

# Oracle® Database

## Database Administrator's Reference



19c for Microsoft Windows

E96296-03

January 2020

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle Database Database Administrator's Reference, 19c for Microsoft Windows

E96296-03

Copyright © 1996, 2020, Oracle and/or its affiliates. All rights reserved.

Primary Authors: Tanaya Bhattacharjee, Sunil Surabhi, Mark Bauer

Contributing Authors: Lance Ashdown

Contributors: Alexander Keh, Sivaselvam Narayanasamy, Ricky Chen, David Collelo, David Friedman, Prakash Jashnani, Sue K. Lee, Rich Long, Satish Panchumarthy, Ravi Thammaiah, Michael Verheij

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

## Preface

---

Audience	xiv
Documentation Accessibility	xiv
Accessing Documentation	xv
Related Documents	xv
Conventions	xv

## Changes in This Release for Oracle Database Administrator's Reference Guide

---

Changes in Oracle Database Release 19c	xvi
--	-----

## 1 Oracle Database Architecture on Windows

---

Overview of Oracle Database on Windows Architecture	1-1
Oracle Automatic Storage Management	1-1
Oracle Automatic Storage Management File Access Control	1-2
Creation of New User Groups and Users for Separation of Database Administration Duties	1-3
About Disk Group User Replacement	1-3
About Changing File Access Control While the File is Open	1-4
Thread-Based Architecture	1-4
File I/O Enhancements	1-6
Overview of Oracle Database Scalability on Windows	1-7
Large User Populations	1-7
Oracle Database Integration with Windows	1-7
Oracle PKI Integration with Windows	1-8
Oracle Fail Safe Integration with Windows	1-8
Oracle Services for Microsoft Transaction Server	1-9

## 2 Database Tools on Windows

---

Choosing a Database Tool	2-1
--------------------------	-----

Database Tools and Operating System Compatibility	2-1
Preferred Database Tools	2-2
Starting Database Tools	2-4
Starting Database Tools in Multiple Oracle Homes	2-4
Starting Tools from Multiple Oracle Homes	2-5
Running Tools with Windows User Account Control	2-5
Starting Database Tools from the Start Menu	2-6
Starting Database Tools from the Command Line	2-7
About Archiving Redo Log Files	2-9
Starting Windows Tools	2-10
Using the Oracle Home User Control Tool	2-10
Using Windows Tools	2-11
Using Event Viewer to Monitor a Database	2-12
Using Microsoft Management Console to Administer a Database	2-12
Using Registry Editor to Modify Configuration Information	2-12
Using Task Manager to Monitor Applications and Processes	2-13
Using Local Users and Groups to Manage Users and Groups	2-13
Using SQL*Loader	2-13
Windows Processing Options	2-14
Control File Conventions	2-14

### 3 Supporting Oracle Home User on Windows

---

Managing Oracle Home User	3-2
Using Oracle Home User for an Oracle Database and Oracle Database Client	3-3
Using Oracle Home User for Multiple Oracle Homes	3-4
Using Oracle Home User During Oracle Database Upgrade	3-5
Converting from Single-Instance Oracle Database to Oracle Real Application Clusters	3-5

### 4 Postinstallation Database Creation on Windows

---

About Oracle Database Naming Conventions	4-1
About Using Oracle Database Configuration Assistant on Windows	4-2
Overview of Database Creation Tasks on Windows Using Command-Line Tools	4-3
About Exporting an Existing Database	4-5
Exporting All Data from an Existing Database	4-6
Deleting Database Files	4-7
Modifying the Initialization Parameter File	4-7
About Creating and Starting an Oracle Database Service	4-9
Creating and Starting an Oracle Database Service	4-9
Access to Oracle Wallets in a File System for Oracle Database Services	4-10

Starting an Oracle Database Instance	4-10
Adding the CREATE DATABASE Statement in a Script	4-10
Running the CREATE DATABASE Script	4-11
About Importing a Database	4-12
Updating ORACLE_SID in the Registry	4-13
Creating the ORACLE_SID Parameter	4-14
Backing Up the New Database	4-14
About Administering an Oracle Database Instance Using ORADIM	4-15
Creating an Instance Using ORADIM	4-17
Starting an Instance and Services Using ORADIM	4-18
Stopping an Instance and Services Using ORADIM	4-19
Editing an Instance Using ORADIM	4-20
Deleting an Instance Using ORADIM	4-20
Manipulating ACLs Using ORADIM	4-21
Manipulating Family Settings to Initialization Parameters using ORADIM	4-22
About Administering an Oracle Database Instance Using Microsoft Management Console Snapin	4-22
Overview of Database Migration from a 32-Bit Windows Computer	4-23
Backing Up a 32-Bit Oracle Database	4-23
Migration Considerations	4-24
Migrating an Oracle Database 11g Release 2 (11.2) or Earlier Database	4-24

## 5 Postinstallation Configuration Tasks on Windows

---

Overview of Windows Firewall	5-2
About Oracle Executables Requiring Windows Firewall Exceptions	5-2
Configuring Windows Firewall Exceptions for Successful Connections to Oracle Software	5-3
Overview of Different Executables Added to the Windows Firewall Exception List	5-4
Configuring the Windows Firewall	5-6
About Backing Up a Database	5-7
Troubleshooting Windows Firewall Exceptions	5-8
About the Need to Reset Passwords for Default Accounts	5-8
About Windows Authenticated Users	5-8
Overview of NTFS File System and Windows Registry Permissions	5-9
Setting File Permissions	5-10
About Default File Permissions Set by Oracle Universal Installer	5-10
About File Permissions Set by Oracle Database Configuration Assistant	5-11
About File Permissions Set by Oracle Database Upgrade Assistant	5-12
About Setting Permissions for Oracle Wallets	5-13
About Setting File System ACLs Manually	5-13

Setting Permissions for Windows Registry Entries	5-13
Setting Permissions for Windows Service Entries	5-13
Setting NTFS File System Security	5-14
Setting Windows Registry Security	5-14
Overview of ReFS File System	5-14
Setting File Permissions	5-15
About Configuring External Job Support for the Scheduler on Windows	5-15
About Oracle Multimedia on Windows	5-16
Configuring Oracle Multimedia on Windows	5-16
About Oracle Text on Windows	5-17
About Oracle Spatial and Graph on Windows	5-18
Configuring Oracle Spatial and Graph on Windows Automatically	5-18
About Advanced Replication on Windows	5-18
About Checking Tablespace and Rollback Segment Requirements	5-19
Adding and Modifying Initialization Parameters	5-19
Monitoring Data Dictionary Tables	5-20

## 6 Administering a Database on Windows

---

About Ways to Manage Oracle Database Services	6-1
Overview of Oracle Database Service Naming Conventions for Multiple Oracle Homes	6-2
Starting Oracle Database Services	6-2
Stopping Oracle Database Services	6-3
Auto-Starting Oracle Database Services	6-3
Starting and Shutting Down a Database with SQL*Plus	6-4
Starting and Shutting Down a Database Using Services	6-5
Starting Multiple Instances	6-7
Creating and Populating Password Files	6-7
Viewing and Hiding the Password File	6-9
Connecting Remotely to the Database	6-10
Connecting to a Database Using SYSDBA Privileges	6-10
About Verifying a Remote Database Using Encrypted Passwords	6-11
About Archiving Redo Log Files	6-11

## 7 Monitoring a Database on Windows

---

Overview of Database Monitoring Tools	7-1
About Event Viewer	7-2
Using Event Viewer	7-2
Managing Event Viewer	7-4
Reading Event Viewer	7-4

About Trace Files	7-6
About Alert Logs	7-6
Viewing Oracle Database Thread Information	7-7

## 8 Tuning Windows to Optimize Oracle Database

---

Overview of Windows Tuning	8-2
Overview of Large Page Support	8-2
Granting Lock Pages in Memory Privilege	8-3
Enabling Large Page Support	8-3
About Reducing Priority of Foreground Applications on Server Console	8-5
About Configuring Windows Server to Be an Application Server	8-6
About Disabling Unnecessary Services	8-6
About the Necessity to Remove Unused Network Protocols	8-7
About the Necessity to Reset Network Protocol Bind Order	8-8
Setting the Order of Multiple Network Interface Cards	8-8
Overview of the Latest Reliable Windows Server Service Pack	8-9
Overview of Hardware or Operating System Striping	8-9
About Multiple Striped Volumes for Sequential and Random Access	8-11
About Multiplex Windows Server Virtual Memory Paging File	8-11
Closing All Unnecessary Foreground Applications	8-12

## 9 Performing Database Backup and Recovery with VSS

---

Overview of Database Backup and Recovery with VSS	9-1
Scope of This Chapter	9-2
Purpose of Database Backup and Recovery with VSS	9-2
Basic Concepts of Database Backup and Recovery with VSS	9-2
Component-Based Shadow Copies	9-3
Volume-Based Shadow Copies	9-3
Oracle VSS Backup Types	9-4
Basic Steps of Backup and Recovery with VSS	9-4
About Installing and Uninstalling the Oracle VSS Writer Service	9-5
About Backing Up a Database	9-6
About Component-Based Backups	9-7
About Backing Up a Database in ARCHIVELOG Mode	9-9
Making Component-Based Backups of an ARCHIVELOG Database	9-9
Making Volume-Based Backups of an ARCHIVELOG Database	9-9
About Backing Up a Database in NOARCHIVELOG Mode	9-10
Making Component-Based Backups of a NOARCHIVELOG Database	9-10
Making Volume-Based Backups of a NOARCHIVELOG Database	9-10

About Restoring and Recovering a Database	9-11
About Restoring and Recovering a Database in ARCHIVELOG Mode	9-11
Restoring the Server Parameter File	9-13
Recovering from the Loss of All Control Files	9-13
Recovering Tablespaces or Data Files	9-13
Recovering All Tablespaces	9-14
Performing Disaster Recovery	9-14
Restoring a Database in NOARCHIVELOG Mode	9-14
Restoring Component-Based Backups of a NOARCHIVELOG Database	9-14
Restoring Volume-Based Backups of a NOARCHIVELOG Database	9-15
About Integrating VSS with Third-Party Requester Applications	9-15
Running Writer Control Commands	9-15
Controlling Commands for Database or All Tablespaces Component	9-16
About Duplicating a Database	9-16
Creating a Nonstandby Database from Shadow Copies	9-16
Creating a Standby Database From Shadow Copies	9-17

## 10 Authenticating Database Users with Windows

---

Overview of Windows Native Authentication	10-1
About Windows Authentication Protocols	10-2
About User Authentication and Role Authorization Methods	10-3
About Using Authentication and Authorization Methods	10-3
Overview of Operating System Authentication Enabled at Installation	10-4

## 11 Administering External Users and Roles on Windows

---

Overview of Manually Administering External Users and Roles	11-1
About Manually Creating an External Operating System User	11-3
Performing External User Authentication Tasks on the Oracle Database Server	11-3
Performing External User Authentication Tasks on the Client Computer	11-5
Overview of Manually Granting Administrator, Operator, and Task-Specific Privileges for Databases	11-6
Running System Privilege Authentication Tasks on the Oracle Database Server	11-8
Running System Privilege Authentication Tasks on the Client Computer	11-8
Managing New Users and User Groups	11-9
Overview of Manually Creating an External Role	11-9
Performing External Role Authorization Tasks on the Oracle Database Server	11-10
Performing External Role Authorization Tasks on the Client Computer	11-11



About Manually Migrating Users	11-12
--------------------------------	-------

## 12 Storing Oracle Wallets in the Windows Registry

---

About Storing Private Keys and Trust Points	12-1
About Storing User's Profile	12-1
About Registry Parameters for Wallet Storage	12-1
About Oracle Wallet Manager	12-2
About Sharing Wallets and sqlnet.ora Files Among Multiple Databases	12-3

## 13 Oracle PKI Integration with Windows

---

About Oracle Public Key Infrastructure	13-1
About Windows Public Key Infrastructure	13-1
About Microsoft Certificate Stores	13-2
About Microsoft Certificate Services	13-2
Using Microsoft Certificate Stores with Oracle PKI Applications	13-3

## 14 Using Oracle Database with Microsoft Active Directory

---

Overview of Microsoft Active Directory Support	14-1
About Microsoft Active Directory	14-1
About Accessing Active Directory	14-2
Overview of Oracle Components That Integrate with Active Directory	14-2
About Directory Naming	14-3
About Automatic Discovery of Directory Servers	14-3
About Integration with Windows Tools	14-3
About User Interface Extensions for Oracle Net Directory Naming	14-4
About Enhancement of Directory Object Type Descriptions	14-4
About Integration with Windows Login Credentials	14-4
Overview of Oracle Directory Objects in Active Directory	14-5
Overview of Requirements for Using Oracle Database with Active Directory	14-6
Creating Oracle Schema Objects	14-6
Creating an OracleContext	14-8
Running Oracle Network Configuration Assistant	14-8
About Directory Naming Software Requirements	14-9
Configuring Client Computers and Oracle Database to Use Active Directory	14-10
About Testing Connectivity	14-11
Testing Connectivity from Client Computers	14-11
Testing Connectivity from Microsoft Tools	14-13
Overview of Access Control List Management for Oracle Directory Objects	14-15
Overview of Security Groups	14-16

About OracleDBCreators	14-16
About OracleNetAdmins	14-16
About Oracle Net Services Objects	14-16
Setting ACLs on Net Service Entries	14-17
Adding and Deleting Security Group Members	14-18

## 15 Oracle Database Specifications for Windows

---

Overview of Initialization Parameter File	15-1
About the Location of the Initialization Parameter File	15-2
About Editing The Initialization Parameter File	15-2
About Oracle Database Configuration Assistant Renaming init.ora	15-3
Using Sample File for Database Creation	15-3
About SGA_MAX_SIZE Parameter	15-4
Overview of Initialization Parameters Without Windows-Specific Values	15-4
Displaying Initialization Parameter Values	15-5
About Unmodifiable Database Initialization Parameters	15-5
About Calculating Database Limits	15-6

## 16 Configuration Parameters and the Registry

---

About Configuration Parameters	16-1
Registry Overview	16-1
Registry Parameters Overview	16-2
About HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME	16-2
MSHELP_TOOLS	16-4
NLS_LANG and Other Globalization Parameters	16-4
ORA_CWD	16-5
ORA_SID_AUTOSTART	16-5
ORA_SID_PFILE	16-5
ORA_SID_SHUTDOWN	16-5
ORA_SID_SHUTDOWN_TIMEOUT	16-5
ORA_SID_SHUTDOWNNTYPE	16-6
ORA_TZFILE	16-6
ORACLE_AFFINITY	16-6
ORACLE_BASE	16-8
ORACLE_GROUP_NAME	16-8
ORACLE_HOME	16-8
ORACLE_HOME_KEY	16-8
ORACLE_HOME_USER	16-8
ORACLE_HOMENAME	16-8

ORACLE_PRRITY	16-8
ORACLE_SID	16-9
OSAUTH_PREFIX_DOMAIN	16-9
RDBMS_ARCHIVE	16-9
RDBMS_CONTROL	16-9
SQLPATH	16-9
About HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE	16-9
INST_LOC	16-10
About HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services	16-10
Parameters for Oracle Database Services	16-10
Overview of Oracle RAC Registry Parameters	16-10
About HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OCR	16-11
Managing Registry Parameters with regedit	16-11
Modifying a Parameter Value with regedit	16-11
Adding a Registry Parameter with regedit	16-12

## 17 Developing Applications for Windows

---

About Finding Information on Application Development for Windows	17-1
About Java Enhancements	17-2
About ODP.NET	17-2
About Oracle Developer Tools for Visual Studio	17-3
About Oracle Providers for ASP.NET	17-3
About XML Support	17-3
About Support for Internet Applications	17-4
About Oracle Services for Microsoft Transaction Server	17-4
About Oracle ODBC Driver	17-5
About Pro*C/C++ and Pro*COBOL Applications	17-5
32-bit to 64-bit Application Migration	17-5
About Building External Procedures	17-6
External Procedures Overview	17-6
Installing and Configuring Oracle Database and Oracle Net Services	17-8
Installing Oracle Database	17-8
Configuring Oracle Net Services	17-8
Writing an External Procedure	17-9
Building a DLL	17-10
Registering an External Procedure	17-10
Restricting Library-Related Privileges to Trusted Users Only	17-12
Executing an External Procedure	17-12
Overview of Multithreaded Agent Architecture	17-13
About Debugging External Procedures	17-13

Using Package DEBUG_EXTPROC	17-14
About Accessing Text Files with UTL_FILE	17-14

## A Storing Tablespaces on Raw Partitions

---

Raw Partition Overview	A-1
About Physical Disk	A-2
About Logical Partition	A-2
About Physical Disk and Logical Partition Considerations	A-3
About Compatibility Issues	A-3
Configuring Disks for Oracle Automatic Storage Management	A-4

## B Oracle Net Services Configuration on Windows

---

About Configuring Oracle Database to Communicate with Oracle ASM	B-1
About Modifying Oracle Net Services Registry Parameters and Subkeys	B-2
About Oracle Net Service Subkeys	B-2
About Listener Requirements	B-2
Running Oracle Net Services	B-3
Overview of Optional Configuration Parameters	B-3
About LOCAL Parameter	B-4
About TNS_ADMIN Parameter	B-4
About USE_SHARED_SOCKET Parameter	B-4
Overview of Advanced Network Configuration	B-5
About Configuring Authentication Method	B-5
About Configuring Security for Named Pipes Protocol	B-5
Modifying Configuration of External Procedures for Higher Security	B-5

## C Running Windows Services

---

About Windows Services for Oracle Database	C-1
About Running Windows Services in Oracle Home	C-2
Additional Privileges Required by Oracle Database Services	C-2
Granting Additional Operating System Privileges Manually	C-3

## D Error Messages on Windows

---

ORA-09275: CONNECT INTERNAL No Longer Supported	D-1
ORA-15252 to ORA-15266: User Replacement Failure on Windows	D-2
ORA-15301 to ORA-15302: Failure to Modify Ownership, Group, and Permission of Opened Files	D-3
OSD-04000 to OSD-04599: Windows-Specific Oracle Database Messages	D-3

File I/O Errors: OSD-04000 to OSD-04099	D-6
Memory Errors: OSD-04100 to OSD-04199	D-10
Process Errors: OSD-04200 to OSD-04299	D-11
Loader Errors: OSD-04300 to OSD-04399	D-14
Semaphore Errors: OSD-04400 to OSD-04499	D-15
Miscellaneous Errors: OSD-04500 to OSD-04599	D-15
DIM-00000 to DIM-00228: ORADIM Command Syntax Errors	D-16
Database Connection Issues	D-27

## E Oracle Database Differences on Windows and UNIX

---

Automatic Startup and Shutdown	E-2
Background Processing and Batch Jobs	E-2
Diagnostic and Tuning Utilities	E-3
Direct Writes to Disk	E-3
Dynamic Link Libraries (DLLs)	E-3
Hot Backups	E-4
Initialization Parameters: Multiple Database Writers	E-4
Installation Accounts and Groups	E-4
Oracle Database Installation	E-5
Memory Resources	E-5
Microsoft Transaction Server	E-5
Multiple Oracle Homes and OFA	E-6
Oracle Home User and Oracle User	E-6
Raw Partitions	E-7
Windows Services	E-7

## Index

---

# Preface

This guide provides platform-specific information about administering and configuring Oracle Database 19c on Microsoft Windows 32-bit and 64-bit platforms.



## See Also:

- *Oracle Database Installation Guide for Microsoft Windows* for information about Oracle Database preinstallation requirements
- *Oracle Database Client Installation Guide for Microsoft Windows* for information about Oracle Database Client preinstallation requirements
- *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)* for information about Oracle Grid Infrastructure and Oracle RAC preinstallation requirements

This Preface contains these topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Accessing Documentation](#)
- [Related Documents](#)
- [Conventions](#)

## Audience

This guide is intended for database administrators, network administrators, security specialists, and developers who use Oracle Database for Windows.

To use this document, you need:

- Oracle-certified Windows operating system software installed and tested
- Knowledge of object-relational database management concepts

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Accessing Documentation

The Oracle Database Online Documentation Library for Windows is not shipped with Oracle Database for Windows. Everything in the library is available for download from the Oracle Technology Network (OTN) at <http://www.oracle.com/technetwork/indexes/documentation/index.html>

You must register online before using OTN; registration is free and can be done at <http://www.oracle.com/technetwork/index.html>

## Related Documents

For more information, see the following documents in the Oracle Database documentation set:

- *Oracle Database Installation Guide for Microsoft Windows*
- *Oracle Database Platform Guide for Microsoft Windows*
- *Oracle Database Concepts*

Many books in the documentation set use the sample schemas of the seed database, which is installed by default when you install Oracle Database. See *Oracle Database Sample Schemas* for information about how these schemas were created and how you can use them yourself.

## Conventions

The following text conventions are used in this document:

Convention	Meaning
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

# Changes in This Release for Oracle Database Administrator's Reference Guide

Changes in Oracle Database Administrator's Reference Guide for Oracle Database 19c.

- [Changes in Oracle Database Release 19c](#)  
New features, deprecated features, and desupported features in this release.

## Changes in Oracle Database Release 19c

New features, deprecated features, and desupported features in this release.

The following are changes in Oracle Database Administrator's Reference Guide for Oracle Database 19c:

- [Desupported Features for Oracle Database 19c](#)  
There are no desupported features at the time of release.
- [Deprecation of Oracle Fail Safe](#)  
Oracle Fail Safe is deprecated as of Oracle Database 19c. It can be desupported and unavailable in a future release.

## Desupported Features for Oracle Database 19c

There are no desupported features at the time of release.

## Deprecation of Oracle Fail Safe

Oracle Fail Safe is deprecated as of Oracle Database 19c. It can be desupported and unavailable in a future release.

Oracle recommends that you evaluate other single-node failover options, such as Oracle RAC One Node.



# 1

## Oracle Database Architecture on Windows

Learn how Oracle Database architecture takes advantage of some more advanced services in Microsoft Windows operating systems.

- [Overview of Oracle Database on Windows Architecture](#)  
Oracle Database on Windows is a stable, reliable, and a high-performing system upon which you can build applications. Each release of the database provides new platform-specific features for high performance on Windows.
- [Overview of Oracle Database Scalability on Windows](#)  
Features in Oracle Database and in the Windows operating system work together to help increase scalability, throughput, and database capacity.
- [Oracle Database Integration with Windows](#)  
Oracle Database is increasingly integrated with Windows, easing maintenance and improving enterprise-level deployment in security, directory, and transaction services.

### Overview of Oracle Database on Windows Architecture

Oracle Database on Windows is a stable, reliable, and a high-performing system upon which you can build applications. Each release of the database provides new platform-specific features for high performance on Windows.

Oracle Database operates the same way on Windows as it does on the other platforms.

- [Oracle Automatic Storage Management](#)  
Oracle Automatic Storage Management (Oracle ASM) is an integrated file system and volume manager expressly built for Oracle Database files.
- [Oracle Automatic Storage Management File Access Control](#)  
Oracle ASM File Access Control restricts the access of files to specific Oracle ASM clients that connect as `SYSDBA`.
- [Thread-Based Architecture](#)  
The internal process architecture of Oracle Database is thread-based. Threads are objects within a process that run program instructions.
- [File I/O Enhancements](#)  
Oracle Database supports 64-bit file I/O to allow the use of files larger than 4 gigabytes (GB).

### Oracle Automatic Storage Management

Oracle Automatic Storage Management (Oracle ASM) is an integrated file system and volume manager expressly built for Oracle Database files.

Oracle ASM provides the performance of raw I/O with the easy management of a file system. It simplifies database administration by eliminating the need for you to directly manage potentially thousands of Oracle Database files. It enables you to divide all

available storage into disk groups. You manage a small set of disk groups, and Oracle ASM automates the placement of the database files within those disk groups.

Oracle recommends that you use Oracle ASM instead of raw files to store data files. It provides the performance benefits of raw files with much better manageability. Oracle ASM is available for both single instance and Oracle Real Application Clusters (Oracle RAC) databases.

You can store Oracle Cluster Registry and voting files in Oracle ASM disk groups and store database data files in the `data` disk group. The voting files and Oracle Cluster Registry are two important components of Oracle Clusterware.

 **Note:**

- You must be logged on either as an Administrator or a user name that is a member of the Administrators group.
- To open Disk Management console, click **Start, Run**, and then enter: `diskmgmt.msc`.
- Storing data files on raw devices is no longer supported. You must use a file system or Oracle Automatic Storage Management.
- NFS or Direct NFS cannot be used for Oracle Clusterware files.

**Related Topics**

- [Storing Tablespaces on Raw Partitions](#)

 **See Also:**

- *Oracle Automatic Storage Management Administrator's Guide*
- *Oracle Database Installation Guide for Microsoft Windows*

## Oracle Automatic Storage Management File Access Control

Oracle ASM File Access Control restricts the access of files to specific Oracle ASM clients that connect as `SYSDBA`.

An Oracle ASM client is a database, which is identified by the name of the user that owns the database instance home. Oracle ASM File Access Control uses this user name to identify a database. Oracle ASM File Access Control restricts access based on the operating system and effective user identification number of a database owner.

- [Creation of New User Groups and Users for Separation of Database Administration Duties](#)  
Oracle Database 19c provides access control to separate the roles on Windows.
- [About Disk Group User Replacement](#)  
The identity of an Oracle ASM user can be changed from one operating system user to another operating system user.

- [About Changing File Access Control While the File is Open](#)  
Oracle Database 19c enables users to change the ownership, permissions, or group membership of a file even while the file is open.

## Creation of New User Groups and Users for Separation of Database Administration Duties

Oracle Database 19c provides access control to separate the roles on Windows.

With Oracle Database services running under the Oracle Home User account instead of the Local System Account, the Oracle ASM access control feature must be enabled to support role separation on Windows. In previous releases, this feature was disabled on Windows because all Oracle Database services ran under Windows Built-in Local System Account.

The new user groups added are `ORA_HOMENAME_DBA`, `ORA_HOMENAME_OPER`, `ORA_HOMENAME_SYSBACKUP`, and so on. For Oracle ASM administration, new groups `ORA_ASMADMIN`, `ORA_ASMDBA` and `ORA_ASMOPER` are automatically created and populated during Oracle Database installation. The Oracle ASM administrator can manage these Windows groups using Windows tools, though you must ensure that the required user names are not removed from these groups.

### Related Topics

- [Managing New Users and User Groups](#)  
Learn how to manage new users and user groups.

#### See Also:

- *Oracle Database Installation Guide for Microsoft Windows*
- *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*

## About Disk Group User Replacement

The identity of an Oracle ASM user can be changed from one operating system user to another operating system user.

It enables end users to change the identity of an Oracle ASM user without having to delete and re-create the user, which requires dropping all the files a user owns. This feature improves the manageability of Oracle ASM users and the files they own. The SQL Statements `ALTER`, `DISKGROUP`, `REPLACE`, `USER`, and a new `ASMCMD` command (`rpusr`) have been added to support user replacement in a disk group.

#### See Also:

- *Oracle Automatic Storage Management Administrator's Guide*

## About Changing File Access Control While the File is Open

Oracle Database 19c enables users to change the ownership, permissions, or group membership of a file even while the file is open.

Since this release, the ASMCMD file access control commands, such as `chgrp`, `chmod`, and `chown`, can run even while the file is open. The SQL statements, such as `ALTER`, `DISKGROUP`, `MODIFY`, and `USERGROUP` commands have also been modified as these SQL statements provide support for these ASMCMD commands.



### See Also:

*Oracle Automatic Storage Management Administrator's Guide*

## Thread-Based Architecture

The internal process architecture of Oracle Database is thread-based. Threads are objects within a process that run program instructions.

Threads allow concurrent operations within a process so that a process can run different parts of its program simultaneously on different processors. A thread-based architecture provides the following advantages:

- Faster context switching
- Simpler System Global Area allocation routine, because it does not require use of shared memory
- Faster spawning of new connections, because threads are created more quickly than processes
- Decreased memory usage, because threads share more data structures than processes

Internally, the code to implement the thread model is compact and separate from the main body of Oracle Database code. Exception handlers and routines track and deallocate resources. They add robustness, with no downtime because of resource leaks or program that does not function as expected.

Oracle Database is not a typical Windows process. On Windows, an Oracle Database or Oracle Automatic Storage Management instance (threads and memory structures) is a Windows service: a background process registered with the operating system. The service is started by Windows and requires no user interaction to start. This enables the database to open automatically at computer startup.

When running multiple Oracle Database or Oracle Automatic Storage Management instances on Windows, each instance runs its own Windows service with multiple component threads. Each thread is required for the database to be available, or is optional and specific to certain platforms. The background processes read and write from various data files, depending on your configuration. Oracle Database architecture on Windows is illustrated in Oracle Database Architecture on Windows. Examples of Oracle Database required threads on Windows are listed in Oracle Database Threads.

Figure 1-1 Oracle Database Architecture on Windows

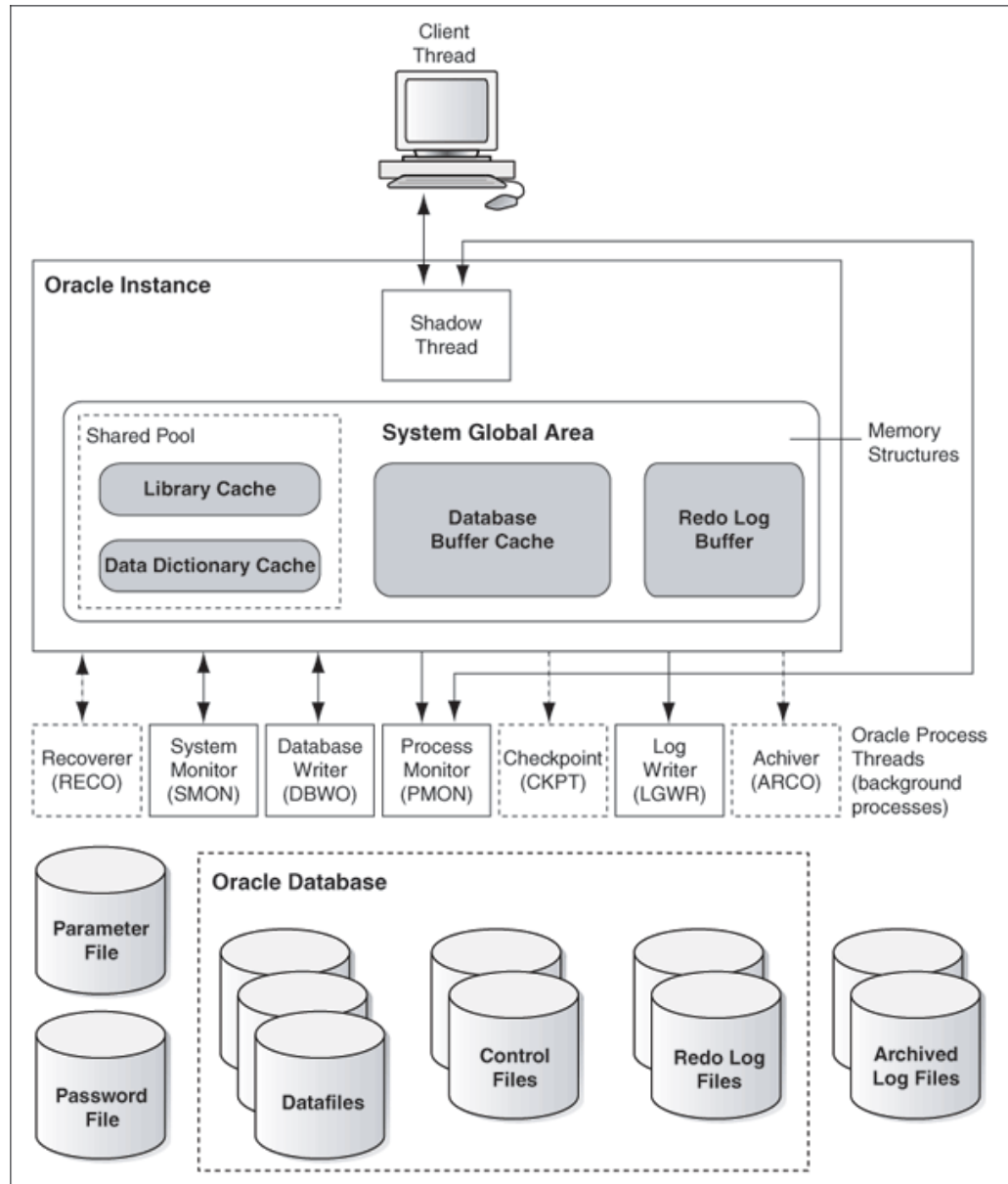


Table 1-1 Oracle Database Threads

Oracle Database Thread	Description	Required/Optional
DBWO	database writer	Required
LGWR	log writer	Required
MMAN	memory manager process	Required
PMON	process monitor	Required
PSPO	process spawner process	Required
SMON	system monitor	Required

**Table 1-1 (Cont.) Oracle Database Threads**

Oracle Database Thread	Description	Required/ Optional
CKPT	checkpoint process (thread on Windows) that runs by default on Windows	Required
ARCO	archive process (or thread on Windows)	Required
RECO	distributed recovery background process	Required

**Note:**

You can view running background processes by entering the following query:

```
SQL> select * from v$bgprocess where paddr <> '00';
```

Oracle Database for Windows is supplied as a set of executables and dynamic link libraries (DLLs). Executable images can be modified using `ORASTACK` to change the size of the stack used by the threads of the Oracle Database process. Oracle recommends that you use this tool only under the guidance of Oracle Support Services.

## File I/O Enhancements

Oracle Database supports 64-bit file I/O to allow the use of files larger than 4 gigabytes (GB).

In addition, physical and logical raw files are supported as data, log, and control files to support Oracle Real Application Clusters (Oracle RAC) on Windows and for those cases where performance must be maximized.

Instead of using the operating system kernel NFS client, you can configure Oracle Database to access NFS V3 servers directly using an Oracle internal Direct NFS client. Through this integration, Oracle can optimize the I/O path between Oracle and the NFS server, resulting in a significantly superior performance. In addition, Direct NFS client simplifies and optimizes the NFS client configuration for database workloads.

Now, the Direct NFS client supports all widely accepted NFS path formats, including both Windows-style and UNIX-style NFS paths.

Volumes mounted through CIFS cannot be used for storing Oracle database files without configuring the Direct NFS client. The atomic write requirements needed for database writes are not guaranteed through the CIFS protocol. Consequently, CIFS can be used only for operating system-level commands such as, copy, move, and so on.

The Direct NFS client currently supports up to four parallel network paths to provide scalability and high availability. The Direct NFS client delivers optimized performance by automatically load balancing requests across all specified paths. If one network path fails, then the Direct NFS client resends commands over any remaining paths ensuring fault tolerance and high availability.

A new parameter called `dnfs_batch_size` is now added to control the number of asynchronous I/O operations that can be queued by an Oracle process when the Direct NFS client is enabled. Set this parameter only if the Direct NFS client is overwhelming the NFS server or the network. This parameter helps the user to manage the load that the Direct NFS client can generate. In typical environments, you must not set this parameter. The default value of this parameter is 4096. To reduce the Direct NFS client load, Oracle recommends a value of 128 that can be changed based on the NFS server performance.

 **See Also:**

Your vendor documentation to complete NFS configuration and mounting

## Overview of Oracle Database Scalability on Windows

Features in Oracle Database and in the Windows operating system work together to help increase scalability, throughput, and database capacity.

- [Large User Populations](#)  
Several features allow Oracle Database to support an increasingly large number of database connections on Windows.

### Large User Populations

Several features allow Oracle Database to support an increasingly large number of database connections on Windows.

- The Shared Server Process, limits the number of threads needed in the Oracle Database process and supports over 10,000 simultaneous connections to a single database instance.
- Oracle Net multiplexing and connection pooling features allow a large configuration to connect more users to a single database instance.
- Oracle RAC raises connection counts dramatically by allowing multiple server computers to access the same database files, increasing the number of user connections by tens of thousands while increasing throughput.

 **Note:**

Oracle RAC is only supported on 64-bit Windows server operating systems.

## Oracle Database Integration with Windows

Oracle Database is increasingly integrated with Windows, easing maintenance and improving enterprise-level deployment in security, directory, and transaction services.

- [Oracle PKI Integration with Windows](#)  
Oracle Advanced Security includes Oracle PKI (public key infrastructure) integration for authentication and single sign-on. You can integrate Oracle-based applications with the PKI authentication and encryption framework, using Oracle Wallet Manager.
- [Oracle Fail Safe Integration with Windows](#)  
Oracle Fail Safe ensures that Oracle Database (and also other Oracle and third-party applications) can be configured and managed for high availability on Windows clusters. An instance runs on only one node at a time.
- [Oracle Services for Microsoft Transaction Server](#)  
Microsoft Transaction Server (MTS) is used in the middle tier as an application server for COM objects and transactions in distributed environments.

## Oracle PKI Integration with Windows

Oracle Advanced Security includes Oracle PKI (public key infrastructure) integration for authentication and single sign-on. You can integrate Oracle-based applications with the PKI authentication and encryption framework, using Oracle Wallet Manager.

## Oracle Fail Safe Integration with Windows

Oracle Fail Safe ensures that Oracle Database (and also other Oracle and third-party applications) can be configured and managed for high availability on Windows clusters. An instance runs on only one node at a time.

A cluster is a group of independent computing systems that operates as a single virtual system, eliminating individual host systems as points of failure. Oracle Fail Safe works with Microsoft Windows Failover Clusters to ensure that if a failure occurs on one cluster system, then workloads running on that system fail over quickly and automatically to a surviving system. Oracle Database combined with Oracle Fail Safe on a Windows cluster ensures protection from both hardware and software failures.

### Note:

Windows server operating systems support the clustering technology. You can install Oracle Fail Safe Manager client on Windows Server operating systems (such as Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2 x64) and on client systems (such as Windows 7, Windows 8, and Windows 8.1).

### See Also:

The Oracle Fail Safe documentation set, which is available on a separate media in the Oracle Database media pack



## Oracle Services for Microsoft Transaction Server

Microsoft Transaction Server (MTS) is used in the middle tier as an application server for COM objects and transactions in distributed environments.

Oracle Services for Microsoft Transaction Server allows Oracle Database to be used as a resource manager in Microsoft Transaction Server-coordinated transactions, providing strong integration between Oracle solutions and Microsoft Transaction Server. Oracle Services for Microsoft Transaction Server can operate with Oracle Database running on any operating system.

Oracle Database takes advantage of a native implementation and also stores recovery information in Oracle Database itself. Oracle Services for Microsoft Transaction Server allows development in all industry wide data access interfaces, including Oracle Call Interface (OCI), ActiveX Data Objects (ADO), ADO.NET, OLE DB, and Open Database Connectivity (ODBC). The Oracle APIs, Oracle Data Provider for .NET and OCI, offer greatest efficiency.

# 2

## Database Tools on Windows

Oracle Database for Windows includes various tools to perform database functions, describes the preferred tools to perform common database administration tasks, and explains how the tools can be started.

Unless otherwise noted, features described in this guide are common to all Oracle Database editions.

- [Choosing a Database Tool](#)  
Database tools is a collective term for tools, utilities, and assistants that you can use to perform database administration tasks.
- [Starting Database Tools](#)  
Oracle Database 19c mandates that the administrator starting all the administration tools such as Oracle Database Configuration Assistant, Oracle Database Upgrade Assistant, Oracle Net Configuration Assistant, and Oracle ASM Configuration Assistant, must be an operating system administrator.
- [Using the Oracle Home User Control Tool](#)  
A new Windows tool, Oracle Home User Control, is a command-line tool that displays the Oracle Home User name associated with the current Oracle home and updates the password for the Windows services for the Oracle home.
- [Using Windows Tools](#)  
You can use Windows tools in the following ways to manage Oracle Database:
- [Using SQL\\*Loader](#)  
Describes Windows-specific information for using SQL\*Loader (SQLLDR).

### Choosing a Database Tool

Database tools is a collective term for tools, utilities, and assistants that you can use to perform database administration tasks.

Some database tools perform similar tasks, though no one database tool performs all the database administration tasks.

- [Database Tools and Operating System Compatibility](#)  
Almost all the database tools are available on all supported versions of Windows.
- [Preferred Database Tools](#)  
Lists the various database tools you can use to perform common database administration tasks.

### Database Tools and Operating System Compatibility

Almost all the database tools are available on all supported versions of Windows.

