation Issues

In order to install Oracle7 on UnixWare 2.0, there are several required changes to the UNIX system parameters. For example, the typical UnixWare 2.0 installation does not allocate enough shared memory to startup the Oracle database engine which is used during the installation. These system parameters must be changed, the kernel re-linked and the system rebooted before Oracle can be installed.

This step is done for you automatically when you run the `pkgadd` command on the `oraconf` package supplied in your Oracle distribution. This is done as part of the Oracle 7.1 installation process and is documented in the installation guide.

Asynchronous I/O

Asynchronous I/O is available on the UnixWare 2.0 operating system. AIO can be used with databases built using RAW devices or file systems and should be utilized to achieve the best performance. Asynchronous I/O allows the DBWR and/or the LGWR process to avoid blocking on I/O. With AIO enabled in the OS and in Oracle, only one DBWR process is necessary in the system. This cuts down on system idle time that may occur when the DBWR is blocked on I/O.

Oracle's use of asynchronous I/O is activated in Oracle7 version 7.1 as follows:

For DBWR	init.ora parameter USE_ASYNC_IO=TRUE
For LGWR	init.ora parameter LGWR_USE_ASYNC_IO=TRUE

In heavy I/O environments the following OS tunables should be set:

NUMAIO	The maximum number of AIO control blocks.
AIO_LISTIO_MAX	The maximum size of list I/O requests.

Note: Make sure that the device file `/dev/async` is available read-write for the Oracle user. This can be done with the OS command `ls -alst /dev/async`. To change this permanently set the mode bits to 666 in the file `/etc/conf/node.d/async`.

Post-Wait

Post-Wait is available on the UnixWare 2.0 operating system and should be used with Oracle7. Post-Wait eliminates much of the use of semaphores. Post-Wait is enabled by default in the operating system but must be enabled in Oracle. To enable Post-Wait in Oracle, the parameters **USE_POST_WAIT_DRIVER = TRUE**, and **POST_WAIT_DEVICE=/dev/pw** should be included in the `init.ora` file. To verify that Post-Wait is active, use the OS command `ipcs`. If there are semaphores being used by Oracle, then Post-Wait is not being used. Post-Wait has been available in Oracle since version 7.0

UNIX Scheduling

Some additional performance can be achieved by placing Oracle processes into the real-time processing group but this can cause other processes to be CPU starved. Compaq recommends that this not be done in normal operation.

I/O Tuning

In most well tuned Oracle systems, I/O is not a limiting factor. In order to assure that this is not a problem, the following factors need to be verified.

- Sequential I/O's are isolated to their own controller volume.
- Random I/O's are balanced across all drives allocated to data and indexes.
- Physical disk I/O limits are not exceeded.

Separate Sequential and Random I/O's

In order to achieve maximum performance on data files being accessed sequentially, the disk(s) need to be dedicated to this purpose. Of primary importance are the Oracle transaction log files, which are accessed in a sequential, write-only fashion. Other partitions with little I/O activity can share the disk(s) with the redo logs, such as the OS partition and swap (unless your machine is memory limited).

In typical, multi-user database systems file access is random. These files should be spread out over as many physical disks as necessary to achieve random I/O rates that do not exceed recommendations. This is best achieved by using the disk striping available with the Compaq SMART Array controller. Spreading out the disk requests among many disks allows a high degree of parallelism to occur on accesses. Using the Compaq SMART Array controller ensures that the load will be balanced equally across the disks. For more information on optimizing array configurations refer to the Compaq White-Paper *Configuring Compaq RAID Technology for Database Servers*.

Layout of Tables and Files

In order to improve performance where disk I/O is a problem, keep in mind the following.

- Redo Log access is 100% sequential I/O and needs to be isolated if possible. Speed of the redo log is essential to the performance of the system. If possible, these drives should be fault tolerant. Hardware fault tolerance provides the maximum performance and reliability. See the Compaq Database Engineering White Paper *Configuring Compaq RAID Technology for Database Servers*.
- Redo log archive files also are 100% sequential I/O and should be isolated for maximum performance.
- Data file access is usually random and needs to be spread across as many drives as necessary. By increasing the number of physical drives, greater I/O rates can be achieved. Using a striped array will assure that the I/O's are well distributed.

