# SQL*DBA to Oracle Server Manager™ Migration Guide

**Release 2.3.2**

Part No. A42573–1

ORACLE®

SQL*DBA to Oracle Server Manager™ Migration Guide, Release 2.3.2

Part No. A42573–1



Contributing Authors: Marlene Lopez, Dennis Lee
Contributors: Eric Belden, Valerie Kane, Daniel Sternbergh, Keri Kruse, Lawrence Chan

# Preface

This section describes the purpose and organization of this guide: *SQL*DBA to Oracle Server Manager Migration Guide.* This section also illustrates the conventions used in this guide.

## Purpose of this Guide

This guide provides you with information that you need to consider when migrating from SQL*DBA to the Oracle Server Manager. This guide also discusses the differences between SQL*DBA and Oracle Server Manager.

## How the *SQL*DBA to Oracle Server Manager Migration Guide* Is Organized

This guide is divided into chapters and appendices, as described below.

**Chapter 1: Oracle Server Manager Overview**

This chapter describes the overall organization and behavior of Server Manager's graphical interface.

**Chapter 2: Migration Considerations**

This chapter explains migration issues that you must aware of when migrating from SQL*DBA to Oracle Server Manager.

**Appendix A: SQL*DBA Compatibility**

This appendix describes the differences in functionality and behavior between Server Manager / Line Mode and SQL*DBA.

**Appendix B: System Monitors**

This appendix provides you with the differences between SQL*DBA and Oracle Server Manager monitors and the underlying SQL queries of the Server Manager Systems Monitors.

## Conventions Used in This Guide

The following sections explain the conventions used in this guide.

### Syntax Diagrams

The syntax diagrams in this guide show the complete syntax for the Server Manager commands. Syntax diagrams are composed of these items:

**Keywords** Keywords are words that have special meanings. In the syntax diagrams in this guide, keywords appear in uppercase. When you specify them, they can be in uppercase or lowercase, but they must be used exactly as they appear in the syntax diagram.

**Parameters** Parameters act as place holders in syntax diagrams. They appear in lowercase. Parameters are usually names of schema objects, Oracle datatypes, or expressions. When you see a parameter in a syntax diagram, you should substitute an object or expression of the appropriate type. Note that parameter names appear in italics in the text of this guide.

## Examples

This guide also contains examples. This is an example of a SELECT statement:

```
SELECT * FROM emp
```

Note that the text of examples appears in a different font than the text of the guide.

Examples in this guide follow these case conventions:

- Keywords, such as CREATE and NUMBER, appear in uppercase.
- Names of database objects and their parts, such as `emp` and `empno`, appear in lowercase.

  However, in the text of this guide, names of database objects and their parts appear in uppercase.

## Special Icons

Special icons are provided to alert you to particular information within the body of this guide and within other manuals.

OSDoc

**Additional Information**: **Operating System–Specific Documentation** Where necessary, this guide refers you to your operating system–specific Oracle documentation for additional information.

**Attention:** The attention icon highlights information that is important to remember when performing the described task.

## Related Publications

*SQL*DBA to Oracle Server Manager Migration Guide* is one of many books that explain various parts of an Oracle database system.

Related publications that may assist you in your migration process are listed below:

- For information about administering the Oracle7 Server, see the *Oracle7 Server Administrator's Guide.*
- For information about developing database applications within the Oracle7 Server, see the *Oracle7 Server Application Developer's Guide.*
- For information on Oracle's SQL commands and functions, see the *Oracle7 Server SQL Reference.*
- For information about Oracle's procedural language extension to SQL, PL / SQL, see the *PL/SQL User's Guide and Reference.*
- For information about using the Oracle Server Manager, see *Oracle Server Manager User's Guide*
- For information about installing the Oracle Server Manager on Windows, see *Oracle Server Manager's for Windows Installation Guide.*

## Your Comments Are Welcome

We value and appreciate your comments as an Oracle user and reader of the manuals. As we write, revise, and evaluate our documentation, your opinions are the most important input we receive. At the back of our printed manuals is a Reader's Comment Form, which we encourage you to use to tell us what you like and dislike about this manual or other Oracle manuals. If the form is not available, please use the following address or FAX number.

Server Manager Product Manager
Oracle Corporation
500 Oracle Parkway
Redwood City, CA 94065
U.S.A.
FAX: 415–506–7200

# Contents

CHAPTER

# 1

# Oracle Server Manager Overview

This chapter introduces Oracle Server Manager and gives you an overview of its major components.

This chapter covers the following topics:

- A description of Server Manager
- An overview of Server Manager's organization

**Attention:** Oracle Corporation has begun to obsolete SQL*DBA. SQL*DBA is being replaced by Oracle Server Manager. It is urged that you begin to migrate to Server Manager.

## What is Server Manager?

Server Manager is Oracle's database administration tool. The graphical component of Server Manager (Server Manager/GUI) allows you to perform database administration tasks with the convenience of a graphical user interface (GUI). The line mode component of Server Manager (Server Manager/LineMode) provides a line mode interface.