The exception is Oracle SQL Developer is available only at Oracle Technology Network (OTN). See

<http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index.html>

## Preferred Database Tools

Lists the various database tools you can use to perform common database administration tasks.

Oracle recommends you use tools listed in the "Preferred Tool" column of the table. After choosing a tool to perform a task, go to Preferred Database Tools from the Start Menu, for instructions on how to start the tool.

### Note:

The `VOLSIZE` parameter for the Export and Import utilities is not supported on Windows. If you attempt to use the utilities with the `VOLSIZE` parameter, then error LRM-00101 occurs. For example:

```
D:\> exp system full=y volsize=100m;
Password: password
LRM-00101: unknown parameter name 'volsize'
EXP-00019: failed to process parameters, type 'EXP HELP=Y' for help
EXP-00000: Export terminated unsuccessfully
```

**Table 2-1 Preferred Database Tools**

Administration Task	Preferred Tool	Other Tools
Create database services	Database Configuration Assistant	ORADIM
Delete database services	Database Configuration Assistant	ORADIM
Change passwords in the database password file	ORAPWD	ORADIM
Update the password of an Oracle Home User	Oracle Home User Control	None
Migrate database users to a directory	User Migration Utility	None
Migrate a database	Oracle Database Upgrade Assistant	Upgrade Information Tool
Export data	Data Pump Export (EXPDP)	Export (EXP)
Import data	Data Pump Import (IMPDP)	Import (IMP)
Load data	Oracle Enterprise Manager Load Wizard	SQL*Loader (SQLLDR)
Back up database	Oracle Enterprise Manager Backup Wizard	Recovery Manager (RMAN) OCOPY
Recover database	Oracle Enterprise Manager Recovery Wizard	Recovery Manager (RMAN) OCOPY
Store encrypted and decrypted Oracle Wallet (Oracle Advanced Security and security PKI integration)	Oracle Wallet Manager	None

Table 2-1 (Cont.) Preferred Database Tools

Administration Task	Preferred Tool	Other Tools
Grant database roles	Oracle Enterprise Manager Database Express	Local Users and Groups SQL*Plus
Create database objects	Oracle Enterprise Manager Cloud Control	SQL*Plus

- ORADIM can set a password only when none was previously set. If a password has been previously set, then ORADIM can change it only by deleting and re-creating Oracle Database services. Now, ORADIM creates the Oracle Database service, Oracle VSS Writer service, and Oracle Scheduler service to run under the Oracle Home User account. If this account is a Local or a Domain User Account, then ORADIM prompts for the password for that account and accepts the same through `stdin`.

It is possible to specify both the Oracle Home User and its password using the `-RUNAS osusr[/ospass]` option to `oradim`. If the given `osusr` is different from the Oracle Home User, then the Oracle Home User is used instead of `osusr` along with the given `ospass`.

- User Migration Utility can migrate local or external users to enterprise users. For more information, see "Using the User Migration Utility" in *Oracle Database Enterprise User Security Administrator's Guide*.
- Oracle Database 11g Release 2 (11.2), Oracle Database 12c Release 1 (12.1), and Oracle Database 12c Release 2 (12.2) are supported for Oracle Database Upgrade Assistant upgrades in Oracle Database 19c. Oracle Database Upgrade Assistant can also be used to apply patch sets.
- Data Pump Export and Data Pump Import are preferred for Oracle Database 10g Release 1 (10.1) and later data; Export and Import are preferred for earlier data.
- If you back up files while you are shutting down the database, then your backup is invalid. You cannot use an invalid backup to restore files at a later date.
- You cannot use earlier versions of Oracle Wallet Manager to manage Oracle Database 10g Release 1 (10.1) and later wallets that contain password-based credentials for authentication to Oracle Internet Directory. These credentials are placed in the wallet when an Oracle Database server is registered in Oracle Internet Directory.

The database wallet that Oracle Database Configuration Assistant automatically generates during database registration can be used only with Oracle Database 10g Release 1 (10.1) or later. You cannot use this database wallet for earlier versions of the database, nor can you use it for Oracle Internet Directory Release 9.0.4 or earlier.

 **See Also:**

- *Oracle Database Enterprise User Security Administrator's Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Database Upgrade Guide*

## Starting Database Tools

Oracle Database 19c mandates that the administrator starting all the administration tools such as Oracle Database Configuration Assistant, Oracle Database Upgrade Assistant, Oracle Net Configuration Assistant, and Oracle ASM Configuration Assistant, must be an operating system administrator.

The administrator must also be a member of the `ORA_DBA` and `ORA_ASMADMIN` group for using the Oracle Database Configuration Assistant and Oracle Database Upgrade Assistant tools when accessing Oracle ASM. The administrator must be a member of the `ORA_ASMADMIN` group for using the Oracle ASM Configuration Assistant tool.

Oracle needs the password of Oracle Home User to create new Windows services for Database, Listener, and other entities. To support this, all the administration tools have been modified to prompt for the password of Oracle Home User that is required only when the Oracle Home User is a Local or a Domain Windows User Account and the password for the Oracle Home User is not stored in the Oracle Wallet.

- [Starting Database Tools in Multiple Oracle Homes](#)  
You can start database tools in multiple Oracle homes.
- [Running Tools with Windows User Account Control](#)  
You must ensure that only trusted applications run on your computer.
- [Starting Database Tools from the Start Menu](#)  
Describes how to start assistants and other tools from the **Start menu**.
- [Starting Database Tools from the Command Line](#)  
Describes how to start Oracle Database tools from the command line, and where to go for further information on using these products.
- [Starting Windows Tools](#)  
Describes how to start each Windows tool and where to go for more information on using these products.

## Starting Database Tools in Multiple Oracle Homes

You can start database tools in multiple Oracle homes.

If you have multiple Oracle homes on your computer from previous releases, then see Appendix B, "Optimal Flexible Architecture" in *Oracle Database Installation Guide for Microsoft Windows* for a description of differences between Oracle homes in different releases.

- [Starting Tools from Multiple Oracle Homes](#)  
Each Oracle home, including the first Oracle home you create on your computer, has a unique `HOMENAME`.

## Starting Tools from Multiple Oracle Homes

Each Oracle home, including the first Oracle home you create on your computer, has a unique *HOMENAME*.

## Running Tools with Windows User Account Control

You must ensure that only trusted applications run on your computer.

To ensure that only trusted applications run on your computer, all Windows operating systems supported for Oracle Database 19c provide User Account Control. If you have enabled this security feature, then, depending on how you have configured it, Oracle Universal Installer prompts you for either your consent or your credentials when installing Oracle Database Client. Provide either the consent or your Windows Administrator credentials as appropriate.

You must have the Administrator privileges to run some configuration tools, or to run any tool or application that writes to any directory within the Oracle home. If User Account Control is enabled, and you are logged in as the local Administrator, then you can successfully run each of these commands in the usual way. However, if you are logged in as a member of the Administrator group, then you must explicitly run these tasks with Windows Administrator privileges.

The following tools must be run with Administrator privileges:

- Oracle Net Configuration Assistant. This tool is available as a Configuration and Migration Tool.
- Oracle OLAP Analytic Workspace Manager and Worksheet. This tool is available as an Integrated Management Tool.
- Oracle Database Configuration Assistant. This tool is available as a Configuration and Migration Tool.
- Oracle Database Wallet Manager. This tool is available as an Integrated Management Tool.
- Oracle Database Upgrade Assistant. This tool is available as a Configuration and Migration Tool.
- Oracle Net Manager. This tool is available as a Configuration and Migration Tool.
- Oracle ASM Configuration Assistant. This tool is available as a Configuration and Migration Tool.
- Oracle ASM Disk Stamping Tool (`asmtool`, `asmtoolg`). This tool is available as a Configuration and Migration Tool.

To run any **Start** menu tool with Administrator privileges:

1. Click the **Start** menu option.
2. Select **All Programs**, then select **Oracle - HOMENAME**.
3. Select the name of the tool, then right-click the name of the tool or application you want to run, and then select **Run as administrator**.

These steps describe how to start a tool as an Administrator from the command prompt:

1. Create a shortcut for the command prompt window on your desktop. An icon for that shortcut appears on the desktop.
2. Right-click the icon for the newly created shortcut, and specify **Run as administrator**.

When you open this window, the title bar reads Administrator: Command Prompt. Commands running within this window are run with Administrator privileges.

 **See Also:**

*Oracle Database Installation Guide for Microsoft Windows*

## Starting Database Tools from the Start Menu

Describes how to start assistants and other tools from the **Start menu**.

It also directs you about further information on using these products.

 **Note:**

When you use an assistant, you must have read and write access to the directory where database files are created or moved to. To create an Oracle Database instance, you must have the administrator privilege.

 **Note:**

All Start menu paths begin with the **Start** menu where you select **All Programs**, then select **Oracle - HOMENAME** and so on.

**Table 2-2 Starting Database Tools from the Start Menu**

Tool	Start Menu Path	More Information
Microsoft ODBC Administrator	From <b>Configuration and Migration Tools</b> , select <b>Microsoft ODBC Administration</b>	Microsoft ODBC Administration online help
Oracle Automatic Storage Management Configuration Assistant	From <b>Configuration and Management Tools</b> , select <b>Automatic Storage Management Configuration Assistant</b>	<i>Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)</i>
Oracle Database Configuration Assistant	From <b>Configuration and Migration Tools</b> , select <b>Database Configuration Assistant</b>	
Oracle Directory Manager	From <b>Integrated Management Tools</b> , select <b>Oracle Directory Manager</b>	<i>Oracle Internet Directory Administrator's Guide</i>

**Table 2-2 (Cont.) Starting Database Tools from the Start Menu**

Tool	Start Menu Path	More Information
Oracle Locale Builder	From <b>Configuration and Migration Tools</b> , select <b>Locale Builder</b>	<i>Oracle Database Globalization Support Guide</i>
Oracle Net Configuration Assistant	From <b>Configuration and Migration Tools</b> , select <b>Net Configuration Assistant</b>	<i>Oracle Database Net Services Administrator's Guide</i>
Oracle Net Manager	From <b>Configuration and Migration Tools</b> , select <b>Net Manager</b>	<i>Oracle Database Net Services Administrator's Guide</i>
Oracle Wallet Manager	From <b>Integrated Management Tools</b> , select <b>Wallet Manager</b>	<i>Oracle Database Enterprise User Security Administrator's Guide</i>
SQL*Plus	From <b>Application Development</b> , select <b>SQL*Plus</b>	<i>SQL*Plus User's Guide and Reference</i>

## Starting Database Tools from the Command Line

Describes how to start Oracle Database tools from the command line, and where to go for further information on using these products.

**Table 2-3 Starting Database Tools from the Command Line**

Tool	Enter at Prompt	More Information
Oracle ASM Disk Stamping Tool	<ul style="list-style-type: none"> <li>C:\&gt; asmtool</li> </ul>	"Marking Disk Partitions for Oracle ASM Before Installation" in <i>Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)</i>
Oracle ASM Disk Stamping Tool (GUI version)	<p>Following are the list of options:</p> <ul style="list-style-type: none"> <li>C:\&gt; asmtool -add</li> <li>C:\&gt; asmtool -addprefix</li> <li>C:\&gt; asmtool -list</li> <li>C:\&gt; asmtool -delete</li> </ul>	
	<ul style="list-style-type: none"> <li>C:\&gt; asmtoolg</li> </ul>	
	<p><b>Note:</b> asmtoolg is the GUI-based tool that performs the same actions as the command-line asmtool tool.</p>	
DBVERIFY	<p>C:\&gt; dbv</p> <p>DBVERIFY starts and prompts you for a file name parameter. To obtain a list of parameters, enter:</p> <p>C:\&gt; dbv help=y</p>	<i>Oracle Database Utilities</i>
Data Pump Export	<p>C:\&gt; expdp user name</p> <p>EXP starts and prompts you for parameters. To obtain a list of these parameters, enter:</p> <p>C:\&gt; exp help=y</p>	<p><i>Oracle Database Utilities</i> for instructions on use of Data Pump Export</p> <p><i>Oracle Database Error Messages Reference</i> for information on error messages</p>
Data Pump Import	<p>C:\&gt; impdp user name</p> <p>IMP starts and prompts you for parameters. To get a list of these parameters, enter:</p> <p>C:\&gt; imp help=y</p>	<p><i>Oracle Database Utilities</i> for instructions on use of Data Pump Import</p> <p><i>Oracle Database Error Messages</i> for information on error messages</p>



Table 2-3 (Cont.) Starting Database Tools from the Command Line

Tool	Enter at Prompt	More Information
Database Configuration Assistant	C:\> dbca Oracle Database Configuration Assistant tool starts in interactive mode. For silent options and other command-line options, enter: C:\> dbca -help	"Starting DBCA" in <i>Oracle Database 2 Day DBA</i>
Database Upgrade Assistant	C:\> dbua Oracle Database Upgrade Assistant wizard starts in interactive mode. For silent options and other command line options enter: C:\> dbua -help	<i>Oracle Database Upgrade Guide</i>
Export	C:\> exp user name EXP starts and prompts you for parameters. To obtain a list of these parameters, enter: C:\> exp help=y	<i>Oracle Database Utilities</i> for instructions on use of Export <i>Oracle Database Error Messages</i> for information on error messages
Import	C:\> imp user name IMP starts and prompts you for parameters. To get a list of these parameters, enter: C:\> imp help=y	<i>Oracle Database Utilities</i> for instructions on use of Import <i>Oracle Database Error Messages</i> for information on error messages
Net Services Configuration	C:\> netca Oracle Net Configuration Assistant tool starts in interactive mode. For silent options and other command-line options, enter: C:\> netca -help	<i>Oracle Database Upgrade Guide</i>
ORADIM	C:\> oradim options To get a list of ORADIM options, enter either of the following: C:\> oradim C:\> oradim -?   -h   -help	
Oracle Wallet Manager	C:\> cd ORACLE_HOME\bin C:\ORACLE_HOME\bin> launch.exe ORACLE_HOME\bin owm.cl	
Password Utility (ORAPWD)	C:\> orapwd Password file is hidden. Use Windows Explorer to see it in a file list. From the <b>View</b> menu, select <b>Options</b> , then select <b>View</b> and then select <b>Show All Files</b> .	
Recovery Manager (RMAN)	C:\> rman parameters	<i>Oracle Database Backup and Recovery User's Guide</i>
SQL*Plus (SQLPLUS)	C:\> sqlplus	<i>SQL*Plus User's Guide and Reference</i>
SQL*Loader (SQLLDR)	C:\> sqlldr SQL*Loader displays a Help screen with available keywords and default values.	<i>Oracle Database Utilities</i> <i>Oracle Database Error Messages</i> <a href="#">"Starting Windows Tools"</a>

**Table 2-3 (Cont.) Starting Database Tools from the Command Line**

Tool	Enter at Prompt	More Information
TKPROF	C:\> tkprof	<i>Oracle Database SQL Tuning Guide</i>
User Migration Utility	C:\> umu <i>parameters</i> To get a list of parameters, enter: C:\> umu help=yes	"Using the User Migration Utility" in <i>Oracle Database Enterprise User Security Administrator's Guide</i>

 **Note:**

- Three special conditions apply when running Export or Import utilities on Windows. First, default values for `BUFFER` and `RECORDLENGTH` parameters are 4 KB and 2 KB respectively. This default `RECORDLENGTH` parameter does not depend on the value of `BUFSIZ` defined in the system header file. If you specify a value larger than `USHRT_MAX` (64 KB), you get a warning message. Second, the `VOLSIZE` parameter is not supported. Third, to export an entire database, you must use the `EXP_FULL_DATABASE` role.
- Oracle Enterprise Manager Database Express is another database tool for managing the database. For information about logging in to Oracle Enterprise Manager Database Express, see "Configuring the HTTP Port for EM Express" in *Oracle Database 2 Day DBA*.
- [About Archiving Redo Log Files](#)  
If you installed Oracle Database through the Typical installation, then it is created in the `NOARCHIVELOG` mode. If you created your database through the Custom option of Oracle Database Configuration Assistant, then you had the choice of either `ARCHIVELOG` or `NOARCHIVELOG`.

## About Archiving Redo Log Files

If you installed Oracle Database through the Typical installation, then it is created in the `NOARCHIVELOG` mode. If you created your database through the Custom option of Oracle Database Configuration Assistant, then you had the choice of either `ARCHIVELOG` or `NOARCHIVELOG`.

In `NOARCHIVELOG` mode, redo logs are not archived. Setting your archive mode to `ARCHIVELOG` and enabling automatic archiving causes redo log files to be archived. This protects Oracle Database from both instance and disk failure.

 **See Also:**

*Oracle Database Administrator's Guide* for more information about "Managing Archived Redo Logs."

## Starting Windows Tools

Describes how to start each Windows tool and where to go for more information on using these products.

**Table 2-4 Starting Windows Tools**

Tool	Start Procedure	More Information
Event Viewer	From the <b>Start</b> menu, select <b>All Programs</b> , then select <b>Administrative Tools</b> and then select <b>Event Viewer</b> .	Your operating system documentation
Local Users and Groups	From the <b>Start</b> menu, select <b>Settings</b> , then select <b>Control Panel</b> . Double-click <b>Administrative Tools</b> . Double-click <b>Computer Management</b> . In the console tree, click <b>Local Users and Groups</b> .	Your operating system documentation
Microsoft Management Console (MMC)	From the <b>Start</b> menu, select <b>All Programs</b> , then select <b>Oracle - HOMENAME</b> , then select <b>Configuration and Migration Tools</b> and then select <b>Administration Assistant for Windows</b> .	Your operating system documentation
Registry Editor	At the command prompt, enter:  C:\> regedit	Your operating system documentation
Task Manager	Right-click the Task bar and select <b>Task Manager</b> .	Your operating system documentation



### Note:

Microsoft Management Console is started whenever Oracle Administration Assistant for Windows is started.

## Using the Oracle Home User Control Tool

A new Windows tool, Oracle Home User Control, is a command-line tool that displays the Oracle Home User name associated with the current Oracle home and updates the password for the Windows services for the Oracle home.

The input password must match the password for the Windows User Account used as the Oracle Home User. So, first use Windows operating system tools to change the Windows password and then use this tool. This tool updates all Windows services used by Oracle to use the new password and updates it in the Oracle Cluster Registry wallet too, if one exists.

Installer also creates a shortcut **Update Password for Oracle Home User** which starts the tool.

The Oracle Home User Control tool accepts the new password at the tool's prompt for password entry and validates the password provided against the password of the Windows User Account. The tool terminates if password validation fails. Moreover, the user starting the `orahomeuserctl` command must have Administrator privileges. The command must be in the following format:

```
orahomeuserctl list
orahomeuserctl updpwd [-user username] [-host hostname1, hostname2, ...] [-log
logfile]
```

For this command, note the following:

- `list`: This utility displays the Oracle Home User name associated with the Oracle home.
- `updpwd`: This utility prompts for a new password and updates the password for all Oracle Database services associated with the named Oracle Home User on the node. When `updpwd` is started on a node within an Oracle RAC installation, then the command first updates the Oracle Cluster Registry wallet with the new password, then updates all Oracle Database services associated with the user on all active nodes within a cluster. If there is no Oracle Cluster Registry Wallet, then the utility updates only all the Oracle Database services.
- `-user`: This option updates the passwords for all services owned by a specific user, or the password of the current Oracle Home User if no user is specified.
- `-host`: This option updates the passwords for all services belonging to the named Oracle Home User on the specified hosts. To update the password on a remote host, the user must be a Windows Domain User.
- `-log`: This option appends the time-stamped results of the password update action to the specified log file for every node and service name receiving the new password. The default log file name and location is `%ORACLE_HOME%\log\orahomeuserctl.log`.

## Using Windows Tools

You can use Windows tools in the following ways to manage Oracle Database:

- [Using Event Viewer to Monitor a Database](#)  
Event Viewer lets you monitor events in your system. An event is an important occurrence in the system or application (such as Oracle Database) that requires user notification.
- [Using Microsoft Management Console to Administer a Database](#)  
Microsoft Management Console provides a central location for network administration.
- [Using Registry Editor to Modify Configuration Information](#)  
Oracle Database stores its configuration information in a structure known as the registry.
- [Using Task Manager to Monitor Applications and Processes](#)  
Task Manager monitors applications and processes.

- [Using Local Users and Groups to Manage Users and Groups](#)  
Local Users and Groups enable you to manage users and groups on Windows.

## Using Event Viewer to Monitor a Database

Event Viewer lets you monitor events in your system. An event is an important occurrence in the system or application (such as Oracle Database) that requires user notification.

While messages for major events can appear on-screen as you work at your computer, events that do not require your immediate attention are recorded by Windows in the Event Viewer log file. You can then view this information at your convenience.

Use Event Viewer to monitor Oracle Database events, such as:

- Initialization of System Global Area for active instance
- Initialization of Program Global Area (PGA) for background processes of the active instance
- Connection to Oracle Database using `AS SYSDBA`

In addition, the operating system audit trail is logged in the Event Viewer log file, which can be viewed using Event Viewer.

## Using Microsoft Management Console to Administer a Database

Microsoft Management Console provides a central location for network administration.

Microsoft Management Console hosts applications (called snap-ins) that administrators can use to manage their networks. Oracle snap-ins enable database administrators to:

- Configure Oracle Database administrators, operators, users, and roles so the Windows operating system can authenticate them
- Configure `OracleServiceSID`
- Modify registry parameters for all Oracle homes on the computer
- Modify the computer host name, user name, and password for the database
- View and terminate an Oracle Database thread

## Using Registry Editor to Modify Configuration Information

Oracle Database stores its configuration information in a structure known as the registry.

You can view and modify this configuration information through Registry Editor. The registry contains configuration information for your computer and must not be accessible for editing by inexperienced users. Only experienced administrators must view and change this information.

Registry Editor displays configuration information in a format similar to Windows Explorer. In the left-hand window is a tree-like format consisting of keys (or folders). When one of these keys is highlighted, parameters and values assigned to that key are displayed in the right-hand window.

When you install products from your media, configuration parameters are automatically entered in the registry. These parameters are read each time your Windows computer is started and whenever an Oracle Database product is started. These parameters include settings for:

- Oracle home directory
- Language
- Company name
- Oracle home subdirectories for individual products
- Individual products such as SQL\*Plus
- Services

## Using Task Manager to Monitor Applications and Processes

Task Manager monitors applications and processes.

Task Manager has the following tabs:

- Applications tab displays what applications run. This is useful for identifying and ending unresponsive tasks. (Oracle Database does not appear as an application because it runs as a service.)
- Processes tab displays details of the currently running processes and their resource usage. Columns are customizable.
- Performance tab graphically displays real-time CPU and memory usage, which is useful for spotting sudden changes.
- Networking tab graphically displays the network traffic taking place over the computer's network connections.

## Using Local Users and Groups to Manage Users and Groups

Local Users and Groups enable you to manage users and groups on Windows.

Specifically, you can:

- Create and modify Local User Accounts
- Create and modify user profiles
- Create, add, and delete local groups

## Using SQL\*Loader

Describes Windows-specific information for using SQL\*Loader (SQLLDR).

- [Windows Processing Options](#)  
Discusses possible values for the operating system-dependent file processing specifications string option (`os_file_proc_clause`).
- [Control File Conventions](#)  
When preparing SQL\*Loader control files (`.ctl`), you must follow certain syntax and notational conventions.

## Windows Processing Options

Discusses possible values for the operating system-dependent file processing specifications string option (`os_file_proc_clause`).



### See Also:

*Oracle Database Utilities*

## Control File Conventions

When preparing SQL\*Loader control files (`.ctl`), you must follow certain syntax and notational conventions.

In the full path descriptions, backslashes do not require escape characters or other special treatment. When embedding a single or a double quotation mark inside a string delimited by double quotation marks, place a backslash escape character before the embedded quotation mark.

When specifying data types in the SQL\*Loader control file, note that the default sizes of native data types shown in Default Sizes of Native Data types are specific to Windows. These data types can be loaded with correct results only between systems where they have the same length in bytes. You cannot override these defaults in the control file. If the byte order is different between the systems, you can indicate the byte order of the data with the `BYTEORDER` parameter, or you can place a byte-order mark (BOM) in the file.

**Table 2-5 Default Sizes of Native Data types**

Native Data Types	Default Field Length
DOUBLE	8
FLOAT	4
INTEGER	4
SMALLINT	2



### Note:

The default listed is correct if `INTEGER` is specified without a size. But `INTEGER(n)` is also allowed. In that case, `n` specifies the size of the `INTEGER` field in bytes.

 **See Also:**

*Oracle Database Utilities* for a complete list of options and instructions on using SQL\*Loader



# 3

## Supporting Oracle Home User on Windows

Oracle Database supports the use of Oracle Home User, specified at the time of Oracle Database installation. Oracle Home User is used to run the Windows services for the Oracle home.

Oracle Home User can be a Windows Built-in Account or a Virtual Account or a standard Windows User Account (not an Administrator account). Oracle Home User cannot be changed post installation.

If a Windows Built-in Account is used, then no user name or password is required during installation and administration. However, if a Windows User Account is used as Oracle Home User, then you must provide the user name and password during installation and some of the administration tasks.

Virtual Accounts allow you to install an Oracle Database and, create and manage Database services without passwords. A Virtual Account can be used as the Oracle Home User for Oracle Database Single Instance installations and does not require a user name or password during installation and administration.

Oracle Home User is different from Oracle Installation User. Oracle Installation User is the user, who installs a given Oracle Home and hence requires administrative privileges. Oracle Home User is used to run the Windows services. You must not log into the Windows system as Oracle Home User to perform administrative tasks.

Note that the Windows administrator privilege is still required to perform Oracle administrative functions such as installation, upgrade, patching, and other functions.

### Note:

A Windows User Account used as Oracle Home User cannot have the administrator privileges as it causes the Oracle Universal Installer to display an error message.

- [Managing Oracle Home User](#)  
If you use a Windows User Account as the Oracle Home User, then you must ensure that this user account is present in the Windows system and its password is managed securely to ensure the proper operation and security of the database.
- [Using Oracle Home User for an Oracle Database and Oracle Database Client](#)  
For a single-instance Oracle Database and Oracle Database Client installations, you can use Built-in Account or a Windows User Account as the Oracle Home User. Single-instance Oracle Database installations may also use a Virtual Account.

- [Using Oracle Home User for Multiple Oracle Homes](#)  
Different Oracle homes on a system can use the same Oracle Home User or use different Oracle Home User names. Note that the earlier releases (11.2 and earlier) of Oracle Database are treated equivalent to using the Windows Built-in Account as the Oracle Home User.
- [Using Oracle Home User During Oracle Database Upgrade](#)  
You can use Oracle Database Upgrade Assistant to upgrade or move databases across Oracle homes if both the Oracle homes use the same Windows User Account as Oracle Home User, or at least one of the Oracle homes is configured to use Windows built-in account as the Oracle Home User.
- [Converting from Single-Instance Oracle Database to Oracle Real Application Clusters](#)  
You can convert from Oracle Database 19c single-instance databases to Oracle RAC using Oracle Database Configuration Assistant, `rconfig`, or Oracle Enterprise Manager.



#### See Also:

- Microsoft documentation for more information on different types of Windows user accounts
- *Oracle Database Installation Guide for Microsoft Windows*

## Managing Oracle Home User

If you use a Windows User Account as the Oracle Home User, then you must ensure that this user account is present in the Windows system and its password is managed securely to ensure the proper operation and security of the database.

You must secure the password of this Windows User Account and ensure that only database administrators have access to this password as one can log on to the database as the database administrator from this Windows User Account. You must also change the password for this Windows User Account at regular intervals for security reasons. You can change the password using Windows tools. However, when you change the password for this Windows User Account, you must also update the password for all Oracle services running under the Windows User Account.

This release has introduced a new Windows utility called the Oracle Home User Control. This is a command-line tool that displays the Oracle Home User name associated with the current Oracle home and updates the password for all Oracle services running under a specific Windows User Account (used as Oracle Home User).

### Related Topics

- [Using the Oracle Home User Control Tool](#)  
A new Windows tool, Oracle Home User Control, is a command-line tool that displays the Oracle Home User name associated with the current Oracle home and updates the password for the Windows services for the Oracle home.

## Using Oracle Home User for an Oracle Database and Oracle Database Client

For a single-instance Oracle Database and Oracle Database Client installations, you can use Built-in Account or a Windows User Account as the Oracle Home User. Single-instance Oracle Database installations may also use a Virtual Account.

Virtual Accounts allow you to install Oracle Database, create, and manage database services without passwords. Windows User Account can be an existing Windows Local User, Windows Domain User, Managed Services Account (MSA), or Group Managed Services Account (gMSA). For a Windows Local User Account or a Windows Domain User Account, you must provide both the user name and password during installation. For a Managed Services Account, you must provide the user name only.

The Group Managed Services Account (gMSA) enables you to install an Oracle Database and, create and manage Database services without passwords. The gMSA is a domain level account that can be used by multiple servers in a domain to run the services using this account.

For a Windows Local User, you also have the option of creating a new Windows user during installation. You must provide the user name and password for the user account and Oracle Universal Installer creates the Windows user during installation. The newly created Windows account is denied interactive logon privileges to the Windows computer. However, a Windows administrator can still manage this account like any other Windows account.

 **Note:**

If a Windows Local User Account is chosen as the Oracle Home User during single-instance Oracle Database installation, Windows NT Native Authentication (NTS) cannot be used for authenticating Windows domain users or users from remote computers.

The Virtual Account option enables you to install an Oracle Database and, create and manage Database services without passwords. User names do not appear on the logon screen.

When the Windows built-in account is chosen as the Oracle Home User, Oracle services for a server home are run using the built-in privileged LocalSystem account. Hence for single-instance Oracle Database installations, Oracle recommends that you use Virtual Account or a standard Windows User Account instead of a Windows built-in account as the Oracle Home User for enhanced security. For Oracle Database Client installations, it is not necessary to use a Windows User Account as Oracle Home User for reasons of security. Even when the Windows built-in account is chosen as the Oracle Home User, Oracle services for a client home are run using the built-in low-privileged LocalService account.

 **See Also:**

- *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Grid Infrastructure Installation Guide for Microsoft Windows x64 (64-Bit)*

## Using Oracle Home User for Multiple Oracle Homes

Different Oracle homes on a system can use the same Oracle Home User or use different Oracle Home User names. Note that the earlier releases (11.2 and earlier) of Oracle Database are treated equivalent to using the Windows Built-in Account as the Oracle Home User.

As the Oracle Home User has complete control over the Oracle base directory for an Oracle home, multiple Oracle homes are allowed to share the same Oracle base only when they use the same Oracle Home User. This is done for security reasons.

However, as an exception, Oracle supports the sharing of an Oracle base directory between a Windows built-in account (server) and a specific Windows User Account, and Windows built-in account (server) and Virtual Account. This enables easier upgrade of Oracle home from the older releases of Oracle Database to Oracle Database 19c as the same Oracle base can be shared, and all the files under the Oracle base can be accessed by the Oracle Home User.

 **Note:**

- When you share an Oracle base between 11g Release 2 (or earlier), 12c Release 2, and Oracle Database 19c, Windows User Account (used as Oracle Home User) is granted full control of the Oracle base and its subdirectories. This means that the Windows User Account for Oracle home can access or update any database files for the earlier release.
- After installing Oracle Database 12c Release 1 (or later) with a Windows User Account or Virtual Account as the Oracle Home User, do not install older versions of Oracle Database and share the same Oracle base directory. During the installation of older releases, ACLs are reset corresponding to the older releases and Oracle Database 12c Release 2 (or later) services may not be able to access the Oracle base directory and files.

On the contrary, if you decide to use a different Oracle base for Oracle Database 19c, there may be some issues in terms of Oracle services accessing the files from the older Oracle base.

### Related Topics

- [Setting File Permissions](#)  
Oracle Universal Installer, Oracle Database Configuration Assistant, and Oracle Database Upgrade Assistant set file permissions when you install or upgrade Oracle Database software.

#### See Also:

- *Oracle Database Installation Guide for Microsoft Windows*

## Using Oracle Home User During Oracle Database Upgrade

You can use Oracle Database Upgrade Assistant to upgrade or move databases across Oracle homes if both the Oracle homes use the same Windows User Account as Oracle Home User, or at least one of the Oracle homes is configured to use Windows built-in account as the Oracle Home User.

You can also use Oracle Database Upgrade Assistant to upgrade or move databases across Oracle homes if both the Oracle homes use Virtual Account.

## Converting from Single-Instance Oracle Database to Oracle Real Application Clusters

You can convert from Oracle Database 19c single-instance databases to Oracle RAC using Oracle Database Configuration Assistant, `rconfig`, or Oracle Enterprise Manager.

For an in-place conversion, you cannot change the Oracle Home User. For an out-of-place conversion, you can change the Oracle Home User only if the Oracle home for the single-instance database is not already configured with a Windows Domain User Account.

#### See Also:

- *Oracle Real Application Clusters Administration and Deployment Guide*

# 4

## Postinstallation Database Creation on Windows

Learn how to create a database after installing Oracle Database, using either Oracle Database Configuration Assistant or command-line tools.

- [About Oracle Database Naming Conventions](#)  
All the mounted Oracle Database servers in a network must have unique database names.
- [About Using Oracle Database Configuration Assistant on Windows](#)  
Oracle recommends you use Oracle Database Configuration Assistant (Oracle DBCA) to create a Database, because it is easier.
- [Overview of Database Creation Tasks on Windows Using Command-Line Tools](#)  
Learn how to create a new database manually. As part of its database software files, Oracle Database provides a sample initialization parameter file, which can you can edit to suit your needs.
- [About Administering an Oracle Database Instance Using ORADIM](#)  
ORADIM is a command-line tool that is available with Oracle Database.
- [About Administering an Oracle Database Instance Using Microsoft Management Console Snapin](#)  
You can perform the administrative activities on an Oracle Database from Microsoft Management Console Snap-In.
- [Overview of Database Migration from a 32-Bit Windows Computer](#)  
Describes the overview of database migration.

### About Oracle Database Naming Conventions

All the mounted Oracle Database servers in a network must have unique database names.

When a database is created, a name is associated with it and stored in its control files. If you provide the database keyword, either in the `CREATE DATABASE` statement or when prompted by Database Configuration Assistant, then that value becomes the name for that database.

If you attempt to mount two Oracle Database servers with the same database name, then you receive the following error during mounting of the second server:

```
ORA-01102: cannot mount database in EXCLUSIVE mode
```

If there are two or more Oracle Database servers on the same computer, but located in different Oracle homes, then the following rules apply:

- Each database name must be unique
- Each SID must be unique

To change the name of an existing database, you must use the `CREATE CONTROLFILE` statement to re-create your control files and specify a new database name.

## About Using Oracle Database Configuration Assistant on Windows

Oracle recommends you use Oracle Database Configuration Assistant (Oracle DBCA) to create a Database, because it is easier.

It offers the same interface and operates the same way on all the supported platforms, so no step-by-step procedures or screenshots are included here.

Oracle DBCA prompts for a password when the Oracle Home User is a Windows Local User Account or when a Windows Domain User Account and the password for Oracle Home User is not stored in Oracle wallet. The main purpose of Oracle Home User is to run Windows services with Windows User Account. However, this user account (Oracle Home User) has a very limited set of operating system-level privileges and must not be used for database administration. Oracle DBCA now provides an interface to create an Oracle Database service under an Oracle Home User, as specified during the process of installation. But Oracle DBCA does not provide an interface to create a new Windows user as the Oracle Home User.

The services created are not allowed to interact with the Windows desktop. `ORADIM`, the Windows utility tool used to create the `OracleServiceSID` - Oracle Database services, is used by Oracle Database Configuration Assistant to create those services on local and remote nodes. Oracle Database Configuration Assistant now accepts a user name and a password to run the service, and also changes the ownership of the files it creates (for example, the password file) so that it can be modified by the Oracle home user.

Oracle DBCA enables you to:

- Create a database
- Configure database options in a database
- Delete a database
- Manage templates

An initialization parameter file is an ASCII text file containing parameters. Use this file to create and modify a database using command-line tools. When you create a database using Oracle DBCA, a server parameter file (SPFILE) is created from the initialization parameter file, and the initialization parameter file is renamed. Oracle does not recognize the renamed file as an initialization parameter file, and it is not used after the instance is started.

If you want to modify an instance created with Oracle DBCA after it starts, you must use `ALTER SYSTEM` statements. You cannot change the server parameter file itself, because it is a binary file that cannot be browsed or edited using a text editor. The location of the newly-created server parameter file is `ORACLE_HOME\database`. The server parameter file name is `spfileSID.ora`.

 **See Also:**

- *Oracle Database Administrator's Guide*
- *Oracle Database 2 Day DBA* for instructions on using Oracle DBCA

## Overview of Database Creation Tasks on Windows Using Command-Line Tools

Learn how to create a new database manually. As part of its database software files, Oracle Database provides a sample initialization parameter file, which you can edit to suit your needs.

You can choose to create database creation scripts using Oracle Database Configuration Assistant.

The following are the types of Database creation tasks:

- Copy an existing database and delete the old database.
- Copy an existing database and keep the old database.
- Create a new database when no database exists on your system.

### Manual Database Creation Tasks

Use Manual Database Creation Tasks to understand the manual tasks involved in creating a new database for each of these database creation categories. Each step is explained in detail in the following subsections.

**Table 4-1 Manual Database Creation Tasks**

Task	Copy existing database and delete old database	Copy existing database and keep old database	Create new database when no database exists on system
<a href="#">About Exporting an Existing Database</a>	Yes	Note 1	Not applicable
<a href="#">Deleting Database Files</a>	Yes	No	Not applicable
<a href="#">Modifying the Initialization Parameter File</a>	Yes	Yes	Yes
<a href="#">Starting an Oracle Database Instance</a>	Yes	Yes	Yes
<a href="#">About Creating and Starting an Oracle Database Service</a>	No	Yes	Yes
<a href="#">Adding the CREATE DATABASE Statement in a Script</a>	Yes	Yes	Yes
<a href="#">Running the CREATE DATABASE Script</a>	Yes	Yes	Yes
<a href="#">About Importing a Database</a>	Yes	Note 2	Not applicable



**Table 4-1 (Cont.) Manual Database Creation Tasks**

Task	Copy existing database and delete old database	Copy existing database and keep old database	Create new database when no database exists on system
<a href="#">Updating ORACLE_SID in the Registry</a>	No	Only if you change the default <i>SID</i>	Yes
<a href="#">Backing Up the New Database</a>	Yes	Yes	Yes

 **Note:**

If you copy data from an existing database to the new database, select Yes; otherwise, No. Also, if you import tables and other objects from the existing database, select Yes; otherwise, No.

An example in the following sections demonstrates how to create a database. In this example, the existing database is the starter database with a *SID* of `orcl` located in directory `C:\app\username\oradata\orcl`. Copy `orcl` to a new database with a database name and *SID* of `prod` located in the directory `C:\app\username\oradata\prod`. Then, delete the starter database `orcl`.

- [About Exporting an Existing Database](#)  
You are required to export an existing database only if you intend to copy its contents to a new database.
- [Deleting Database Files](#)  
Deleting database files is required only when you copy an existing database to a new database to replace the old database.
- [Modifying the Initialization Parameter File](#)  
Describes how to modify the initialization parameter file.
- [About Creating and Starting an Oracle Database Service](#)  
Learn how to create and start an Oracle Database service.
- [Starting an Oracle Database Instance](#)  
Learn how to start an instance without mounting a database.
- [Adding the CREATE DATABASE Statement in a Script](#)  
The `CREATE DATABASE` statement is a SQL statement that creates the database.
- [Running the CREATE DATABASE Script](#)  
Use this procedure to run the `CREATE DATABASE` script.
- [About Importing a Database](#)  
Learn how to use Data Pump Import or Import.
- [Updating ORACLE\\_SID in the Registry](#)  
If this is the first database on your computer or if you intend to make the new database the default database, then you must make a change in the registry.
- [Creating the ORACLE\\_SID Parameter](#)  
If you do not yet have the parameter `ORACLE_SID`, because this is the first database on your system, then you must create it.

- [Backing Up the New Database](#)  
Use this procedure to prevent data loss.

## About Exporting an Existing Database

You are required to export an existing database only if you intend to copy its contents to a new database.

If you are working with data from an earlier Oracle release, then you can use Export for this task. If you are using Oracle Database 10g Release 1 (10.1) or later data, then Oracle recommends that you use Data Pump Export because it supports new Oracle Database 10g Release 1 (10.1) or later features, such as floating points.