When configuring raw disk partitions be sure to start all partitions on even 1KB addresses. Since Oracle performs all of its disk I/O in 2KB blocks, aligning the disk partition to start on an even address will ensure that a 2KB I/O will not span multiple disks. The Compaq IDA, IDA-2 and SMART arrays stripe in 16KB chunks. If a partition starts on an odd boundary, a 2KB I/O will take the 1st 1KB from the last 1KB of a 16KB stripe and 1KB from the beginning of the next 16KB stripe. This is inefficient and should be avoided.

Don't exceed I/O limits

Try not to overload any individual disk with random I/Os. Compaq recommends random I/Os not exceed 50 I/Os per second per drive for 2GB drives and not exceed 40 I/Os per second per drive for 1GB and 500MB drives. To determine the I/O rate per driver first determine the number of I/O's per second to each logical volume. This can be done with *sar -d*, *rtpm* or 3rd party tools. Take the number of I/O's per second to each logical volume and divide by the number of physical disks in that logical volume. This will provide the number of I/O's per second per disk. If this number exceeds recommended I/O's per second rating adding more physical disks should improve average system performance. The following *sar* command provides an example for determining the I/O, r+w/s, for each logical controller volume on a system.

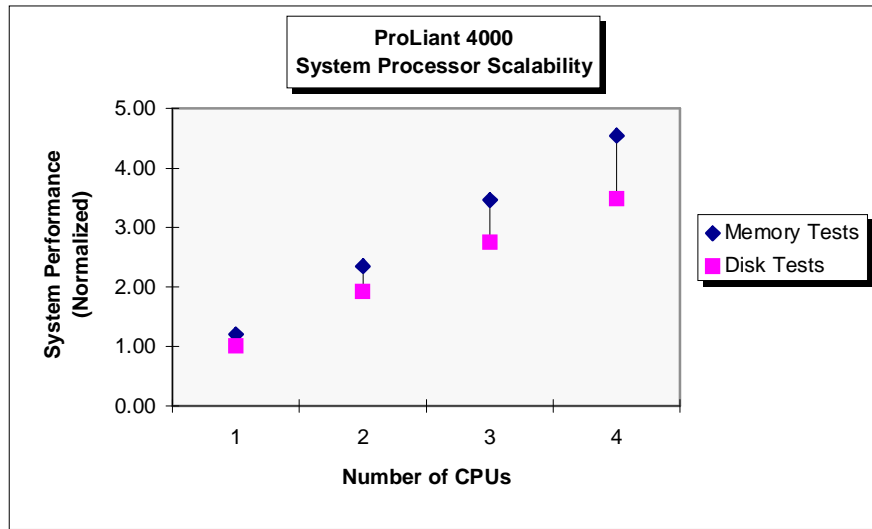
```
sar -d 25 1
```

```
UNIX_SV oracle1 4.2MP 2.0 i386 01/29/95
```

10:51:47	device	%busy	avque	r+w/s	blks/s	await	avserv
10:52:12	sd011	26.13	218.81	156.95	712.22	362.65	1.66
	sd012	100.00	56.07	295.67	1178.64	681.90	12.38
	sd013	100.00	89.28	331.34	1319.40	1140.51	12.92
	sd014	100.00	109.97	260.40	1029.66	1605.28	14.73

CPU Scalability

Oracle7 for UnixWare 2.0 provides excellent CPU scalability. By adding additional CPUs, performance can be dramatically improved, provided that you are CPU-bound. While no OS or hardware provides 100% scalability, UnixWare 2.0 and Oracle7 provide a very high level of scalability. Since most well tuned Oracle instances are CPU bound, most applications will see significant improvement. If the system is I/O bound, adding CPUs will not offer significant improvement; however, the parallel options Oracle7 version 7.1 may increase performance even in an I/O bound system. The following graphical representation of CPU scalability on the Compaq ProLiant 4000 using tests that are both disk and memory intensive