### Server Manager and Database Administration

Server Manager is designed to replace SQL*DBA and provide administrative functionality through an easy–to–use interface. You can use Server Manager to:

- Perform traditional administrative tasks, such as database startup, shutdown, backup, and recovery. Rather than manually entering the SQL commands to perform these tasks, you can use Server Manager's graphical interface to execute the commands quickly and conveniently.
- Concurrently perform multiple tasks. Because you can open multiple windows simultaneously in Server Manager, you can perform multiple administrative and non–administrative tasks concurrently.
- Administer multiple databases. You can use Server Manager to administer a single database or to simultaneously administer multiple databases.
- Centralize database administration tasks. You can administer both local and remote databases running on any Oracle platform at any location worldwide. In addition, these Oracle platforms can be connected by any network protocol(s) supported by SQL*Net and the MultiProtocol Interchange.
- Dynamically execute SQL, PL/SQL, and Server Manager commands. You can use Server Manager to enter, edit, and execute statements. Server Manager also maintains a history of statements executed. Thus, you can re–execute statements without retyping them, a particularly useful feature if you need to execute lengthy statements repeatedly.
- Perform administrative tasks using Server Manager's line mode interface when a graphical user interface is unavailable or undesirable.

**Portability**

Server Manager is available for multiple GUI environments, yet adopts the native look and feel of the platform on which it is running. So, Server Manager running on Motif looks like a Motif application and Server Manager running on Windows looks like a Windows application.

**Supported Oracle Server Releases**

You can use Server Manager to administer any database running Oracle7 release 7.0 or later. You can also simultaneously administer different databases running different releases of Oracle7.

**Server Manager/LineMode**

Server Manager/Line Mode provides line mode when your environment does not support a graphical user interface or when a command line interface is desirable. In line mode, you can explicitly execute commands on a command line.

You may want to use Server Manager in line mode when a graphical device is unavailable (such as when dialing–in from a non–GUI terminal) or when performing unattended operations (such as when running nightly batch jobs or batch scripts that do not require user intervention).

## Using Server Manager

To use Server Manager to administer a database, you must install Server Manager on your personal computer. You must also install several views on the databases you wish to administer.

> **Note:** For more information about Server Manager, see the *Oracle Server Manager User's Guide*

## Server Manager Organization

Server Manager's graphical interface is divided into three major components:

- Administration Manager
- SQL Worksheet
- System Monitors

### Administration Manager

The Administration Manager is the primary administrative component of Server Manager. You will probably use the Administration Manager to perform most of your administrative tasks.

The following figure illustrates the Administration Manager with the Tablespaces folder open.

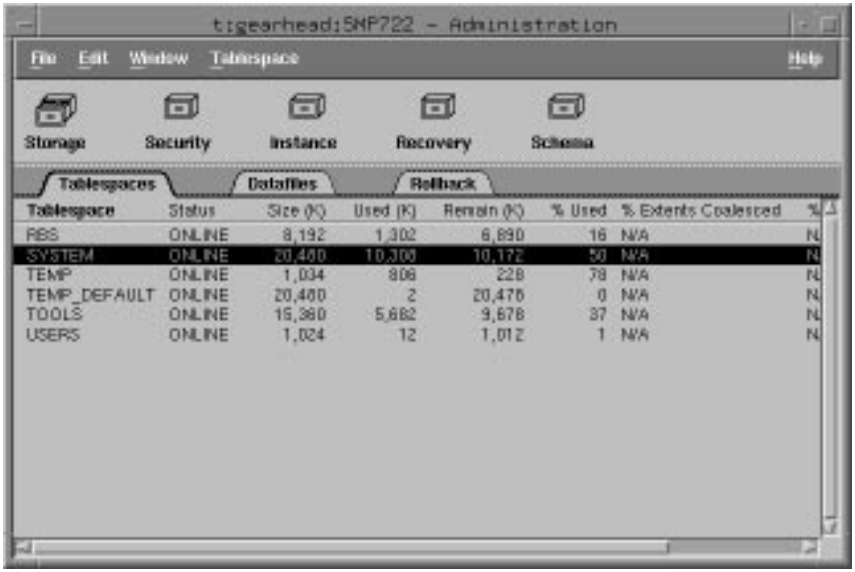


**Figure 1 – 1 Administration Manager**

## SQL Worksheet

A SQL Worksheet allows you to dynamically enter SQL statements, PL/SQL code, and Server Manager commands. You can also run scripts from a SQL Worksheet.

The following figure illustrates the SQL Worksheet.

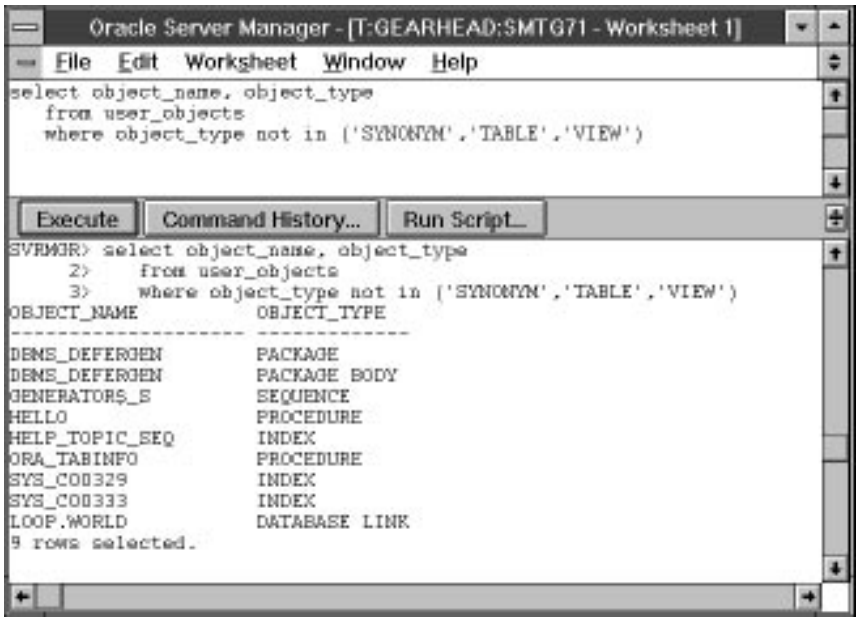


**Figure 1 – 2 SQL Worksheet**

### System Monitors

The System Monitors allow you to gather and inspect performance statistics. These statistics can be useful for tuning your database.