Although you can start Data Pump Export or Export in either the parameter mode or an interactive mode, Oracle recommends parameter mode. Interactive mode provides less functionality than the parameter mode and exists for backward compatibility only.

The syntax for Data Pump Export parameter mode is:

```
C:\> expdp SYSTEM DUMPFILE=myexp.dmp FULL=y LOGFILE=myexp.log  
Password: password
```

The syntax for Data Pump Export interactive mode is:

```
C:\> expdp SYSTEM  
Password: password
```

Enter only the command `expdp SYSTEM` to begin an interactive session and let Data Pump Export prompt you for information it needs.

### Note:

If you use the parameter mode, then Data Pump Export considers the file names and the directory names to be invalid if they contain one or more blank spaces. The workaround is to enclose the full path in the `DUMPFILE=` parameter in triple quotation marks. For example:

```
DUMPFILE=" "C:\program files\export.dmp" " "
```

If Data Pump Export is used in an interactive mode, then the file name or the directory name can contain a space without quotation marks.

The syntax for Export parameter mode is:

```
C:\> exp SYSTEM FILE=myexp.dmp FULL=y LOG=myexp.log  
Password: password
```

The syntax for the Export interactive mode is:

```
C:\> exp SYSTEM  
Password: password
```

Enter only the command `exp SYSTEM` to begin an interactive session and let Export prompt you for information it needs.

 **Note:**

If you use the parameter mode, then Export considers the file names and the directory names to be invalid if they contain one or more blank spaces. The workaround is to enclose the full path in the `FILE=` parameter in triple quotation marks. For example:

```
FILE=" "C:\program files\export.dmp" "
```

If Export is used in an interactive mode, then the file name or the directory name can contain a space without quotation marks.

- [Exporting All Data from an Existing Database](#)  
Describes how to export all data from an existing database to a new database.

 **See Also:**

*Oracle Database Utilities* for more information about using Data Pump Export or Export

## Exporting All Data from an Existing Database

Describes how to export all data from an existing database to a new database.

To export:

1. Set `ORACLE_SID` to the database service of the database whose contents you intend to export. For example, if the database you intend to export is the starter database `orcl`, then enter the following at the command prompt. Note that there are no spaces around the equal sign (=) character.

```
C:\> set ORACLE_SID=orcl
```

2. If the existing database is Oracle Database 10g Release 1 (10.1) or later, then start Data Pump Export from the command prompt:

```
C:\> expdp SYSTEM DUMPFILE=myexp.dmp FULL=y LOG=myexp.log  
Password: password
```

You now have a full database export of the starter database `orcl` in the file `myexp.dmp`. All messages from Data Pump Export are logged in file `myexp.log`.

3. If the existing database is earlier than Oracle Database 10g Release 1 (10.1), then start Export from the command prompt:

```
C:\> exp SYSTEM FILE=myexp.dmp FULL=y LOG=myexp.log  
Password: password
```

You now have a full database export of the starter database `orcl` in the file `myexp.dmp`. All messages from Export are logged in the file `myexp.log`.

## Deleting Database Files

Deleting database files is required only when you copy an existing database to a new database to replace the old database.

In the following example, you delete the database files of the starter database `orcl`.

To delete database files:

1. Shut down starter database `orcl` at the command prompt:

```
C:\> oradim -SHUTDOWN -SID orcl -SHUTTYPE inst -SHUTMODE immediate
```

2. Delete the following files from the directory `C:\app\username\oradata\orcl`:

```
control01.ctl  
control02.ctl  
control03.ctl  
index01.dbf  
drsys01.dbf  
cwmlite01.dbf  
example01.dbf  
system01.dbf  
temp01.dbf  
tools01.dbf  
undotbs01.dbf  
user01.dbf  
xdbs01.dbf  
redo01.log  
redo02.log  
redo03.log
```

## Modifying the Initialization Parameter File

Describes how to modify the initialization parameter file.

To use the starter database `orcl` as the basis for your new database:

1. Copy `ORACLE_BASE\admin\orcl\pfile\init.ora`.
2. Place the copy in `ORACLE_BASE\admin\prod\pfile\init.ora`.
3. Modify the file by performing the following tasks:

 **Note:**

Starting with Oracle9i Release 2 (9.2), nesting of quotation marks using the backslash (\) escape character is no longer supported. This affects how Oracle Database interprets the parameter values in your initialization parameter file. For example, if you specified `CONTROL_FILES = "ctlfile\1.ora"` in releases before release 9.2, the file name was interpreted as `ctlfile'1.ora`. Starting with release 9.2, the file name will be interpreted as `ctlfile\1.ora`.

Oracle highly recommends modifying your parameter files to remove such references and other methods of nesting quotation marks in the initialization parameter values.

- a. If you do not have an existing database on your system, then you cannot copy an existing initialization parameter file to use as the basis for your new initialization parameter file. However, you can use the sample initialization parameter file `initsmpl.ora` provided in:

```
ORACLE_HOME\admin\sample\pfile
```

This is the basis for the initialization parameter file for the database `prod`.

- b. If you use the `initsmpl.ora` file as the basis for the initialization parameter file, then the following parameters must be set to the indicated values, otherwise you cannot start database `prod`:

```
DB_NAME=prod.domain
```

The parameter `DB_NAME` indicates the database name and must match the name used in the `CREATE DATABASE` statement. Give a unique database name to each database. You can use eight characters for a database name. The name is not required to match the `SID` of the database service.

```
INSTANCE_NAME=prod.domain
```

```
SERVICE_NAMES=prod.domain
```

```
CONTROL_FILES = ( "C:\app\username\oradata\prod\control01.ctl", "C:\app\username\oradata\prod\control02.ctl", "C:\app\username\oradata\prod\control03.ctl" )
```

The parameter `CONTROL_FILES` lists the database control files. You do not have to control files on your file system at this point, because control files are created when you run the `CREATE DATABASE` statement. Ensure that you specify the complete path and the file name, including the drive letter.

```
DB_FILES=100
```

Modifying the initialization parameter `DB_FILES` is not required, but it is recommended to optimize performance. Set this parameter to the same number as the value of the `MAXDATAFILES` option of the `CREATE DATABASE` statement. The value of 100 is used for this example.

The `DIAGNOSTIC_DEST` initialization parameter sets the location of the Automatic Diagnostic Repository (ADR), which is a directory structure stored outside of the database. The ADR is used in problem diagnostics.

Use `DIAGNOSTIC_DEST = ORACLE_HOME\log` if the environment variable `ORACLE_BASE` is not set.

Use `DIAGNOSTIC_DEST = ORACLE_BASE` variable if the environment variable `ORACLE_BASE` is set.

 **See Also:**

- [Oracle Database Installation Guide for Microsoft Windows](#) for information about ADR
- [Oracle Database Reference](#) for information about other initialization parameters that you can add or modify

### Related Topics

- [Adding the CREATE DATABASE Statement in a Script](#)  
The `CREATE DATABASE` statement is a SQL statement that creates the database.

## About Creating and Starting an Oracle Database Service

Learn how to create and start an Oracle Database service.

Perform either of the following steps:

- Copy an existing database to a new database and keep the old database
- Create a new database when you have no other database to copy

Before you create the database, first create a Windows service to run the database. This service is the Oracle Database process, `oracle.exe`, installed in the form of a Windows service.

Use `ORADIM` to create the service.

- [Creating and Starting an Oracle Database Service](#)  
Learn how to create and start an Oracle Database service.
- [Access to Oracle Wallets in a File System for Oracle Database Services](#)  
Discusses about accessing Oracle Wallets.

### Related Topics

- [About Administering an Oracle Database Instance Using ORADIM](#)  
`ORADIM` is a command-line tool that is available with Oracle Database.

## Creating and Starting an Oracle Database Service

Learn how to create and start an Oracle Database service.

To create and start an Oracle Database service:

1. Run `ORADIM` from the command prompt:

```
C:\> oradim -NEW -SID prod -STARTMODE manual  
-PFILE "C:\app\username\admin\prod\pfile\init.ora"
```

Note that the previously created initialization parameter file is specified, with complete path, including drive name. You can check if the service is started in the Services window of the Control Panel. ORADIM automatically creates Oracle Database services under the Oracle Home User account. If the Oracle Home User account is a Windows Local User Account or a Windows Domain User Account, then ORADIM prompts for its password.

2. Set the value of `ORACLE_SID` to `prod`. Note that there are no spaces around the equal sign (=) character:

```
C:\> set ORACLE_SID=prod
```

## Access to Oracle Wallets in a File System for Oracle Database Services

Discusses about accessing Oracle Wallets.

When an Oracle wallet is created in the file system, only the user creating the wallet is granted access to that wallet by wallet creation tools. Therefore, Oracle Database services (running as the Windows User Account) might not be able to access the wallet unless you explicitly grant access to the wallet using Windows tools.

### Related Topics

- [About Setting File System ACLs Manually](#)  
As Oracle Database services now run under a standard Windows User Account, a file might not be accessible by Oracle Database services unless the file system Access Control Lists (ACLs) grant access to the file.

## Starting an Oracle Database Instance

Learn how to start an instance without mounting a database.

Start an instance without mounting a database.

```
SQL> STARTUP NOMOUNT
```

You must not specify the `PFILE` clause in this example, because the initialization parameter file is stored in the default location. At this point, there is no database. Only the System Global Area (SGA) is created and the background processes are started in preparation for the creation of a new database.

## Adding the CREATE DATABASE Statement in a Script

The `CREATE DATABASE` statement is a SQL statement that creates the database.

A script containing this statement can be used anytime you create a database.

The `CREATE DATABASE` statement has the following parameters:

- `MAXDATAFILES` - default value: 32, maximum value: 65534
- `MAXLOGFILES` - default value: 32, maximum value: 255

The `CHARACTER SET` parameter determines the database character set of the new database. The default value is `US7ASCII`, however the recommended value is `AL32UTF8`. `AL32UTF8` is the Oracle implementation of the Unicode Standard character set in UTF-8 encoding form. Unicode is suitable for storing text in practically any written language of the world.

When you run the `CREATE DATABASE` statement, Oracle Database performs several operations depending upon the clauses that you specified in the `CREATE DATABASE` statement or the initialization parameters that you have set.

 **Note:**

Oracle Managed Files is a feature that works with the `CREATE DATABASE` statement to simplify administration of Oracle Database. Oracle Managed Files eliminates the requirement to directly manage operating system files comprising an Oracle Database server, because you specify operations in terms of database objects rather than file names.

To create the database `prod`, copy and save the following statement in a file named `script_name.sql`:

```
CREATE DATABASE prod
USER SYS IDENTIFIED BY sys_password
USER SYSTEM IDENTIFIED BY system_password
MAXLOGFILES 5
MAXDATAFILES 100
DATAFILE 'C:\app\username\oradata\prod\system01.dbf' SIZE 325M REUSE
AUTOEXTEND ON NEXT 10240K MAXSIZE UNLIMITED
UNDO TABLESPACE "UNDOTBS" DATAFILE 'app\username\oradata\prod\undotbs01.dbf'
SIZE 200M REUSE AUTOEXTEND ON NEXT 5120K MAXSIZE UNLIMITED
CHARACTER SET AL32UTF8
logfile 'C:\app\username\oradata\prod\redo01.log' size 100M reuse,
       'C:\app\username\oradata\prod\redo02.log' size 100M reuse,
       'C:\app\username\oradata\prod\redo03.log' size 100M reuse
Oracle Database Installation Guide for Microsoft Windows EXTENT MANAGEMENT LOCAL;
```

 **See Also:**

- *Oracle Database Administrator's Guide* for more information about using Oracle Managed Files
- *Oracle Database Installation Guide for Microsoft Windows* for more information about recommended database character sets

## Running the CREATE DATABASE Script

Use this procedure to run the `CREATE DATABASE` script.

To use the SQL script to create a database:

1. Verify that the service is started in the Control Panel. In this example, the service name is `OracleServicePROD`, and its status column must display `Started`. If not, then select the service name and select **Start**.

You can also check the status of the service by entering the following at the command prompt:

```
C:\> net START
```



A list of all the Windows services currently running on the system appears. If OracleServicePROD is missing from the list, then enter:

```
C:\> net START OracleServicePROD
```

**2. Make PROD the current SID:**

```
C:\> set ORACLE_SID=PROD
```

**3. Add ORACLE\_HOME\bin to your PATH environment variable:**

```
set PATH=ORACLE_BASE\ORACLE_HOME\bin;%PATH%
```

**4. Start SQL\*Plus from the command prompt, and connect to the database as SYSDBA:**

```
C:\> sqlplus /NOLOG
SQL> CONNECT / AS SYSDBA
```

The message `connected` appears.

**5. Turn on spooling to save messages:**

```
SQL> SPOOL script_name.log
```

**6. Run the script `script_name.sql` that you created in *Adding the CREATE DATABASE Statement in a Script*.**

```
SQL> C:\app\username\product\12.2.0\dbhome_1\rdbms\admin\script_name.sql;
```

If the database is successfully created, then the instance is started and the following message appears numerous times: `Statement processed`

### Related Topics

- [Adding the CREATE DATABASE Statement in a Script](#)

The `CREATE DATABASE` statement is a SQL statement that creates the database.

## About Importing a Database

Learn how to use Data Pump Import or Import.

You can use Data Pump Import (for Oracle Database 10g Release 1 (10.1) or later data) or Import (for earlier data) to import the full export created into the new database. Although you can start Data Pump Import or Import using either the parameter mode or the interactive mode, Oracle recommends the parameter mode because it provides more functionality. Interactive mode exists solely for backward compatibility.

The syntax for Data Pump Import parameter mode is:

```
C:\> impdp SYSTEM DUMPFILE=myexp.dmp FULL=y LOG=myexp.log
Password: password
```

The syntax for Data Pump Import interactive mode is:

```
C:\> impdp SYSTEM
Password: password
```

Enter only `impdp SYSTEM` to begin an interactive session and let Data Pump Import prompt you for information it needs.

 **Note:**

- If you use the parameter mode, then Data Pump Import considers the file names and the directory names to be invalid if they contain one or more blank spaces. The workaround is to enclose the full path in the `DUMPFILE=` parameter in triple quotation marks. For example:

```
DUMPFILE=" "C:\program files\export.dmp" " "
```

If you use Data Pump Import in an interactive mode, then the file name or the directory name can contain a space without the quotation marks.

- If the original database from which the export file was generated contains a tablespace that is not in the new database, then Import tries to create that tablespace with associated data files.

The easy solution is to ensure that both the databases contain the same tablespaces. Data files are not required to be identical. Only the tablespace names are important.

**Related Topics**

- [About Exporting an Existing Database](#)  
You are required to export an existing database only if you intend to copy its contents to a new database.

 **See Also:**

*Oracle Database Utilities* for more information about using Data Pump Import or Import

## Updating ORACLE\_SID in the Registry

If this is the first database on your computer or if you intend to make the new database the default database, then you must make a change in the registry.

Perform the following steps:

1. Start Registry Editor at the command prompt:

```
C:\> regedit
```

The Registry Editor window appears.

2. Select the subkey `\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0` for the first Oracle home on your computer. For subsequent installations to different Oracle homes on the same computer, the path is `\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOMEID` where ID is the unique number identifying the Oracle home.
3. Locate the parameter `ORACLE_SID` on the right side of the Registry Editor window.
4. Double-click the parameter name and change the data to the new *SID*, which is `prod` in this example.

### Related Topics

- [Configuration Parameters and the Registry](#)

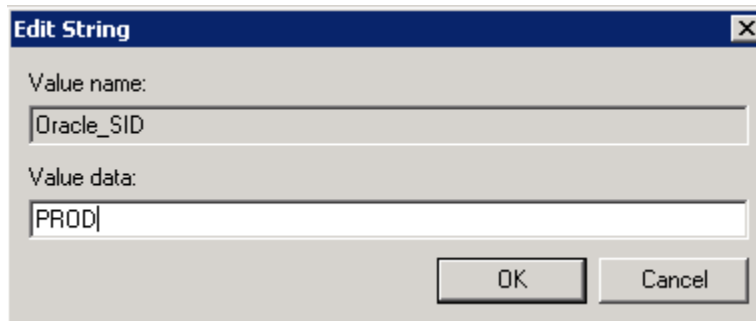
## Creating the ORACLE\_SID Parameter

If you do not yet have the parameter `ORACLE_SID`, because this is the first database on your system, then you must create it.

To create the parameter `ORACLE_SID`:

1. Select **New** from the **Edit** menu.
2. Select **Expandable String Value** from the menu list.
3. A **New Value #1** expandable string value name is created on the right pane of the Registry Editor window of data type `REG_EXPAND_SZ`.
4. Right-click the parameter, select **Rename** to rename it to `ORACLE_SID` and press **Enter**.
5. Double-click the `ORACLE_SID` entry to change the value data to the new `SID`.

An Edit String dialog box appears:



6. Enter **PROD** in the **Value data** field.
7. Click **OK**.  
Registry Editor adds parameter `ORACLE_SID`.
8. Select **Exit** from the **File** menu.  
Registry Editor exits.

## Backing Up the New Database

Use this procedure to prevent data loss.

### Note:

If anything goes wrong while operating the new database without a backup, then you must repeat the database creation procedure. Back up your database now to prevent loss of data.

To back up the new database:

1. Shut down the database instance and stop the service:

```
C:\> oradim -SHUTDOWN -SID prod -SHUTTYPE srvc,inst -SHUTMODE immediate
```

 **Note:**

Although ORADIM returns the prompt immediately, you must wait for the database and the service to stop completely before continuing to Step 2. Wait until the Control Panel indicates service `OracleServicePROD` has stopped. If you do not do this, then the backup is useless because it was taken while data was being written to data files.

2. Back up database files using the tool of your choice.

Database files consist of the initialization parameter file, control files, online redo log files, and data files.

When the backup is complete, you can start the database again, create users and objects, if necessary, make any other changes, and use the database.

Back up the database after making any significant changes, such as switching into the archiving mode or adding a tablespace or a data file.

 **See Also:**

- *Oracle Database Concepts*
- *Oracle Database Backup and Recovery User's Guide*
- *Oracle Database 2 Day DBA*

 **Note:**

Do not store database files on a compressed drive. This can result in write errors and a decreased performance.

## About Administering an Oracle Database Instance Using ORADIM

ORADIM is a command-line tool that is available with Oracle Database.

Use ORADIM only if you are manually creating, deleting, or modifying databases. Oracle Database Configuration Assistant is an easier tool to use for this purpose.

ORADIM creates Oracle Database service, Oracle VSS Writer service, and Oracle Scheduler service to run under the Oracle Home User account. If this account is a Windows Local User Account or a Windows Domain User Account, then ORADIM prompts for a password for that account and accepts the same through `stdin`.

Specify both the Oracle Home User and its password using the `-RUNAS osusr[/ospass]` option to `oradim`. If the given `osusr` is different from the Oracle Home User, then use the Oracle Home User instead of `osusr` along with the given `ospass`.

The following sections describe ORADIM commands and parameters. Note that each command is preceded by a dash (-). To get a list of the ORADIM parameters, enter:

```
oradim -? | -h | -help
```



#### Note:

Specifying `oradim` without any options also returns a list of ORADIM parameters and descriptions.

When you use ORADIM, a log file called `oradim.log` opens in `ORACLE_HOME\database`, or in the directory specified by registry parameter `ORA_CWD`. All operations, whether successful or failed, are logged in this file. You must check this file to verify success of an operation.

If you have installed an Oracle Database service on Windows, then when logging in as the SYSTEM user (LocalSystem), with the startup mode set to Automatic, it is possible that the Oracle Database service starts but the database does not start automatically. The following error message is written to the file `ORADIM.LOG` in the directory `ORACLE_HOME\database`:

```
ORA-12640: Authentication adapter initialization failed
```

Oracle Enterprise Management Agent, Oracle Enterprise Manager Management Server, and Oracle Internet Directory fails, because they cannot connect to the database for the same reason.

To work around this issue, perform the following tasks:

1. Modify `SQLNET.ORA`

You can modify `SQLNET.ORA`, by doing either of the following:

- Remove the line `sqlnet.authentication_services=(NTS)`
- Change the line `sqlnet.authentication_services=(NONE)`

2. Start the database after the service starts.

You can start the database manually after the Oracle Database service has started, using SQL\*Plus and connecting as `SYSDBA`.

3. Start the service as a specific user.



#### See Also:

Your operating system documentation for instructions on starting the services

- [Creating an Instance Using ORADIM](#)  
Learn how to create an Oracle Database instance using ORADIM.

- [Starting an Instance and Services Using ORADIM](#)  
Learn how to start an instance and services using ORADIM.
- [Stopping an Instance and Services Using ORADIM](#)  
Learn how to stop an instance and services using ORADIM.
- [Editing an Instance Using ORADIM](#)  
Learn how to edit an instance to change such values as instance name, startup mode, shutdown mode, and shutdown type using ORADIM.
- [Deleting an Instance Using ORADIM](#)  
Learn how to delete an instance using ORADIM.
- [Manipulating ACLs Using ORADIM](#)  
Learn how to manipulate ACLs using ORADIM.
- [Manipulating Family Settings to Initialization Parameters using ORADIM](#)  
Learn how to manipulate family settings to initializing parameters.

## Creating an Instance Using ORADIM

Learn how to create an Oracle Database instance using ORADIM.

To use ORADIM to create an instance, enter:

```
oradim [-NEW -SID SID] | -SRVC service_name | -ASMSID SID | -ASMSRVC service_name
[-SYSPWD password][-MAXUSERS number][-STARTMODE auto | manual] [-SRVCSTART system |
demand]
[-PFILE filename | -SPFILE] [-SHUTMODE normal | immediate | abort] [-TIMEOUT
secs] [-RUNAS osusr[/ospass]]
```

For this command, note the following:

- `-NEW` indicates that you are creating a new instance. This is a mandatory parameter.
- `-SID SID` is the name of the instance to create.
- `-SRVC service_name` is the name of the service to create (OracleServiceSID).
- `-ASMSID SID` is the name of the Oracle Automatic Storage Management instance to create.
- `-ASMSRVC service_name` is the name of the Oracle Automatic Storage Management service to create.
- `-SYSPWD password` is the system password.
- `-MAXUSERS number` is the number of users defined in the password file. The default is 5.
- `-STARTMODE auto | manual` indicates whether to start the instance when the Oracle Database service is started. The default is `manual`.
- `-SRVCSTART system | demand` indicates whether to start the Oracle Database service upon computer restart. Default is `demand`. Here, `system` specifies that the service be configured to automatically start when the system boots or reboots. `Demand` specifies that the user has to explicitly start the service.
- `-PFILE filename` is the initialization parameter file to be used with this instance. Ensure that you specify the complete path name of this file, including the drive letter.

- `-SPFILE` indicates that a server parameter file (SPFILE) be used during startup instead of a PFILE.
- `-SHUTDOWNMODE` specifies how to stop an instance. It requires an argument and the default is `immediate`. If `SHUTDOWNMODE` is omitted, then there is no attempt made to shutdown the instance when the service is shut down.
- `-TIMEOUT secs` sets the maximum time to wait (in seconds) before the service for a particular `SID` stops. The default is 90 seconds. It cannot be used without the `SHUTDOWN` argument.
- `-RUNAS osusr[/ospass]` ("run as") makes it possible to specify both the Oracle Home User and its password. If the given `osusr` is different from the Oracle Home User, then the Oracle Home User is used instead of the `osusr` along with the given `ospass`.

Though the `ospass` can be specified on the command line, Oracle recommends accepting `ospass` through `stdin`.

ORADIM creates Oracle Database service, Oracle VSS Writer service, and Oracle Scheduler service to run under the Oracle Home User account. If this account is a Windows Local User Account or Windows Domain User Account, then ORADIM prompts for the password for that account and accepts the same through `stdin`.

 **Note:**

For simplicity in demonstrating this feature, this example does not perform the password management techniques that a deployed system typically uses. In a production environment, follow the Oracle Database password management guidelines, and disable any sample accounts.

To create an instance called `PROD`, for example, enter:

```
C:\> oradim -NEW -SID prod -STARTMODE auto -PFILE C:\app\username\admin\prod\pfile
\init.ora
```

 **See Also:**

*Oracle Database Security Guide* for password management guidelines and other security recommendations.

## Starting an Instance and Services Using ORADIM

Learn how to start an instance and services using ORADIM.

To use ORADIM to start an instance and services, enter:

```
oradim -STARTUP -SID SID | -ASMSID SID [-SYSPWD password] [-STARTTYPE srvc |
inst | srvc,inst] [-PFILE filename | -SPFILE]
```

For this command, note the following:

- `-STARTUP` indicates that you are starting an instance that already exists. This is a mandatory parameter.
- `-SID SID` is the name of the instance to start.
- `-ASMSID SID` is the name of the Oracle Automatic Storage Management instance to start.
- `-STARTTYPE srv, inst` indicates whether to start the service or the instance. One or both values can be specified. If it is not specified, then the registry is checked for the current setting.
  - `-STARTTYPE srv` is the equivalent of running `net start oracleservicesid` from the command line.
  - `-STARTTYPE inst` is the equivalent of running `startup` within SQL\*Plus.
- `-PFILE filename` is the initialization parameter file to be used with this instance. Ensure that you specify the complete path name of this file, including drive letter.
- `-SPFILE` indicates that a server parameter file (SPFILE) be used during startup instead of a PFILE.

To start an instance called `puma`, for example, enter:

```
C:\> oradim -STARTUP -SID puma -STARTTYPE inst -PFILE C:\app\username\admin\prod
\pfile\init.ora
```

## Stopping an Instance and Services Using ORADIM

Learn how to stop an instance and services using ORADIM.

To use ORADIM to stop an instance, enter:

```
oradim -SHUTDOWN -SID SID | -ASMSID SID [-SYSPWD password]
[-SHUTTYPE srv | inst | srv,inst] [-SHUTMODE normal | immediate | abort]
```

For this command, note the following:

- `-SHUTDOWN` indicates that you are stopping an instance. This is a mandatory parameter.
- `-SID SID` specifies the name of the instance to stop.
- `-ASMSID SID` is the name of the Oracle Automatic Storage Management instance to stop.
- `-SHUTTYPE srv, inst` indicates whether to stop the service or the instance. One or both values can be specified. If it is not specified, then the registry is checked for the current setting.
- `-SHUTMODE` specifies how to stop an instance. This is an optional parameter. If you do not specify how to stop an instance, then `immediate` is the default mode.

To stop an instance called `puma`, for example, enter:

```
C:\> oradim -SHUTDOWN -SID puma -SHUTTYPE srv,inst
```



## Editing an Instance Using ORADIM

Learn how to edit an instance to change such values as instance name, startup mode, shutdown mode, and shutdown type using ORADIM.

To use ORADIM to modify an instance, enter:

```
oradim -EDIT -SID SID | -ASMSID SID [-SYSPWD password] [-STARTMODE auto | manual] [-SRVCSTART system | demand] [-PFILE filename | -SPFILE][SHUTMODE normal | immediate | abort] [SHUTTYPE srvc | inst | srvc,inst]
```

For this command, note the following:

- `-EDIT` indicates that you are modifying an instance. This is a mandatory parameter.
- `-SID SID` specifies the name of the instance to modify. This is a mandatory parameter.
- `-ASMSID SID` is the name of the Oracle Automatic Storage Management instance to modify.
- `-STARTMODE` indicates whether to start the instance when the Oracle Database service is started. The default is `manual`.
- `-SRVCSTART system | demand` indicates whether to start the Oracle Database service on computer restart. The default is `demand`.
- `-PFILE filename` specifies the initialization parameter file to be used with this instance. Ensure that you specify the complete path name of this file, including the drive letter.
- `-SPFILE` indicates that a server parameter file (SPFILE) be used during startup instead of a PFILE.
- `-SHUTMODE` specifies how to stop an instance. This is an optional parameter. If you do not specify how to stop an instance, then `immediate` is the default mode.
- `-SHUTTYPE` indicates whether to stop the service or the instance. One or both values can be specified. If it is not specified, then the registry is checked for the current setting.

To specify a new initialization parameter file for the instance `prod`, for example, enter:

```
C:\> oradim -EDIT -SID prod -PFILE C:\app\username\product\12.2.0\admin\lynx\pfile\init.ora
```

## Deleting an Instance Using ORADIM

Learn how to delete an instance using ORADIM.

To use ORADIM to delete an instance, enter:

```
oradim -DELETE -SID SID | -ASMSID SID | -SRVC service_name | -ASMSRVC service_name
```

For this command, note the following:

- `-DELETE` indicates that you are deleting an instance or service. This is a mandatory parameter.
- `-SID SID` specifies the name of the `SID` to delete.

- `-SRVC service_name` specifies the name of the service to delete (OracleServiceSID). The user must specify either *SID* or *SRVC*.
- `-ASMSID SID` is the name of the Oracle Automatic Storage Management instance to delete.
- `-ASMSRVC service_name` is the name of the Oracle Automatic Storage Management service to delete.

To delete an instance called `prod`, for example, enter:

```
C:\> oradim -DELETE -SID prod
```

## Manipulating ACLs Using ORADIM

Learn how to manipulate ACLs using ORADIM.

To use ORADIM to manipulate ACL, enter:

```
oradim -ACL -setperm|-addperm|-removeperm dbfiles|diag|registry -USER username  
-OBJTYPE file|dir|registry -OBJPATH object-path -RECURSE true|false [-HOST  
hostname]
```

For this command, note the following:

- `-ACL` indicates that you are manipulating ACL on an object. This is a mandatory parameter.
- `-setperm | -addperm | -removeperm dbfiles | diag | registry-` indicates that you are setting, adding, or removing ACLs on the specified object. `dbfiles` is for database files, `diag` is for database, oracle-base & logs and `registry` is for registry key. Set one of these based on the object on which the ACL is set. This is a mandatory parameter.
- `-USER username` indicates the user for whom the ACLs are granted. This must not be essentially the service user of the current oracle home. This is a mandatory parameter.
- `-OBJTYPE file|dir|registry` Set the object type to `file/dir/registry` based on the object on which the ACLs are set. This is a mandatory parameter.
- `-RECURSE true / false` indicates whether the ACL is applicable to all objects within the specified object. This is a mandatory parameter.
- `-HOST hostname` This can be used to remotely set ACLs on the specified host. This is limited to the scope of what windows supports remotely. Another way of doing this is to use the windows allowed conventions without using the `-HOST` option. For example, `\\<hostame>\c$\oracle\rdbms\admin\abc.txt`. This is optional.

To set ACL on a file named `abc.txt`, for example, enter:

```
C:\> oradim -acl -setperm dbfiles -user winusr -objtype file -objpath c:  
\a.txt -recurse true
```

To add ACL on a registry key, for example, enter:

```
C:\>oradim -acl -addperm registry -USER wingen -OBJTYPE registry -OBJPATH  
MACHINE\SOFTWARE\ORACLE\KEY_Oradb12Home1 -RECURSE true
```

## Manipulating Family Settings to Initialization Parameters using ORADIM

Learn how to manipulate family settings to initializing parameters.

To use ORADIM to add family support to the initialization parameters, enter:

```
oradim -FAMILY -set|-delete value [-SID sid | -ASMSID sid | -MGMTDBSID sid | -IOSSID sid | -APXSID sid ]
```

For this command, note the following:

- **-FAMILY:** Indicates that you are manipulating family settings. This is a mandatory parameter.
- **-set | -delete value:** Should be used to set/delete value <HKLM>/Software/Oracle/<Current\_ORACLE\_HOME>/ORACLE\_FAMILY. Set creates the above registry key and sets its value with the one specified. If the key exists already, then its value is updated. Delete removes the entry.
- **[-SID sid | -ASMSID sid | -MGMTDBSID sid | -IOSSID sid | -APXSID sid ]:** If one of these is specified, then the registry entry set/delete is <HKLM>/Software/Oracle/<Current\_ORACLE\_HOME>/ORACLE\_<sid>\_FAMILY. This is optional.

To make `inst1` as part of the family `prod` for example, enter:

```
c:\>oradim -FAMILY -set prod -SID inst1
```

This creates the registry entry <HKLM>/Software/Oracle/<Current\_ORACLE\_HOME>/ORACLE\_inst1\_FAMILY = prod.

## About Administering an Oracle Database Instance Using Microsoft Management Console Snapin

You can perform the administrative activities on an Oracle Database from Microsoft Management Console Snap-In.

The Oracle Instance Manager Snap-In provides centralized management of instances for all Oracle Database Homes.

You can locate the Oracle Instance Manager Snap-In in the path `ORACLE_HOME\MMC Snap-Ins\oradim` or by clicking on the Oracle Instance Manager shortcut in the Oracle Home.

The Snap-In lists the Oracle Database Homes in the scope pane and clicking on them displays the Oracle Database services for the selected Oracle Database Home in the results pane. You can perform all the operations that are done in ORADIM using the Snap-In.

Right-click the Oracle Database Home in the scope pane to Create an Instance, ACL, and Family options. Right-click the service in the result pane to view the Edit, Delete, Startup, and Shutdown options.

A dialog box for the selected item appears where you choose the options and on clicking **OK**, the action is done. You can use the Snap-In only with the administrator privileges.

 **Note:**

Oracle Instance Manager Snap-In for information about Snap-Ins by pressing F1 or clicking Help

## Overview of Database Migration from a 32-Bit Windows Computer

Describes the overview of database migration.

- [Backing Up a 32-Bit Oracle Database](#)  
Describes how to back up a 32-bit Oracle home database.
- [Migration Considerations](#)  
While upgrading an ASM disk group from Oracle Database 11g to Oracle Database 19c on Windows platforms, all the current files on the disk group are shown as being accessible to all users.
- [Migrating an Oracle Database 11g Release 2 \(11.2\) or Earlier Database](#)  
To migrate Oracle Database 11g Release 2 (11.2) or earlier database for 32-bit Windows to Oracle Database 19c for 64-bit Windows, perform the following steps:

 **See Also:**

*Oracle Database Upgrade Guide* for information about upgrading an earlier release of Oracle Database to Oracle Database 19c

## Backing Up a 32-Bit Oracle Database

Describes how to back up a 32-bit Oracle home database.

To back up:

1. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

2. Connect to the database instance as SYSDBA:

```
SQL> CONNECT / AS SYSDBA;
```

3. Create a `.trc` file to use as a template to re-create the control files on the 64-bit computer:

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE;
```

4. Shut down the database:

```
SQL> SHUTDOWN IMMEDIATE;
```

5. Perform a full offline backup of the database.

 **See Also:**

*Oracle Database Backup and Recovery User's Guide* for an overview of backup and recovery solutions

## Migration Considerations

While upgrading an ASM disk group from Oracle Database 11g to Oracle Database 19c on Windows platforms, all the current files on the disk group are shown as being accessible to all users.

Thus, a user with `SYSASM` privileges must modify the user ownership, group membership, and permissions of the current files accordingly, so that the files are owned by their respective database users only.

## Migrating an Oracle Database 11g Release 2 (11.2) or Earlier Database

To migrate Oracle Database 11g Release 2 (11.2) or earlier database for 32-bit Windows to Oracle Database 19c for 64-bit Windows, perform the following steps:

1. Install Oracle Database 19c for 64-bit Windows.
2. Create the new Oracle Database 19c service at the command prompt:

```
C:\> ORADIM -NEW -SID SID [-INTPWD PASSWORD ] -MAXUSERS USERS
-STARTMODE AUTO -PFILE ORACLE_HOME\DATABASE\INITSID.ORA
```

The following table provides more information about the values you must supply.

Parameter	Description
<i>SID</i>	<i>SID</i> of the database you are migrating.
<i>PASSWORD</i>	Password for the new Oracle Database 19c for 64-bit Windows database. This is the password for the user connected with the <code>SYSDBA</code> privileges. The <code>-INTPWD</code> option is not required. If you do not specify it, then operating system authentication is used, and no password is required.
<i>USERS</i>	Maximum number of users who can be granted <code>SYSDBA</code> and <code>SYSOPER</code> privileges.
<i>ORACLE_HOME</i>	Oracle home directory. Ensure that you specify the full path name with the <code>-PFILE</code> option, including the drive letter of the Oracle home directory.

3. Copy the 32-bit data files to the new 64-bit Oracle home.
4. Copy the 32-bit configuration files to the 64-bit Oracle home.
  - a. If your 32-bit initialization parameter file has an `IFILE` (include file) entry, then copy the file specified by the `IFILE` entry to the 64-bit Oracle home and edit the `IFILE` entry in the initialization parameter file to point to its new location.

- b. If you have a password file that resides in the 32-bit Oracle home, then copy the password file to the 64-bit Oracle home. The default 32-bit password file is located in `ORACLE_HOME\database\pwdSID.ora`, where `SID` is your Oracle instance ID.
5. If 19c Oracle home uses a Windows User Account as the Oracle Home User, then add this Oracle Home User to the list of users and grant file permissions or directory permissions for all the database files, `init.ora` files, oracle password files and so on.
6. Add the `_SYSTEM_TRIG_ENABLED = false` parameter to the `ORACLE_HOME\database\ORACLE_SID\init.ora` file in the 64-bit Oracle home before changing the word size.
7. Remove this parameter from the initialization file after the word size change is complete.
8. Go to the 64-bit `ORACLE_HOME\rdbms\admin` directory from the command prompt.
9. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

10. Connect to the database instance as SYSDBA:

```
SQL> CONNECT / AS SYSDBA;
```

11. Re-create the 64-bit control files using the `CREATE CONTROLFILE` command. This creates the new control file in the `ORACLE_HOME\database` directory.

Here is an example of a database named `orcl32` on a 32-bit computer migrating to `orcl64` on a 64-bit computer:

```
CREATE CONTROLFILE REUSE DATABASE "T1" NORESETLOGS NOARCHIVELOG
    MAXLOGFILES 32
    MAXLOGMEMBERS 2
    MAXDATAFILES 32
    MAXINSTANCES 16
    MAXLOGHISTORY 1815
LOGFILE
    GROUP 1 'C:\app\username\oradata\orcl64\REDO03.LOG' SIZE 1M,
    # was 'C:\app\username\oradata\orcl32\...LOG'
    # on the 32-bit computer
    GROUP 2 'C:\app\username\oradata\orcl64\REDO02.LOG' SIZE 1M,
    GROUP 3 'C:\app\username\oradata\orcl64\REDO01.LOG' SIZE 1M
DATAFILE
    'C:\app\username\oradata\orcl64\SYSTEM01.DBF',
    # was 'C:\app\username\oradata\orcl32\...DBF'
    # on the 32-bit computer
    'C:\app\username\oradata\orcl64\RBS01.DBF',
    'C:\app\username\oradata\orcl64\USERS01.DBF',
    'C:\app\username\oradata\orcl64\TEMP01.DBF',
    'C:\app\username\oradata\orcl64\TOOLS01.DBF',
    'C:\app\username\oradata\orcl64\INDX01.DBF',
    'C:\app\username\oradata\orcl64\DR01.DBF'
CHARACTER SET AL32UTF8;
```

12. Alter the `init` file from the 32-bit computer to include the new control file generated in the preceding step.
13. Shut down the database on the 64-bit computer:

```
SQL> SHUTDOWN IMMEDIATE;
```

14. Start the database migration:

```
SQL> STARTUP MIGRATE;
```

15. Migrate the database.

 **Note:**

Upgrading to Oracle Database 19c is supported only when the same Windows User Account is used as the Oracle Home User in both the source and destination Oracle homes, or when the home from which the database is being upgraded uses a Windows built-in account.

16. Shut down the database:

```
SQL> SHUTDOWN IMMEDIATE;
```

17. Restart the database:

```
SQL> STARTUP OPEN;
```

### Related Topics

- [Backing Up a 32-Bit Oracle Database](#)  
Describes how to back up a 32-bit Oracle home database.

 **See Also:**

- *Oracle Database Installation Guide for Microsoft Windows*
- *Tasks to Complete Only After Manually Upgrading Oracle Database* for more information about changing word size, *Upgrading Oracle Database* about migrating the database, and *Prerequisites for Preparing Oracle Home on Windows* for information about prerequisites for preparing a new Oracle home on Windows in *Oracle Database Upgrade Guide*

# 5

## Postinstallation Configuration Tasks on Windows

Learn about the configuration tasks that you can perform to increase security, and other configuration tasks before using Oracle Multimedia and other Oracle options.

### Note:

Directory path examples in this chapter follow Optimal Flexible Architecture (OFA) guidelines. If you specified non-OFA compliant directories during installation, then your directory paths differ. See Appendix B, "Optimal Flexible Architecture" in *Oracle Database Installation Guide for Microsoft Windows* for more information.

- [Overview of Windows Firewall](#)  
All newer Windows operating systems, by default enable the Windows Firewall to block virtually all TCP network ports to the incoming connections.
- [About the Need to Reset Passwords for Default Accounts](#)  
Oracle Database installs with many default accounts.
- [About Windows Authenticated Users](#)  
Authenticated Users group is a Windows built-in group that cannot be modified and includes all the users whose identities were authenticated when they logged on.
- [Overview of NTFS File System and Windows Registry Permissions](#)  
Oracle recommends that you configure Oracle Database files, directories, and registry settings to provide full control to authorized database administrators (DBAs).
- [Overview of ReFS File System](#)  
The ReFS prevents corruption of the file metadata that occurs in standard NTFS volumes which makes data inaccessible.
- [About Configuring External Job Support for the Scheduler on Windows](#)  
This release includes Oracle Scheduler (the Scheduler), which provides enterprise scheduling functionality.
- [About Oracle Multimedia on Windows](#)  
Oracle Multimedia (formerly Oracle interMedia) is a feature that enables Oracle Database to store, manage, and retrieve images.
- [About Oracle Text on Windows](#)  
Oracle Text enables text queries through SQL and **PI/SQL** from most Oracle interfaces.
- [About Oracle Spatial and Graph on Windows](#)  
Oracle Spatial and Graph makes storage, retrieval, and manipulation of spatial data easier and more intuitive to users.



- [About Advanced Replication on Windows](#)  
There are many configuration and usage possibilities with Advanced Replication.

## Overview of Windows Firewall

All newer Windows operating systems, by default enable the Windows Firewall to block virtually all TCP network ports to the incoming connections.

As a result, any Oracle products that listen for incoming connections on a TCP port do not receive any of those connection requests, and the clients making those connections report errors.

Depending upon which Oracle products are installed and how they are used, the products require some postinstallation configuration of the Windows Firewall to function on these operating systems.

- [About Oracle Executables Requiring Windows Firewall Exceptions](#)  
If the Oracle Database executables are in use and accepting connections from a remote client computer, then Oracle recommends that you add them to the Windows Firewall exceptions list to ensure correct operation.
- [Configuring the Windows Firewall](#)  
Oracle recommends configuring the Windows Firewall if the following conditions are true.
- [Troubleshooting Windows Firewall Exceptions](#)  
Perform the following steps to troubleshoot Windows Firewall exceptions.

## About Oracle Executables Requiring Windows Firewall Exceptions

If the Oracle Database executables are in use and accepting connections from a remote client computer, then Oracle recommends that you add them to the Windows Firewall exceptions list to ensure correct operation.

Except as noted, these Oracle executables can be found in the `ORACLE_HOME\bin` directory.

### Note:

If multiple Oracle homes are in use, then you need several firewall exceptions for the same executable: one for each home from which that executable loads.

- [Configuring Windows Firewall Exceptions for Successful Connections to Oracle Software](#)  
Learn about configuring Windows Firewall exceptions.
- [Overview of Different Executables Added to the Windows Firewall Exception List](#)  
Lists the executables that listen on TCP ports on Windows, along with a brief description of the executable.

 **See Also:**

*Oracle Real Application Clusters Installation Guide*

## Configuring Windows Firewall Exceptions for Successful Connections to Oracle Software

Learn about configuring Windows Firewall exceptions.

You must configure exceptions for the Windows Firewall if your system meets *all* of the following conditions:

- Oracle server-side components are installed on a Windows server operating system. The list of components includes Oracle Database, Oracle Grid infrastructure, network listeners, or any web servers or services.
- The Windows system in question accepts connections from other machines over the network. If no other machines connect to the Windows system to access the Oracle software, then no postinstallation configuration steps are required and the Oracle software functions as expected.
- The Windows system in question is configured to run the Windows Firewall. If the Windows Firewall is not enabled, then no postinstallation configuration steps are required.

If all the conditions are met, then the Windows Firewall must be configured to allow successful incoming connections to the Oracle software. To enable Oracle software to accept connection requests, Windows Firewall must be configured by either opening up the specific static TCP ports in the firewall or by creating exceptions for specific executables so they can receive the connection requests on any ports they choose. This firewall configuration can be done by one of the following methods:

- From the **Start** menu:
  1. Click **Run** and enter **firewall.cpl**. This opens the **Windows Firewall Control Panel** applet.
  2. Complete one of the following operating system-specific steps to allow a program through the Windows Firewall:
    - On Windows 8, Windows 8.1, Windows Server 2012, or Windows Server 2012 R2 x64, click **Allow an app or feature through Windows Firewall**. Click **Change Settings**.
    - On Windows 7 or Windows Server 2008 R2, click **Allow a program or feature through Windows Firewall**. Click **Change Settings**, **Allow Another Program**.
    - On Windows Server 2008, click **Allow a program through Windows Firewall**.
  3. On the **Exceptions** tab, click **Add Program** to create exceptions for the Oracle software.
- From the command prompt, use the `netsh firewall add...` command.

When Windows notifies you that a foreground application is attempting to listen on a port, and gives you the opportunity to create an exception for that executable, if you

choose to create the exception in this way, then the effect is the same as creating an exception for the executable either through Control Panel or from the command line.

## Overview of Different Executables Added to the Windows Firewall Exception List

Lists the executables that listen on TCP ports on Windows, along with a brief description of the executable.

Oracle recommends that these executables (if in use and accepting connections from a remote, client computer) be added to the exceptions list for the Windows Firewall to ensure correct operation. In addition, if multiple Oracle homes are in use, then create firewall exceptions for the same executable, for example, `oracle.exe`, multiple times, once for each Oracle home from which that executable loads.

- [About Firewall Exceptions for Oracle Database](#)  
For a basic database operation and connectivity from remote clients (SQL\*Plus, OCI, ODBC, OLE DB applications, and so on), add the following executables to the Windows Firewall exception list:
- [About Firewall Exceptions for Oracle Database Examples](#)  
After installing Oracle Database Examples, add the following executables to the Windows Firewall exception list:
- [About Firewall Exceptions for Oracle Gateways](#)  
If your Oracle database interacts with non-Oracle software through a gateway, then you must add the gateway executable to the Windows Firewall exception list.
- [About Firewall Exceptions for Oracle Clusterware and Oracle ASM](#)  
If you installed Oracle Grid Infrastructure on the nodes in your cluster, then you can enable the Windows Firewall only after adding the following executables and ports to the Firewall exception list.
- [About Firewall Exceptions for Other Oracle Products](#)  
In addition to all the previously listed exceptions, if you use any of the Oracle software listed, then you must create an exception for Windows Firewall for the associated executable.

### About Firewall Exceptions for Oracle Database

For a basic database operation and connectivity from remote clients (SQL\*Plus, OCI, ODBC, OLE DB applications, and so on), add the following executables to the Windows Firewall exception list:

- `Oracle_home\bin\oracle.exe` - Oracle Database executable
- `Oracle_home\bin\tnslsnr.exe` - Oracle Listener

For remote monitoring capabilities to be available for a database running on Windows, the following executables must be added to the Windows Firewall exception list:

- `Oracle_home\bin\emagent.exe` - Oracle Database Control
- `Oracle_home\jdk\bin\java.exe` - Java Virtual Machine

### About Firewall Exceptions for Oracle Database Examples

After installing Oracle Database Examples, add the following executables to the Windows Firewall exception list:

- *Oracle\_home\opmn\bin\opmn.exe* - Oracle Process Manager
- *Oracle\_home\jdk\bin\java.exe* - Java Virtual Machine

## About Firewall Exceptions for Oracle Gateways

If your Oracle database interacts with non-Oracle software through a gateway, then you must add the gateway executable to the Windows Firewall exception list.

**Table 5-1 Oracle Executables Requiring Windows Firewall Exceptions**

File Name	Executable Name
omtsreco.exe	Oracle Services for Microsoft Transaction Server
dg4sybs.exe	Oracle Database Gateway for Sybase
dg4tera.exe	Oracle Database Gateway for Teradata
dg4msql.exe	Oracle Database Gateway for SQL Server
dg4db2.exe	Oracle Database Gateway for DRDA
pg4arv.exe	Oracle Database Gateway for APPC
pg4t4ic.exe	Oracle Database Gateway for APPC
dg4mqs.exe	Oracle Database Gateway for WebSphere MQ
dg4mqc.exe	Oracle Database Gateway for WebSphere MQ
dg4odbc.exe	Oracle Database Gateway for ODBC

## About Firewall Exceptions for Oracle Clusterware and Oracle ASM

If you installed Oracle Grid Infrastructure on the nodes in your cluster, then you can enable the Windows Firewall only after adding the following executables and ports to the Firewall exception list.

The Firewall Exception list must be updated on each node.

- *Grid\_home\bin\gpnpd.exe* - Grid Plug and Play daemon
- *Grid\_home\bin\oracle.exe* - Oracle ASM executable (if using Oracle ASM for storage)
- *Grid\_home\bin\racgvip.exe* - Virtual Internet Protocol Configuration Assistant
- *Grid\_home\bin\evmd.exe* - OracleEVMSERVICE
- *Grid\_home\bin\crsd.exe* - OracleCRSERVICE
- *Grid\_home\bin\ocssd.exe* - OracleCSSERVICE
- *Grid\_home\bin\octssd.exe* - Cluster Time Synchronization Service daemon
- *Grid\_home\bin\mDNSResponder.exe* - multicast-DNS Responder Daemon
- *Grid\_home\bin\gipcd.exe* - Grid IPC daemon
- *Grid\_home\bin\gnsd.exe* - Grid Naming Service daemon
- *Grid\_home\bin\ohasd.exe* - OracleOHSERVICE

- *Grid\_home\bin\TNSLSNR.EXE* - SCAN listener and local listener for Oracle Database and Oracle ASM
- *Grid\_home\opmn\bin\ons.exe* - Oracle Notification Service
- *Grid\_home\jdk\jre\bin\java.exe* - Java Virtual Machine

## About Firewall Exceptions for Other Oracle Products

In addition to all the previously listed exceptions, if you use any of the Oracle software listed, then you must create an exception for Windows Firewall for the associated executable.

**Table 5-2 Other Oracle Software Products Requiring Windows Firewall Exceptions**

Oracle Software Product	Executable Name
Data Guard Manager	dgmgrl.exe
Oracle Internet Directory LDAP Server	oidldapd.exe
External Procedural Calls	extproc.exe

## Configuring the Windows Firewall

Oracle recommends configuring the Windows Firewall if the following conditions are true.

Configure the Windows Firewall if:

- Oracle server-side components are installed.  
These components include Oracle Database, network listeners, and any web servers or services.
- The computer handles connections from other computers over a network.  
If no other computers connect to the computer with the Oracle software, then no postinstallation configuration steps are required and the Oracle software functions as expected.
- The Windows Firewall is enabled.  
If the Windows Firewall is not enabled, then no postinstallation configuration steps are required.

If all of the conditions are met, then you must configure the Windows Firewall either by opening specific static TCP ports in the firewall or by creating exceptions for specific executables so that they are able to receive connection requests on any ports they choose. Postinstallation configuration for the Windows Firewall can be done by one of following methods:

- From the **Control Panel**, select **Windows Firewall** and then select **Exceptions**.
- Or enter `netsh firewall add...` at the command line.

Alternatively, Windows informs you if a foreground application is attempting to listen on a port, and it prompts you to create an exception for that executable. If you choose to do so, then the effect is the same as creating an exception for the executable either in the Control Panel or from the command line.

 **Note:**

Windows Server 2008 and later operating systems do not provide any information about applications attempting to listen on a port. Instead, a security audit event is logged to signal that an application is blocked.

- [About Backing Up a Database](#)

The technique for backing up a database depends on the archiving mode of the database and whether you are making a component-based or a volume-based backup.

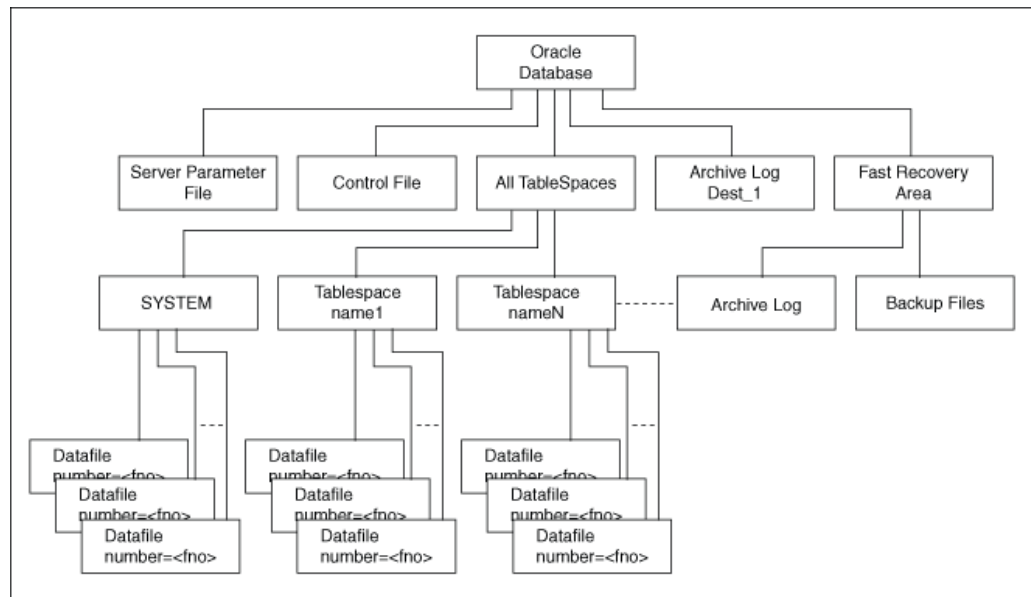
## About Backing Up a Database

The technique for backing up a database depends on the archiving mode of the database and whether you are making a component-based or a volume-based backup.

Oracle recommends shadow copies taken in a component mode for backing up the Oracle Database using VSS writer. The Oracle VSS writer defines the components that include the set of database files. The Oracle VSS writer then saves the redo generated during hot backup mode when the snapshot was created in the backup writer metadata document.

The component hierarchy defined by the Oracle VSS writer is illustrated in Oracle VSS Writer Component Hierarchy.

**Figure 5-1 Oracle VSS Writer Component Hierarchy**



### Related Topics

- [About Component-Based Backups](#)  
The components supported by the Oracle VSS writer are listed in Components Backed Up by the Oracle VSS Writer.

## Troubleshooting Windows Firewall Exceptions

Perform the following steps to troubleshoot Windows Firewall exceptions.

Perform the following steps to troubleshoot Windows Firewall exceptions:

1. Examine Oracle configuration files (such as \*.conf files), the Oracle key in the Windows registry, and network configuration files in `ORACLE_HOME\network\admin`.
2. Pay particular attention to any executable listed in `ORACLE_HOME\network\admin\listener.ora` in a `PROGRAM=` clause. Each of these must be granted an exception in the Windows Firewall, because a connection can be made through the TNS Listener to that executable.
3. Examine Oracle trace files, log files, and other sources of diagnostic information for details on failed connection attempts. Log and trace files on the database client computer might contain useful error codes or troubleshooting information for failed connection attempts. The Windows Firewall log file on the server might contain useful information as well.
4. If the preceding troubleshooting steps do not resolve a specific configuration issue, then provide the output from command `netsh firewall show state verbose=enable` to My Oracle Support for diagnosis and problem resolution at: <https://support.oracle.com/>

## About the Need to Reset Passwords for Default Accounts

Oracle Database installs with many default accounts.

Oracle Database Configuration Assistant locks and removes most default database accounts upon successful installation. Oracle recommends changing all user passwords *immediately* after installation.

### See Also:

*Oracle Database Administrator's Guide*

## About Windows Authenticated Users

Authenticated Users group is a Windows built-in group that cannot be modified and includes all the users whose identities were authenticated when they logged on.

Membership is controlled by the operating system. The SID for Authenticated Users is S-1-5-11.

## Overview of NTFS File System and Windows Registry Permissions

Oracle recommends that you configure Oracle Database files, directories, and registry settings to provide full control to authorized database administrators (DBAs).

If you have created a database using Oracle Database Configuration Assistant or upgraded a database using Oracle Database Upgrade Assistant, then no further action is required.

Learn about the permissions automatically set by Oracle Universal Installer, Oracle Database Configuration Assistant, and Oracle Database Upgrade Assistant and the steps to set these permissions manually.

In addition to the various groups listed in Oracle Database software installation creates the following groups for Oracle internal use and sets permissions on files and registry entries for these groups to ensure that the Oracle software functions properly. The group memberships and permissions set for the following groups must not be changed or removed:

- `ORA_INSTALL`
- `ORA_GRID_LISTENERS`
- `ORA_CLIENT_LISTENERS`
- `ORA_HOMENAME_SVCSIDS`
- [Setting File Permissions](#)  
Oracle Universal Installer, Oracle Database Configuration Assistant, and Oracle Database Upgrade Assistant set file permissions when you install or upgrade Oracle Database software.
- [Setting Permissions for Windows Registry Entries](#)  
Oracle Universal Installer sets the permissions for Windows registry entries pertaining to Oracle Database software.
- [Setting Permissions for Windows Service Entries](#)  
Oracle Universal Installer sets the following permissions to users and user groups for Windows service entries for Oracle Database services.
- [Setting NTFS File System Security](#)  
Use this procedure to set the NTFS file system security.
- [Setting Windows Registry Security](#)  
Oracle recommends that you remove write permissions from users who are *not* Oracle Database DBAs or system administrators in the `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE` directory of the Windows registry.



 **See Also:**

- Your operating system documentation for more information about modifying NTFS file system and Windows registry settings
- *Oracle Database Installation Guide for Microsoft Windows*

## Setting File Permissions

Oracle Universal Installer, Oracle Database Configuration Assistant, and Oracle Database Upgrade Assistant set file permissions when you install or upgrade Oracle Database software.

- [About Default File Permissions Set by Oracle Universal Installer](#)  
During Oracle Database installation, by default Oracle Universal Installer installs software in the *ORACLE\_HOME* directory.
- [About File Permissions Set by Oracle Database Configuration Assistant](#)  
During Oracle Database configuration, Oracle Database Configuration Assistant installs files and directories in the following default locations, where *database\_name* is the database name or *SID*.
- [About File Permissions Set by Oracle Database Upgrade Assistant](#)  
When an earlier version of the database is upgraded to Oracle Database 19c, Oracle Database Upgrade Assistant installs software in the following directories, where *database\_name* is the database name or *SID*.
- [About Setting Permissions for Oracle Wallets](#)  
When an Oracle Wallet is created in the file system, the user creating the wallet is granted access to the wallet by wallet creation tools.
- [About Setting File System ACLs Manually](#)  
As Oracle Database services now run under a standard Windows User Account, a file might not be accessible by Oracle Database services unless the file system Access Control Lists (ACLs) grant access to the file.

## About Default File Permissions Set by Oracle Universal Installer

During Oracle Database installation, by default Oracle Universal Installer installs software in the *ORACLE\_HOME* directory.

Oracle Universal Installer sets the following permissions to this directory, and to all files and directories under this directory:

### For the Oracle Grid Infrastructure home:

- Full control - Administrators, SYSTEM, ORA\_GRID\_LISTENERS, Oracle Installation User, Oracle Home User
- Read, execute, and list content - Authenticated Users

### For the Database ORACLE\_HOME:

- Full control - Administrators, SYSTEM, Oracle Installation User, Oracle Home User, or ORA\_<HomeName>\_SVCACCTS group for Virtual Account homes.

- Read, execute, and list content - Authenticated Users

#### For the Client ORACLE\_HOME:

- Full control - Administrators, SYSTEM, Oracle Installation User, ORA\_HOMENAME\_SVCSIDS or the Oracle Home User
- Read, execute, and list content - Authenticated Users

Oracle Universal Installer sets the following permissions to the ORACLE\_BASE directory, and to all the files and directories under this directory with the exception of database files, wallets, and so on:

- Full control - Administrators, SYSTEM, Oracle Installation User, Oracle Home User or ORA\_<HomeName>\_SVCACCTS group for Virtual Account homes.
- Full control - ORA\_GRID\_LISTENERS if the ORACLE\_BASE is for the Oracle Grid Infrastructure ORACLE\_HOME
- Full control - ORA\_HOMENAME\_SVCSIDS or Oracle Home User if the ORACLE\_BASE is for a Client ORACLE\_HOME

#### Note:

If these accounts already exist and have more restrictive permissions, then most restrictive permissions are retained. If accounts other than Administrators, SYSTEM, Authenticated Users, and the Oracle groups mentioned exist, then the permissions for these accounts are removed.

#### See Also:

- *Oracle Grid Infrastructure Installation Guide for Microsoft Windows x64 (64-Bit)*
- *Oracle Database Installation Guide for Microsoft Windows*

## About File Permissions Set by Oracle Database Configuration Assistant

During Oracle Database configuration, Oracle Database Configuration Assistant installs files and directories in the following default locations, where *database\_name* is the database name or *SID*.

- ORACLE\_BASE\admin\database\_name (administration file directories)
- ORACLE\_BASE\oradata\database\_name (database file directories)
- ORACLE\_BASE\oradata\database\_name (redo log files and control files)
- ORACLE\_HOME\database (SPFILESID.ORA)

Oracle Database Configuration Assistant sets the following permission to these directories, and to all the files and directories under these directories:

- Full control Administrators, SYSTEM, Oracle Home User or ORA\_<HomeName>\_SVCACCTS group for Virtual Account homes

 **Note:**

If these accounts already exist and have more restrictive permissions, then the most restrictive permissions are retained. If accounts other than Administrators, SYSTEM, and Oracle Home User already exist, then the permissions for these accounts are removed.

## About File Permissions Set by Oracle Database Upgrade Assistant

When an earlier version of the database is upgraded to Oracle Database 19c, Oracle Database Upgrade Assistant installs software in the following directories, where *database\_name* is the database name or *SID*.

When an earlier version of the database is upgraded to Oracle Database 19c, Oracle Database Upgrade Assistant installs software in the following directories, where *database\_name* is the database name or *SID*:

- `ORACLE_BASE\admin\database_name` (administration files)
- `ORACLE_BASE\oradata\database_name` (database file directories)
- `ORACLE_BASE\oradata\database_name` (redo log files and control files)
- `ORACLE_BASE\ORACLE_HOME\database` (SPFILESID.ORA)

Oracle Database Upgrade Assistant sets the following permissions to these directories, and to all files and directories under these directories:

- Full control Administrators, SYSTEM, Oracle Home User or ORA\_<HomeName>\_SVCACCTS group for Virtual Account homes

 **Note:**

If these accounts already exist and have more restrictive permissions, then the most restrictive permissions are retained. If accounts other than Administrators, SYSTEM, and Oracle Home User already exist, then the permissions for these accounts are removed.

Oracle Database Upgrade Assistant can also configure Oracle Enterprise Manager. If the **Enable daily backup** option is selected while configuring Oracle Enterprise Manager, then Oracle Database Upgrade Assistant shows a separate screen asking for Fast Recovery Area. Oracle Database Upgrade Assistant tries to create the directory structure (if it does not exist) in the specified file system location. Oracle Database Upgrade Assistant also puts the same set of file permissions to this location. The default location shown by Oracle Database Upgrade Assistant for Fast Recovery Area is:

- `ORACLE_BASE\recovery_area`

## About Setting Permissions for Oracle Wallets

When an Oracle Wallet is created in the file system, the user creating the wallet is granted access to the wallet by wallet creation tools.

Oracle Database Windows services may run under a standard Windows User Account or Virtual Account and might not be able to access to the wallet. You may need to change the file system ACL for the wallet file manually to grant access to database and listener services.

## About Setting File System ACLs Manually

As Oracle Database services now run under a standard Windows User Account, a file might not be accessible by Oracle Database services unless the file system Access Control Lists (ACLs) grant access to the file.

Though Oracle installation configures the ACLs in a way to ensure that you do not have to change ACLs manually for typical usage, it is necessary to change ACLs manually, for example, to manually upgrade databases, and database files not in Oracle base, or to grant access to wallets in the file system.

The rules to set file system ACLs manually are:

- To allow Oracle Database service access to a file: Grant access to Oracle Home User for the file when a Windows User Account is used as the Oracle Home User. If a Windows built-in account is used as the Oracle Home User, then no such permission is necessary because the Oracle Database services run under the administrative account.
- To allow Oracle Grid Listeners services access to a file: Grant access to `ORA_GRID_LISTENERS` group for the file.
- To allow Oracle services from a client `ORACLE_HOME` access to a file: Grant access to Oracle Home User for the file when a Windows User Account is used as the Oracle Home User for the client home. If a Windows built-in account is used as the Oracle Home User, then grant access to the `ORA_HOMENAME_SVCSIDS` group for the file.

## Setting Permissions for Windows Registry Entries

Oracle Universal Installer sets the permissions for Windows registry entries pertaining to Oracle Database software.

Follow the guidelines listed below to set the permissions for Windows registry entries:

- All users have read permissions.
- Local administrators and Oracle Installation User have full control.

## Setting Permissions for Windows Service Entries

Oracle Universal Installer sets the following permissions to users and user groups for Windows service entries for Oracle Database services.

The guidelines to set permissions to users and user groups for Windows service entries for Oracle Database services are:

- `ORA_DBA` and `ORA_HOMENAME_DBA` group users have start and stop privileges for Windows service entries.
- Local System Account and local administrators have full control of Windows service entries.

## Setting NTFS File System Security

Use this procedure to set the NTFS file system security.

To ensure that only authorized users have full file system permissions:

1. Go to Windows Explorer.
2. Set the following permissions for each directory or file based on the information provided in the earlier sections.

### See Also:

Your operating system online help for more information about how to modify NTFS file system and registry settings

## Setting Windows Registry Security

Oracle recommends that you remove write permissions from users who are *not* Oracle Database DBAs or system administrators in the `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE` directory of the Windows registry.

To remove write permissions:

1. Open the registry.
2. Go to `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE`.
3. Select **Permissions** from the **Edit** menu.  
The Permissions for Oracle dialog box appears.
4. Remove write privileges from any users who are not Oracle Database DBAs or system administrators. Note that the `SYSTEM` account must have Full Control, because some Oracle Database services run as `SYSTEM`.
5. Ensure that user accounts that must run Oracle applications have read privileges.
6. Select **OK**.
7. Exit the registry.

## Overview of ReFS File System

The ReFS prevents corruption of the file metadata that occurs in standard NTFS volumes which makes data inaccessible.

This release supports Oracle Database installation on Resilient File System (ReFS). ReFS uses checksums for file metadata, and an allocate-on-write method to update data which minimizes the risk of corruption.

ReFS supports volumes from 256 zettabytes to a maximum of 4 petabytes. ReFS stores and protects data from common errors that cause data loss. ReFS is resilient to power outages. ReFS also protects data based on volumes, directory, and files.

- [Setting File Permissions](#)

## Setting File Permissions

Oracle Universal Installer, Oracle Database Configuration Assistant, and Oracle Database Upgrade Assistant set file permissions when Oracle Database software is installed or upgraded.

This section contains these topics:

- [About Default File Permissions Set by Oracle Universal Installer](#)
- [About File Permissions Set by Oracle Database Configuration Assistant](#)
- [About Setting File System ACLs Manually](#)

## About Configuring External Job Support for the Scheduler on Windows

This release includes Oracle Scheduler (the Scheduler), which provides enterprise scheduling functionality.

This release includes Oracle Scheduler (the Scheduler), which provides enterprise scheduling functionality. External jobs performed by the user are started using the `OracleJobSchedulerSID` service. This service is disabled by default. To use the external jobs functionality, the administrator must set the user name and password for the user account under which this service must run and enable the service.

ORADIM creates the Oracle Database service, Oracle VSS Writer service, and Oracle Scheduler service to run under the Oracle Home User account. If this account is a Windows Local User or a Windows Domain User Account, then ORADIM prompts for the password for that account and accepts the same through `stdin`. It is possible to specify both the Oracle Home User and its password using the `-RUNAS osusr[/ospass]` option to `oradim`. If the given `osusr` is different from the Oracle Home User, then Oracle Home User is used instead of `osusr` along with the given `ospass`.

Restricting execution of external jobs to a low-privileged user prevents unauthorized database users from gaining operating system-level privileges, but it also places restrictions on the kinds of jobs that can be run. Jobs requiring a higher level of operating system privileges cannot be run by this mechanism.

Enabling and starting the `OracleJobSchedulerSID` service is required only for compatibility with Oracle Database 10g Release 1 and Release 2, for local external jobs that do not use credentials. This service is not required if all local external jobs use credentials. For improved security, Oracle recommends that all local external jobs use credentials.



**See Also:**

*Oracle Database Administrator's Guide*

## About Oracle Multimedia on Windows

Oracle Multimedia (formerly Oracle interMedia) is a feature that enables Oracle Database to store, manage, and retrieve images.

Oracle Multimedia also helps DICOM format medical images and other DICOM data, audio, video, or other heterogeneous media data in an integrated fashion with other enterprise information. Oracle Multimedia extends Oracle Database reliability, availability, and data management to multimedia content in traditional, Internet, electronic commerce, medical, and media-rich applications.

If you install Standard Edition 2, or Enterprise Edition, then Oracle Database Configuration Assistant starts automatically at the end of installation. If you choose any Oracle Database Configuration Assistant installation type other than Customized, then Oracle Multimedia does not require manual configuration. All tasks described in this section are performed automatically.

If you select Customized installation, then Oracle Database Configuration Assistant guides you through configuration of Oracle Multimedia.

- [Configuring Oracle Multimedia on Windows](#)  
Use this procedure to configure Oracle Multimedia.

## Configuring Oracle Multimedia on Windows

Use this procedure to configure Oracle Multimedia.

If you are creating and configuring a database manually, then you can configure Oracle Multimedia as follows:

1. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

2. Connect to Oracle Database with account SYSDBA:

```
SQL> CONNECT / AS SYSDBA
```

3. Start the database (if necessary):

```
SQL> STARTUP
```

4. Run the script `ordinst.sql`:

```
SQL> ORACLE_HOME\ord\admin\ordinst.sql SYS AUX SYS AUX
```

5. Run the script `iminst.sql`:

```
SQL> ORACLE_HOME\ord\im\admin\catim.sql
```

6. Exit SQL\*Plus:

```
SQL> EXIT
```

 **Note:**

If you manually copy your Oracle8i `listener.ora` and `tnsnames.ora` files into your Oracle Database network directory, then you must modify network configuration files `tnsnames.ora` and `listener.ora` on your server to enable calls to work and Oracle Multimedia to function properly.

 **See Also:**

*Oracle Net Services Administrator's Guide*

## About Oracle Text on Windows

Oracle Text enables text queries through SQL and **PI/SQL** from most Oracle interfaces.

Oracle Text enables text queries through SQL and **PI/SQL** from most Oracle interfaces. By installing Oracle Text with an Oracle Database server, client tools such as SQL\*Plus and Pro\*C/C++ are able to retrieve and manipulate text in Oracle Database.

Oracle Text manages textual data in concert with traditional data types in Oracle Database. When text is inserted, updated, or deleted, Oracle Text automatically manages the change.

If you install Oracle Text from the media and do not have a previous release of Oracle Text installed, then Oracle Database is already configured for use with Oracle Text if one of the following is true:

- You created the database by using Oracle Database Configuration Assistant in standalone mode, and selected the Typical database creation type.
- The database is a **starter database** that you created by using Oracle Universal Installer (OUI) and selected the **Create and configure a database** option in "Select Installation Option" window.

 **See Also:**

- *Oracle Text Application Developer's Guide*
- *Oracle Database Upgrade Guide*
- *Oracle Database Installation Guide for Microsoft Windows*

### Configuring Oracle Text Using Database Configuration Assistant

To use Oracle Database Configuration Assistant to configure Oracle Database for use with Oracle Text at the time you create the database, select Oracle Text as the option to configure when prompted.

To configure the database at a later time:



1. Start Database Configuration Assistant.  
From the **Start** menu, select **All Programs**, then select **Oracle - HOMENAME**, then select **Configuration and Migration Tools**, and then select **Database Configuration Assistant**.
2. Select **Configure Database Options**.
3. Select the database to modify when prompted.
4. Select **Oracle Text** as the option to configure when prompted.

## About Oracle Spatial and Graph on Windows

Oracle Spatial and Graph makes storage, retrieval, and manipulation of spatial data easier and more intuitive to users.

One example of spatial data is a road map. A road map is a two-dimensional object that contains points, lines, and polygons representing cities, roads, and political boundaries such as states. A road map represents geographic information. Locations of cities, roads, and political boundaries are projected onto a two-dimensional display or piece of paper, preserving relative positions and relative distances of objects.

- [Configuring Oracle Spatial and Graph on Windows Automatically](#)  
Learn how to configure Oracle Spatial and Graph automatically on Windows.

## Configuring Oracle Spatial and Graph on Windows Automatically

Learn how to configure Oracle Spatial and Graph automatically on Windows.

If you install Oracle Spatial and Graph through Enterprise Edition, then no manual configuration is required. All Oracle Spatial and Graph configuration tasks are performed automatically.

If you install both Oracle Spatial and Graph and Oracle Database together through Enterprise Edition or Standard Edition 2 installation, then Database Configuration Assistant starts automatically at the end of installation. If you select **Custom** installation and select **Create new database**, then the assistant asks if Oracle Spatial and Graph is to be configured automatically.

If you install Oracle Spatial and Graph during a separate installation from Enterprise Edition, then you must either start Oracle Database Configuration Assistant and select **Configure** database options or configure Oracle Spatial and Graph manually.



### See Also:

*Oracle Spatial and Graph Developer's Guide*

## About Advanced Replication on Windows

There are many configuration and usage possibilities with Advanced Replication.

Oracle Database installs packages and procedures automatically rather than as a separate manual process.

This section describes how to manually configure Advanced Replication in Oracle Database. Follow the instructions only if you add Advanced Replication to an installation of Oracle Database that was not previously configured with this feature.

Configuring Advanced Replication consists of the following steps:

- [About Checking Tablespace and Rollback Segment Requirements](#)
- [Adding and Modifying Initialization Parameters](#)
- [Monitoring Data Dictionary Tables](#)

## About Checking Tablespace and Rollback Segment Requirements

**Table 5-3 Advanced Replication Tablespace/Rollback Segment Requirements**

Tablespace/Rollback Segment	Minimum Free Space
SYSTEM	20 MB
UNDOTBS	10 MB
RBS	5 MB
TEMP	10 MB
USERS	No specific requirement

### Note:

Replication triggers and procedures are stored here.

### See Also:

*Oracle Database Administrator's Guide* for more information on tablespace

## Adding and Modifying Initialization Parameters

If you use Advanced Replication, then certain parameter values must be added to the initialization parameter file, and others must be set to recommended values.

**Table 5-4 Advanced Replication Initialization Parameters**

Parameter Name	Recommended Value	Site
JAVA_POOL_SIZE	50 MB	master
DISTRIBUTED_LOCK_TIMEOUT	300 seconds	master
GLOBAL_NAMES	TRUE	master
OPEN_LINKS	4	master
PROCESSES	Add 9 to current value	master

**Table 5-4 (Cont.) Advanced Replication Initialization Parameters**

Parameter Name	Recommended Value	Site
JOB_QUEUE-PROCESSES	2	master
JOB_QUEUE_PROCESSES	2	materialized view

**Note**

Depends on the number of n-way sites.

## Monitoring Data Dictionary Tables

If you use Advanced Replication and intend to set up a large number of replicated objects, then you are required to monitor the following data dictionary tables with the SQL `SELECT` argument:

- ARGUMENT\$
- IDL\_CHAR\$
- IDL\_UB1\$
- IDL\_UB2\$
- IDL\_SB4\$
- I\_ARGUMENT1
- I\_SOURCE1I\$
- SOURCE\$
- TRIGGER

If necessary, increase the storage parameters to accommodate storage requirements of large numbers of replicated objects.

# 6

## Administering a Database on Windows

Learn how to administer Oracle Database for Windows.

- [About Ways to Manage Oracle Database Services](#)  
Learn how to manage the services that Oracle Database installs on your computer.
- [Starting and Shutting Down a Database with SQL\\*Plus](#)  
Learn how to start and shut down a database with SQL \*Plus.
- [Starting and Shutting Down a Database Using Services](#)  
Learn how to start and shut down a database using services.
- [Starting Multiple Instances](#)  
Learn about how to start multiple database instances.
- [Creating and Populating Password Files](#)  
Use Password Utility to create password files. Password Utility is automatically installed with Oracle Database utilities.
- [Connecting Remotely to the Database](#)  
Learn how to connect to Oracle Database remotely.
- [About Archiving Redo Log Files](#)  
If you installed Oracle Database through the Typical installation, then it is created in the NOARCHIVELOG mode. If you created your database through the Custom option of Oracle Database Configuration Assistant, then you had the choice of either ARCHIVELOG or NOARCHIVELOG.

### About Ways to Manage Oracle Database Services

Learn how to manage the services that Oracle Database installs on your computer.

- [Overview of Oracle Database Service Naming Conventions for Multiple Oracle Homes](#)  
Oracle Database for Windows lets you have multiple Oracle homes on a single computer.
- [Starting Oracle Database Services](#)  
Oracle Database services must be started for you to use Oracle Database and its products.
- [Stopping Oracle Database Services](#)  
On occasion (for example, when reinstalling Oracle Database), you must stop Oracle Database services.
- [Auto-Starting Oracle Database Services](#)  
Oracle Database services can be set to start automatically whenever you start the Windows computer.

## Overview of Oracle Database Service Naming Conventions for Multiple Oracle Homes

Oracle Database for Windows lets you have multiple Oracle homes on a single computer.

This feature, described in Appendix B, "Optimal Flexible Architecture", in *Oracle Database Installation Guide for Microsoft Windows*, affects Oracle Services naming conventions. As you perform installations into Oracle home directories:

- You must accept the default Oracle home name provided or specify a different name for each Oracle home directory.
- You are prompted to give a system identifier and a global database name for each database installation.



### See Also:

*Oracle Database Installation Guide for Microsoft Windows*

## Starting Oracle Database Services

Oracle Database services must be started for you to use Oracle Database and its products.

You can start Oracle Database services by:

- Using the Control Panel
- Using the Command Prompt

### Using the Control Panel

To start Oracle Database services from the Control Panel:

1. Access your Windows Services dialog box.



### See Also:

Your operating system documentation for instructions

2. Find the service to start in the list, select it, and click **Start**.  
If you cannot find `OracleServiceSID` in the list, then use `ORADIM` to create it.
3. Click **Close** to exit the Services dialog box.

### Using the Command Prompt

To start Oracle Database services from the command prompt, enter:

```
C:\> NET START service
```

The variable `service` is a specific service name, such as `OracleServiceORCL`.

### Related Topics

- [Starting and Shutting Down a Database Using Services](#)  
Learn how to start and shut down a database using services.

## Stopping Oracle Database Services

On occasion (for example, when reinstalling Oracle Database), you must stop Oracle Database services.

You can stop Oracle Database services from three different locations:

- Using the Control Panel
- Using the Command Prompt

### Using the Control Panel

To stop Oracle Database services from the Control Panel:

1. Access your Windows Services dialog box.



#### See Also:

Your operating system documentation for instructions

2. Select `OracleHOMENAME(TNS)Listener` and click **Stop**.

`OracleHOMENAME(TNS)Listener` is stopped.

3. Select `OracleServiceSID` and click **Stop**.

4. Click **OK**.

`OracleServiceSID` is stopped.

### Using the Command Prompt

To stop Oracle Database services from the command prompt, enter:

```
C:\> net STOP service
```

The variable `service` is a specific service name, such as `OracleServiceORCL`.

### Related Topics

- [Starting and Shutting Down a Database Using Services](#)  
Learn how to start and shut down a database using services.

## Auto-Starting Oracle Database Services

Oracle Database services can be set to start automatically whenever you start the Windows computer.

You can turn auto-start on or off by using the Control Panel.

## Using the Control Panel

To use the Control Panel to configure when and how Oracle Database is started:

1. Access your Windows Services dialog box.

### See Also:

Your operating system documentation for instructions

2. Select the service **OracleServiceSID** and click **Startup**.
3. Select **Automatic** from the **Startup Type** field.
4. Click **OK**.
5. Click **Close** to exit the Services dialog box.

# Starting and Shutting Down a Database with SQL\*Plus

Learn how to start and shut down a database with SQL \*Plus.

These instructions assume that a database instance has been created.

### Note:

Directory path examples in this chapter follow Optimal Flexible Architecture (OFA) guidelines. If you specified directories during installation that do not comply with OFA guidelines, then your directory paths differ.

To start or shut down Oracle Database:

1. Go to your Oracle Database server.
2. Start SQL\*Plus at the command prompt:

```
C:\> sqlplus /NOLOG
```

3. Connect to Oracle Database with username SYSDBA:

```
SQL> CONNECT / AS SYSDBA
```

4. To start a database, enter:

```
SQL> STARTUP [PFILE=path\filename]
```

This command uses the initialization parameter file specified in *path\filename*. To start a database using a file named *init2.ora* located in *C:\app\username\product\11.2.0\admin\orcl\pfile*, enter:

```
SQL> STARTUP PFILE=C:\app\username\product\11.2.0\admin\orcl\pfile\init2.ora
```

If no *PFILE* is specified, then the command looks for an *SPFILE* in *ORACLE\_HOME\database*. If the command finds one, then the command uses it to start the

database. If it does not find an `SPFILE`, then it uses the default initialization parameter file located in `ORACLE_BASE\ADMIN\db_name\pfile`.

5. To stop a database, enter:

```
SQL> SHUTDOWN [mode]
```

The mode is `normal`, `immediate`, or `abort`.

In a `normal` shutdown, Oracle Database waits for all currently connected users to disconnect and disallows any new connections before shutting down. This is the default mode.

In an `immediate` shutdown, Oracle Database terminates and rolls back active transactions, disconnects clients, and shuts down.

In an `abort` shutdown, Oracle Database terminates active transactions and disconnects users; it does not roll back transactions. The database performs automatic recovery and rollback the next time it is started. Use this mode only in emergencies.

 See Also:

*Oracle Database Installation Guide for Microsoft Windows* for more information about "Optimal Flexible Architecture"

## Starting and Shutting Down a Database Using Services

Learn how to start and shut down a database using services.

You can start or shut down Oracle Database by starting or stopping the service `OracleServiceSID` in the Control Panel. Starting `OracleServiceSID` is equivalent to using the `STARTUP` command or manually entering:

```
C:\> oradim -STARTUP -SID SID [-STARTTYPE srvc | inst | srvc,inst] [-PFILE filename | -SPFILE]
```

Stopping `OracleServiceSID` is equivalent to using the `SHUTDOWN` command or manually entering:

```
C:\> oradim -SHUTDOWN -SID SID [-SHUTTYPE srvc | inst | srvc,inst] [-SHUTMODE normal | immediate | abort]
```

You can enable starting and stopping Oracle Database through `OracleServiceSID` by setting registry parameters.

### Setting Registry Parameters

To start or stop Oracle Database through Oracle Database services, set the following registry parameters to the indicated values:

- `ORA_SID_AUTOSTART`

When set to `true`, the default value, this parameter causes Oracle Database to start when `OracleServiceSID` is started.

- `ORA_SID_PFILE`



This parameter sets the full path to the initialization parameter file. If this entry is not present, then ORADIM tries to start the database with an `SPFILE` or `PFILE` from `ORACLE_HOME\database`.

- `ORA_SHUTDOWN`

When set to `true`, this parameter enables the selected instance of Oracle Database to be shut down when `OracleServiceSID` is stopped. This includes any database in the current Oracle home. The default value is `false`.

- `ORA_SID_SHUTDOWN`

When set to `true`, the default value, this parameter causes the instance of Oracle Database identified by the `SID` value to shut down when `OracleServiceSID` is stopped manually—using either the Control Panel or Net stop command.

 **Note:**

If `ORA_SHUTDOWN` or `ORA_SID_SHUTDOWN` is set to `false`, then manually shutting down `OracleServiceSID` still shuts down Oracle Database. But it is an abnormal shutdown, and Oracle does not recommend it.

The following two registry parameters are optional:

- `ORA_SID_SHUTDOWNTYPE`

This parameter controls database shutdown mode. Set it to `a` (abort), `i` (immediate), or `n` (normal). The default mode is `i` (immediate) if you do not set this parameter.

- `ORA_SID_SHUTDOWN_TIMEOUT`

This parameter sets the maximum time to wait before the service for a particular `SID` stops.

The registry location of these required and optional parameters are determined by the number of Oracle home directories on your computer. If you have only one Oracle home directory, then these parameters belong in:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0
```

If you have multiple Oracle home directories, then these parameters belong in:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOMEID
```

The variable `ID` is incremented for each additional Oracle home directory on your computer.

 **Note:**

If you use ORADIM to create or edit instances, then it automatically sets the relevant registry parameters to their appropriate values.

### Starting or Stopping OracleServiceSID from the Control Panel

1. To start the database, start **OracleServiceSID**.

This automatically starts ORADIM and enters the `-STARTUP` command using the initialization parameter file identified by `ORA_SID_PFILE`.

2. To stop the database, stop **OracleServicesID**.

This automatically stops ORADIM, which enters the `-SHUTDOWN` command in the mode indicated by `ORA_SID_SHUTDOWNTYPE`, and shuts down Oracle Database.

#### See Also:

Your operating system documentation for instructions on starting and stopping services.

#### Related Topics

- [Configuration Parameters and the Registry](#)

## Starting Multiple Instances

Learn about how to start multiple database instances.

Perform the following steps to start service for multiple Oracle Database instance:

1. Start the service for each instance using ORADIM or the Services dialog of the Control Panel.
2. At the command prompt, set the `ORACLE_SID` configuration parameter to the `SID` for the first instance to run:

```
C:\> SET ORACLE_SID=SID
```

The variable `SID` is the name of the Oracle Database instance.

3. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

4. Connect AS SYSDBA:

```
SQL> CONNECT / AS SYSDBA
```

5. Start up the first instance:

```
SQL> STARTUP PFILE=ORACLE_BASE\admin\db_name\pfile\init.ora
```

The variable `ORACLE_BASE` is `c:\app\username` (unless you changed it during installation) and `db_name` is the name of the instance.

6. Repeat Step 2 through Step 5 for the other instances to run.

## Creating and Populating Password Files

Use Password Utility to create password files. Password Utility is automatically installed with Oracle Database utilities.

Password files are located in the directory `ORACLE_HOME\database` and are named `PWDsid.ora`, where `SID` identifies the Oracle Database instance. Password files can be used for local or remote connections to Oracle Database.

To create and populate a password file:

**1. Create a password file with Password Utility:**

```
C:\> orapwd FILE=PWDsid.ora ENTRIES=max_users
```

- `FILE` specifies the password file name.
- `SID` identifies the database instance.
- `ENTRIES` sets the maximum number of entries in the password file. This corresponds to maximum number of distinct users allowed to connect to the database simultaneously with either the `SYSDBA` or the `SYSOPER` DBA privilege.

**2. Set the initialization parameter file parameter `REMOTE_LOGIN_PASSWORDFILE` to `exclusive`, `shared`, or `none`.**

The value `exclusive` specifies that only one instance can use the password file and that the password file contains names other than `SYS`. In search of the password file, Oracle Database looks in the registry for the value of the parameter `ORA_SID_PWFIL`. If no value is specified, then Oracle Database looks in the registry for the value of the parameter `ORA_PWFIL`, which points to a file containing user names, passwords, and privileges. If that is not set, then Oracle Database uses the default:

```
ORACLE_HOME\DATABASE\PWDsid.ORA.
```

The default value is `shared`. It specifies that multiple instances (for example, an Oracle RAC environment) can use the password file. However, the only user recognized by the password file is `SYS`. Other users cannot log in with `SYSOPER` or `SYSDBA` privileges even if those privileges are granted in the password file. The `shared` value of this parameter affords backward compatibility with earlier Oracle releases. Oracle Database looks for the same files as it does when the value is `exclusive`.

The value `none` specifies that Oracle Database ignores the password file and that authentication of privileged users is handled by the Windows operating system.

**3. Start SQL\*Plus:**

```
C:\> sqlplus /NOLOG
```

**4. Connect AS `SYSDBA`:**

```
SQL> CONNECT / AS SYSDBA
```

For an Oracle ASM instance, connect AS `SYSASM`:

```
SQL> CONNECT / AS SYSASM
```

**5. Start Oracle Database:**

```
SQL> STARTUP
```

**6. Grant appropriate privileges to each user. Users who must perform database administration, for example, are granted the `SYSDBA` privilege:**

```
SQL> GRANT SYSDBA TO db_administrator;
```

For an Oracle ASM instance:

```
SQL> GRANT SYSASM TO SYS;
```

If the grant is successful, then the following message is displayed:

```
Statement Processed.
```

This adds `smith` to the password file and enables `smith` to connect to the database with `SYSDBA` privileges. Use SQL\*Plus to add or delete user names, user passwords, and user privileges in password files.

 **Note:**

Copying or manually moving password files might result in ORADIM being unable to find a password to start an instance.

- [Viewing and Hiding the Password File](#)  
Use this procedure to make the password file visible or invisible from different locations.

## Viewing and Hiding the Password File

Use this procedure to make the password file visible or invisible from different locations.

The password file is not automatically hidden. It can be made invisible and visible again from two different locations:

- Using Command Prompt
- Using Windows Explorer

 **Note:**

The password file must be visible before it can be moved, copied, or deleted.

### Using Command Prompt

1. To see the password file, enter:

```
ORACLE_HOME\database> attrib
```

The password file is displayed as `PWDsid.ora`:

```
A      ORACLE_HOME\database\oradba.exe
A      ORACLE_HOME\database\oradim.log
A      ORACLE_HOME\database\PWDsid.ora
A      ORACLE_HOME\database\SPFILEsid.ora
```

2. To make the password file invisible, enter:

```
ORACLE_HOME\database> attrib +H PWDsid.ora
```

3. To see the effect of the change, enter:

```
ORACLE_HOME\database> attrib
```

The password file is now hidden:

```
A      ORACLE_HOME\database\oradba.exe
A      ORACLE_HOME\database\oradim.log
A  H   ORACLE_HOME\database\PWDsid.ora
A      ORACLE_HOME\database\SPFILEsid.ora
```

4. To make the password file visible again, enter:

```
ORACLE_HOME\database> attrib -H PWDsid.ora
```

### Using Windows Explorer

To make the password file invisible or visible again:

1. Go to the directory `ORACLE_HOME\database`.
2. Right-click **PWDsid.ora**.
3. Select **Properties**.  
The `PWDsid.ora` Properties dialog box opens.
4. In **Attributes**, check or clear the check box next to **Hidden**.
5. Click **OK**.

To view or hide an invisible password file:

1. Go to the directory `ORACLE_HOME\database`.
2. Select **Folder Options** from the **Tools** main menu.
3. In the Folder Options window, select the **View** tab.
4. To view an invisible password file, select **Show hidden files and folders**.
5. To hide a visible password file, select **Do not show hidden files and folders**.
6. Click **OK**.

## Connecting Remotely to the Database

Learn how to connect to Oracle Database remotely.

There are many steps you must remember while connecting to the database remotely.

- [Connecting to a Database Using SYSDBA Privileges](#)  
When connecting to the starter database from a remote computer as `SYS`, you must use a different password from the one described in *Oracle Database Installation Guide for Microsoft Windows* when logging on with `SYSDBA` privileges.
- [About Verifying a Remote Database Using Encrypted Passwords](#)  
Learn how to verify a remote database using encrypted passwords.

## Connecting to a Database Using SYSDBA Privileges

When connecting to the starter database from a remote computer as `SYS`, you must use a different password from the one described in *Oracle Database Installation Guide for Microsoft Windows* when logging on with `SYSDBA` privileges.

This is because the password file enables database access in this situation and it requires the password `oracle` for this purpose.

## About Verifying a Remote Database Using Encrypted Passwords

Learn how to verify a remote database using encrypted passwords.

With Oracle Database, the password used to verify a remote database connection is automatically encrypted. Whenever a user attempts a remote login, Oracle Database encrypts the password before sending it to the remote database. If the connection fails, then the failure is noted in the operating system audit log.

### Note:

The configuration parameter `ORA_ENCRYPT_LOGIN` is retained for backward compatibility and is set to `true` by default.

### Related Topics

- [Configuration Parameters and the Registry](#)

## About Archiving Redo Log Files

If you installed Oracle Database through the Typical installation, then it is created in the `NOARCHIVELOG` mode. If you created your database through the Custom option of Oracle Database Configuration Assistant, then you had the choice of either `ARCHIVELOG` or `NOARCHIVELOG`.

In `NOARCHIVELOG` mode, redo logs are not archived. Setting your archive mode to `ARCHIVELOG` and enabling automatic archiving causes redo log files to be archived. This protects Oracle Database from both instance and disk failure.

### See Also:

*Oracle Database Administrator's Guide* for more information about "Managing Archived Redo Logs."

# 7

## Monitoring a Database on Windows

Learn how to monitor Oracle Database for Windows.

- [Overview of Database Monitoring Tools](#)  
Database Monitoring Tools describes tools that enable you to monitor Oracle Database.
- [About Event Viewer](#)  
Oracle Database for Windows problems and other significant occurrences are recorded as events in an application event log.
- [About Trace Files](#)  
Oracle Database for Windows background threads use trace files to record occurrences and exceptions of database operations, and errors.
- [About Alert Logs](#)  
Alert logs contain important information about error messages and exceptions that occur during database operations.
- [Viewing Oracle Database Thread Information](#)  
To view information about Oracle Database threads using Oracle Administration Assistant for Windows, you must either enable Windows native authentication for the database or run the utility `ocfgutil.exe` with arguments *username* and *password*.

## Overview of Database Monitoring Tools

Database Monitoring Tools describes tools that enable you to monitor Oracle Database.

**Table 7-1 Database Monitoring Tools**

Tool	Functionality
Event Viewer	Monitor database events.
Trace Files	Record occurrences and exceptions of the database operations.
Alert Logs	Record important information about error messages and exceptions during database operations.
Oracle Enterprise Manager Database Management Packs	Monitor and tune using tools with a real-time graphical performance information. <b>See Also:</b> Your Oracle Enterprise Manager documentation set for more information
Oracle Administration Assistant for Windows	View information about or terminate any Oracle thread.

 **Note:**

A 64-bit version of Oracle Enterprise Manager Database Express is available on 64-bit Windows. Oracle Enterprise Manager Database Express can manage a 32-bit Windows database from a remote Linux or Windows 64-bit computer.

 **See Also:**

*Oracle Database Performance Tuning Guide*

## About Event Viewer

Oracle Database for Windows problems and other significant occurrences are recorded as events in an application event log.

View and manage these recorded events in Event Viewer.

- [Using Event Viewer](#)  
Learn how to use Event Viewer.
- [Managing Event Viewer](#)  
Learn how to manage Event Viewer.
- [Reading Event Viewer](#)  
Learn how to read an Event Viewer.

## Using Event Viewer

Learn how to use Event Viewer.

To access Event Viewer:

1. From the **Start** menu, select **All Programs**, then select **Administrative Tools**, and then select **Event Viewer**.  
The Event Viewer window appears.
2. Select **Windows Logs**.
3. Double-click **Application** to open the Application view window.

Application View Window displays the Application view window, Application View Definitions shows what is recorded in each column, and Event Viewer Icons interprets icons that appear on the left hand side of the viewer.



Figure 7-1 Application View Window

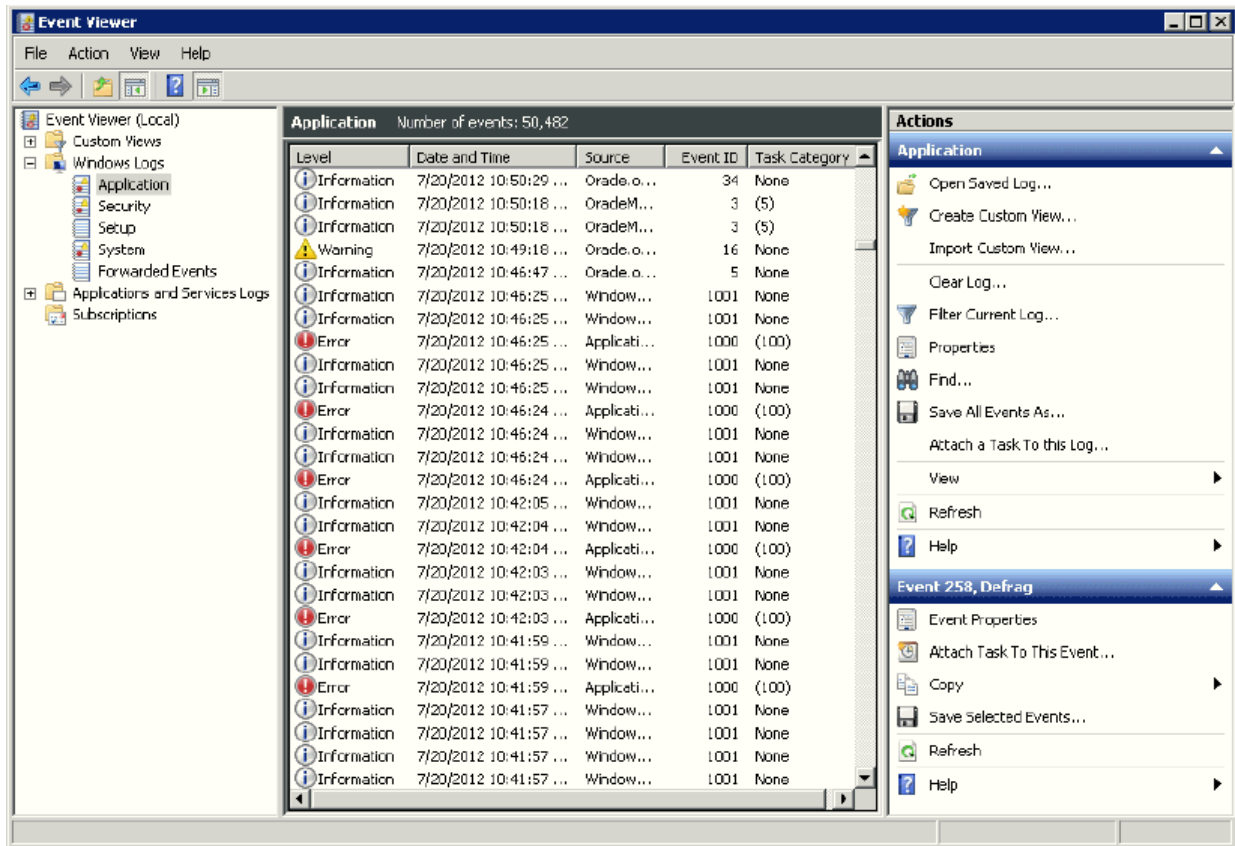


Table 7-2 Application View Definitions

Column Name	Definition
Date and Time	Date and time at which an event took place
Source	Application that recorded an event
Event ID	Unique number assigned to an event
Task Category	Classification of events

Table 7-3 Event Viewer Icons

Icon	Event Type	Suggested Action
Exclamation Point in Red Circle	Error	Error identification. Always check these icons.
Lowercase "i" in Blue Circle	Information	Noncritical system events. Check these icons only to track a specific event.
Exclamation Point in Yellow Triangle	Warning	Special events, such as instance termination or services shutdown. Investigate these icons, but they are usually noncritical.

## Managing Event Viewer

Learn how to manage Event Viewer.

Setting `AUDIT_TRAIL` to `db` or `os` causes more records to be written to Event Viewer. This can fill up the Event Viewer log file. Follow these procedures to increase log file size:

1. Right-click the event log in which you want to set size, and select **Properties**.  
The event Log Properties window appears.
2. Use the up and down arrow keys to set the size you want in the **Maximum log size** box.
3. Under **When maximum event log size is reached**, select one of the options that you want. The options are as follows:
  - Overwrite events as needed (oldest events first)
  - Archive the log when full, do not overwrite events
  - Do not overwrite events (clear log manually)
4. If you want to clear the log contents, click **Clear Log**.
5. Click **OK**.

You return to Event Viewer.

### Note:

Audit information cannot be spooled to a file. `AUDIT_FILE_DEST` is supported on Windows to write XML format audit files when `AUDIT_TRAIL` is set to `XML` or `XML, EXTENDED` format and thus must be added to the initialization parameter file.

## Reading Event Viewer

Learn how to read an Event Viewer.

Oracle Database for Windows events are displayed with a source of `Oracle.SID`.

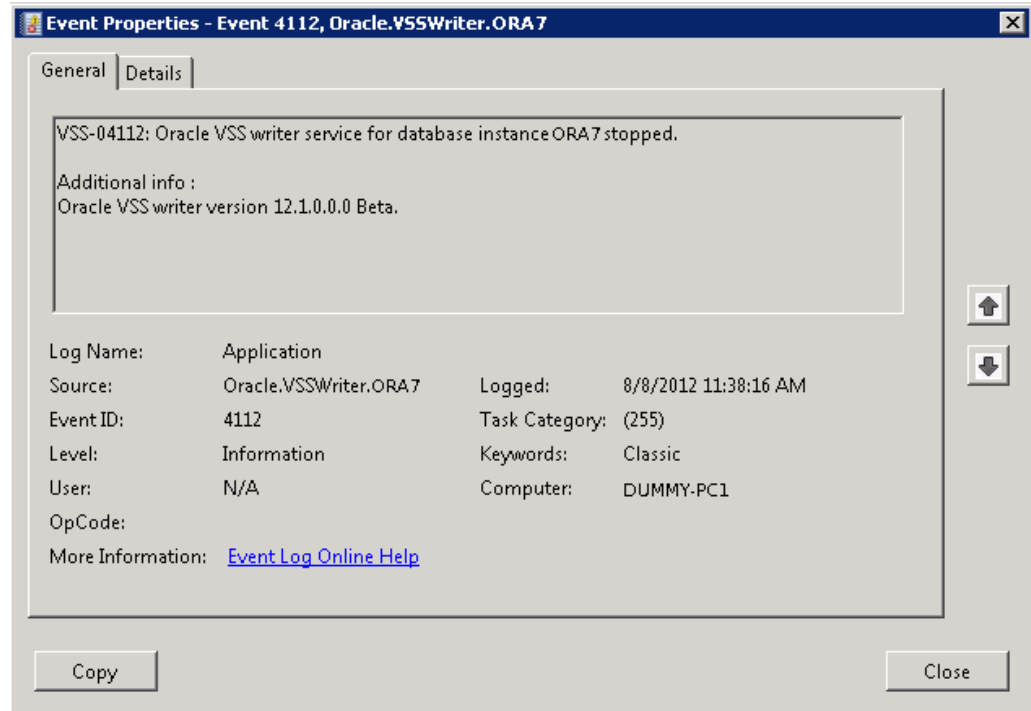
Event number 34 specifies an audit trail event. These events are recorded if the parameter `AUDIT_TRAIL` is set to `db` (`true`) or `os` in the initialization parameter file. Option `os` enables systemwide auditing and causes audited records to be written to Event Viewer. Option `db` enables systemwide auditing and causes audited records to be written to the database audit trail (table `SYS.AUD$`). Some records, however, are written to Event Viewer.

Event numbers other than 34 specify general database activities, such as an instance being started or stopped.

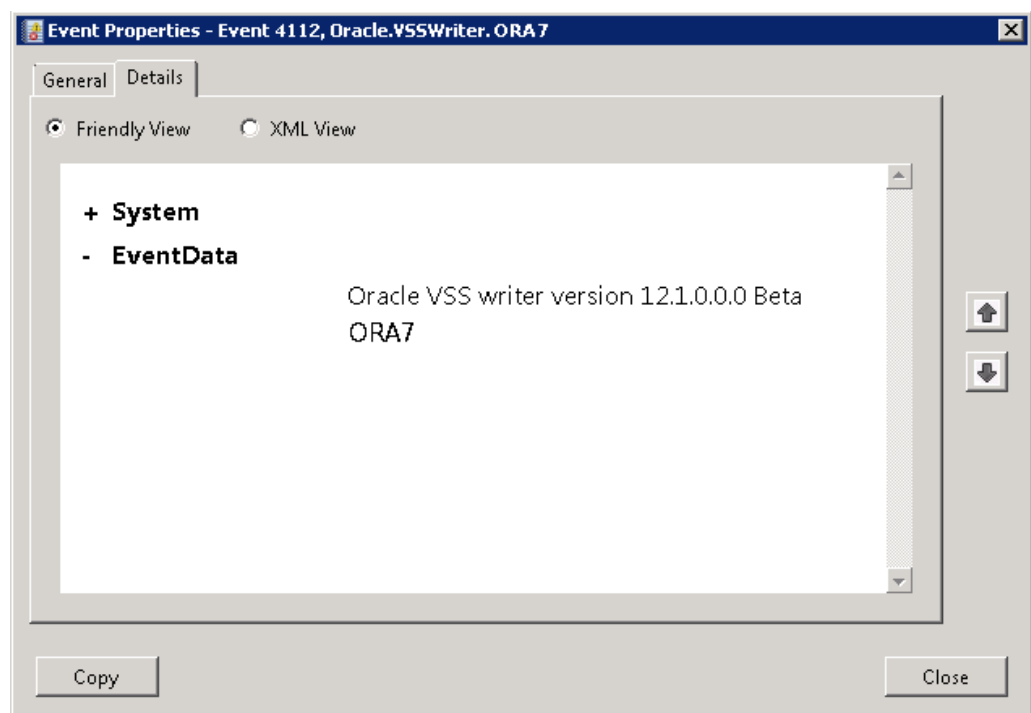
When you double-click an icon in Event Viewer, the Event Properties dialog box appears with more information about the selected event. Event Properties General Tab, for example, shows details about Event ID 4112. In the **General** tab, you find a text description of the event. In the **Details** tab, you can select **Friendly View** to see

the System and Event Data in words or **XML View** to see the same information in XML format, as shown in Event Properties Details Tab.

**Figure 7-2 Event Properties General Tab**



**Figure 7-3 Event Properties Details Tab**



**See Also:**

Microsoft operating system documentation for more information about using Event Viewer

## About Trace Files

Oracle Database for Windows background threads use trace files to record occurrences and exceptions of database operations, and errors.

Background thread trace files are created and stored in the Automatic Diagnostic Repository (ADR) directory specified by the parameter `DIAGNOSTIC_DEST` in the initialization parameter file.

Oracle Database creates a different trace file for each foreground and background thread. The name of the trace file contains the name of the thread, followed by the extension ".trc". The following are examples of foreground trace file names:

- `ops_ora_5804.trc`
- `ops_ora_4160.trc`

The following are the examples of the background trace file names:

- `ops_pmon_1556.trc`
- `ops_mmon_3768.trc`
- `ops_lgwr_2356.trc`
- `ops_dbw0_132.trc`

Trace files are also created for user threads and stored in the ADR directory specified by the parameter `DIAGNOSTIC_DEST` in the initialization parameter file. Trace files for user threads have the form `oraxxxxx.trc`, where `xxxxx` is a 5-digit number indicating the Windows thread ID.

## About Alert Logs

Alert logs contain important information about error messages and exceptions that occur during database operations.

Each Oracle Database for Windows instance has one alert log; information is appended to the file each time you start the instance. All threads can write to the **alert log**.

For example, when automatic archiving of redo logs is halted because no disk space is available, a message is placed in the alert log. The alert log is the first place to check if something goes wrong with the database and the cause is not immediately obvious.

The alert log is named `alert_SID.log` and is found in the ADR directory specified by the parameter `DIAGNOSTIC_DEST` in the initialization parameter file. Alert logs must be deleted or archived periodically.

### Related Topics

- [Modifying the Initialization Parameter File](#)  
Describes how to modify the initialization parameter file.

#### See Also:

*Oracle Database Installation Guide for Microsoft Windows* in the "ADMIN Directory" section.

## Viewing Oracle Database Thread Information

To view information about Oracle Database threads using Oracle Administration Assistant for Windows, you must either enable Windows native authentication for the database or run the utility `ocfgutil.exe` with arguments `username` and `password`.

The utility stores the user name and password in the following registry location:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraConfig
```

When the Windows native authentication is not enabled, Oracle Remote Configuration Agent retrieves the user name and password from this registry key to log in to the database.

To view information about Oracle Database threads using Oracle Administration Assistant for Windows:

1. From the **Start** menu, select **All Programs**, then select **Oracle - HOMENAME**, then select **Configuration and Migration Tools**, and then select **Administration Assistant for Windows**.
2. Right-click the *SID*, where *SID* is a specific instance name, such as `orcl`.
3. Select **Process Information**.

The Process Information dialog box appears, listing name, type, user, thread ID, and CPU usage for each Oracle Database thread.

4. To terminate a thread, select it, and click **Kill Thread**.

This list displays information about Oracle threads.

Name	Type	User	Thread ID	CPU	%
PMON	Background	SYS	1596	0:00:02	0%
DBW0	Background	SYS	1512	0:00:02	0%
LGWR	Background	SYS	1556	0:01:56	0%
<b>CKPT</b>	Background	SYS	<b>1552</b>	<b>0:00:21</b>	<b>0%</b>
SMON	Background	SYS	1520	0:00:45	0%
RECO	Background	SYS	1516	0:00:00	0%
CJQ0	Foreground	SYS	1572	0:00:05	0%

Buttons: Kill Thread, OK, Help

# 8

## Tuning Windows to Optimize Oracle Database

Learn how to tune the Windows Server operating system to ensure that Oracle Database is running in the best possible environment.

### Note:

For the sake of brevity, this discussion uses the collective term *Windows Server* when a statement applies to all supported Windows operating systems.

- [Overview of Windows Tuning](#)  
Learn about tuning Windows Server operating system.
- [Overview of Large Page Support](#)  
Large page support is a feature of Oracle Database.
- [About Reducing Priority of Foreground Applications on Server Console](#)  
One of the settings provided by default during the installation of a Windows Server gives the interactive foreground applications a priority over every background process.
- [About Configuring Windows Server to Be an Application Server](#)  
Windows memory manager divides up system memory into three different pools, described in Windows Server Memory Shares.
- [About Disabling Unnecessary Services](#)  
After you have significantly reduced the file cache, you can retrieve additional physical memory for Oracle Database by disabling services not needed for core operating system functionality.
- [About the Necessity to Remove Unused Network Protocols](#)  
Remove all unnecessary network protocols on Windows so that the processing time can be concentrated on servicing only critical protocols.
- [About the Necessity to Reset Network Protocol Bind Order](#)  
If multiple protocols must be installed on the server, you can give the protocol most frequently used by Oracle Database the highest priority by resetting the network protocol bind order.
- [Setting the Order of Multiple Network Interface Cards](#)  
If you have public and private network interface cards (NICs) on a single Windows computer and they are not in the correct order, then you might experience problems with any configuration (Oracle Enterprise Manager, for example) that uses `gethostname`.

- [Overview of the Latest Reliable Windows Server Service Pack](#)  
Microsoft releases operating system patches, called Service Packs, on a quarterly basis. Service Packs are a collection of bug fixes and product enhancements to the basic Windows Server release.
- [Overview of Hardware or Operating System Striping](#)  
Learn about data striping, which is an effective means of reducing the impact of slow hard drives.
- [About Multiplex Windows Server Virtual Memory Paging File](#)  
Discusses about Multiplexing the Windows Server virtual memory paging file to boost system performance.
- [Closing All Unnecessary Foreground Applications](#)  
Learn about closing all unnecessary foreground applications.

## Overview of Windows Tuning

Learn about tuning Windows Server operating system.

Windows Server operating systems offer considerably fewer tuning adjustments than UNIX systems. This difference constrains system administrators when they try to optimize Windows Server performance, but it also makes Windows Server easier to use.

You can make Windows Server a better application server environment for Oracle Database. Most of the operating system specific procedures described in this chapter enable Oracle Database to reserve more system resources, such as CPU, memory, and disk I/O.

In addition, because Oracle Database is a high-performance database management system that effectively uses resources of your Windows computer, it must not also serve as any of the following:

- Primary or backup domain controller
- File or print server
- Remote access server
- Router

These configurations consume network, memory, and CPU resources. In addition, the Windows computer that is running Oracle Database should not be locally accessed with a high frequency or intensively used for local user processing, unless it has enough resources to accommodate all this activity.

## Overview of Large Page Support

Large page support is a feature of Oracle Database.

It provides a performance boost for memory-intensive database instances running on Windows Server. By taking advantage of newly introduced operating system support, Oracle Database can now make more efficient use of processor memory addressing resources. Specifically, when large page support is enabled, the CPUs in the system access the Oracle Database buffers in RAM more quickly. Instead of addressing the buffers in 4KB increments, the CPUs are told to use 2 MB page sizes in Physical Address Extension (PAE) mode and 4MB page sizes in non-PAE mode when addressing the database buffers.



This feature is particularly useful when the Oracle buffer cache is several gigabytes. Smaller-sized configurations still see a gain when using large pages, but the gain will not be as great as when the database is accessing large amounts of memory.

If the service is running as a user instead of the default `SYSTEM` user, then the administrator must grant the "Lock pages in memory" privilege to the user. This privilege is not enabled by default when Windows is installed.

- [Granting Lock Pages in Memory Privilege](#)  
Use this procedure to grant lock pages in memory privilege.
- [Enabling Large Page Support](#)  
To take advantage of large pages, the amount of physical memory must be greater than the amount of System Global Area (SGA) specified in the parameter file.

## Granting Lock Pages in Memory Privilege

Use this procedure to grant lock pages in memory privilege.

To grant `lockMemoryPrivilege`, perform the following steps:

1. From the **Start** menu, select **Control Panel**.  
The Control Panel window opens.
2. Double-click **Administrative Tools**.  
The Administrative Tools window opens.
3. Double-click **Local Security Policy**.  
The Local Security Policy window opens.
4. In the left pane of the Local Security Policy window, expand **Local Policies** and select **User Rights Assignment**.
5. In the right pane of the Local Security Policy window, double-click **Lock pages in memory**.  
The Lock pages in memory Properties window opens.
6. Click **Add User or Group**.  
The Select Users, Computers, Service Accounts, or Groups dialog box opens.
7. Enter Oracle Home User name in **Enter the object names to select** field and click **Check Names**.
8. Click **OK** to close the Select Users, Computers, Service Accounts, or Groups dialog box.
9. Click **OK** to close the Lock pages in memory Properties window.

## Enabling Large Page Support

To take advantage of large pages, the amount of physical memory must be greater than the amount of System Global Area (SGA) specified in the parameter file.

Large pages might not be allocated always during instance startup. Large pages are supported in 2 modes:

- Regular mode: All of the SGA is attempted to be allocated in large pages. If the required amount of large pages are not available, then the instance does not come up.
- Mixed mode: All of the SGA is attempted to be allocated in large pages. If no more large pages are available, then the subsequent allocations are done using regular pages. So the SGA allocation can be a mixed set of large pages and regular pages.

The mixed mode also supports a time parameter (in `msecs`). If a large page allocation took more time than the `msecs` specified by this time parameter, then subsequent allocations are made using regular pages. This parameter is helpful when the database startup time might be too long due to the entire SGA being allocated using large pages.

 **Note:**

In regular mode, large page usage locks the entire SGA into physical memory. Physical memory is not released during a shrink operation. In mixed mode, only the large pages from the SGA are locked in physical memory, not released during a shrink operation and regular pages remain pageable.

 **See Also:**

Your operating system documentation for restrictions on allocating large pages

To enable large page support:

1. Go to the directory `ORACLE_HOME\bin\oracle.key`.
2. Open the `oracle.key` in a text editor and record the value found. It is set by Oracle Universal Installer. The default is:

```
SOFTWARE\ORACLE\KEY_HOMENAME
```

3. Start Registry Editor at the command prompt:

```
C:\> regedit
```

 **Note:**

Although Registry Editor lets you view and modify registry keys and parameter values, you usually are not required to do so. In fact, you can render your system useless if you make incorrect changes. Therefore, only advanced users must edit the registry. Back up your system before making any changes in the registry.

`oracle.key` file must not be modified or removed. Oracle binaries open it to determine the location in the registry where their variables are stored.