Memory Tuning

Oracle DB Block Buffers

In order to reduce excess I/O, tune the Oracle data cache to provide maximum use of the resources provided. The Oracle buffers are tuned with the parameter **DB_BLOCK_BUFFERS**. One way to utilize the system's resources efficiently is to provide the optimum balance between the Oracle data cache and the UNIX file system buffers. If your system is using raw devices for all Oracle database files and redo log files, and there is little OS activity, the file system buffers can be tuned down and the additional memory used for Oracle buffers. The file system buffers are tuned with the OS parameters **NBUF** and **BUFHWM**. The OS dynamically allocates filesystem buffers in chunks of **NBUF** with the total amount limited by the tunable **BUFHWM** (in kilobytes). The tunable **BUFHWM** is tuned automatically. To see the current value use the OS command `sysdef`.

Do not tune the Oracle buffers up at the expense of swapping. Swapping will degrade system performance more than the advantages acquired by tuning up the Oracle buffers. Keep in mind that the Oracle processes and shadow processes also take significant memory resources. See the Oracle7 for UnixWare 2.0 Installation and Configuration Guide for specifics on disk and memory requirements for each of the Oracle products.

SGA Size

Since the Oracle System Global Area, SGA, resides in shared memory, shared memory must be large enough to hold the SGA. The SGA is made up of the following components:

Fixed Size + Variable Size + DB Block Buffers + Redo Buffers = Size of SGA

Fixed Size is determined by the products that you have installed.

Variable Size is determined by the majority of the init.ora parameters.

DB Block Buffers is determined by the **DB_BLOCK_BUFFERS*DB_BUFFER_SIZE** parameters. By default **DB_BUFFER_SIZE=2048** bytes and need not be changed.

Note: If you are using AIO and Oracle can lock down memory for AIO, the DB buffers will not be swappable.

The Redo Buffer size is determined by the **LOG_BUFFERS** parameter. **LOG_BUFFERS** is specified in bytes.

The amount of shared memory allowed in UnixWare needs to exceed the amount of shared memory required for Oracle. The maximum amount of shared memory allowed in a system is determined by several tunable parameters. The OS tunables **SVMMLIM** and **HVMMLIM**, sets the maximum amount of memory available to a user on the system and needs to exceed this SGA size. The OS tunable **SHMMAX** sets the maximum size of a single shared memory segment. The maximum amount of shared memory that can be used by a single process is equal to the OS tunables **SHMMAX*SHMSEG** (maximum shared memory segment size * maximum number of shared memory segments). Check to see that only one shared memory segment is being allocated. If more than one segment is being allocated for a single SGA it is less efficient than having one large shared memory segment and you should increase **SHMMAX**. This can be verified with the OS command *ipcs*.

ipcs -b

IPC status from /dev/kmem as of Sun Jan 29 10:33:19 1995

T	ID	KEY	MODE	OWNER	GROUP	QBYTES
---	----	-----	------	-------	-------	--------

Message Queues:

T	ID	KEY	MODE	OWNER	GROUP	SEGSZ
---	----	-----	------	-------	-------	-------

Shared Memory:

m	900	0x10043232	--rw-r----	oracle	dba	50745344
---	-----	------------	------------	--------	-----	----------

T	ID	KEY	MODE	OWNER	GROUP	NSEMS
---	----	-----	------	-------	-------	-------

Semaphores:

Note: Only one shared memory segment has been allocated with a size of 50745344. No semaphores have been allocated indicating that Post-Wait is in operation.