The following figure illustrates an example of using filters in the SQL Area monitor.

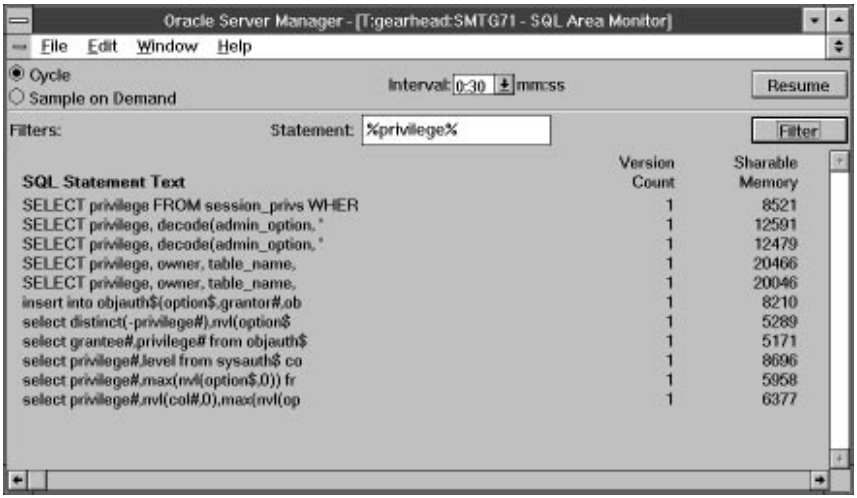


**Figure 1 – 3 SQL Area Monitor**

### Invoking System Monitors

You can start a system monitor from the file menu or by executing the MONITOR command in a SQL Worksheet.

> **Note:** For more information about Server Manager and System Monitors, see Chapter 10 in the *Oracle Server Manager User's Guide.*

## Server Manager in Line Mode

In line mode you can execute the Server Manager commands, execute SQL statements, PL/SQL code, and run SQL scripts.

> **Note:** For more information about Server Manager in Line Mode, see *Oracle Server Manager User's Guide.*

CHAPTER

# 2 Migration Considerations

This chapter describes items to consider when you migrate from SQL*DBA to Oracle Server Manager:

- Launching Server Manager compared to SQL*DBA
- Running SQL scripts

**Attention:** Beginning with Oracle7 release 7.3, SQL*DBA will not be included as part of the database product set. Oracle Server Manager will be the only database administration tool provided with Oracle7 release 7.3.

## Launching Applications

Several differences exist between launching Server Manager and launching SQL*DBA.

### Modes

**SQL*DBA**

SQL*DBA used a variety of methods for starting up in different modes, including command line parameters or environment variables.

**Server Manager**

Server Manager graphical mode and the Server Manager line mode tool are separate applications.

### Initialization

**SQL*DBA**

When initializing, SQL*DBA looked for a SQL script and executed the commands. This occurred before SQL*DBA either executed the user–specified SQL script or accepted user input, depending on the SQL*DBA startup parameters. The location of the script was operating system dependent. For example, on UNIX, the file name was:

```
$ORACLE_HOME/rdbms/admin/sqldba.sql
```

If this file was not present then SQL*DBA would generate error DBA–00383.

**Server Manager**

Server Manager does not use a startup script. DBA's using graphical mode Server Manager can take advantage of its launch in context feature to perform many initialization steps. DBA's using Server Manager line mode can also startup and execute a script by using the COMMAND parameter. Refer to the *Oracle Server Manager User's Guide* for more information about the launch in context feature and using Server Manager in line mode.

**Attention:** Server Manager requires a specific order for STARTUP parameters. The following table summarizes Server Manager STARTUP parmeters and where they must appear:

| STARTUP Parameters | Position |
|---|---|
| FORCE | Anywhere after STARTUP |
| RESTRICT | Anywhere after STARTUP |

| | |
|---|---|
| PFILE=<filename> | Anywhere after STARTUP |
| OPEN \| MOUNT \| NOMOUNT | Anywhere after STARTUP |
| RECOVER | Immediately following OPEN |
| <database name> | Immediately following OPEN or MOUNT |
| EXCLUSIVE \| SHARED / PARALLEL | Anywhere after STARTUP |
| RETRY | Immediately following SHARED / PARALLEL |
| QUIET | Anywhere after STARTUP |

**Display of Line Numbers**

SQL*DBA supported an environment variable for turning off the display of line numbers when entering multi–line commands. Server Manager does not support this feature.

> **Note:** For additional information about compatibility issues between SQL*DBA and Server Manager, refer to Appendix A in this guide.

## SQL Scripts

Server Manager is designed to be compatible with existing SQL*DBA scripts. However, during migration, some SQL scripts might need minor modifications.

### LOGWIDTH

SQL*DBA allowed you to specify the width of spooled log files using the LOGWIDTH parameter. Server Manager does not support this parameter and only inserts end–of–line characters at the end of the lines.

### STARTUP

If an error occurred during an intermediate step of a STARTUP command, SQL*DBA would shut down the database. Server Manager, in the same scenario, leaves the database in the last successfully completed state.