4. Go to the `HKEY_LOCAL_MACHINE` file.

Locate the key corresponding to the value found in the `oracle.key` file. In the default case, for example, locate:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME
```

5. Create *one* of the following, depending on where you want to enable large page support:

- `ORA_LPENABLE` to enable large page support on all instances. Its value decides the mode of large pages for all Oracle database instances on the computer.
- `ORA_SID_LPENABLE` to enable large page support on a specific instance. Its value decides the mode of large pages for the specific database instance.

Set the value of the above registry entry to 1 for regular mode and 2 for mixed mode.

The time parameter for mixed mode is optional. To specify this time parameter which is instance specific, create `ORA_SID_LPMAXTIME` and set the value to the desired number of milliseconds.

When this time parameter is specified for an instance and the large pages allocation takes more than the specified milliseconds, then the rest of the SGA is allocated using regular pages.

6. Exit Registry Editor.

By default, Oracle allocates the minimum available large page size when using large pages. The minimum available large page size, 16 MB, is obtained by using the `GetLargePageMinimum` function.

 **Note:**

Do not set the initialization parameter `lock_sga` when large pages are enabled. Large page usage locks the entire SGA into physical memory. When used with the parameter `lock_sga`, database startup fails with an error because the operating system automatically locks. That is, it prevents memory from being paged to disk when large pages are requested. Physical memory is not released during a shrink operation.

## About Reducing Priority of Foreground Applications on Server Console

One of the settings provided by default during the installation of a Windows Server gives the interactive foreground applications a priority over every background process.

To prevent foreground applications on the server console from taking excessive processor time away from Oracle Database, you can reduce the priority for foreground applications.



**See Also:**

Your operating system documentation for instructions on reducing priority of foreground applications

## About Configuring Windows Server to Be an Application Server

Windows memory manager divides up system memory into three different pools, described in Windows Server Memory Shares.

**Table 8-1 Windows Server Memory Shares**

Pool	Percent of Total Memory
Kernel and other system services	9%
File cache	41%
Paged memory	50%

Windows Server memory manager tries to balance each application's usage of memory by dynamically paging memory between a physical RAM and a virtual memory paging file. If an application is particularly memory-intensive (like Oracle Database) or if a large number of applications run concurrently, then combined memory requirements of the applications might exceed physical memory capacity.

The large proportion of memory reserved for file caching (41%) can be quite beneficial to the file and print servers. But it might not be advantageous to the application servers that often run memory-intensive network applications. A Windows Server file cache is particularly unnecessary for Oracle Database, which performs its own caching through the System Global Area.

You can reset the Windows Server memory model from the default file and print server, with its large file cache, to a network applications model, with a reduced file cache and more physical memory available for Oracle Database.



**See Also:**

Your operating system documentation for instructions

## About Disabling Unnecessary Services

After you have significantly reduced the file cache, you can retrieve additional physical memory for Oracle Database by disabling services not needed for core operating system functionality.

Unnecessary services include:

- License Logging Service
- Plug and Play
- Remote Access Autodial Manager
- Remote Access Connection Manager
- Remote Access Server
- Telephony Service

Consult with your systems administrator to identify other unnecessary services.

Do *not* disable any of the following services:

- Alerter
- Computer Browser
- EventLog
- Messenger
- OracleServiceSID
- OracleHOMENAME\_TNSListener
- Remote Procedure Call (RPC) Service
- Server
- Spooler
- TCP/IP NetBS Helper
- Workstation

#### Related Topics

- [About Configuring Windows Server to Be an Application Server](#)  
Windows memory manager divides up system memory into three different pools, described in Windows Server Memory Shares.

 **See Also:**

Your operating system documentation for instructions on disabling unnecessary services

## About the Necessity to Remove Unused Network Protocols

Remove all unnecessary network protocols on Windows so that the processing time can be concentrated on servicing only critical protocols.



#### See Also:

Your operating system documentation for instructions on removing unnecessary network protocols

## About the Necessity to Reset Network Protocol Bind Order

If multiple protocols must be installed on the server, you can give the protocol most frequently used by Oracle Database the highest priority by resetting the network protocol bind order.



#### See Also:

Your operating system documentation for instructions on resetting the network protocol bind order

## Setting the Order of Multiple Network Interface Cards

If you have public and private network interface cards (NICs) on a single Windows computer and they are not in the correct order, then you might experience problems with any configuration (Oracle Enterprise Manager, for example) that uses `gethostname`.

If the private NIC is seen first by Windows, then a `gethostname` call returns the host name of the private interconnect. Whichever tool is calling `gethostname` has configuration or connectivity problems stemming from this nonpublic network information.

You can determine the current order of your NICs, and change it if necessary, as follows:

1. From the **Start** menu, select **Control Panel**.
2. Select **Network and Internet**, and then select **Network and Sharing Center**.
3. Change the network adapter settings, depending on your operating system:
  - For Windows Server 2008, select **Manage adapter settings**.
  - For Windows Server 2008 R2, select **Change adapter settings**.
4. Click **Organize**, then **Layout**, and then select **Menu bar**.
5. From the **Advanced** menu, click **Advanced Settings**. The Advanced Settings window opens.
6. From the Connections window in the Adapters and Binding tab, select the network adapter that you want.
7. Move this network adapter to the top or the bottom of the list by using the up and down arrow buttons.
8. Click **OK**.

## Overview of the Latest Reliable Windows Server Service Pack

Microsoft releases operating system patches, called Service Packs, on a quarterly basis. Service Packs are a collection of bug fixes and product enhancements to the basic Windows Server release.

In general, apply Service Packs as soon as it is safe to do so, because they fix bugs and can improve Windows Server performance or functionality.

While Service Packs are supposed to fix bugs, they sometimes introduce new problems as well. In general, it is safest to wait a few weeks after a Service Pack is released before implementing it. This allows time for other field sites to report any problems with the Service Pack release.

The latest Windows Server Service Packs can be downloaded as self-extracting archives from <http://support.microsoft.com>

Unless you can be certain that the Service Pack works without flaws on Windows Server, create an `Uninstall` directory. This enables the Service Pack to be removed and the original configuration to be restored.

Service Pack files overwrite similarly-named files in the previous Windows Server configuration. However, Service Pack files can be overwritten in turn by setup programs that copy files from the original installation media.

For example, installing a new network protocol or a printer driver usually requires copying files from the original Windows Server installation media. When Service Pack files are comprehensively or selectively overwritten, the Service Pack must be reapplied.

## Overview of Hardware or Operating System Striping

Learn about data striping, which is an effective means of reducing the impact of slow hard drives.

Compared to CPU and memory speeds, hard disk drives are extremely slow. Now that the hard disk drives are relatively inexpensive, Oracle recommends that Windows Server use logical volumes composed of striped physical disks. Data striping is an effective means of reducing the impact of relatively slow hard drives by distributing file I/O across a number of hard drives simultaneously.

### Note:

Data striping through the operating system is not permitted with Oracle Automatic Storage Management (Oracle ASM), which does its own striping. Hardware striping is allowed with Oracle ASM, but it is not necessary.

Striping data across a number of disks is one example of a redundant array of inexpensive disks (RAID). There are several different types of RAID, also referred to as RAID levels, ranging from high performance to high reliability. The three most

common RAID levels in Oracle Database installations are RAID-0, RAID-1, and RAID-5. Descriptions of each RAID level are in RAID Levels in Oracle Database Installations, which shows each level's read and write penalties.

**Table 8-2 RAID Levels in Oracle Database Installations**

RAID Level	Read Penalty Note 1	Write Penalty Note 2
0 (Disk Striping)	1:1	1:1
1 (Disk Mirroring)	1:1	2:1
0 + 1	1:1	2:1
5 (Distributed Data Gathering)	1:1	4:1

**Note 1**

Read penalty is the ratio of I/O operations to read requests.

**Note 2**

Write penalty is the ratio of I/O operations to write requests.

**About Disk Striping**

RAID level 0 enables high-performance, fault-intolerant disk striping. Multiple physical hard disks are aggregated into a logical whole, either by a disk controller or through the operating system. Data operations against the logical volume are broken down into as many chunks as there are physical drives in the array, making simultaneous use of all disks. Given identical hard disks, if one hard disk has a throughput rate of `DISKRATE` operations/second, then a RAID 0 logical volume has a rate of:

`(DISKRATE * [number of physical drives in array]) operations/second`

The downside of RAID 0 is its lack of fault tolerance. If one disk in the logical volume fails, the whole volume fails and must be restored from a backup.

**About Disk Mirroring**

RAID level 1 enables fault-tolerant disk mirroring with some chance of a performance penalty. Essentially, every write to a mirrored disk is duplicated on another drive dedicated to this purpose (the mirror drive). If the mirrored disk fails, the mirror drive is brought online in real time. After the faulty drive is replaced, the mirror configuration can be reestablished.

The read penalty for RAID level 1 is nominally 1:1, but it might benefit from split reads on some controllers. When the controller knows which mirror can be accessed fastest, for example, it can lower search times by directing I/O operations to that disk.

**About Disk Striping Plus Mirroring**

RAID level 0+1 enables mirroring of an array of striped hard disks. This is a blend of RAID 0 and RAID 1, offering high-performing fault tolerance.

**About Distributed Data Guarding**

RAID level 5, also known as disk striping with parity, eliminates the costly requirement to mirror. In RAID 5, multiple hard disks are aggregated into a striped logical volume, similar to RAID 0, but each drive contains parity information such that any single drive



failure is tolerated. With one failed drive, a RAID-5 system can allow continued access to data, although access times are greatly reduced due to on-the-fly rebuilding of bytes from parity information. RAID-5 solutions usually allow hot-swapping of faulty drives with replacements, triggering a rebuild of the failed drive's data onto the replacement from parity information.

The write penalty of 4:1 results from 2 reads and 2 writes during parity calculation.

- [About Multiple Striped Volumes for Sequential and Random Access](#)  
If there are enough physical disks in Windows Server, create at least two striped volumes (in addition to a standalone hard disk or striped volume for the operating system).

 **See Also:**

*Oracle Database Administrator's Guide* in "Using Automatic Memory Management."

## About Multiple Striped Volumes for Sequential and Random Access

If there are enough physical disks in Windows Server, create at least two striped volumes (in addition to a standalone hard disk or striped volume for the operating system).

One striped volume can be used for sequential data access, and the other can be used for random data access.

Oracle Database redo logs and archived redo logs, for example, are written in sequential order. Because of a reduced head movement, hard disks perform best when reading or writing sequential data.

Oracle Database data files, however, are usually accessed in random order. Random access in a hard disk results in significant head movement, translating to slower data access.

Unless redo logs are separated from data files (at physical device level), undo file I/O contention may result, increasing access times for both types of files.

## About Multiplex Windows Server Virtual Memory Paging File

Discusses about Multiplexing the Windows Server virtual memory paging file to boost system performance.

Some virtual memory paging is likely even if Oracle Database is the only network application running on Windows Server, because Windows Server memory manager attempts to move seldom-used pages to disk to free up more physical memory for hot pages.

Multiplexing the Windows Server virtual memory paging file is a good strategy to boost overall system performance. Splitting the paging file onto at least two different physical volumes (or logical volumes as long as underlying physical volumes do not overlap) provides a significant performance boost to virtual memory swapping operations.

Even though this is a good technique to increase speed of virtual memory paging, too much paging activity is still a performance hit and must be corrected by adding more RAM to the server.

#### About General Page File Sizing Tip

Oracle recommends setting virtual memory to one times the size of RAM, if physical memory is between 2GB and 16GB. If physical memory is more than 16 GB, then set virtual memory to 16 GB.

Configurations where combined size is two to four times the size of physical RAM are not uncommon. Minimize paging as much as possible. But situations in which the operating system runs out of or low on paging space are to be avoided at all costs. Adequately-sized paging files spaced across physical disks spread out I/O most efficiently, because the operating system spreads paging evenly across page files.



#### Note:

Internal read/write batch size for Windows is 4K.

## Closing All Unnecessary Foreground Applications

Learn about closing all unnecessary foreground applications.

After applying procedures from the previous sections, remember to close any unnecessary foreground applications by:

- Removing all applications from Startup folders of Windows Server console operators
- Minimizing the window when executing long-running scripts from a command prompt, so that Windows Server can focus on the operation and not on a flood of window repaint messages
- Disabling screen savers, which can quickly saturate the CPU. If a screen saver must be run, choose Blank Screen, which uses the least amount of processing time

# 9

## Performing Database Backup and Recovery with VSS

Learn how to use Volume Shadow Copy Service (VSS) applications to back up and recover an Oracle Database.

- [Overview of Database Backup and Recovery with VSS](#)  
Learn about the basic concepts and tasks involved in backup and recovery with component-based shadow copies.
- [Basic Concepts of Database Backup and Recovery with VSS](#)  
VSS is an infrastructure on Windows server platforms that enables applications to create shadow copies.
- [Basic Steps of Backup and Recovery with VSS](#)  
Learn how to perform the basic steps of backup and recovery with VSS.
- [About Installing and Uninstalling the Oracle VSS Writer Service](#)  
The Oracle VSS writer runs separately from the Oracle database instance. From the perspective of the database, the VSS writer is simply an OCI client.
- [About Backing Up a Database](#)  
The technique for backing up a database depends on the archiving mode of the database and whether you are making a component-based or a volume-based backup.
- [About Restoring and Recovering a Database](#)  
Learn how to restore and recover VSS snapshots. As in the case of backups, the procedure depends on the archiving mode of the database and the type of snapshot that you are restoring.
- [About Integrating VSS with Third-Party Requester Applications](#)  
Oracle VSS writer allows third-party requester applications to control the behavior of recovery and backup sessions.
- [About Duplicating a Database](#)  
If your VSS shadow copies are transportable, then you can use these shadow copies to duplicate the primary database.

### Overview of Database Backup and Recovery with VSS

Learn about the basic concepts and tasks involved in backup and recovery with component-based shadow copies.

- [Purpose of Database Backup and Recovery with VSS](#)  
VSS provides a Windows-specific interface that enables coordination between requesters that back up data, writers that update data on disk, and providers that manage storage.
- [Scope of This Chapter](#)  
Learn how to perform database backup and recovery in the VSS infrastructure.

## Scope of This Chapter

Learn how to perform database backup and recovery in the VSS infrastructure.

This chapter assumes that you are familiar with VSS applications and the Oracle Database backup and recovery principles and techniques. This chapter does not attempt to provide an introduction to backup and recovery.



### See Also:

*Oracle Database Backup and Recovery User's Guide*

## Purpose of Database Backup and Recovery with VSS

VSS provides a Windows-specific interface that enables coordination between requesters that back up data, writers that update data on disk, and providers that manage storage.

Oracle Database functions as a writer that is integrated with VSS-enabled applications.

You can use VSS-enabled software and storage systems on Windows to back up and restore an Oracle Database. A key benefit is the ability to use a VSS-enabled application to make an online backup of the whole database.

## Basic Concepts of Database Backup and Recovery with VSS

VSS is an infrastructure on Windows server platforms that enables applications to create shadow copies.

A shadow copy is a consistent snapshot of the data held on a volume or component at a well-defined point in time. A shadow copy set is a collection of shadow copies that are all taken at the same time. VSS identifies each shadow copy and shadow copy set by a persistent Global Unique Identifier (GUID).

VSS provides the following infrastructure for running VSS applications:

- Coordinates activities of requesters, providers, and writers in the creation and use of shadow copies
- Furnishes the default system provider
- Implements low-level driver functionality necessary for any provider to work

A VSS requester is an application that requests VSS services to create shadow copies. Typically, VSS requesters are backup applications. Requesters communicate with writers to gather system data and signal writers to prepare data for backup.

A VSS provider manages storage volumes and creates shadow copies on demand. In response to a requester, a provider generates COM events to signal applications of an impending shadow copy and creates and maintains this copy until it is no longer

needed. During the life cycle of the shadow copy, the provider effectively supports two independent copies: the disk that is actively updated and a fixed copy that is stable for backup.

A VSS writer is an application or a service that writes data to a disk and cooperates with VSS providers and requesters. During backups, writers ensure that data is in the proper state for a shadow copy.

The Oracle VSS writer is a Windows service that coordinates an Oracle Database instance and other VSS components. The writer service, which is started under the user account with `SYSDBA` privileges, runs separately from the database instance. If the writer service runs as a non administrator user, then the user should also be a part of Backup Operators group. You must use third-party requesters to perform backup and recovery within the VSS infrastructure.

As explained in the following sections, the Oracle VSS writer supports both volume-based and component-based shadow copies. You can use these shadow copies in a backup and recovery strategy or to create a copy of your original database. You can use the duplicate database for testing or as a standby database.

- [Component-Based Shadow Copies](#)  
The Oracle VSS writer supports component-based shadow copies, which are sets of database files.
- [Volume-Based Shadow Copies](#)  
The Oracle VSS writer supports volume-based shadow copies, which are snapshots of complete drive or volumes.
- [Oracle VSS Backup Types](#)  
Oracle VSS writer supports log, copy, full, differential, and incremental backups.

## Component-Based Shadow Copies

The Oracle VSS writer supports component-based shadow copies, which are sets of database files.

The recommended technique for backing up an Oracle Database with VSS writer is to create shadow copies of components. During a backup, the Oracle VSS writer saves the redo generated during snapshot creation in a metadata document. During a restore operation, the writer automatically extracts the redo from the metadata document and applies it to files restored from a snapshot.

## Volume-Based Shadow Copies

The Oracle VSS writer supports volume-based shadow copies, which are snapshots of complete drive or volumes.

Oracle Database places the files that it manages in a state suitable to create shadow copies. For example, the data files are placed in hot backup mode and a new snapshot control file is created for a database in `ARCHIVELOG` mode. Oracle VSS writer excludes files such as the current control file and online redo logs from the shadow copies. The writer also returns an error if the snapshot cannot be taken. For example, if a `NOARCHIVELOG` database is open in read/write mode, then the writer returns an error indicating that the snapshot is not possible.



**Note:**

Oracle Automatic Storage Management files and raw files are not supported for Oracle VSS snapshots.

## Oracle VSS Backup Types

Oracle VSS writer supports log, copy, full, differential, and incremental backups.

The VSS writer uses time stamp mechanism for incremental and differential backups and stores a time stamp in the backup document using `SetBackupStamp()` API. This backup stamp is used by Oracle VSS writer during incremental or differential backups to specify changed files since the last full or incremental backup using `AddDifferencedFilesByLastModifyTime()` API.

Oracle VSS writer also stores backup metadata and restore metadata, which must be available during restore operations so that the VSS writer can perform intelligent postrestore operations. In case of full or copy backup, the restore metadata contains important redo information to make the restored files consistent. Hence, it is imperative that Oracle VSS writer is called during restore operations to perform the recovery operations.

## Basic Steps of Backup and Recovery with VSS

Learn how to perform the basic steps of backup and recovery with VSS.

The Oracle VSS writer is installed automatically as part of the database.

In the most typical backup scenario, you select the `Oracle Database` component in your VSS-enabled application and create a shadow copy. The shadow copy contains the database files, control files, and server parameter file. If the database is in `ARCHIVELOG` mode, then you can create the shadow copy when the database is open or closed; otherwise, only when closed.

In a typical recovery scenario, you select the `Oracle Database` component in your VSS-enabled application and restore it. Afterward, you can open the database either in read-only mode or with the `RESETLOGS` option. The Oracle VSS writer also supports applications that perform point-in-time recovery.

To restore a subset of database files, you can select individual components and restore them. The Oracle VSS writer performs the appropriate actions automatically in the postrestore phase so that the file can be used (or brought online) at the end of restore operation. For example, if you select a data file component for restore, then the writer automatically recovers the data file by using `RMAN`.

### Related Topics

- [About Installing and Uninstalling the Oracle VSS Writer Service](#)  
The Oracle VSS writer runs separately from the Oracle database instance. From the perspective of the database, the VSS writer is simply an OCI client.

- [About Backing Up a Database](#)  
The technique for backing up a database depends on the archiving mode of the database and whether you are making a component-based or a volume-based backup.
- [About Restoring and Recovering a Database](#)  
Learn how to restore and recover VSS snapshots. As in the case of backups, the procedure depends on the archiving mode of the database and the type of snapshot that you are restoring.

## About Installing and Uninstalling the Oracle VSS Writer Service

The Oracle VSS writer runs separately from the Oracle database instance. From the perspective of the database, the VSS writer is simply an OCI client.

Oracle VSS writer instances are created automatically during the setup of an instance. `oradim.exe` utility that sets up an instance, also starts Oracle's VSS writer utility to setup VSS writer instance for managing the given Oracle instance. In addition, the Oracle VSS writer provides command-line options to install and uninstall the writer service. If `/user` option is used but `/password` option not used, then `oravssw` waits for password through `stdin`. During installation, you can specify the Windows account under which the service must be started. The writer uses operating system authentication when connecting to a database instance. Thus, the Windows user must be able to log in as `SYSDBA` to the Oracle database instances managed by the writer service.

Oracle VSS is supported on the same operating systems that are supported by Oracle Database. See *Oracle Database Installation Guide for Microsoft Windows* for the list of supported operating systems.

The command-line syntaxes for the Oracle VSS writer are as follows:

```
oravssw {/q [/start | /stop | /status]}|
oravssw {SID [/tl trace_level] [/tf trace_file]}|
oravssw {SID [/i {/user:userid /password:password}]}|
oravssw {SID [/d]}
```

### Note:

You can change the user ID and password using the Services snap-in.

**Table 9-1 Oracle VSS Writer Options**

Option	Description
<i>SID</i>	<i>SID</i> of the Oracle instance to which the service connects.
<i>/i {/user:userid/ password:password}</i>	Installs Oracle VSS writer service for a specified <i>SID</i> .
<i>/q</i>	Queries the Oracle VSS writer services. But when not used with options like <i>/start</i> or <i>/status</i> or <i>/stop</i> , it just displays the list of Oracle VSS writer services.

**Table 9-1 (Cont.) Oracle VSS Writer Options**

Option	Description
<code>/status</code>	Displays the current status of all Oracle writer services and can be used only with the <code>/q</code> option.
<code>/start</code>	Starts all Oracle VSS writer services and can be used only with the <code>/q</code> option.
<code>/stop</code>	Stops all Oracle VSS writer services and can be used only with the <code>/q</code> option.
<code>/tl</code>	Specifies the trace level for Oracle VSS writer for a specified <i>SID</i> .
<code>/tf</code>	Specifies the trace file name for Oracle VSS writer for a specified <i>SID</i> .
<code>/d</code>	Uninstalls Oracle VSS writer service for a specified <i>SID</i> .

In Installing Oracle VSS Writer, you install the service so that it connects to the `prod1` instance.

 **Note:**

- Any errors during operation of the Oracle VSS writer are reported by means of Windows System Event logging APIs. You can view these errors with the Windows Event Viewer.
- Oracle Database 10g Release 2 supports Oracle VSS snapshots only when Oracle VSS writer 11g or later is configured to manage the 10.2 database. See My Oracle Support Note 580558.1 at <https://support.oracle.com> for more information about installing Oracle VSS writer for use with 9i and 10g databases.

**Example 9-1 Installing Oracle VSS Writer**

```
oravssw prod1 /i
```

## About Backing Up a Database

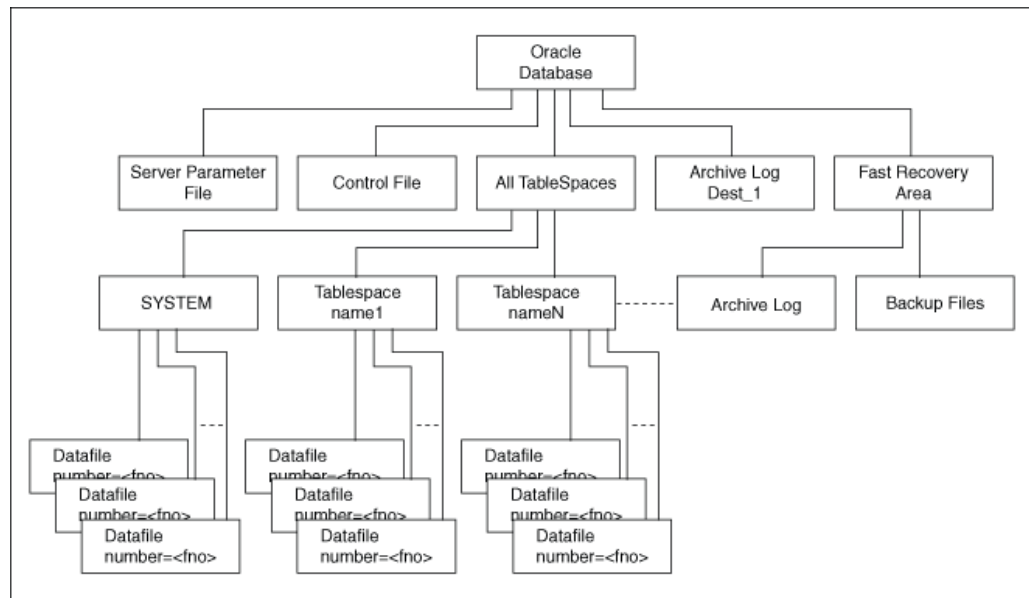
The technique for backing up a database depends on the archiving mode of the database and whether you are making a component-based or a volume-based backup.

Oracle recommends shadow copies taken in a component mode for backing up the Oracle Database using VSS writer. The Oracle VSS writer defines the components that include the set of database files. The Oracle VSS writer then saves the redo generated during hot backup mode when the snapshot was created in the backup writer metadata document.

The component hierarchy defined by the Oracle VSS writer is illustrated in Oracle VSS Writer Component Hierarchy.



**Figure 9-1 Oracle VSS Writer Component Hierarchy**



- [About Component-Based Backups](#)  
The components supported by the Oracle VSS writer are listed in Components Backed Up by the Oracle VSS Writer.
- [About Backing Up a Database in ARCHIVELOG Mode](#)  
Learn about backing up a database in ARCHIVELOG mode.
- [About Backing Up a Database in NOARCHIVELOG Mode](#)  
Learn about backing up a database in NOARCHIVELOG mode.

**Related Topics**

- [About Component-Based Backups](#)  
The components supported by the Oracle VSS writer are listed in Components Backed Up by the Oracle VSS Writer.

## About Component-Based Backups

The components supported by the Oracle VSS writer are listed in Components Backed Up by the Oracle VSS Writer.

The name of the component is the value returned by an `OnIdentify` VSS message. The `Selectable for Backup` column indicates whether a component is eligible to be selected in VSS shadow copies.

**Table 9-2 Components Backed Up by the Oracle VSS Writer**

Component	Description	Selectable for Backup
Oracle Database	Contains the database files, control files, and server parameter file.	Yes

**Table 9-2 (Cont.) Components Backed Up by the Oracle VSS Writer**

Component	Description	Selectable for Backup
Control File	Contains the snapshot location of the control file for a database running in ARCHIVELOG mode, or the current control file locations when all database files are recovered to a consistent SCN.  <b>Note:</b> The files included in the Control File component determine whether current control files or snapshot control files are excluded. For example, if the database is opened in read-only mode, then the snapshot control file location is excluded from the snapshot.	No
Server Parameter File	Contains the location of the server parameter file, if the instance is using one.	No
All TableSpaces	Includes all tablespaces in a snapshot.	No
<i>tablespace_names</i>	Selects individual TableSpaces.	No
Data file number= <i>n</i>	Selects individual data files. The metadata contains RESETLOGS information, tablespace number, tablespace name, and DBID.	No
ArchiveLogDest_ <i>n</i>	Selects individual local archiving destinations other than the fast recovery area.	Yes
Fast Recovery Area	Includes all backup files and archived logs in the Fast Recovery Area in the VSS snapshot.  Files backed up by VSS from the Fast Recovery Area can be subjected to deletion under space constraints.	Yes
Archived Logs	Logs in Fast Recovery Area	No
Backup Files	Backs up from Fast Recovery Area	No

You can select only Oracle Database, ArchiveLogDest\_*n*, and Fast Recovery Area, exposed by the Oracle VSS writer during a backup. The availability of the components in Components Backed Up by the Oracle VSS Writer may depend on the database state. For example, if the database is in NOARCHIVELOG mode, then the Archived Logs component is not returned. Likewise, if the instance is not started with a server parameter file, then the Server Parameter File component is not returned.

 **Note:**

The components that are not marked as Selectable for Backup are implicitly included by components marked as Selectable for Backup.

When you select Oracle Database component for backup or restore, all other components are implicitly selected for backup or restore. This implies that all files that are part of the selected component are candidates for backup or restore.

ArchiveLogDest\_*n* and Fast Recovery Area components are defined to contain only log or incremental files. This means that the requester must backup files from these components only when creating a log for incremental or differential backup. Likewise,

the requester must restore files from these components only when restoring from log or incremental or differential backups.

The files in all other components other than `ArchiveLogDest_n` and `Fast Recovery Area` define database files. This means that the requester must backup files from these components only when creating a full or copy backup. Likewise, the requester must restore files from these components only when restoring from full or copy backup.

## About Backing Up a Database in ARCHIVELOG Mode

Learn about backing up a database in ARCHIVELOG mode.

The procedures assume that the database is open read/write. You can also make closed and consistent backups.

- [Making Component-Based Backups of an ARCHIVELOG Database](#)  
Explains how to back up the entire database. You can back up only `Oracle Database`, `ArchiveLogDest_n`, and `Fast Recovery Area`, listed in `Components Backed Up by the Oracle VSS Writer`.
- [Making Volume-Based Backups of an ARCHIVELOG Database](#)  
To make volume-based shadow copies of Oracle database when the database is open in read/write mode, the archived redo logs must be physically located on a separate volume from the volume containing the Oracle data files, control files, server parameter file, and online redo logs.

### Related Topics

- [About Backing Up a Database in NOARCHIVELOG Mode](#)  
Learn about backing up a database in NOARCHIVELOG mode.

## Making Component-Based Backups of an ARCHIVELOG Database

Explains how to back up the entire database. You can back up only `Oracle Database`, `ArchiveLogDest_n`, and `Fast Recovery Area`, listed in `Components Backed Up by the Oracle VSS Writer`.

To back up the entire database:

1. Start a SQL\*Plus session on the target database and make sure the database is open `READ WRITE`.
2. Use a third-party VSS requester to select the `Oracle Database` component.
3. Create a snapshot of the database.

Oracle VSS writer includes the server parameter file, control file, and data files in the snapshot. The online redo logs are not included in the snapshot.

## Making Volume-Based Backups of an ARCHIVELOG Database

To make volume-based shadow copies of Oracle database when the database is open in read/write mode, the archived redo logs must be physically located on a separate volume from the volume containing the Oracle data files, control files, server parameter file, and online redo logs.

To back up the database and archived redo logs by volume:

1. Start a SQL\*Plus session on the target database and make sure the database is open `READ WRITE`.
2. Use a third-party VSS requester to select the volumes where the data files, control files, and server parameter file are physically located.
3. Create a snapshot of the database files.  
Oracle VSS writer includes the server parameter file, control file, and data files in the snapshot. The online redo logs are not included in the snapshot. Note that you can restore the server parameter file individually, but the control files and data files must be always restored together.
4. Use a third-party VSS requester to select the volumes where all of the archived redo logs (or the fast recovery area) are physically located.
5. Create a snapshot of the archived redo logs.

## About Backing Up a Database in NOARCHIVELOG Mode

Learn about backing up a database in NOARCHIVELOG mode.

For an Oracle database running in NOARCHIVELOG mode, the database must be in a consistent state when you create a VSS snapshot. Backups made while the database is open read/write are not supported.

- [Making Component-Based Backups of a NOARCHIVELOG Database](#)  
For an Oracle database in NOARCHIVELOG mode, the only supported component-based VSS snapshot is of Oracle Database when the type is full, default, or copy.
- [Making Volume-Based Backups of a NOARCHIVELOG Database](#)  
Learn how to make volume-based backups of a NOARCHIVELOG database.

## Making Component-Based Backups of a NOARCHIVELOG Database

For an Oracle database in NOARCHIVELOG mode, the only supported component-based VSS snapshot is of Oracle Database when the type is full, default, or copy.

To back up the database by component:

1. Start a SQL\*Plus session on the target database and place the database in a consistent state. For example, enter the following commands:

```
SHUTDOWN  
STARTUP MOUNT
```

2. Use a third-party VSS requester to select the Oracle Database component.
3. Create a volume-based VSS snapshot.

Oracle VSS writer includes the server parameter file, control file, and data files in the snapshot. The online redo logs are not included in the snapshot.

## Making Volume-Based Backups of a NOARCHIVELOG Database

Learn how to make volume-based backups of a NOARCHIVELOG database.

To back up the database by volume:

1. Start a SQL\*Plus session on the target database and place the database in a consistent state. For example, enter the following commands:

SHUTDOWN  
STARTUP MOUNT

2. Use a third-party VSS requester to select the volumes where the data files, control files, and server parameter file are physically located.
3. Create a volume-based VSS snapshot.

Oracle VSS writer includes the server parameter file, control file, and data files in the snapshot. The online redo logs are not included in the snapshot. Note that you can restore the server parameter file individually, but the control files and data files must be always restored together.

## About Restoring and Recovering a Database

Learn how to restore and recover VSS snapshots. As in the case of backups, the procedure depends on the archiving mode of the database and the type of snapshot that you are restoring.

- [About Restoring and Recovering a Database in ARCHIVELOG Mode](#)  
You can select the components listed in Components Usable in a Restore Operation in a restore and recovery operation.
- [Restoring a Database in NOARCHIVELOG Mode](#)  
For an Oracle Database running in NOARCHIVELOG mode, archived redo logs are not generated. So, media recovery is not possible.

## About Restoring and Recovering a Database in ARCHIVELOG Mode

You can select the components listed in Components Usable in a Restore Operation in a restore and recovery operation.

The table describes the validations that Oracle VSS writer performs for the components during the pre-restore phase, and the actions that it performs after the restore completes.

**Table 9-3 Components Usable in a Restore Operation**

Component	Pre-Restore Phase	PostRestore Phase	Section
Server Parameter File	Verifies that the database instance is not started. Otherwise, the writer returns a pre-restore failure.	Ensures that the database is started NOMOUNT. If the server parameter file is restored to the default location for the Oracle home, then the instance starts NOMOUNT automatically. Otherwise, you must set <code>ORA_SID_PFILE</code> to the location of the text-based initialization parameter file that points to the location of the server parameter file.	<a href="#">"Restoring the Server Parameter File "</a>

**Table 9-3 (Cont.) Components Usable in a Restore Operation**

Component	Pre-Restore Phase	PostRestore Phase	Section
Control File	Verifies that the instance is either started NOMOUNT or not started. If the instance is not started, the writer either starts the instance with the <code>ORA_SID_PFILE</code> instance parameter file, or uses the initialization parameter file or server parameter file in the default location.	Mounts control file after replicating control file to all the current control file locations pointed to by the instance.	<a href="#">"Recovering from the Loss of All Control Files"</a>
Tablespace or data file component	Verifies that the database must be mounted or the specified data files or tablespaces must be offline.	Performs complete recovery of these tablespaces or data files. The requester application can override the default recovery behavior.	<a href="#">"Recovering Tablespaces or Data Files"</a>
All Tablespaces	Verifies that the database is mounted.	Extracts redo from the backup writer metadata document and performs incomplete recovery on all the restored data files up to the time of snapshot creation. The requester application can override the default recovery behavior.	<a href="#">"Recovering All Tablespaces"</a>
Oracle Database	Verifies that the instance is not started.	Starts the database instance, mounts the control file, and performs recovery. See the descriptions of postrestore behavior for Server Parameter File, Control File, and All Tablespaces.	<a href="#">"Performing Disaster Recovery"</a> and <a href="#">"Restoring Component-Based Backups of a NOARCHIVELOG Database"</a>
Archived redo log or fast recovery area	None.	Does not perform default recovery of this component. Nevertheless, the requester application can run required RMAN commands.	

- [Restoring the Server Parameter File](#)  
Use this procedure to restore the server parameter file.
- [Recovering from the Loss of All Control Files](#)  
This procedure explains how to recover from the loss of all multiplexed control files.
- [Recovering Tablespaces or Data Files](#)  
This procedure explains how to recover from the loss of one or more tablespaces or data files. This procedure assumes that not all data files are lost.
- [Recovering All Tablespaces](#)  
This procedure explains how to recover from the loss of all tablespaces.
- [Performing Disaster Recovery](#)  
This procedure explains how to recover from the loss of the server parameter file, control file, and all data files.

## Restoring the Server Parameter File

Use this procedure to restore the server parameter file.

To restore the server parameter file:

1. Select the component named `Server Parameter File` from a VSS snapshot.
2. Restore the server parameter file.

Oracle VSS writer restores the server parameter file to the original location from where it was copied. You can also restore it to a new location.

## Recovering from the Loss of All Control Files

This procedure explains how to recover from the loss of all multiplexed control files.

To recover from the loss of all control files:

1. Ensure that the database is in `NOMOUNT` state or can be started in `NOMOUNT` state by the Oracle VSS writer.
2. Select the component named `Control File` from a VSS snapshot.
3. Restore the component containing the lost control file.

The Oracle VSS writer automatically mounts the database with the restored control files. If only the control file must be recovered, then the VSS requester application can ask the Oracle writer to perform complete recovery.

4. Restore and recover other database components if necessary.
5. Open the database with the `RESETLOGS` option.

## Recovering Tablespaces or Data Files

This procedure explains how to recover from the loss of one or more tablespaces or data files. This procedure assumes that not all data files are lost.

To recover from the loss of all tablespaces or data files:

1. Ensure that the database is either mounted or open. If the database is open, then take the data files or tablespaces needing recovery offline with the `ALTER DATABASE . . . OFFLINE` statement.
2. If the archived redo logs are required for recovery of the data files or tablespaces, then restore the archived redo logs.
3. Select the components from the VSS snapshot that contains the lost data files, or all data files in the lost tablespaces.
4. Restore the component containing the lost data files.

The Oracle VSS writer automatically recovers the restored data files. If some archived logs are missing, then you can restore the logs and recover the data files with `SQL*Plus` or `RMAN`.

5. Bring the offline data files or tablespaces back online.

## Recovering All Tablespaces

This procedure explains how to recover from the loss of all tablespaces.

To recover all data files:

1. Ensure that the database is mounted.
2. If the archived redo logs are required for recovery of the data files or tablespaces, then restore the archived redo logs.
3. Select the component named `All Tablespaces` from a VSS snapshot.
4. Restore the tablespaces.

The Oracle VSS writer automatically recovers the restored data files. If some archived logs are missing, then you can restore the logs and recover the data files with SQL\*Plus or RMAN.

5. Open the database.

## Performing Disaster Recovery

This procedure explains how to recover from the loss of the server parameter file, control file, and all data files.

To perform disaster recovery:

1. Ensure that the instance is not started.
2. If the archived redo logs are required for recovery of the data files or tablespaces, then restore the archived redo logs.
3. Select the component named `Oracle Database` from a VSS snapshot.
4. Restore the database.

The Oracle VSS writer automatically starts the instance, mount the database, and recovers the restored data files. If some archived logs are missing, then you can restore the logs and recover the data files with SQL\*Plus or RMAN.

5. Open the database with the `RESETLOGS` option.

## Restoring a Database in NOARCHIVELOG Mode

For an Oracle Database running in `NOARCHIVELOG` mode, archived redo logs are not generated. So, media recovery is not possible.

- [Restoring Component-Based Backups of a NOARCHIVELOG Database](#)  
Use this procedure to restore a component-based backup.
- [Restoring Volume-Based Backups of a NOARCHIVELOG Database](#)  
Use this procedure to restore a volume-based backup.

## Restoring Component-Based Backups of a NOARCHIVELOG Database

Use this procedure to restore a component-based backup.

To restore a component-based backup:



1. Use a third-party VSS requester to select the Oracle Database component.  
The Oracle VSS writer automatically restores the data files and mounts the database.
2. Open the database with the `RESETLOGS` option.

## Restoring Volume-Based Backups of a NOARCHIVELOG Database

Use this procedure to restore a volume-based backup.

To restore a volume-based backup:

1. Use a third-party VSS requester to select the volumes where the data files, control files, and server parameter file are physically located.
2. Restore all volumes where data files and logs are located.
3. Open the database with the `RESETLOGS` option.

## About Integrating VSS with Third-Party Requester Applications

Oracle VSS writer allows third-party requester applications to control the behavior of recovery and backup sessions.

Third-party requester applications use VSS API `setBackupOptions` or `setRestoreOptions` to pass an appropriate string to the writer. The writer uses `getBackupOptions` or `getRestoreOptions` to get the string set from the requester to perform the pre or post backup and restore actions.

- [Running Writer Control Commands](#)  
The writer control commands are applicable to all the restored components during the postrestore phase.
- [Controlling Commands for Database or All Tablespaces Component](#)  
The `POST_WTRCMD=UNTIL_SNAPSHOT` command instructs the writer to perform recovery to the snapshot creation time.

## Running Writer Control Commands

The writer control commands are applicable to all the restored components during the postrestore phase.

The format is as follows:

```
OP1=CMD1, OP2=CMD2, . . .
```

Run the commands in the following sequence:

1. `POST_WTRCMD=NORECOVER`  
This command instructs the writer to not perform any postrestore recovery activities defined in the default postrestore recovery operations for the restored component. Otherwise, the postrestore phase default actions are performed.
2. `POST_RMANCMD=cmdstr`

This command instructs the writer to run specific RMAN commands, instead of the default operations, after the current operation.

3. `PRE_SQLCMD=cmdstr`

This command instructs the writer to run specific SQL commands in `OnPrepareBackup` or `OnPreRestore` callback, before performing any other validations. The command is used to stop MRP on a standby database before VSS snapshot is created or to shut down database instance creating a cold backup of the database.

4. `POST_SQLCMD=cmdstr`

This command instructs the writer to run specific SQL commands in `PostSnapshot` or `PostRestore` callback. This command is used to restart MRP on standby database after VSS snapshot is created or to restart the database instance after the cold backup of the database is performed.

## Controlling Commands for Database or All Tablespaces Component

The `POST_WTRCMD=UNTIL_SNAPSHOT` command instructs the writer to perform recovery to the snapshot creation time.

Run the following command:

```
POST_WTRCMD=UNTIL_SNAPSHOT
```

This command instructs the writer to perform recovery to the snapshot creation time. The writer extracts the system change number of the redo logs stored in the database component and performs recovery until the system change number.

## About Duplicating a Database

If your VSS shadow copies are transportable, then you can use these shadow copies to duplicate the primary database.

In the context of this chapter, duplication refers to the creation of a new database out of the shadow copies for a different database. A duplicate database created from shadow copies can either be a nonstandby database or a standby database for use in a Data Guard environment. Note that RMAN duplication, which makes use of the `Duplicate` command, is a different procedure.

- [Creating a Nonstandby Database from Shadow Copies](#)  
Use this procedure to create a nonstandby database from shadow copies.
- [Creating a Standby Database From Shadow Copies](#)  
Use this procedure to create a standby database from shadow copies.

## Creating a Nonstandby Database from Shadow Copies

Use this procedure to create a nonstandby database from shadow copies.

This section assumes that you are duplicating the database on a host with the same file system structure as the primary database.

To create a nonstandby database from shadow copies:

1. Restore the database on the new host.

2. Start a SQL\*Plus session on the duplicate database and obtain the DBID. You can query the DBID as follows:

```
SELECT DBID FROM V$DATABASE;
```

3. Shut down the database consistently. You can shut down the database as follows:

```
SHUTDOWN;
```

4. Use the DBNEWID utility to change the DBID.
5. Open the database.
6. Start a SQL\*Plus session on the duplicate database and query the DBID. You can query the DBID as follows:

```
SELECT DBID FROM V$DATABASE;
```

**See Also:**

*Oracle Database Utilities* for information about how to use DBNEWID

**Related Topics**

- [Performing Disaster Recovery](#)

This procedure explains how to recover from the loss of the server parameter file, control file, and all data files.

## Creating a Standby Database From Shadow Copies

Use this procedure to create a standby database from shadow copies.

This section assumes that you have created a standby database on a host with the same file system structure as the primary database. This section also assumes that you have read *Oracle Data Guard Concepts and Administration* and are familiar with standby database creation and maintenance.

To create a standby database from shadow copies:

1. Restore the database on the standby host.
2. Start a SQL\*Plus session on the new database and a new standby control file must be obtained from primary database. You can create the control file with the SQL statement `ALTER DATABASE CREATE STANDBY CONTROLFILE.`
3. Start the instance and mount the standby control file.

**Related Topics**

- [Performing Disaster Recovery](#)

This procedure explains how to recover from the loss of the server parameter file, control file, and all data files.

**See Also:**

*Oracle Data Guard Concepts and Administration*

# 10

## Authenticating Database Users with Windows

Learn about the authentication of Oracle Database users with Windows operating systems.

- [Overview of Windows Native Authentication](#)  
Oracle Database can use Windows user login credentials to authenticate database users.
- [About Windows Authentication Protocols](#)  
The Windows native authentication adapter works with Windows authentication protocols to enable access to Oracle Database.
- [About User Authentication and Role Authorization Methods](#)  
Describes how user login credentials are authenticated and database roles are authorized in Windows domains.
- [Overview of Operating System Authentication Enabled at Installation](#)  
When you install Oracle Database, a special Windows local group called `ORA_DBA` is created (if it does not already exist from an earlier Oracle Database installation) and the Oracle Installation User is automatically added to it.

### Overview of Windows Native Authentication

Oracle Database can use Windows user login credentials to authenticate database users.

Benefits include:

- Enabling users to connect to Oracle Database without supplying a username or password
- Centralizing Oracle Database user authentication and role authorization information in Windows, which frees Oracle Database from storing or managing user passwords or role information

The Windows native authentication adapter (automatically installed with Oracle Net Services) enables database user authentication through Windows. This enables client computers to make secure connections to Oracle Database on a Windows server. The server then permits the user to perform database actions on the server.

#### Note:

Current user database links are not supported with Windows native authentication.

 **See Also:**

- *Oracle Database Security Guide*
- *Oracle Internet Directory Administrator's Guide*

## About Windows Authentication Protocols

The Windows native authentication adapter works with Windows authentication protocols to enable access to Oracle Database.

The NTS authentication adapter no longer supports the use of NTLM to authenticate Windows domain users. Thus the NTS cannot be used to authenticate users in old Windows NT domains or domains with old Windows NT domain controllers. However, local connections and Oracle Database services running as a Windows Local User continues to be authenticated using NTLM.

If you use the Windows Local User Account as the Oracle Home User for an Oracle Database home, then Windows Native Authentication (NTS) cannot be used for authenticating Windows domain users or users from remote computers.

Client server must not specify an authentication protocol while trying to connect to Oracle Database. Instead, Oracle Database determines the protocol to use which is completely transparent to the user. The only Oracle Database requirement is to ensure that the parameter `SQLNET.AUTHENTICATION_SERVICES` in client and database server contains `nts` in the following file:

```
ORACLE_HOME\network\admin\sqlnet.ora
```

This is the default setting for both client computer and database server after installation.

In a typical installation, Oracle Database network includes client computers and database servers, and computers on this network may use different Oracle Database software releases on different domains of Windows operating systems. This combination of different releases means that the authentication protocol being used can vary.

### Related Topics

- [About Configuring Oracle Database to Communicate with Oracle ASM](#)  
Oracle Databases that use Oracle Automatic Storage Management (Oracle ASM) and the databases that are managed by Oracle Grid infrastructure must use Windows native authentication, which is enabled by default.

 **See Also:**

Your operating system documentation for more information on authentication protocol

## About User Authentication and Role Authorization Methods

Describes how user login credentials are authenticated and database roles are authorized in Windows domains.

User authentication and role authorization are defined in User Authentication and Role Authorization Defined.

**Table 10-1 User Authentication and Role Authorization Defined**

Feature	Description	More Information
User authentication	Process by which the database uses the user's Windows login credentials to authenticate the user.	<i>Oracle Database 2 Day DBA</i>
Role authorization	Process of granting an assigned set of roles to authenticated users.	<i>Oracle Database 2 Day DBA</i>

Oracle Database supports user authentication and role authorization in Windows domains. Basic Features of User Authentication and Role Authorization describes these basic features.

**Table 10-2 Basic Features of User Authentication and Role Authorization**

Feature	Description
Authentication of external users	Users are authenticated by the database using the user's Windows login credentials enabling them to access Oracle Database without being prompted for additional login credentials.
Authorization of external roles	Roles are authorized using Windows local groups. Once an external role is created, you can grant or revoke that role to a database user. Initialization parameter <code>OS_ROLES</code> is set to <code>false</code> by default. You must set <code>OS_ROLES</code> to <code>true</code> to authorize external roles.

- [About Using Authentication and Authorization Methods](#)  
User Authentication and Role Authorization Methods describes user authentication and role authorization methods to use based on your Oracle Database environment:

## About Using Authentication and Authorization Methods

User Authentication and Role Authorization Methods describes user authentication and role authorization methods to use based on your Oracle Database environment:

**Table 10-3 User Authentication and Role Authorization Methods**

Method	Database Environment
Enterprise users and roles	<p>You have many users connecting to multiple databases. Enterprise users have the same identity across multiple databases. Enterprise users require use of a directory server.</p> <p>Use enterprise roles in environments where enterprise users assigned to these roles are located in many geographic regions and must access multiple databases. Each enterprise role can be assigned to multiple enterprise user in the directory. If you do not use enterprise roles, then you must assign database roles manually to each database user. Enterprise roles require use of a directory server.</p>
External users and roles	<p>You have a smaller number of users accessing a limited number of databases. External users must be created individually in each database and do not require use of a directory server.</p> <p>External roles must also be created individually in each database, and do not require use of a directory server. External roles are authorized using group membership of the users in local groups on the system.</p>

 **See Also:**

*Oracle Database Enterprise User Security Administrator's Guide* for more information on Enterprise users and roles

## Overview of Operating System Authentication Enabled at Installation

When you install Oracle Database, a special Windows local group called `ORA_DBA` is created (if it does not already exist from an earlier Oracle Database installation) and the Oracle Installation User is automatically added to it.

Members of local group `ORA_DBA` automatically receive the `SYSDBA` privilege. The `ORA_DBA` group is also created for each Oracle home called `ORA_HOMENAME_DBA` group. This group is automatically populated with the Oracle Home User for the Oracle home.

 **Note:**

If you use a domain account for database administration, then that domain account must be granted local administrative privileges and `ORA_DBA` membership explicitly. It is not sufficient for the domain account to inherit these memberships from another group. You must ensure that the user performing the installation is in the same domain as this domain account. If not, it results in an NTS authentication failure.

Membership in `ORA_DBA` enables you to:

- Connect to local Oracle Database servers without a password with the command

```
SQL> CONNECT / AS SYSDBA
```

- Connect to remote Oracle Database servers without a password with the command

```
SQL> CONNECT /@net_service_name AS SYSDBA
```

where *net\_service\_name* is the net service name of the remote Oracle Database server

- Perform database administration procedures such as starting and shutting down local databases
- Add additional Windows users to `ORA_DBA`, enabling them to have the `SYSDBA` privilege



# 11

## Administering External Users and Roles on Windows

External users and roles are in general defined by something external to Oracle Database.

In a Windows environment, they are defined by the operating system.

Learn about the external user and external role creation and management using a combination of Oracle Database command-line tools, Registry Editor, and other Windows tools.

- [Overview of Manually Administering External Users and Roles](#)  
Manually configure administrators, operators, users, and roles to be authenticated by the operating system.

### See Also:

*Oracle Database Enterprise User Security Administrator's Guide* for more information about tools available for administering enterprise users and roles

## Overview of Manually Administering External Users and Roles

Manually configure administrators, operators, users, and roles to be authenticated by the operating system.

Manual configuration involves using Oracle Database command-line tools, editing the registry, and creating local groups in Active Directory Users and Computers.

All of the following can be manually configured to access Oracle Database without a password:

- External operating system users
- Windows database administrators (with `SYSDBA` privilege)
- Windows database operators (with `SYSOPER` privilege)

In addition, you can manually create and grant local and external database roles to Windows domain users and global groups.

This section describes:

- [About Manually Creating an External Operating System User](#)

Describes how to authenticate external operating system users (not database administrators) using Windows, so that a password is not required when accessing

the database. When you use Windows to authenticate external operating system users, your database relies solely on the operating system to restrict access to database user names.

- [Overview of Manually Granting Administrator, Operator, and Task-Specific Privileges for Databases](#)

Describes how to enable Windows to grant the database administrator (SYSDBA), database operator (SYSOPER), database administrator for ASM (SYSASM), and new task-specific and less privileged than the ORA\_DBA/SYSDBA system privileges to administrators.

- [Managing New Users and User Groups](#)

During Oracle Database installation, ORA\_INSTALL, ORA\_DBA, ORA\_OPER, ORA\_HOMENAME\_DBA, ORA\_HOMENAME\_OPER, ORA\_HOMENAME\_SYSDG, ORA\_HOMENAME\_SYSBACKUP, ORA\_HOMENAME\_SYSKM, ORA\_ASMADMIN, ORA\_ASMDBA, and ORA\_ASMOPER user groups are automatically created with the required privileges.

- [Overview of Manually Creating an External Role](#)

Describes how to grant Oracle Database roles to users directly through Windows (known as external roles). When you use Windows to authenticate users, Windows local groups can grant these users external roles.

- [About Manually Migrating Users](#)

You can migrate local or external users to enterprise users with User Migration Utility. Migrating from a database user model to an enterprise user model provides solutions to administrative, security, and usability challenges in an enterprise environment. In an enterprise user model, all user information is moved to an LDAP directory service, which provides the following benefits:

 **Note:**

Use extreme care when manually configuring administrators, operators, users, and roles to be authenticated by the operating system.

- [About Manually Creating an External Operating System User](#)

Describes how to authenticate external operating system users (not database administrators) using Windows, so that a password is not required when accessing the database.

- [Overview of Manually Granting Administrator, Operator, and Task-Specific Privileges for Databases](#)

Describes how to enable Windows to grant the database administrator (SYSDBA), database operator (SYSOPER), database administrator for ASM (SYSASM), and new task-specific and less privileged than the ORA\_DBA/SYSDBA system privileges to administrators.

- [Managing New Users and User Groups](#)

Learn how to manage new users and user groups.

- [Overview of Manually Creating an External Role](#)

Describes how to grant Oracle Database roles to users directly through Windows (known as external roles).

- [About Manually Migrating Users](#)  
You can migrate local or external users to enterprise users with User Migration Utility.

## About Manually Creating an External Operating System User

Describes how to authenticate external operating system users (not database administrators) using Windows, so that a password is not required when accessing the database.

When you use Windows to authenticate external operating system users, your database relies solely on the operating system to restrict access to database user names.

Note that if a Windows Local User is used as the Oracle Home User for an Oracle home, then external user authentication of the Windows Local users is only supported from the same computer. Oracle recommends using Windows Domain User or Windows built-in user as the Oracle Home User to support external authentication of the Windows Domain User from the same computer or a different computer.

In the following procedure, two Windows user names are authenticated:

- Local user `jones`
- Domain user `jones` on domain `sales`

Local user `jones` logs into its local Windows client computer to access an Oracle Database server, which can be on a different computer. To access other databases and resources on other computers, the local user must provide a user name and password each time.

Domain user `jones` on domain `sales` logs into a `sales` domain that includes many other Windows computers and resources, one of which contains an Oracle Database server. The domain user can access all the resources the domain provides with a single user name and password.

- [Performing External User Authentication Tasks on the Oracle Database Server](#)  
Use this procedure to perform external user authentication tasks.
- [Performing External User Authentication Tasks on the Client Computer](#)  
Use this procedure to perform external user authentication tasks on the client computer.

## Performing External User Authentication Tasks on the Oracle Database Server

Use this procedure to perform external user authentication tasks.

Perform the following external user authentication tasks on the Oracle Database server:

1. Add parameter `OS_AUTHENT_PREFIX` to your `init.ora` file.

The `OS_AUTHENT_PREFIX` value is prefixed to local or domain user names attempting to connect to the server with the user's operating system name and password. The prefixed user name is compared with Oracle Database user names in the database when a connection request is attempted. Using parameter `OS_AUTHENT_PREFIX` with Windows native authentication methods is the recommended method for performing secure, trusted client connections to your server.

2. Set a value for `OS_AUTHENT_PREFIX`. Your choices are:

- Any character string

If you specify `xyz`, as in this procedure's example, then `xyz` is prefixed to the beginning of the Windows user name (for example, `xyzjones` for local user `jones` or `xyzsales\jones` for domain user `jones` on domain `sales`). String values are case insensitive.

- "" (two double quotes with no space between)

This option is recommended, because it eliminates the need for any prefix to Windows user names (for example, `jones` for local user `jones` or `sales\jones` for domain user `jones` on domain `sales`).

- No value specified

If you do not specify a value for `OS_AUTHENT_PREFIX`, it defaults to `OPS$` (for example, `OPS$jones` for local user `jones` or `OPS$sales\jones` for domain user `jones` on domain `sales`).

 **Note:**

On Windows and Linux, create a database user name with characters in upper case for successful authentication. Authentication fails when you use characters with lower case. For example, authentication fails when creating a database user name with characters in lower case `ops$sales\jones`, but is successful when creating a database user name with upper case characters such as `OPS$SALES\JONES`.

3. Create a Windows local user name for `jones` with the Computer Management tool, or create a Windows domain user name for `jones` with Active Directory Users and Computers (if the appropriate name does not currently exist). See your operating system documentation for detailed instructions.

4. Ensure that parameter `SQLNET.AUTHENTICATN_SERVICES` in file `sqlnet.ora` contains `nts`.

5. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

6. Connect to the database with the SYSTEM database administrator (DBA) name:

```
SQL> CONNECT SYSTEM
Enter password: system_password
```

Unless you have changed it, the SYSTEM password is `MANAGER` by default.

7. Create a local external user by entering:

```
SQL> CREATE USER xyzjones IDENTIFIED EXTERNALLY;
```

where `xyz` is the value you chose for initialization parameter `OS_AUTHENT_PREFIX`, and `jones` is the Windows local user name.

8. Grant a local external user database roles by entering:

```
SQL> GRANT DBA TO xyzjones;
```

 **Note:**

External authentication of Windows Local users is supported from the same computer only. While external authentication of Windows Domain user is supported from the same computer or a different computer.

9. Create a domain external user by entering:

```
SQL> CREATE USER "XYZSALES\JONES" IDENTIFIED EXTERNALLY;
```

where XYZ is the value you chose for initialization parameter `OS_AUTHENT_PREFIX`, and SALES\JONES is the domain name and Windows domain user name. Double quotes are required and the entire syntax must be in uppercase.

10. Grant a domain external user database roles by entering:

```
SQL> GRANT DBA TO "XYZSALES\JONES";
```

Double quotes are required and the entire syntax must be in uppercase.

11. Log on to the Windows system using the Windows local user `jones` or domain user SALES\JONES.

12. Connect to the database with the SYSDBA name:

```
SQL> CONNECT / AS SYSDBA
```

13. Shut down the database:

```
SQL> SHUTDOWN
```

14. Restart the database:

```
SQL> STARTUP
```

This causes the change to parameter `OS_AUTHENT_PREFIX` to take effect.

## Performing External User Authentication Tasks on the Client Computer

Use this procedure to perform external user authentication tasks on the client computer.

Perform the following external user authentication tasks on the client computer:

1. Ensure that parameter `SQLNET.AUTHENTICATN_SERVICES` in file `sqlnet.ora` contains `nts`.
2. Use Oracle Net Configuration Assistant to configure a network connection from your client computer to the Windows server on which Oracle Database is installed.
3. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

4. Connect to your Windows server:

```
SQL> CONNECT /@connect_identifier
```

where `connect_identifier` is the net service name for Oracle Database.

Oracle Database searches the data dictionary for an automatic login user name corresponding to the Windows local or domain user name, verifies it, and enables connection as `xyzjones` or `xyzsales\jones`.

5. Verify that you have connected to Oracle Database as domain user `jones` by viewing the roles assigned.

```
SQL> SELECT * FROM USER_ROLE_PRIVS;
```

which outputs for local user `jones`:

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
XYZJONES	DBA	NO	YES	NO

1 row selected.

or, for domain user `jones`:

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
XYZSALES\JONES	DBA	NO	YES	NO

1 row selected.

Because the Oracle Database user name is the whole name `xyzjones` or `xyzsales\jones`, each object created by `xyzjones` or `xyzsales\jones` (that is, table, view, index, and so on) is prefixed by this name. For another user to reference the table `shark` owned by `xyzjones`, for example, the user must enter:

```
SQL> SELECT * FROM xyzjones.shark
```

#### Note:

Automatic authorization is supported for all Oracle Net protocols.

#### See Also:

*Oracle Database Net Services Administrator's Guide*

#### Related Topics

- [Performing External User Authentication Tasks on the Oracle Database Server](#)  
Use this procedure to perform external user authentication tasks.

## Overview of Manually Granting Administrator, Operator, and Task-Specific Privileges for Databases

Describes how to enable Windows to grant the database administrator (`SYSDBA`), database operator (`SYSOPER`), database administrator for ASM (`SYSASM`), and new task-specific and less privileged than the `ORA_DBA/SYSDBA` system privileges to administrators.

With these privileges, the administrator can issue the following commands from a client computer and connect to Oracle Database without entering a password:

```

SQL> CONNECT / AS SYSOPER
SQL> CONNECT / AS SYSDBA
SQL> CONNECT / AS SYSASM
SQL> CONNECT / AS SYSBACKUP
SQL> CONNECT / AS SYSDG
SQL> CONNECT / AS SYSKM

```

To enable this feature, a Windows local or a domain user name of the administrator must belong to one of the Windows local groups listed in Windows Local Groups with SYSDBA, SYSOPER, SYSASM, SYSDG, SYSBACKUP, and SYSKM Privileges.

**Table 11-1 Windows Local Groups with SYSDBA, SYSOPER, SYSASM, SYSDG, SYSBACKUP, and SYSKM Privileges**

Local Group	System Privileges
ORA_OPER	SYSOPER privileges for all databases on a computer
ORA_DBA Note	SYSDBA privileges for all databases on a computer
ORA_SID_OPER	SYSOPER privileges for a single database (identified by <i>SID</i> )
ORA_SID_DBA	SYSDBA privileges for a single database (identified by <i>SID</i> )
ORA_HOMENAME_DBA	SYSDBA privileges for all database instances of the specified Oracle home.
ORA_HOMENAME_OPER	SYSOPER privileges for starting up and shutting down all databases instances that run from a specified Oracle home.
ORA_HOMENAME_SYSDG	SYSDG privilege for all database instances that run from the particular Oracle home
ORA_HOMENAME_SYSBACKUP	SYSBACKUP privilege for all database instances that run from the particular Oracle home
ORA_HOMENAME_SYSKM	SYSKM privilege for all database instances that run from the particular Oracle home
ORA_ASMADMIN	SYSASM privileges for all ASM instances on a computer
ORA_ASMDBA	SYSDBA privileges for all ASM instances on a computer
ORA_ASMOPER	SYSOPER privileges for all ASM instances on a computer

#### Note

All the groups mentioned in the table above are automatically created during installation and the Oracle Home User is automatically added to `ORA_HOMENAME_DBA` group. See section "[Overview of Operating System Authentication Enabled at Installation](#)" for information.

#### See Also:

- *Oracle Database Administrator's Guide*
- *Oracle Automatic Storage Management Administrator's Guide*

The manual procedure for enabling administrators to connect as `SYSOPER`, `SYSDBA`, `SYSASM`, `SYSDBG`, `SYSKM`, or `SYSBACKUP` without a password is divided into two sets of tasks performed on different computers:

- [Running System Privilege Authentication Tasks on the Oracle Database Server](#)  
Learn about running system privilege authentication tasks on the Oracle Database server.
- [Running System Privilege Authentication Tasks on the Client Computer](#)  
Learn about running system privilege authentication tasks on the client computer.

## Running System Privilege Authentication Tasks on the Oracle Database Server

Learn about running system privilege authentication tasks on the Oracle Database server.

Perform the following steps:

1. Add your administrator user names to this group. The client logs in using one of these user names so that it is granted the required system privilege.

### See Also:

Your operating system documentation for instructions on managing users and groups

2. Ensure that parameter `SQLNET.AUTHENTICATN_SERVICES` in file `sqlnet.ora` contains `nts`.

## Running System Privilege Authentication Tasks on the Client Computer

Learn about running system privilege authentication tasks on the client computer.

Perform the following steps:

1. Log in as a Windows domain user who is a member of one of the Windows local group on the server, according to the system privilege that you want Windows to grant. The administrator must add this domain user to the required Windows local group. Windows local group membership is created on the server system where Oracle Database runs.
2. Ensure that the parameter `SQLNET.AUTHENTICATN_SERVICES` in file `sqlnet.ora` contains `nts`.
3. Use Oracle Net Configuration Assistant to configure a network connection from your client computer to the Windows server on which Oracle Database is installed.

4. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

5. Connect to Oracle Database:

```
SQL> SET INSTANCE net_service_name
```

where `net_service_name` is the Oracle Net net service name for Oracle Database.

6. Enter either of the following SQL\*Plus commands so that you connect to the database with the required system privilege:



```
SQL> CONNECT / AS SYSOPER
SQL> CONNECT / AS SYSDBA
SQL> CONNECT / AS SYSASM
SQL> CONNECT / AS SYSDBG
SQL> CONNECT / AS SYSKM
SQL> CONNECT / AS SYSBACKUP
```

You are now connected to the Windows server. If you connect with SYSDBA, you are given DBA privileges.

#### See Also:

*Oracle Database Net Services Administrator's Guide*

#### Related Topics

- [Running System Privilege Authentication Tasks on the Oracle Database Server](#)  
Learn about running system privilege authentication tasks on the Oracle Database server.

## Managing New Users and User Groups

Learn how to manage new users and user groups.

During Oracle Database installation, ORA\_INSTALL, ORA\_DBA, ORA\_OPER, ORA\_HOMENAME\_DBA, ORA\_HOMENAME\_OPER, ORA\_HOMENAME\_SYSDG, ORA\_HOMENAME\_SYSBACKUP, ORA\_HOMENAME\_SYSKM, ORA\_ASMADMIN, ORA\_ASMDBA, and ORA\_ASMOPER user groups are automatically created with the required privileges.

#### See Also:

- *Oracle Database Installation Guide for Microsoft Windows*
- *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*

## Overview of Manually Creating an External Role

Describes how to grant Oracle Database roles to users directly through Windows (known as external roles).

When you use Windows to authenticate users, Windows local groups can grant these users external roles.

All privileges for these roles are active when the user connects. When using external roles, all roles are granted and managed through the operating system. You cannot use both external roles and Oracle Database roles at the same time.

Consider the following example. With external roles enabled, you log on to a Windows domain with domain user name sales\jones (sales is the domain name and jones is the domain user name). You then connect to Oracle Database as Oracle Database

user `smith`. In this case, you receive the roles granted to `sales\jones` but *not* the roles granted to `smith`.

The procedure for manually creating an external role is divided into two sets of authorization tasks performed on different computers:

- [Performing External Role Authorization Tasks on the Oracle Database Server](#)  
Learn how to perform external role authorization tasks on the Oracle Database server.
- [Performing External Role Authorization Tasks on the Client Computer](#)  
Learn how to perform external role authorization tasks on the client computer.

## Performing External Role Authorization Tasks on the Oracle Database Server

Learn how to perform external role authorization tasks on the Oracle Database server.

Perform the following steps:

1. Add initialization parameter `OS_ROLES` to the `init.ora` file.
2. Set `OS_ROLES` to `true`.

The default setting for this parameter is `false`.

3. Ensure that parameter `SQLNET.AUTHENTICATN_SERVICES` in file `sqlnet.ora` contains `nts`.
4. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

5. Connect to your Windows server:

```
SQL> CONNECT / AS SYSDBA
```

6. Create a new database role. You can give this new role whatever name you want. In this example the role is named `DBSALES3`:

```
SQL> CREATE ROLE DBSALES3 IDENTIFIED EXTERNALLY;
```

7. Grant to `DBSALES3` whatever Oracle Database roles are appropriate to your database environment:

```
SQL> GRANT DBA TO DBSALES3 WITH ADMIN OPTN;
```

8. Connect to the database as `SYSDBA`:

```
SQL> CONNECT / AS SYSDBA
```

9. Shut down the database:

```
SQL> SHUTDOWN
```

10. Restart the database:

```
SQL> STARTUP
```

11. Create a Windows local group with the following syntax:

```
ORA_sid_rolename[_D][_A]
```

For this command, note the following:

- *sid* identifies the database instance
- *rolename* identifies the database role granted

- `D` indicates that this database role is to be a default role of the database user
- `A` indicates that this database role includes `ADMIN OPTN`

Characters `D` and `A` are optional. If specified, they must be preceded by an underscore.

For this example, `ORA_orcl_dbsales3_D` is created.

## 12. Add one or more Windows local or domain user names to this group.

You can create multiple database roles and grant them to several possible Windows groups with differing options, as shown in the following table. Users connecting to the `ORCL` instance and authenticated by Windows as members of all four of these Windows local groups has the privileges associated with `dbsales3` and `dbsales4` by default (because of option `_D`). If these users first connect as members of `dbsales3` or `dbsales4` and use the `SET ROLE` command, then they can also gain access to database roles `dbsales1` and `dbsales2`. But if these users try to connect with `dbsales1` or `dbsales2` without first connecting with a default role, they are unable to connect. Finally, these users can grant `dbsales2` and `dbsales4` to other roles (because of option `_A`).

Database Roles	Windows Groups
<code>dbsales1</code>	<code>ORA_ORCL_dbsales1</code>
<code>dbsales2</code>	<code>ORA_ORCL_dbsales2_a</code>
<code>dbsales3</code>	<code>ORA_ORCL_dbsales3_d</code>
<code>dbsales4</code>	<code>ORA_ORCL_dbsales4_da</code>

### Note:

When Oracle Database converts the group name to a role name, it changes the name to uppercase.

### See Also:

Your operating system documentation for instructions on managing users and groups

## Performing External Role Authorization Tasks on the Client Computer

Learn how to perform external role authorization tasks on the client computer.

Perform the following steps:

1. Create a Windows local or a domain user name with the same user name and password that exist on the Windows server (if the appropriate user name does not currently exist).
2. Ensure that parameter `SQLNET.AUTHENTICATN_SERVICES` in file `sqlnet.ora` contains `nts`.

3. Use Oracle Net Configuration Assistant to configure a network connection from your client computer to Oracle Database.

4. Start SQL\*Plus:

```
C:\> sqlplus /NOLOG
```

5. Connect to the correct instance:

```
SQL> SET INSTANCE connect_identifier
```

where *connect\_identifier* is the net service name for the Oracle Database connection that you created in Step 3.

6. Connect to Oracle Database:

```
SQL> CONNECT SMITH  
Enter password: password
```

You are connected to the Windows server over net service with Oracle Database user name *smith*. Roles applied to Oracle Database user name *smith* consist of all roles defined for the Windows user name that were previously mapped to the database roles (in this case, *ORA\_DBSALES3\_D*). All roles available under an authenticated connection are determined by the Windows user name and the Oracle-specific Windows local groups to which the user belongs (for example, *ORA\_SID\_DBSALES1* or *ORA\_SID\_DBSALES4\_DA*).

#### Note:

OSDBA and OSOPER are the generic names for the two special operating system groups that control database administrator logins when using operating system authentication.

#### See Also:

- *Oracle Database Administrator's Guide*
- *Oracle Database Net Services Administrator's Guide*

#### Related Topics

- [Overview of Manually Granting Administrator, Operator, and Task-Specific Privileges for Databases](#)  
Describes how to enable Windows to grant the database administrator (*SYSDBA*), database operator (*SYSOPER*), database administrator for ASM (*SYSASM*), and new task-specific and less privileged than the *ORA\_DBA/SYSDBA* system privileges to administrators.

## About Manually Migrating Users

You can migrate local or external users to enterprise users with User Migration Utility.

Migrating from a database user model to an enterprise user model provides solutions to administrative, security, and usability challenges in an enterprise environment. In an

enterprise user model, all user information is moved to an LDAP directory service, which provides the following benefits:

- Centralized storage and management of user information
- Centralized user authentication
- Enhanced security

User Migration Utility is a command-line tool. Its syntax is of the form:

```
C:\ umu parameters
```

To get a list of User Migration Utility parameters, enter:

```
C:\ umu help=yes
```

 **See Also:**

*Oracle Database Enterprise User Security Administrator's Guide* in "Using the User Migration Utility."

# 12

## Storing Oracle Wallets in the Windows Registry

Learn about storing and retrieving of Oracle Wallets in the Windows registry.

- [About Storing Private Keys and Trust Points](#)  
Oracle Wallets store private keys, trust points, and digital certificates used in public key applications for authentication and encryption.
- [About Storing User's Profile](#)  
In Windows domain, a user's profile is stored on the local computer.
- [About Registry Parameters for Wallet Storage](#)  
Parameter `WALLET_LOCATN` in file `sqlnet.ora` specifies the location of the obfuscated Oracle Wallet for use by Oracle PKI applications.

### About Storing Private Keys and Trust Points

Oracle Wallets store private keys, trust points, and digital certificates used in public key applications for authentication and encryption.

Oracle Wallet Manager creates and manages Oracle Wallets. Oracle public key applications use obfuscated Oracle Wallets for authentication and encryption.

### About Storing User's Profile

In Windows domain, a user's profile is stored on the local computer.

When a local user logs on to that computer, that user's profile on the local computer is uploaded into the user profile in that computer's registry. When a user logs out, that user's profile stored on the local file system is updated, ensuring that the Windows Domain user or the Windows Local user always has the most recent user profile version.

### About Registry Parameters for Wallet Storage

Parameter `WALLET_LOCATN` in file `sqlnet.ora` specifies the location of the obfuscated Oracle Wallet for use by Oracle PKI applications.

For example, the `WALLET_LOCATN` parameter for storing an Oracle Wallet in the registry in:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP
```

is located in:

```
WALLET_LOCATN = (SOURCE= (METHOD=REG) (METHOD_DATA= (KEY=SALESAPP)))
```

Continuing the example, the encrypted Oracle Wallet is stored in the registry in:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\EWALLET.P12
```

and the changed Oracle Wallet stored in:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\SALESAPP\CWALLET.SSO
```

On Windows operating systems, if there is no value specified for parameter `WALLET_LOCATN`, then Oracle PKI applications first look for the changed wallet in registry key:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\DEFAULT
```

If no obfuscated wallet is found there, Oracle PKI applications look for it in the file system of the local computer at location:

```
%USERPROFILE%\ORACLE\WALLETS
```

Regardless of location, wallets are always stored in the same format. All functionality is the same except for the location of the wallets.

- [About Oracle Wallet Manager](#)  
Oracle Wallet Manager creates and manages Oracle Wallets.

## About Oracle Wallet Manager

Oracle Wallet Manager creates and manages Oracle Wallets.

If you want to use the Windows registry for Oracle Wallets, then you must select the Use Windows System Registry check box. If Windows System Registry is selected, then the tool shows a list of existing keys when it opens a wallet or saves a new wallet. The list appears in:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS
```

You can select one of the existing locations or enter the name for a new location (registry key). If you enter a new key called `key1`, for example, then the tool creates the following registry key:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\KEY1
```

The encrypted wallet is stored in:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\KEY1\EWALLET.P12
```

The obfuscated wallet is stored in:

```
\\HKEY_CURRENT_USER\SOFTWARE\ORACLE\WALLETS\KEY1\CWALLET.SSO
```

If you do not select the Use Windows System Registry check box, then the tool displays all the available drives and directories on the local computer. You can select one of the existing directories or enter a new directory. The tool stores the encrypted or the obfuscated wallet in the selected directory or creates the directory if it does not exist.

- [About Sharing Wallets and sqlnet.ora Files Among Multiple Databases](#)  
Multiple nonreplicated databases cannot share wallets.

 **See Also:**

*Oracle Database Enterprise User Security Administrator's Guide* for more information about using Oracle Wallet Manager

## About Sharing Wallets and `sqlnet.ora` Files Among Multiple Databases

Multiple nonreplicated databases cannot share wallets.

If `sqlnet.ora` files contain wallet location information, then databases also cannot share `sqlnet.ora` files.

The only exception to this rule is password-authenticated or Kerberos-authenticated enterprise user security with default database-to-directory connection configuration that uses passwords. This configuration keeps database wallets in the default location, where Database Configuration Assistant creates them. In this situation, no wallet location information is stored in the `sqlnet.ora` file, and the wallet can be shared among multiple databases.

 **Note:**

If SSL is used for enterprise user authentication, then the wallet location must be specified in the `sqlnet.ora` file. So, `sqlnet.ora` files cannot be shared by multiple databases for SSL-authenticated enterprise users.



# 13

## Oracle PKI Integration with Windows

Describes Windows public key infrastructure.

Learn about the integration of Oracle public key infrastructure (PKI) with public key infrastructure (Windows PKI) on Windows operating systems.

- [About Oracle Public Key Infrastructure](#)  
Learn about Oracle public key infrastructure (PKI).
- [About Windows Public Key Infrastructure](#)  
Learn about Windows public key infrastructure.

### About Oracle Public Key Infrastructure

Learn about Oracle public key infrastructure (PKI).

Oracle public key infrastructure (PKI) is used by Oracle Enterprise Security Manager, LDAP-enabled Oracle Enterprise Manager, Oracle's Secure Socket Layer (SSL) authentication, Oracle Database, and Oracle WebLogic Server.

#### Note:

- Oracle Security Manager is installed only with Oracle Database Client.
- Microsoft Windows does not support an SSL key length of less than 1024 bits any longer. Attempting to use these smaller key lengths with Oracle software on Windows returns an error.

#### See Also:

For more information, see <https://support.microsoft.com/en-us/kb/2661254>

Oracle PKI includes the following components:

- Oracle Wallets
- Oracle Wallet Manager (OWM)

Oracle Wallets store digital certificates, trust points, and private keys used in public key applications for encryption, decryption, digital signature, and verification.

### About Windows Public Key Infrastructure

Learn about Windows public key infrastructure.

 **Note:**

Microsoft Certificate Store integration works only with digital certificates that use Microsoft Enhanced Cryptographic Provider. To create these certificates, you must install Windows High Encryption Pack and select Microsoft Enhanced Cryptographic Provider. Also, when there are multiple certificates available for the same key usage (signature/key exchange), the first certificate retrieved is used for Oracle SSL.

- [About Microsoft Certificate Stores](#)  
Microsoft Certificate Stores are repositories for storing digital certificates and their associated properties.
- [About Microsoft Certificate Services](#)  
Learn about Microsoft Certificate Services (MCS) and its associated modules.
- [Using Microsoft Certificate Stores with Oracle PKI Applications](#)  
Wallet Resource Locator (WRL) specifies that parameter `WALLET_LOCATN` in file `sqlnet.ora` identifies a particular PKI.

## About Microsoft Certificate Stores

Microsoft Certificate Stores are repositories for storing digital certificates and their associated properties.

Windows operating systems store digital certificates and certificate revocation lists in logical and physical stores. Logical stores contain pointers to public key objects in physical stores. Logical stores enable public key objects to be shared between users, computers, and services without requiring storage of duplicates of objects for each user, computer, or services. Public key objects are physically stored in the certificate authority of the local computer or, for some user certificates, in Active Directory. Standard system certificate stores defined by Microsoft include:

- MY or Personal
- CA
- ROOT

MY or Personal holds a user's certificates for which the associated private key is available. The MY certificate store maintains certificate properties that indicate the Cryptographic Service Provider (CSP) associated with the private key. An application uses this information to obtain the private key from the CSP for the associated certificate. CA holds issuing or intermediate certificate authority (CA) certificates. ROOT holds only self-signed CA certificates for trusted root CAs.

## About Microsoft Certificate Services

Learn about Microsoft Certificate Services (MCS) and its associated modules.

Microsoft Certificate Services (MCS) consists of the following modules:

- Server Engine
- Intermediary

- Policy

Server Engine handles all certificate requests. It interacts with other modules at each processing stage to ensure that the proper action is taken based on the state of the request. The Intermediary module receives requests for new certificate from clients and then submits them to Server Engine. The Policy module contains the set of rules controlling the issuance of certificates. This module may be upgraded or customized as needed.

## Using Microsoft Certificate Stores with Oracle PKI Applications

Wallet Resource Locator (WRL) specifies that parameter `WALLET_LOCATN` in file `sqlnet.ora` identifies a particular PKI.

You can choose between using Oracle Wallet or Microsoft Certificate Stores by setting parameter `WALLET_LOCATN` in `sqlnet.ora`. To use credentials from Microsoft Certificate Stores, set parameter `WALLET_LOCATN` in `sqlnet.ora` to:

```
WALLET_LOCATN = (SOURCE = (METHOD=MCS))
```

The Oracle application uses Oracle's TCP/IP with SSL protocol (TCPS) to connect to Oracle Server. The SSL protocol uses X.509 certificates and trust points from the user's Microsoft Certificate Store for SSL authentication.

# Using Oracle Database with Microsoft Active Directory

Learn how to configure and use Microsoft Active Directory as the LDAP directory.

- [Overview of Microsoft Active Directory Support](#)  
Describes how Microsoft Active Directory is used as an LDAP directory server by Oracle Database.
- [Overview of Oracle Components That Integrate with Active Directory](#)  
The following Oracle Database features support or have been specifically designed to integrate with Active Directory:
- [Overview of Requirements for Using Oracle Database with Active Directory](#)  
To use Net Directory Naming with Active Directory, you must have a supported Windows operating system and Oracle software releases, and you must create Oracle schema objects and an Oracle Context.
- [Configuring Client Computers and Oracle Database to Use Active Directory](#)  
Oracle Net Configuration Assistant enables you to configure client computers and Oracle Database to access a directory server.
- [About Testing Connectivity](#)  
Describes how to connect to an Oracle Database server through Active Directory.
- [Overview of Access Control List Management for Oracle Directory Objects](#)  
Identifies the security groups specific to Oracle directory objects within Active Directory and explains how to add and delete security group members.

## Overview of Microsoft Active Directory Support

Describes how Microsoft Active Directory is used as an LDAP directory server by Oracle Database.

- [About Microsoft Active Directory](#)  
Active Directory is the LDAP-compliant directory server included with Windows server operating systems.
- [About Accessing Active Directory](#)  
When using Oracle features that support Active Directory, ensure that the Active Directory computer can be successfully reached using all possible TCP/IP host name forms to reach the domain controller.

## About Microsoft Active Directory

Active Directory is the LDAP-compliant directory server included with Windows server operating systems.

Active Directory stores all Windows operating system information, including users, groups, and policies. Active Directory also stores information about network resources (such as databases) and makes this information available to application users and

network administrators. Active Directory enables users to access network resources with a single login. The scope of Active Directory can range from storing all the resources of a small computer network to storing all the resources of several wide areas networks (WANs).

## About Accessing Active Directory

When using Oracle features that support Active Directory, ensure that the Active Directory computer can be successfully reached using all possible TCP/IP host name forms to reach the domain controller.

For example, if the host name of the domain controller is `server1` in the domain `example.com`, then ensure that you can ping that computer using all of the following:

- `server1.example.com`
- `example.com`
- `server1`

Active Directory often issues referrals back to itself in one or more of these forms, depending upon the operation being performed. If any of the forms cannot reach the Active Directory computer, then some LDAP operations may fail.

## Overview of Oracle Components That Integrate with Active Directory

The following Oracle Database features support or have been specifically designed to integrate with Active Directory:

- [About Directory Naming](#)  
Oracle Database provides Oracle Net Services directory naming, which makes use of a directory server.
- [About Automatic Discovery of Directory Servers](#)  
Oracle Net Configuration Assistant provides automatic discovery of directory servers.
- [About Integration with Windows Tools](#)  
Describes about the Windows integration tools.
- [About User Interface Extensions for Oracle Net Directory Naming](#)  
The property menus of Oracle Database service and net service name objects in Windows Explorer and Active Directory Users and Computers have been enhanced.
- [About Enhancement of Directory Object Type Descriptions](#)  
Oracle directory object type descriptions in Active Directory have been enhanced to make them easier to understand.
- [About Integration with Windows Login Credentials](#)  
Oracle Database and configuration tools can use the login credentials of the Windows user currently logged on to connect to Active Directory without having to reenter the login credentials.
- [Overview of Oracle Directory Objects in Active Directory](#)  
Learn about Oracle directory objects in Active Directory.

## About Directory Naming

Oracle Database provides Oracle Net Services directory naming, which makes use of a directory server.

This feature has been enabled to work with Microsoft Active Directory. Directory Naming enables clients to connect to the database making use of information stored centrally in an LDAP-compliant directory server such as Active Directory. For example, any net service name previously stored in the `tnsnames.ora` file can now be stored in Active Directory.

## About Automatic Discovery of Directory Servers

Oracle Net Configuration Assistant provides automatic discovery of directory servers.

When you select Active Directory as the directory server type, Oracle Net Configuration Assistant automatically discovers the directory server location and performs related tasks.

### Related Topics

- [Configuring Client Computers and Oracle Database to Use Active Directory](#)  
Oracle Net Configuration Assistant enables you to configure client computers and Oracle Database to access a directory server.

## About Integration with Windows Tools

Describes about the Windows integration tools.