By increasing the amount of shared memory allocated, you are reducing the amount of memory available to user processes. Be careful not to reduce user process memory to a point where swapping may occur. Swapping can be detected by noting the available space on the swap file with the UNIX command */etc/swap -l* or by noting swapping activity with *sar -r*, *sar -w*, or with *rpm*.

```
sar -r 5 5
```

```
UNIX_SV oracle1 4.2MP 2.0 i386 01/29/95
```

10:09:54	freemem	freeswp
10:09:59	28640	196608
10:10:04	28627	196608
10:10:09	28637	196608
10:10:15	28637	196608
10:10:20	28637	196608
Average	28636	196608

These values are given in number of 4k pages. As you can see there is plenty of free memory. If the amount of freemem dips below 100 and freeswp decreases, then swapping is occurring.

```
sar -w
```

```
UNIX_SV oracle1 4.2MP 2.0 i386 01/29/95
```

	swpin/s	bswin/s	swpot/s	bswot/s	pswch/s
10:09:28	0.00	0.0	0.00	0.0	1536
10:09:33	0.00	0.0	0.00	0.0	1510
10:09:38	0.00	0.0	0.00	0.0	1512
10:09:43	0.00	0.0	0.00	0.0	1558
10:09:48	0.00	0.0	0.00	0.0	1520
10:09:54	0.00	0.0	0.00	0.0	1527
Average	0.00	0.0	0.00	0.0	1527

Note: The blocks swapped in and out per second (bswin/s & bswot/s) are given in 512 byte blocks. The number of swap ins and swap outs per second is zero. This indicates that no swapping is occurring.

PSE Pages

To take the most efficient use of system memory on an Intel Pentium processor you should take advantage of PSE memory. PSE memory allocates contiguous memory with a 4 Mbyte page size that can be used for shared memory. The larger page size allows more efficient page lookup on Intel Pentium processors. PSE pages are allocated in 4 Mbyte pages and are configured with the kernel parameter **PSE_PHYSMEM**. **PSE_PHYSMEM** is specified in bytes but is rounded up to the nearest 4 Mbyte. To allow Oracle to use this shared memory area both **PSE_PHYSMEM** and **SHMMAX** must be tuned to a size sufficient to hold the Oracle SGA.

User Capacity

There are several areas of tuning necessary based on the number of users that you want to connect. There are the Oracle parameters necessary to allow a specified number of connections into the database engine, and there may be several areas of OS tuning required based on your configuration.

The init.ora parameter **PROCESSES** must be adjusted. This parameter specifies the number of processes that Oracle can create. This number also includes the Oracle background processes. The number of Oracle background processes will typically be 4 + the number of DB_WRITERS. The OS equivalent of this is **MAXUP**. **MAXUP** specifies the maximum number of processes allowed on the system on a per user basis. Since all of the background processes are owned by Oracle, this number must be greater than the **PROCESSES** parameter in init.ora. The OS parameter **NPROC** specifies the maximum number of processes allowed on the system. **NPROC** must be at least 50 greater than **MAXUP** to allow for other OS and user processes to run. Both **MAXUP** and **NPROC** are tune automatically based on the amount of memory in your system. To check on the current value, use the OS command sysdef.

Network Tuning

SQL*Net

If the network appears to be operational but a connection via SQL*Net cannot be established, it is possible that the SQL*Net libraries are not linked into the Oracle executables. In order to relink the executables, cd to \$ORACLE_HOME/rdbms/lib and use the command *make -f oracle.mk install*. Stop all Oracle processes while doing this. If an executable file is open, it cannot be replaced. Once this is completed, try to reconnecting to Oracle.

TCP/IP

Under normal loads, TCP/IP most likely does not need to be tuned. Under extreme loads, the OS tunable **TCPWINDOW** may have to be increased. If SQL*Net TCP connections are lost under heavy load, increase this parameter. **TCPWINDOW** can be increased to 24576 or greater.

Management Considerations

Management of the OS, the database, and the hardware is a great concern. If the system is used in a mission critical role, it is especially important that the administrator is notified of any fault in the system and that corrective action is taken as quickly as possible. In the past few months several new products have been introduced to help with this task. As more new products are introduced and evaluated, this document will be revised in order to provide current information.

Compaq Insight Manager (CIM)

CIM is a Windows based utility that uses SNMP in conjunction with OS and Driver Agents on the server to report hardware failures and system degradation due to a hardware problem. CIM can be configured to page the system administrator if a component is failing. Using CIM pre-failure warranty will allow a hardware component to be replaced under warranty before it fails. CIM monitors system hardware and a few OS components.