> **Note:** This may have implications for the types of procedures you choose to use in recovering from a failed startup.

**PFILE**

Specific to the optional PFILE=filename parameter in both SQL*DBA and Server Manager, the equal sign ( = ) character was optional for SQL*DBA but is required for Server Manager.

### Line Mode

SQL*DBA line mode aborted processing a SQL statement if a blank line was input. It was possible to have a malformed statement in a SQL script which did not cause an error because it was followed by a blank line. This caused the command to be aborted and SQL*DBA to proceed to the next command. A common example of this occurred when an improper comment syntax had been used in the script.

For example, a script which uses a SQL comment:

```
/*   . . .   */
```

in between valid SQL statements is incorrect. But if the comment was followed by a blank line, the command processing would be aborted and no error would be generated.

Server Manager does not abort processing a SQL statement if a blank line is inserted into it. As a result, a script which contained a problem that remained hidden while using SQL*DBA may generate an error while using Server Manager.

### Spool Files

The name of the default spool file is different for the Server Manager than for SQL*DBA. The default spool file is the file to which output is spooled if no filename is specified with the SPOOL command.

The default spool filename for Server Manager is system specific.

APPENDIX

# A

# SQL*DBA Compatibility

This appendix lists the differences between Oracle7 SQL*DBA and Server Manager.

## Functional Differences

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| **DESCRIBE** for PL/SQL | Lists information about the procedure argument in a tabular form | Lists the definition of the procedure argument |
| **PRINT** <*no args*> | Does nothing | Prints all defined variables |
| **PRINT** (before connect) | `SQLDBA> print a`<br>`DBA-00302: not con-`<br>`nected to a database` | `SVRMGR> print a`<br>`A`<br>`----------`<br>`10` |
| **SET CHARWIDTH** | Displays a character string equivalent to the number of bytes required. For example, if CHARWIDTH is set to 14 and a two–byte character set is used, seven characters are displayed. | Displays the number of characters specified by CHARWIDTH regardless of the number of bytes used by the character string. For example, if CHARWIDTH is set to 14, 14 characters are displayed. |
| **SET COMPATIBILITY NATIVE** | Unsupported | Server Manager uses the native SQL of the database to which you are connected |
| **SHOW** <*no args*> | `SQLDBA> show`<br>`show`<br>`*`<br>`DBA-00115: unexpected`<br>`end of command` | Prints current Server Manager settings |
| **SPOOL** <*no args*> | `SQLDBA> spool`<br>`File sqldba.log`<br>`opened...`<br>No prompt | `SVRMGR> spool`<br>`Name for the spool file:`<br>`(svrmgr.log)`<br><br>Prompts for a filename. The default is SVRMGR.LOG. |

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| **STARTUP** | If any step of STARTUP fails, SQL*DBA backs out and leaves the database in a shutdown state. | If any step of STARTUP fails, Server Manager leaves the database in its current state.<br><br>For example, if an error is encountered after the mount stage, Server Manager leaves the database mounted. |
| **VARIABLE** *a* **CHAR** | Default size is 1 | Default size is CHARWIDTH |
| **@**<*no args*> | `SQLDBA> @`<br>`ORA-07225: ...`<br>Prints an error | `SVRMGR> @`<br>`Name of script file:`<br>Prompts for the name of the file |
| Empty line | `SQLDBA> garbage`<br>`2>`<br>`SQLDBA>;`<br>`SQLDBA>` | `SVRMGR> garbage`<br>`2>`<br>`3>;`<br>`garbage *`<br><br>`ORA-00900: invalid SQL statement`<br>`SVRMGR>` |

## Cosmetic Differences

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| Prompt | `SQLDBA>` | `SVRMGR>` |
| Product Banner | `SQL*DBA` | `Server Manager` |
| Connect to idle instance | `SQLDBA> connect internal`<br>`connected.` | `SVRMGR> connect internal`<br>`connected to an idle instance.` |

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| **PRINT** *a* | A<br>----------<br>10<br>1 row selected.<br>The variable name is space padded: "A ". | A<br>----------<br>10<br><br>The variable name is not space padded: "A". Also, no feedback is listed. |
| **SET** | SQLDBA> set echo on<br>SQLDBA> | SVRMGR> set echo on<br>Echo ON<br>SVRMGR><br>Server Manager echoes the new value |
| Error Message Prefix | DBA-*xxxxx*: ... | MGR-*xxxxx*: ... |
| Wrapping text | Writes 80 characters and then wraps | Writes the entire width and relies on the display device to deal with wrapping. This is better for wide terminals. |

## Unsupported Options

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| **RECOVERY** prompt | | FROM <*log source*> clause is unsupported |
| **SET FLAGGER**... | SET FLAGGER ENTRY | Unsupported: Use the ALTER SESSION SET FLAGGER statement |
| **SHOW FLAGGER** | Shows FLAGGER value | Unsupported |
| **SET LABELWIDTH** | | Unsupported |
| **SET RETRIES** | | Unsupported |

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| **SHOW VARIABLES**<br>**SHOW VAR** | Shows all defined variables | Unsupported: Use the PRINT command with no arguments |
| **SHUTDOWN** *dbname* | | Unsupported |

## Obsolete Functionality

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| **SET ARRAYSIZE** | Changes the size of each fetch call | No longer used: Server Manager automatically chooses optimum fetch size. |
| **SET CYCLE** | Sets the monitor interval | No longer used |
| **SET/SHOW DISPWIDTH** | Line mode | No longer used |
| **SET HISTORY** | Screen mode | No longer used |
| **SET LINES** | Screen mode | No longer used |
| **SET/SHOW LOGWIDTH** | Line mode | No longer used |
| **SET TERM** | Screen mode | No longer used |

## Operational Differences

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| Executing a SQL script from the operating system command line | `>sqldba @script.sql` | `>svrmgrl command = @script.sql` |
| Executing a server command from the operating system command line | Unsupported | `>svrmgrl command="connect x/y@z"` |

| *Feature* | *SQL*DBA* | *Server Manager* |
|---|---|---|
| Entering a Server Manager command spanning multiple lines | Unsupported | Type a backslash (\) at the end of each line to indicate continuation.<br><br>`SVRMGR> connect \`<br>`  2> myrdb/`*`password`* |
| Entering a quoted string spanning multiple lines | | Type a backslash (\) at the end of each line to indicate continuation.<br><br>`SVRMGR> select *`<br>`  2> from emp`<br>`  3> where ename = 'A\`<br>`  4> BCDEF';` |
| Exiting the application | SQLDBA> exit | `SVRMGR> exit`<br>or<br>`SVRMGR> quit` |
| Aborting a command | Hit <Return> twice<br>or<br>Enter a period (.) by itself on a line. | Enter a period (.) by itself on a line. This procedure applies to Server Manager l<br>Line Mode only.<br><br>`SVRMGR> delete from emp`<br>`  2> .`<br>`SVRMGR>` |
| Executing the current SQL statement | | Enter a slash (/) by itself on a line.<br><br>`SVRMGR> select *`<br>`  2> from emp`<br>`  3> /` |

APPENDIX

# B System Monitors

This appendix provides you with an overview of the differences between Server Manager System Monitors and SQL*DBA System Monitors.

This appendix describes:

- the underlying SQL queries of the Server Manager System Monitors
- the enhancements that Server Manager adds to the system monitors

## Overview of System Monitors

The following table describes the Server Manager monitors. For a detailed description of each monitor, see Chapters 10 and 11 in the *Oracle Server Manager User's Guide.*

| ***Server Manager Monitor Name*** | ***Monitor Function*** |
|---|---|
| Circuit Monitor | Displays information about the shared servers' virtual circuits, through which users connect to the database. |
| Dispatcher Monitor | Displays information about the shared server's dispatcher processes. |
| File I/O Monitor | Displays the read and write activity for each file in the database. |
| Latch Monitor | Displays the latches on shared internal data structures in the SGA. |
| Library Cache Monitor | Displays the activity in and the effectiveness of the library cache. |
| Lock Monitor | Displays current database locks. |
| Process Monitor | Displays all background and user processes currently accessing the database through the current instance. |
| Queue Monitor | Displays information on the multi–threaded server's message queues. |
| Rollback Monitor | Displays the current status of rollback segments. |
| Session Monitor | Displays statistics and information for each user session in the current instance. |
| Shared Server Monitor | Displays statistics for the shared server processes. |
| SQL Area Monitor | Displays the memory used to process SQL statements that have been or are being executed. |
| System I/O Monitor | Displays the percentage of I/O activity generated by each background and user process that is accessing the database through the current instance. |
| System Statistics Monitor | Displays runtime statistics for the system. **Note: S**QL*DBAs Session Statistics Monitor has been merged into this monitor. |
| Table Access Monitor | Displays objects being accessed and the session ID that is accessing them. |
| Tablespace Monitor | Displays information about the tablespaces created in the database**. Note:** This monitor is only available with Server Manager. |

## Simple Monitors

Simple monitors display the results of queries against the V$ tables.

To improve readability, the following query elements do not appear:

- PL/SQL functions which serve to format the output or handle special cases
- only the join parts of WHERE clauses appear
- parts of the SQL for filtering on the monitor filters were taken out

**Note:** Some functions have comments in square brackets to describe the functionality of deleted PL/SQL such as formatting and DECODEs.

### Circuit Monitor (Server Manager, SQL*DBA)

```
SELECT   RAWTOHEX(C.CIRCUIT),
         D.NAME,
         S1.NAME,
         S.SID,
         S.SERIAL#,
         C.STATUS,
         C.QUEUE,
         C.MESSAGES,
         C.BYTES
FROM     V$CIRCUIT C, V$DISPATCHER D, V$SHARED_SERVER S1, V$SESSION S
WHERE    C.DISPATCHER = D.PADDR(+) AND C.SERVER = S1.PADDR(+) AND
             C.SADDR = S.SADDR(+)
(SQL*DBA: ORDER BY C.CIRCUIT)
```

### Dispatcher Monitor (Server Manager, SQL*DBA)

```
SELECT   NAME,
         STATUS,
         ACCEPT,
         MESSAGES,
         BYTES,
         OWNED,
         IDLE [formatted to days:hrs:min:sec],
         BUSY [formatted to days:hrs:min:sec],
         BUSY/(IDLE+BUSY),
FROM     V$DISPATCHER
(SQL*DBA: ORDER BY NAME)
```

### Library Cache Monitor (Server Manager, SQL*DBA)

```
SELECT   NAMESPACE,
         GETS,
         GETHITS,
         GETHITRATIO,
         PINS,
         PINHITS,
         PINHITRATIO,
         RELOADS,
         INVALIDATIONS
FROM     V$LIBRARYCACHE
(SQL*DBA: ORDER BY NAMESPACE)
```