Oracle Server Manager

Oracle Server Manager is delivered as part of Oracle7 version 7.1. Server Manager provides a graphical interface in order to administer database functions as well as gather some performance information. Server Manager operates over SQL*Net, therefore eliminating the need to execute these functions locally.

DB General

DB General SQL Watch from Bradmark provides an MS Windows based tool that uses SQL*Net to monitor various parameters in the Oracle7 database. The console runs on MS Windows but can connect to Oracle7 databases on any OS which supports network protocols TCP/IP, SPX, or named pipes.

Conclusion

The information in this paper is not a complete tuning guide but a supplement to other tuning information provided by Oracle and Novell. To achieve an optimal configuration, there are several factors to include. Tuning the application, tuning the hardware, tuning the OS, and tuning the network are all areas that must be carefully planned and tuned. The tuning process is iterative and will be done several times in order to achieve the most optimal performance possible. We hope that the information provided in this paper will help in this process. The information given is based on experience in tuning Oracle7 on UnixWare 2.0, however, each configuration is unique. Although all of the hints given here have been tested extensively, do not assume that tuning a specific parameter will always give the desired result. Do not be afraid to experiment.

We welcome feedback on your configurations and experiences to improve our information products in the future. Please send us any comments or suggestions on the attached form, attaching addition sheets if necessary. This will help us tailor future information products to your needs, and will enable us to make future revisions of this document and related new information products available to you.

References

ORACLE for UNIX Performance Tuning Tips, Oracle, Part No. 53134-0293. This book offers many performance tips for Oracle7 in a UNIX environment. The focus is on tuning Oracle and tuning your application. This book gives many details about Oracle7 for UNIX including Oracle defaults and installation issues.

DB General, Bradmark Technologies Inc. (800) 275-2723

User Registration/Evaluation Form

Please fill out and return to us this registration/evaluation form to help us keep you up to date with future revisions of this document and related new information products. Your effort will help us improve the quality of the future information products.

Name: _____
Title: _____
Company: _____
Address: _____

Phone Number: _____

Please evaluate the quality of this document:

	Excellent				Poor
Technical Accuracy	1	2	3	4	5
Organization	1	2	3	4	5
Clarity	1	2	3	4	5
Completeness	1	2	3	4	5
Timeliness	1	2	3	4	5

Please indicate the type of environment you have at your site:

Operating Systems	RDBMS	Processing Type
<input type="checkbox"/> SCO Unix	<input type="checkbox"/> Microsoft SQL Server	<input type="checkbox"/> On-Line Transaction Processing
<input type="checkbox"/> Microsoft Windows NT	<input type="checkbox"/> Sybase System 10	<input type="checkbox"/> Decision Support
<input type="checkbox"/> IBM OS/2	<input type="checkbox"/> Oracle 7	<input type="checkbox"/> Batch Processing
<input type="checkbox"/> Novell NetWare	<input type="checkbox"/> Other:	<input type="checkbox"/> Other:
<input type="checkbox"/> Novell UnixWare		
<input type="checkbox"/> Other:		

Please indicate the type of information you would like us to provide in the future:

Topic	Operating Systems	RDBMS
<input type="checkbox"/> Configuration and Tuning	<input type="checkbox"/> Microsoft Windows NT	<input type="checkbox"/> Microsoft SQL Server
<input type="checkbox"/> Capacity Planning	<input type="checkbox"/> Novell NetWare	<input type="checkbox"/> Sybase System 10
<input type="checkbox"/> Integration Information	<input type="checkbox"/> IBM OS/2	<input type="checkbox"/> Oracle 7
<input type="checkbox"/> Competitive Analysis	<input type="checkbox"/> Novell UnixWare	<input type="checkbox"/> Other:
<input type="checkbox"/> Systems Management	<input type="checkbox"/> SCO Unix	
<input type="checkbox"/> Other:	<input type="checkbox"/> Other:	

Additional Comments:

Return to:

Database Engineering
 Compaq Computer Corporation
 MailCode 090803
 20555 SH 249
 Houston, Texas 77070