### Lock Monitor (Server Manager, SQL*DBA)

```
SELECT   S.USERNAME,
         S.SID,
         S.SERIAL#,
         L.TYPE,
         L.ID1,
         L.ID2,
         DECODE(L.LMODE, [mode abbreviations for mode type numbers],
             [default] '?'),
         DECODE(L.REQUEST, [mode abbreviations for mode type numbers],
        [default] '?')
FROM     V$LOCK L, V$SESSION S
WHERE    L.SID=S.SID
(SQL*DBA: ORDER BY S.SID, L.TYPE)
```

### Process Monitor (Server Manager)

```
SELECT   P.PID,
         P.SPID,
         DECODE (P.USERNAME,
                 '?', DECODE(S.USERNAME,
                             NULL, P.USERNAME,
                             '(' || S.USERNAME || ')'),
                 P.USERNAME),
         NVL(S.TERMINAL, P.TERMINAL),
         P.LATCHWAIT,
         NVL(S.PROGRAM, P.PROGRAM)
FROM     V$PROCESS P, V$SESSION S
WHERE    P.ADDR = S.PADDR(+)
```

### Process Monitor (SQL*DBA)

```
SELECT   P.PID,
         P.SPID,
         P.USERNAME,
         P.LATCHWAIT,
         DECODE(S.TERMINAL, NULL, P.TERMINAL, S.TERMINAL),
         DECODE(S.PROGRAM, NULL, P.PROGRAM, S.PROGRAM)
FROM     V$PROCESS P, V$SESSION S
WHERE    P.ADDR = S.PADDR
ORDER BY PID
```

### Queue Monitor (Server Manager, SQL*DBA)

```
SELECT   RAWTOHEX(PADDR),
         TYPE,
         QUEUED,
         TOTALQ,
         DECODE(TOTALQ, 0, 0, WAIT/TOTALQ/100)
FROM     V$QUEUE
(SQL*DBA: ORDER BY PADDR)
```

### Session Monitor (Server Manager)

```
SELECT   S.SID,
         S.SERIAL#,
         P.PID,
         S.STATUS,
         S.USERNAME,
         LOCKWAIT,
         DECODE (S.COMMAND, 0, 'NONE', NVL(A.NAME, 'UNKNOWN'))
FROM     V$SESSION S, V$PROCESS P, AUDIT_ACTIONS A
WHERE    S.PADDR = P.ADDR AND A.ACTION (+) = S.COMMAND
```

### Session Monitor (SQL*DBA)

```
SELECT   S.SID,
         S.SERIAL#,
         P.PID,
         S.STATUS,
         S.USERNAME,
         LOCKWAIT,
         DECODE(S.COMMAND, [command names for each command number],
            [default] 'UNKNOWN')
FROM     V$SESSION S, V$PROCESS P
WHERE    S.PADDR = P.ADDR
ORDER BY S.SID, S.SERIAL#
```

### Shared Server Monitor (Server Manager, SQL*DBA)

```
SELECT   NAME,
         STATUS,
         REQUESTS,
         IDLE [formatted to days:hrs:min:sec],
         BUSY [formatted to days:hrs:min:sec],
         BUSY/(IDLE+BUSY),
         RAWTOHEX(CIRCUIT)
FROM     V$SHARED_SERVER
(SQL*DBA: ORDER BY NAME)
```

### SQL Area Monitor (Server Manager, SQL*DBA)

```
SELECT   SQL_TEXT,
         VERSION_COUNT,
         SHARABLE_MEM,
         PERSISTENT_MEM,
         RUNTIME_MEM
FROM     V$SQLAREA
(SQL*DBA: ORDER BY SQL_TEXT)
```

### Table Access Monitor (Server Manager, SQL*DBA)

```
SELECT   SID,
         OWNER,
         OBJECT
FROM     V$ACCESS
(SQL*DBA: ORDER BY SID,OWNER)
```

### Tablespace Monitor (Server Manager only)

```
SELECT   DS.TABLESPACE_NAME,
         DECODE(DS.STATUS, 'ONLINE', DS.STATUS, NLS_INITCAP(DS.STATUS)),
         SUM(DF.BLOCKS),
         SUM(DT.BLOCKS)
FROM     SYS.DBA_TABLESPACES DS, SYS.DBA_TS_QUOTAS DT, SYS.DBA_DATA_FILES DF
WHERE    DS.STATUS IN ('ONLINE', 'OFFLINE') AND
         DS.TABLESPACE_NAME = DF.TABLESPACE_NAME AND
         DF.TABLESPACE_NAME = DT.TABLESPACE_NAME (+)
GROUP BY DS.TABLESPACE_NAME, DS.STATUS
```

## Complex System Monitors

Complex monitors perform calculations on the queried value and display the calculated values instead of or in addition to the queried column values.

The following SQL*DBA monitors display quantities derived from the queried values retrieved from the dynamic performance (V$) tables.

| | |
|---|---|
| `DELTA(:1)` | the current retrieved value for :1 minus the previous retrieved value for :1. |
| `RATE(DELTA(:1))` | DELTA(:1) divided by the amount of time between the queries (note: RATE requires DELTA). |
| `PCTOFTOTAL(:1)` | the value in the current row of column :1 expressed as a percentage of the sum of all values in the column. |
| `NORMALIZED(:1)` | current retrieved value for :1 minus the retrieved value for :1 when the monitor was started. |
| `TAVG(:1)` | NORMALIZED (:1) divided by total time a monitor has been running. |
| `TMIN(:1)` | smallest value of TAVG(:1) since the monitor was started. |
| `TMAX(:1)` | largest value of TAVG(:1) since the monitor was started. |

## File I/O Monitor (Server Manager, SQL*DBA)

```
SELECT   NAME,          :1
         PHYRDS,        :2
         PHYWRTS,       :3
         PHYBLKRD,      :4
        PHYBLKWRT,     :5
         READTIM,       :6
         WRITETIM       :7
FROM     V$DBFILE DF, V$FILESTAT FS
WHERE    DF.FILE#=FS.FILE#
ORDER BY DF.FILE#
```

| Server Manager Columns | SQL*DBA Columns | Column Calculations |
|---|---|---|
| Datafile | Data File | :1 |
| | -- Request Rate | |
| Request Reads/s | Reads/s | RATE(DELTA(:2)) |
| Request Writes/s | Writes/s | RATE(DELTA(:3)) |
| | -- Batch Size | |
| Batch Blks/rd | blks/rd | DELTA(:4)/DELTA(:2) |
| Batch Blks/wt | blks/wt | DELTA(:5)/DELTA(:3) |
| | -- Response Time | |
| Resp Time ms/rd | ms/rd | DELTA(:6)/DELTA(:2) |
| Resp Time ms/wt | ms/wt | DELTA(:7)/DELTA(:3) |
| | -- Total Blocks | |
| | Read | :2 |
| | Written | :3 |

## Latch Monitor (Server Manager)

```
SELECT   L.NAME,             :1
         LH.PID,             :2
         L.GETS,             :3
         L.MISSES,           :4
         L.SLEEPS,           :5
         L.IMMEDIATE_GETS,   :6
         L.IMMEDIATE_MISSES  :7
FROM     V$LATCH L, V$LATCHHOLDER LH
WHERE    L.ADDR = LH.LADDR(+)
```

| Server Manager Columns | Column Calculations |
|---|---|
| Latch Name | :1 |
| Holder PID | :2 |
| -- Willing-to-Wait-Requests | |
| Gets (Wait) | DELTA(:3) |
| Misses (Wait) | DELTA(:4) |
| Sleeps (Wait) | DELTA(:5) |
| -- No-Wait-Requests | |
| Gets (No Wait) | DELTA(:6) |
| Misses (No Wait) | DELTA(:7) |

## Latch Monitor (SQL*DBA)

```
SELECT   LN.NAME,            :1
         LH.PID,             :2
         L.GETS,             :3
         L.MISSES,           :4
         L.SLEEPS,           :5
         L.IMMEDIATE_GETS,   :6
         L.IMMEDIATE_MISSES  :7
FROM     V$LATCH L, V$LATCHNAME LN, V$LATCHHOLDER LH
WHERE    L.LATCH#=LN.LATCH# AND L.ADDR = LH.LADDR(+)
ORDER BY L.LEVEL#,L.LATCH#
```

| SQL*DBA Column | Column Calculations |
|---|---|
| Latch Name | :1 |
| Holder PID | :2 |
| Gets | DELTA(:3) |
| Misses | DELTA(:4) |
| Sleeps | DELTA(:5) |
| Gets | DELTA(:6) |
| Misses | DELTA(:7) |

## Rollback Monitor (Server Manager)

```
SELECT   RN.NAME,          :1
         RN.USN,           :2
         RS.RSSIZE,        :3
         RS.EXTENTS,       :4
         RS.XACTS,         :5
         RS.WRITES,        :6
         RS.GETS,          :7
         RS.WAITS,         :8
         RS.OPTSIZE,       :9
         RS.HWMSIZE,       :10
         RS.AVEACTIVE,     :11
         RS.AVESHRINK,     :12
         RS.WRAPS,         :13
         RS.EXTENDS,       :14
         RS.SHRINKS,       :15
         RS.STATUS         :16
FROM     V$ROLLNAME RN, V$ROLLSTAT RS
WHERE    RN.USN = RS.USN
```

| Server Manager Columns | Column Calculations |
|---|---|
| Rollback Segment | :1 |
| RS ID | :2 |
| Size (bytes) | :3 |
| Extents | :4 |
| Active Xactions | :5 |
| Write Rate (bytes/s) | RATE(DELTA(:6)) |
| Header Gets | RATE(DELTA(:7)) |
| Header Waits | RATE(DELTA(:8)) |
| Optimal Size | :9 |
| HWM Size | :10 |
| Average Active | :11 |
| Average Shrink | :12 |
| Wraps | :13 |
| Extends | :14 |
| Shrinks | :15 |
| Status | :16 |

## Rollback Monitor (SQL*DBA)

```
SELECT   RN.USN,          :1
         RN.NAME,         :2
         RS.STATUS        :3
         RS.RSSIZE,       :4
         RS.EXTENTS,      :5
         RS.XACTS,        :6
         RS.WRITES,       :7
        RS.GETS,         :8
         RS.WAITS,        :9
         RS.OPTSIZE,      :10
         RS.HWMSIZE,      :11
         RS.AVEACTIVE,    :12
         RS.AVESHRINK,    :13
        RS.WRAPS,        :14
         RS.EXTENDS,      :15
         RS.SHRINKS,      :16
FROM     V$ROLLNAME RN, V$ROLLSTAT RS
WHERE    RN.USN=RS.USN
ORDER BY RN.USN
```

| SQL*DBA Columns | Column Calculations |
|---|---|
| ID | :1 |
| Rollback Segment | :2 |
| Status | :3 |
| Size (bytes) | :4 |
| Extents | :5 |
| Active Xactions | :6 |
| Write Rate (bytes/s) | RATE(DELTA(:7)) |
| -- Header | |
| Gets/s | RATE(DELTA(:8)) |
| Waits/s | RATE(DELTA(:9)) |
| -- Sizes (bytes) | |
| Optimal | :10 |
| Highwater | :11 |
| -- Avg Sizes (bytes) | |
| Active | :12 |
| Shrunk | :13 |
| -- Occurrences | |
| Wraps | :14 |
| Extends | :15 |
| Shrinks | :16 |

## Session Statistics Monitor (SQL*DBA only)

```
SELECT   DISTINCT S.NAME,    :1
         ST.VALUE            :2
FROM     V$STATNAME S, V$SESSTAT ST, V$SESSION SS
WHERE    S.STATISTIC# = ST.STATISTIC# AND ST.SID = SS.SID
ORDER BY S.NAME, ST.VALUE
```

| Columns | Column Calculations |
|---|---|
| Statistic Name | :1 |
| Current | DELTA(:2) |
| Average | TAVG(DELTA(:2)) |
| Minimum | TMIN(DELTA(:2)) |
| Maximum | TMAX(DELTA(:2)) |
| Total | :2 |

## System I/O Monitor (Server Manager)

```
SELECT   P.PID,                                                          :1
         S.SID,                                                          :2
         P.USERNAME,                                                     :3
         ((I.BLOCK_GETS + I.CONSISTENT_GETS) /
            SUM (I.BLOCK_GETS + I.CONSISTENT_GETS + .0001)) * 100,       :4
         (I.PHYSICAL_READS / SUM (I.PHYSICAL_READS + .0001)) * 100,      :5
         ((I.BLOCK_CHANGES + I.CONSISTENT_CHANGES) /
            SUM (I.BLOCK_CHANGES + I.CONSISTENT_CHANGES + .0001)) * 100,  :6
         ((I.BLOCK_GETS + I.CONSISTENT_GETS) /
              SUM (I.BLOCK_GETS + I.CONSISTENT_GETS + .0001)) * 100,     :7
         (I.PHYSICAL_READS / SUM (I.PHYSICAL_READS + .0001)) * 100,      :8
         ((I.BLOCK_CHANGES + I.CONSISTENT_CHANGES) /
              SUM (I.BLOCK_CHANGES + I.CONSISTENT_CHANGES + .0001)) * 100 :9
FROM     V$PROCESS P, V$SESSION S, V$SESS_IO I
WHERE    P.ADDR=S.PADDR AND I.SID = S.SID  \
GROUP BY P.PID, S.SID, P.USERNAME, I.BLOCK_GETS, I.CONSISTENT_GETS,
         I.PHYSICAL_READS, I.BLOCK_CHANGES, I.CONSISTENT_CHANGES
```

| Server Manager Columns | Column Calculations |
|---|---|
| Process ID | :1 |
| Session ID | :2 |
| System Username | :3 |
| Change in % logical reads | DELTA(:4) |
| Change in % physical reads | DELTA(:5) |
| Change in % logical writes | DELTA(:6) |
| Total % logical reads | :7 |
| Total % physical reads | :8 |
| Total % logical writes | :9 |

## System I/O Monitor (SQL*DBA)

```
SELECT   P.PID,                                    :1
         S.SID,                                    :2
         I.BLOCK_GETS + I.CONSISTENT_GETS,         :3
        I.PHYSICAL_READS,                         :4
         I.BLOCK_CHANGES + I.CONSISTENT_CHANGES   :5
FROM     V$PROCESS P, V$SESSION S, V$SESS_IO I
WHERE    P.ADDR=S.PADDR AND I.SID=S.SID
ORDER BY P.PID, S.SID
```

| SQL*DBA Columns | Column Calculations |
|---|---|
| Process ID | :1 |
| Session ID | :2 |
| -- Interval | |
| %logical reads | PCTOFTOTAL(DELTA(:3)) |
| %physical reads | PCTOFTOTAL(DELTA(:4)) |
| %logical writes | PCTOFTOTAL(DELTA(:5)) |
| -- Cumulative | |
| %logical reads | PCTOFTOTAL(NORMALIZED(:3)) |
| %physical reads | PCTOFTOTAL(NORMALIZED(:4)) |
| %logical writes | PCTOFTOTAL(NORMALIZED(:5)) |

## System Statistics Monitor (Server Manager, SQL*DBA)

```
SELECT   S.NAME,    :1
         S.VALUE    :2
FROM     V$SYSSTAT S
ORDER BY S.NAME, S.VALUE
```

| Server Manager Columns | Column Calculations |
|---|---|
| Statistic Name | :1 |
| Total | :2 |
| Current | DELTA(:2) |
| Average | TAVG(DELTA(:2)) |
| Minimum | TMIN(DELTA(:2)) |
| Maximum | TMAX(DELTA(:2)) |

| SQL*DBA Columns | Column Calculations |
|---|---|
| Statistic Name | :1 |
| Current | DELTA(:2) |
| Average | TAVG(DELTA(:2)) |
| Minimum | TMIN(DELTA(:2)) |
| Maximum | TMAX(DELTA(:2)) |
| Total | :2 |

# Reader's Comment Form

**Name of Document: SQL*DBA to Oracle Server Manager™ Migration Guide**
**Part No. A42573–1**

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the topic, chapter, and page number below:

_____

_____

_____

_____

_____

_____

_____

_____

Please send your comments to:

SMP Documentation Manager
Oracle Corporation
500 Oracle Parkway
Redwood City, CA 94065 U.S.A.
Fax: (415) 506–7200

If you would like a reply, please give your name, address, and telephone number below:

_____

_____

_____

Thank you for helping us improve our documentation.