Oracle Database services, net service names, and enterprise role entries in Active Directory can be displayed and tested in the following Windows tools:

- Windows Explorer
- Active Directory Users and Computers

Windows Explorer displays the hierarchical structure of files, directories, and local and network drives on your computer. It can display and test Oracle Database service and net service name objects.

Active Directory Users and Computers is an administrative tool installed on Windows servers configured as domain controllers. This tool enables you to add, modify, delete, and organize Windows accounts and groups, and publish resources in the directory of your organization. Like Windows Explorer, it can display and test Oracle Database service and net service name objects. Additionally, it can manage access control.

### Related Topics

- [Testing Connectivity from Microsoft Tools](#)  
Learn how you can test connectivity to an Oracle Database server from Microsoft tools.
- [Overview of Access Control List Management for Oracle Directory Objects](#)  
Identifies the security groups specific to Oracle directory objects within Active Directory and explains how to add and delete security group members.

## About User Interface Extensions for Oracle Net Directory Naming

The property menus of Oracle Database service and net service name objects in Windows Explorer and Active Directory Users and Computers have been enhanced.

When you right-click these Oracle directory objects, you now see two new options for testing connectivity:

- Test
- Connect with SQL\*Plus

The Test option tests whether the user name, password, and net service name you initially entered can actually connect to Oracle Database. The Connect with SQL\*Plus option starts SQL\*Plus, which enables you to perform database administration, run scripts, and so on.

### Related Topics

- [Testing Connectivity from Microsoft Tools](#)  
Learn how you can test connectivity to an Oracle Database server from Microsoft tools.

## About Enhancement of Directory Object Type Descriptions

Oracle directory object type descriptions in Active Directory have been enhanced to make them easier to understand.

In the right pane of Oracle Directory Objects in Active Directory Users and Computers, for example, the Type column reveals that `sales` is an Oracle Net Service name.

## About Integration with Windows Login Credentials

Oracle Database and configuration tools can use the login credentials of the Windows user currently logged on to connect to Active Directory without having to reenter the login credentials.

This feature has the following benefits:

- Oracle clients and databases can securely connect to Active Directory and retrieve the net service name.
- Oracle configuration tools can connect automatically to Active Directory and configure Oracle Database and net service name objects. The enabled tools include Oracle Net Configuration Assistant and Database Configuration Assistant.
- Oracle clients can make secure access over the internet to avoid anonymous binds to the directory. The enhanced security enables the sites to restrict access to Database Service by setting access control (ACL) on Database Service DN in Directory Server. The enhancement gives clients the option to use authenticated binds for LDAP name lookup. Clients have access to Database Service object if the object (DN of Database Service Entry) has been configured with restrictive access control.

### Configuration on machines that require authenticated name lookups

Add the following entry in `sqlnet.ora` to enable authenticated name lookup:

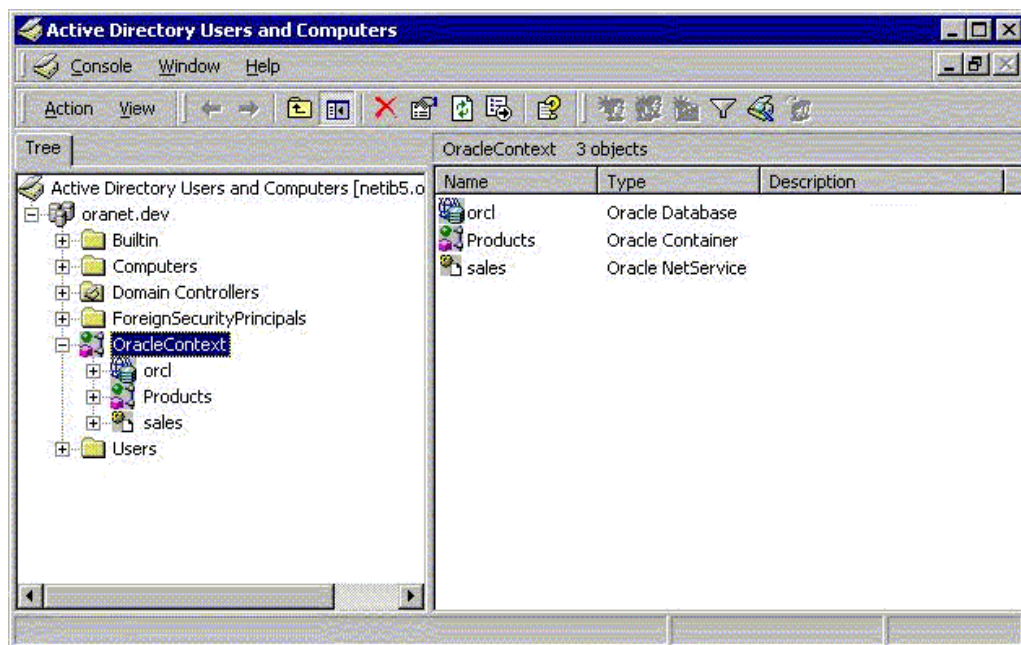
```
names.ldap_authenticate_bind = TRUE
```

## Overview of Oracle Directory Objects in Active Directory

Learn about Oracle directory objects in Active Directory.

If Oracle Database and Oracle Net Services are installed and configured to access Active Directory, then Active Directory Users and Computers displays Oracle directory objects, as illustrated in Oracle Directory Objects in Active Directory Users and Computers:

**Figure 14-1 Oracle Directory Objects in Active Directory Users and Computers**



Oracle Directory Objects describes the Oracle directory objects appearing in Oracle Directory Objects in Active Directory Users and Computers.

**Table 14-1 Oracle Directory Objects**

Object	Description
oranet.dev	The domain in which you created your Oracle Context. This domain (also known as the administrative context) contains various Oracle entries to support directory naming. Oracle Net Configuration Assistant automatically discovers this information during Oracle Database integration with Active Directory.
OracleContext	The top-level Oracle entry in the Active Directory tree. It contains Oracle Database service and net service name object information. All Oracle software information is placed in this folder.
orcl	The Oracle Database service name used in this example.
Products	Folder for Oracle product information.



**Table 14-1 (Cont.) Oracle Directory Objects**

Object	Description
sales	The net service name object used in this example.
Users	Folder for the Oracle security groups. Enterprise users and roles created with Oracle Enterprise Security Manager also appear in this folder.

**Related Topics**

- [Overview of Access Control List Management for Oracle Directory Objects](#)  
 Identifies the security groups specific to Oracle directory objects within Active Directory and explains how to add and delete security group members.

## Overview of Requirements for Using Oracle Database with Active Directory

To use Net Directory Naming with Active Directory, you must have a supported Windows operating system and Oracle software releases, and you must create Oracle schema objects and an Oracle Context.

 **Note:**

- The Oracle schema objects and Oracle Context can both be created by running Oracle Net Configuration Assistant.
- Regardless of the Oracle Database Client and Oracle Database releases you are using, you must be running in a Windows Server domain to integrate Net Directory Naming with Active Directory.

- [Creating Oracle Schema Objects](#)  
 You must create Oracle schema objects to use net directory naming features with Active Directory.
- [Creating an OracleContext](#)  
 You must create an Oracle Context to use net directory naming features with Active Directory.
- [About Directory Naming Software Requirements](#)  
 Directory naming method maps connect identifiers to connect descriptors contained in Microsoft Active Directory server.

## Creating Oracle Schema Objects

You must create Oracle schema objects to use net directory naming features with Active Directory.

Schema objects are sets of rules for Oracle Net Services and Oracle Database entries and their attributes stored in Active Directory. The following restrictions apply to creating Oracle schema objects to use with Active Directory:

- Only one Oracle schema object can be created for each forest.
- The Windows server domain controller must be the operations master that allows schema updates. See your operating system documentation for instructions.

To create an Oracle schema object:

1. Log in as a member of Schema Administrator group or as a member who has rights to update the schema into schema master domain. The logged in client computer must be a part of the schema master domain. Schema master domain administrators are schema administrators by default.
2. Use Oracle Net Configuration Assistant to create the Oracle schema object. You can create your schema object during or after database installation.

If the Active Directory display is not configured to accept all 24 default languages, then Oracle schema object creation can fail while Oracle Net Configuration Assistant is configuring Active Directory as the directory server. Before running Oracle Net Configuration Assistant to complete directory access configuration, verify that the display specifiers for all 24 languages are populated by entering the following at the command prompt:

```
ldifde -p OneLevel -d cn=DisplaySpecifiers,cn=Configuration,domain context -f temp file
```

For this command, note the following:

- *domain context* is the domain context for this Active Directory server.  
For example, `dc=example,dc=com`.
- *temp file* is a file where you want to put the output.

If the command reports that fewer than 24 entries were found, then you can still use Oracle Net Configuration Assistant. However, the report indicates that the Oracle schema object creation failed, rather than simply reporting that display specifiers for some languages were not created.

When the Oracle Net Configuration Assistant report shows failure due to less than 24 entries found, create display specifiers manually.

### Creating Display Specifiers Manually

When Oracle Net Configuration Assistant creates the Oracle schema object in Active Directory, the display specifiers for Oracle entries are not created. This means you cannot view Oracle database entries in Active Directory interfaces.

You can manually add these entries into Active Directory after the Oracle schema object has been created by doing the following, using the same Windows user identification you used when creating the Oracle schema object with Net Configuration Assistant:

1. Open a command shell.
2. Change directory to `ORACLE_HOME\ldap\schema\ad`.
3. Copy `adDisplaySpecifiers_us.sbs` to `adDisplaySpecifiers_us.ldif`.
4. Copy `adDisplaySpecifiers_other.sbs` to `adDisplaySpecifiers_other.ldif`.
5. Edit each of these `.ldif` files, replacing all occurrences of `%s_AdDomainDN%` with the domain DN for the specific Active Directory into which you want to load the display specifiers (for example, `dc=example,dc=com`).

6. Run the following commands:

```
ldapmodify -h ad hostname -Z -f adDisplaySpecifiers_us.ldif  
ldapmodify -h ad hostname -Z -f adDisplaySpecifiers_other.ldif
```

where `ad hostname` is the host name of the Active Directory domain controller to which you want to load the display specifiers.

**Related Topics**

- [About Automatic Discovery of Directory Servers](#)  
Oracle Net Configuration Assistant provides automatic discovery of directory servers.

## Creating an OracleContext

You must create an Oracle Context to use net directory naming features with Active Directory.

Oracle Context is the top-level Oracle entry in the Active Directory tree. It contains Oracle Database service and Oracle Net service name object information.

- You can create only one Oracle Context for each Windows server domain (administrative context).
- You must have the necessary permissions to create domain and enterprise objects to create the Oracle Context in Active Directory with Oracle Net Configuration Assistant.
- Use Oracle Net Configuration Assistant to create your Oracle Context. You can create the Oracle Context during or after Oracle Database Custom installation.
- [Running Oracle Network Configuration Assistant](#)  
Oracle Net Configuration Assistant is a graphical, wizard-based tool used to configure and manage Oracle Network configurations.

 **See Also:**

- *Oracle Database Installation Guide for Microsoft Windows* for installation procedures
- *Oracle Database Net Services Administrator's Guide* for configuration procedures

## Running Oracle Network Configuration Assistant

Oracle Net Configuration Assistant is a graphical, wizard-based tool used to configure and manage Oracle Network configurations.

To start Oracle Net Configuration Assistant:

1. Click **Start**, then click **All Programs**.
2. Click **Oracle - HOMENAME, Configuration and Migration Tools**, then **Net Configuration Assistant**.

3. Select the **Directory Usage Configuration** option, then click **Next**.
4. Select Directory Type **Microsoft Active Directory**, then click **Next**.

 **Note:**

The Microsoft Active Directory configuration option is only available in the Windows version of Oracle Net Configuration Assistant.

5. Select the option to configure the directory server for Oracle usage and to create or upgrade the Oracle Schema and Context, then click **Next**.
6. Enter the Active Directory host name, then click **Next**.
7. Select the option to upgrade the Oracle Schema, then click **Next**.

The next page must denote successful Directory configuration.

```
Directory usage configuration complete!
The distinguished name of your default Oracle Context is:
cn=OracleContext,DC=home,DC=com
```

8. Click **Next**, then click **Finish**.
9. The earlier message may only denote partial success:

```
The Assistant is unable to create or upgrade the Oracle Schema
for the following reason: ConfigException: Oracle Schema creation
was successful, but Active Directory Display Specifier creation
failed.oracle.net.config.ConfigException; TNS-04420: Problem
running LDAPMODIFY
```

Click **OK**, then click **Finish**.

10. If you receive the preceding error, disregard the message and rerun Oracle Net Configuration Assistant using the originally supplied values.

The wizard must complete denoting successful Directory configuration:

```
Directory usage configuration complete!
The distinguished name of your default Oracle Context is:
cn=OracleContext,DC=home,DC=com
```

Click **Next**, then click **Finish**.

## About Directory Naming Software Requirements

Directory naming method maps connect identifiers to connect descriptors contained in Microsoft Active Directory server.

A directory server provides central administration of database services and net service names, making it easier to add or relocate services.

Use Oracle Enterprise Manager or Oracle Net Manager to create net service names. To use Microsoft Active Directory naming method, the Oracle Database Client must run on supported Windows operating systems. You must have Oracle Database that is required for registering the database service as an object in Active Directory. The database server can run on any of the supported operating system, not necessarily Windows operating system.

By default, directory naming adaptor connects anonymously to active directory. Authenticated naming method requires client computer to be a part of the active directory domain to resolve a database service or net service name to a connect descriptor stored in a central directory server of its domain.

`NAMES.LDAP_AUTHENTICATE_BIND=true` parameter in `sqlnet.ora` file enables authenticated naming method.

 **See Also:**

*Oracle Database Net Services Administrator's Guide*

## Configuring Client Computers and Oracle Database to Use Active Directory

Oracle Net Configuration Assistant enables you to configure client computers and Oracle Database to access a directory server.

When you choose directory access configuration from Oracle Net Configuration Assistant, it prompts you to specify a directory server type to use. When you select Active Directory as the directory server type, the Automatic Discovery of Directory Servers feature of Oracle Net Configuration Assistant automatically:

- Discovers the Active Directory server location
- Configures access to the Active Directory server
- Creates the Oracle context (also known as your domain)

 **Note:**

Oracle Net Configuration Assistant does not configure `DIRECTORY_SERVERS` parameter in `ldap.ora`, in which case, clients automatically discover the Active Directory server for Net Naming.

If the Active Directory server already has an Oracle Context, then select the following nondefault option:

Select the directory server you want to use, and configure the directory server for Oracle usage. (Create or upgrade Oracle schema objects and Oracle Context as necessary.)

Oracle Net Configuration Assistant reports that the Oracle Context does not exist. Ignore this and choose to create the Oracle Context anyway. Directory access configuration completes without trying to re-create the existing Oracle Context.

 **Note:**

Regardless of the Oracle Database Client and Oracle Database releases you are using, you must be running a Windows Server domain to take advantage of the automatic directory server discovery features of Oracle Net Configuration Assistant. Oracle Net Configuration Assistant does not automatically discover your directory server, and instead prompts you for additional information, such as the Active Directory location.

**Related Topics**

- [Creating Oracle Schema Objects](#)  
You must create Oracle schema objects to use net directory naming features with Active Directory.
- [Overview of Requirements for Using Oracle Database with Active Directory](#)  
To use Net Directory Naming with Active Directory, you must have a supported Windows operating system and Oracle software releases, and you must create Oracle schema objects and an Oracle Context.
- [About Automatic Discovery of Directory Servers](#)  
Oracle Net Configuration Assistant provides automatic discovery of directory servers.

 **See Also:**

*Oracle Database Net Services Administrator's Guide* for configuration procedures

## About Testing Connectivity

Describes how to connect to an Oracle Database server through Active Directory.

- [Testing Connectivity from Client Computers](#)  
When using Oracle Net directory naming, client computers connect to a database by specifying the database or net service name entry that appears in the Oracle Context.
- [Testing Connectivity from Microsoft Tools](#)  
Learn how you can test connectivity to an Oracle Database server from Microsoft tools.

## Testing Connectivity from Client Computers

When using Oracle Net directory naming, client computers connect to a database by specifying the database or net service name entry that appears in the Oracle Context.

For example, if the database entry under the Oracle Context in Active Directory is `orcl`, and the client and the database are in the same domain, then a user connects to the database through SQL\*Plus by entering the following connect string:

```
SQL> CONNECT username@orcl
Enter password: password
```

If the client and the database are in different domains, then a user connects to the database through SQL\*Plus by entering:

```
SQL> CONNECT username@orcl.domain
Enter password: password
```

where *domain* is the domain in which the Oracle Database server is located.

The LDAP naming adapter has an internal function called **simplified naming**, which attempts to translate a DNS-style name into an x500 (LDAP) style name (DN) based on the naming convention used in `ldap.ora:DEFAULT_ADMIN_CONTEXT`.

It relies on `ldap.ora:default_admin_context` using either an **org** form or a **domain component (dc)** form. This cues the mechanism to use either of the following conventions to convert the domain name to an x500 DN:

- 'dc=, dc='
- 'ou=, o='
- 'ou=, o=, c='

For example,

```
SQL> CONNECT SMITH@hr.example.com
Enter password: password
```

The following values for `default_admin_context` results in the associated DN:

```
DEFAULT_ADMIN_CONTEXT="o=stdev"
```

The resulting DN is

```
cn=HR,cn=OracleContext,ou=EXAMPLE,o=COM
```

```
DEFAULT_ADMIN_CONTEXT="dc=oracle, dc=com"
```

The resulting DN is

```
cn=HR,cn=OracleContext,dc=EXAMPLE,dc=COM
```

```
DEFAULT_ADMIN_CONTEXT="o=oracle,c=us"
```

The resulting DN is

```
cn=HR,cn=OracleContext,o=EXAMPLE,c=COM
```

 **Note:**

The value of the `default_admin_context` is not used literally, since the queried-name is given in a fully qualified form. The `default_admin_context` determines which style DN is produced, or which side to use when converting each domain in the given DN component.

DNS-style conventions enable client users to access an Oracle Database server through a directory server by entering minimal connection information, even when the client computer and Oracle Database server are in separate domains. Names following the X.500 convention are longer, especially when the client and Oracle Database server are located in different domains (also known as administrative contexts).

 **See Also:**

- *Oracle Database Net Services Administrator's Guide* for more information about Configuration Management Concepts
- *Oracle Database Installation Guide for Microsoft Windows* for more information about Minimum Requirements for Passwords

## Testing Connectivity from Microsoft Tools

Learn how you can test connectivity to an Oracle Database server from Microsoft tools.

Oracle directory objects in Active Directory are integrated with the following Microsoft tools:

- Windows Explorer
- Active Directory Users and Computers

You can test connectivity to an Oracle Database server from within these Microsoft tools by connecting to it, or you can just test the connection with actually connecting. To test connectivity:

1. Start Windows Explorer or Active Directory Users and Computers.

To start Windows Explorer:

- a. From the **Start** menu, select **All Programs**, then select **Accessories**, and then select **Windows Explorer**.
- b. Expand **Network**.
- c. Expand **Directory**.

To start Active Directory Users and Computers:

From the **Start** menu, select **All Programs**, then select **Administrative Tools**, and then select **Active Directory Users and Computers**.

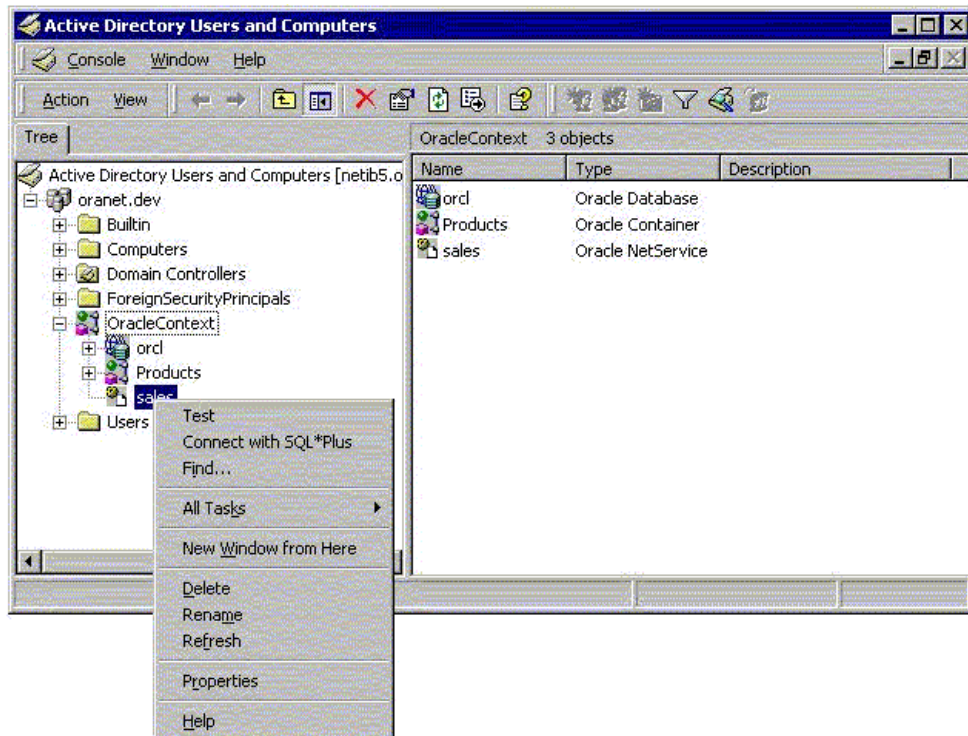


 **Note:**

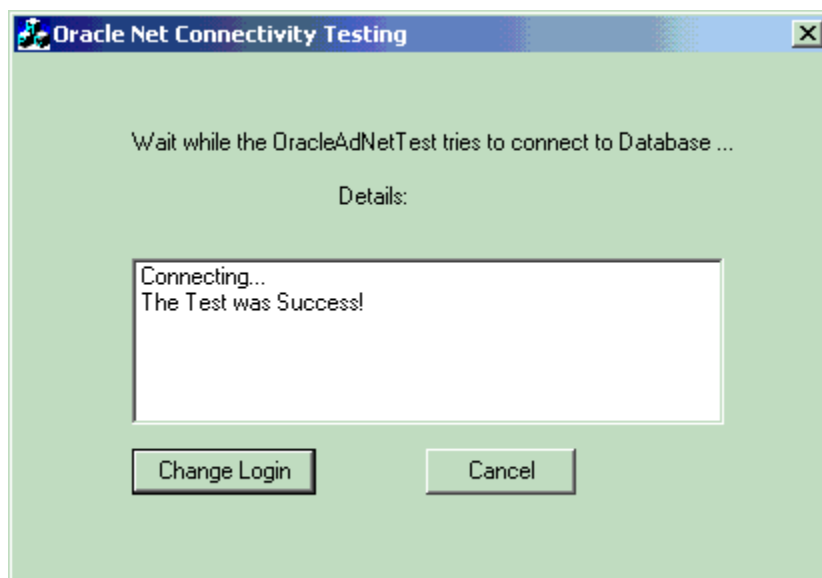
All clients accessing an Oracle Database server through Active Directory require read access on all net service name objects in the Oracle Context. If Oracle Net is not configured to require authentication for name lookup, then clients must be able to authenticate anonymously with Active Directory. With Windows Server domain, this requires changing the Active Directory default setting so that anonymous access is allowed. If anonymous access is not going to be allowed to this directory the clients must be configured to authenticate and net service objects must have access control definitions that allow clients to read them as appropriate.

2. Expand the domain in which your Oracle Context is located.
3. Expand your Oracle Context.
4. Right-click a database service or Oracle Net Service name object.

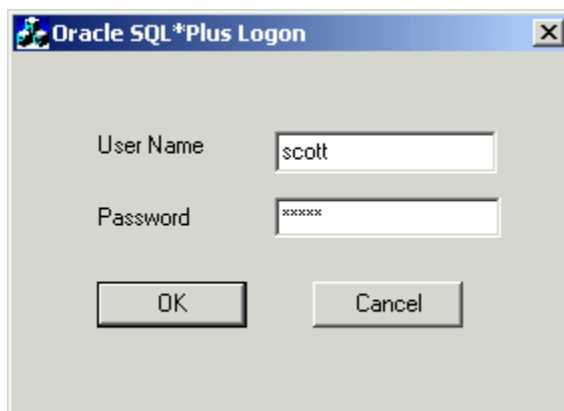
A menu appears with several options. This section covers only the **Test** and **Connect with SQL\*Plus** options.



5. If you want to test the database connection without actually connecting to it, then choose **Test**. A status message appears describing the status of your connection attempt.



6. If you want to test the database connection by actually connecting to it, then choose **Connect with SQL\*Plus**. The Oracle SQL\*Plus Logon dialog appears.



7. Enter your user name and password, then click **OK**. A status message appears describing the status of your connection attempt.

 **See also:**

*Oracle Database Net Services Administrator's Guide* for more information

## Overview of Access Control List Management for Oracle Directory Objects

Identifies the security groups specific to Oracle directory objects within Active Directory and explains how to add and delete security group members.

- [Overview of Security Groups](#)  
Security groups are automatically created when the Oracle Context is created in Active Directory.
- [Setting ACLs on Net Service Entries](#)  
Use the Microsoft `Dsacls.exe` tool to set ACLs on directory objects.
- [Adding and Deleting Security Group Members](#)  
Learn how to add or remove users in the security groups with Active Directory Users and Computers.

## Overview of Security Groups

Security groups are automatically created when the Oracle Context is created in Active Directory.

The user configuring access (and thus creating the Oracle Context) is automatically added to each group.

- [About OracleDBCreators](#)  
The `OracleDBCreators` group is for the person registering the Oracle Database server.
- [About OracleNetAdmins](#)  
Describes the various tasks that the users in this group can perform.
- [About Oracle Net Services Objects](#)  
In Oracle Database Client 11g or later, directory clients may optionally be configured to authenticate with the directory while resolving DB names to connect strings.

### About OracleDBCreators

The `OracleDBCreators` group is for the person registering the Oracle Database server.

The domain administrator is automatically a member of this group. Users in this group can:

- Create new Oracle Database objects in the Oracle Context.
- Modify the Oracle Database objects that they create.
- Read, but not modify, the membership for this group.

### About OracleNetAdmins

Describes the various tasks that the users in this group can perform.

Users in the `OracleNetAdmins` group can:

- Create, modify, and read Oracle Net Services objects and attributes.
- Read the group membership of this group.

### About Oracle Net Services Objects

In Oracle Database Client 11g or later, directory clients may optionally be configured to authenticate with the directory while resolving DB names to connect strings.

This makes it possible for Oracle Net Services objects to be protected using ACLs.

There are many ways in which the identities of users may be defined in the directory, and how those users or certain groups of users may be given access to some or all Net Services. Oracle Database supplies no predefined groups, and has no procedures in the config tools for defining read-access restrictions on this data. Therefore, administrators must use standard object management tools from their directory system to manually create any necessary groups and ACLs. Existing identity structures may be referred to by Net Service ACLs.

The access definitions for objects are complex and may involve security properties which are inherited from parent nodes in the Directory Information Tree (DIT).

Oracle recommends that the administrators should refer to the relevant tools and documentation for the directory system they are using, and formulate or integrate access management for Oracle Net Services objects into a directory-wide policy and security implementation.

 **Note:**

Pre-11g clients can only bind to the directory as **anonymous**, so any ACL protection on Net Services disables older clients. Access Control can only be implemented if all clients requiring access to these objects are 11g or later.

## Setting ACLs on Net Service Entries

Use the Microsoft `Dsacls.exe` tool to set ACLs on directory objects.

The `Dsacls.exe` command-line tool displays and changes permissions (access control entries) in the Access Control List (ACL) of objects in Active Directory. This command-line tool is included with the support tools on the CD-ROM.

Examples:

To enable an anonymous generic read on the `orcl` service, run the following command:

```
dsaccls "CN=orcl,CN=OracleContext,OU=Example,O=Com" /G "anonymous logon":GR
```

To enable a generic read on the `orcl` service for the user `smith` in the `EXAMPLE` domain, run the following command:

```
dsaccls "CN=orcl,CN=OracleContext,OU=Example,O=Com" /G example\smith:GR
```

To disable an anonymous generic read on the `orcl` service, run the following command:

```
dsaccls "CN=orcl,CN=OracleContext,OU=Example,O=Com" /R "anonymous logon"
```

To disable a generic read on the `orcl` service for the user `smith` in the `EXAMPLE` domain, run the following command:

```
dsaccls "CN=orcl,CN=OracleContext,OU=Example,O=com" /R example\smith
```



### See Also:

<http://support.microsoft.com/kb/281146> for a complete description of the `Dsacls.exe` tool

## Adding and Deleting Security Group Members

Learn how to add or remove users in the security groups with Active Directory Users and Computers.

You can add or remove users in the security groups with Active Directory Users and Computers.

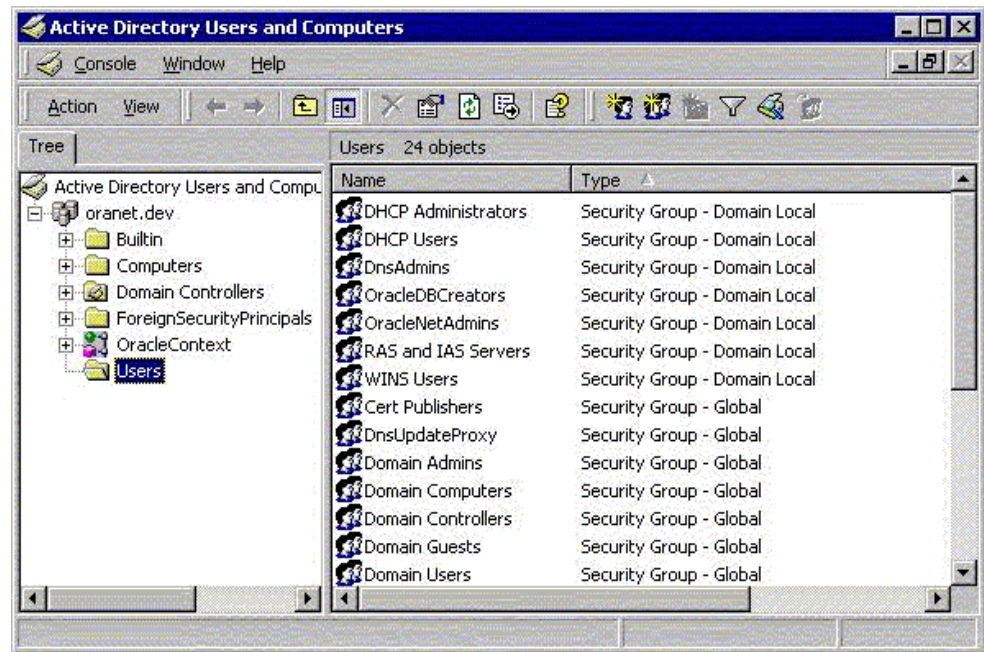


### Note:

Use Active Directory Users and Computers to perform the procedures described in this section. Windows Explorer does not provide the necessary functionality.

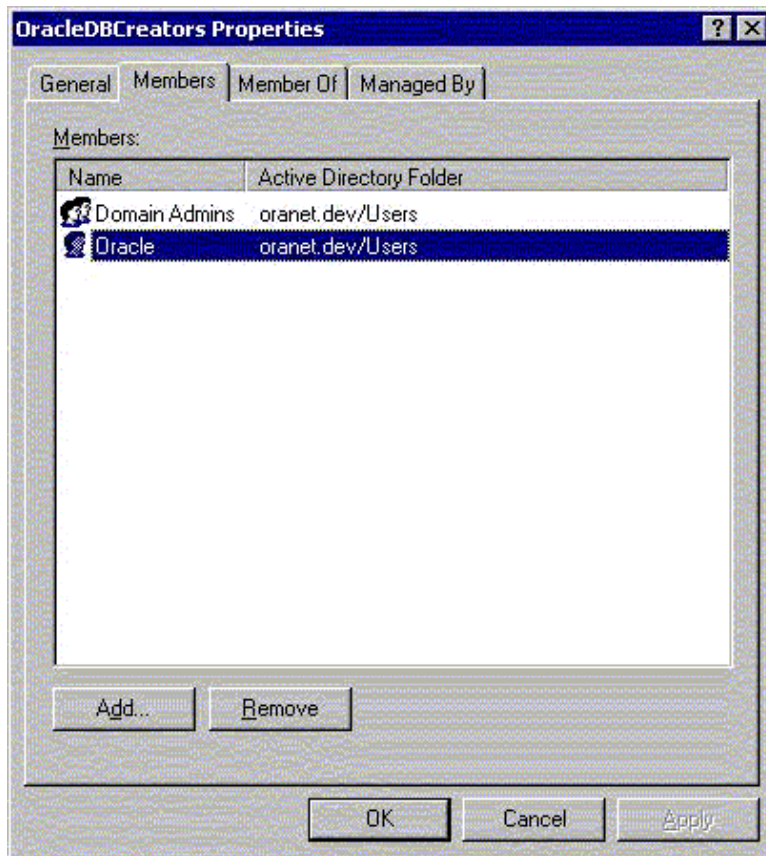
To add or remove users:

1. From the **Start** menu, select **All Programs**, then select **Administrative Tools**, and then select **Active Directory Users and Computers**.
2. Choose **Advanced Features** from the **View** main menu.  
This enables you to view and edit information that is usually hidden.
3. Expand the domain (administrative context) in which your Oracle Context is located.
4. Expand **Users**.  
The security groups appear in the right window pane.



5. Right-click the Oracle security group that you want to view or modify.  
A menu appears with several options.
6. Choose **Properties**.
7. Choose the **Members** tab.

The Properties dialog for the group you selected appears (in this example, OracleDBCreators).



8. To add users, click **Add**.  
The Select Users, Computers, Service Accounts, or Groups dialog appears.
9. Select the users or groups you want to add and click **Add**.  
Your selections appear in the Select Users, Computers, Service Accounts, or Groups dialog.
10. To remove a user, select the user name from the Members list and click **Remove**.
11. When you are finished adding and removing users, click **OK**.

# Oracle Database Specifications for Windows

Oracle Database for Windows uses initialization parameters to enable various features of the database every time an instance is started.

- [Overview of Initialization Parameter File](#)  
An initialization parameter file is an ASCII text file containing parameters.
- [Using Sample File for Database Creation](#)  
Oracle Database provides an annotated sample initialization parameter file with alternative values for initialization parameters.
- [About SGA\\_MAX\\_SIZE Parameter](#)  
Parameter `SGA_MAX_SIZE` holds the maximum size that System Global Area (SGA) can reach for a particular instance.
- [Overview of Initialization Parameters Without Windows-Specific Values](#)  
Describes the overview of initialization parameters.
- [Displaying Initialization Parameter Values](#)  
Learn how you can view Windows-specific parameter values.
- [About Unmodifiable Database Initialization Parameters](#)  
Check the initialization parameters in the Unmodifiable Database Initialization Parameters when creating a new database.
- [About Calculating Database Limits](#)  
Use the size guidelines in this section to calculate Oracle Database limits.

## Overview of Initialization Parameter File

An initialization parameter file is an ASCII text file containing parameters.

By changing parameters and values in an initialization parameter file, you can specify, for example:

- Amount of memory Oracle Database uses
- Whether to archive filled online redo logs
- Which control files currently exist

Every database instance has a corresponding initialization parameter file and an `ORACLE_SID` registry parameter that points to the system identifier for the instance.

The initialization parameter file name takes the form `init.ora`. A single instance might have several initialization parameter files, each having some differences that affect system performance.



 **Note:**

Your `init.ora` file for initialization parameters is set by Oracle Universal Installer during database installation. These parameter settings may vary depending on your hardware configuration.

- [About the Location of the Initialization Parameter File](#)  
Describes the location of the initialization parameter file.
- [About Editing The Initialization Parameter File](#)  
To customize Oracle Database functions, you may be required to edit the initialization parameter file.
- [About Oracle Database Configuration Assistant Renaming init.ora](#)  
When you create a database using Oracle Database Configuration Assistant (Oracle DBCA), a Server Parameter File (SPFILE) is created from the initialization parameter file, and the initialization parameter file is renamed.

 **See Also:**

*Oracle Database Reference* for descriptions of all initialization parameters and instructions for setting and displaying their values

## About the Location of the Initialization Parameter File

Describes the location of the initialization parameter file.

If you do not specify a different initialization file with the option `PFILE` at database startup, then by default Oracle Database uses initialization parameter files located in

`ORACLE_HOME\Database\init.ora`

 **Note:**

If you create a database manually using an SQL script, then you are required to create an initialization parameter file or copy an existing initialization parameter file and modify the contents. If you use Database Configuration Assistant to create a database, then the initialization parameter file is automatically created for you.

## About Editing The Initialization Parameter File

To customize Oracle Database functions, you may be required to edit the initialization parameter file.

Use only an ASCII text editor to modify the file.

## About Oracle Database Configuration Assistant Renaming init.ora

When you create a database using Oracle Database Configuration Assistant (Oracle DBCA), a Server Parameter File (SPFILE) is created from the initialization parameter file, and the initialization parameter file is renamed.

Oracle does not recognize the renamed file as an initialization parameter file, and it is not used after the instance is started.

If you want to modify an instance created with Oracle DBCA after it starts, you must use the `ALTER SYSTEM` statement. You cannot change the Server Parameter File itself, because it is a binary file that cannot be browsed or edited using a text editor. The location of the newly-created Server Parameter File is `ORACLE_HOME\database`. The Server Parameter File file name is `spfileSID.ora`.



### See Also:

*Oracle Database Administrator's Guide*

## Using Sample File for Database Creation

Oracle Database provides an annotated sample initialization parameter file with alternative values for initialization parameters.

These values and annotations are preceded by the comment signs (`#`), which prevent them from being processed. To activate a particular parameter, remove the preceding `#` sign. To clear a particular parameter, edit the initialization parameter file to add a comment sign. The sample file is called `initsmpl.ora` and is located in

```
ORACLE_HOME\admin\sample\pfile.
```

If you installed a starter database, then the initialization parameter file used by the starter database is located in the same directory. You can use either `initsmpl.ora` or the starter database `init.ora` as a basis for creating a new Oracle Database initialization parameter file.

To use the sample file `initsmpl.ora` as part of the database creation:

1. Rename the sample file `init.ora`.
2. Edit this file to reflect the correct location of your database control files and the name of your database, as a minimum.

Here are two examples of activation and de-activation of alternative parameters. Several initialization parameters are specified with three different values to create a small, medium, or large System Global Area, respectively. The parameter that creates a small SGA is active in this first example:

```
db_block_buffers = 200 # SMALL
# db_block_buffers = 550 # MEDIUM
# db_block_buffers = 3200 # LARGE
```

To create a medium-sized SGA, comment out the small parameter definition and activate the medium parameter definition. Edit the initialization parameter file as in this second example:

```
# db_block_buffers = 200 # SMALL
  db_block_buffers = 550 # MEDIUM
# db_block_buffers = 3200 # LARGE
```

## About SGA\_MAX\_SIZE Parameter

Parameter `SGA_MAX_SIZE` holds the maximum size that System Global Area (SGA) can reach for a particular instance.

Oracle Database can change its SGA configuration while the instance is running. This allows sizes of the buffer cache, shared pool, and the large pool to be changed without an instance shutdown.

Oracle Database can start the instances unconfigured and allow the instance to use more memory by growing SGA up to a maximum of `SGA_MAX_SIZE`. If no `SGA_MAX_SIZE` value is specified, then Oracle Database selects a default value that is the sum of all components specified or defaulted at initialization time. If `SGA_MAX_SIZE` specified in the initialization parameter file is less than the sum of all components specified or defaulted to at initialization time, then the setting of `SGA_MAX_SIZE` in the initialization parameter file serves as an upper bound.

Memory allocated for the SGA of an instance is displayed on an instance startup when using Oracle Enterprise Manager (or SQL\*Plus). You can also display the SGA size of the current instance by using the SQL\*Plus `SHOW` statement with the SGA clause.

### See Also:

- *Oracle Database Performance Tuning Guide* for more information about SGA initialization parameters
- *Oracle Database Concepts* for more information about SGA and its components

## Overview of Initialization Parameters Without Windows-Specific Values

Describes the overview of initialization parameters.

*Oracle Database Reference* describes default values for many initialization parameters as being operating system-specific. However, not all parameters that it describes as having operating system-specific values affect Windows. In these cases, Windows uses either the default value set in the Oracle Database kernel or does not use the parameter. Initialization Parameters Without Windows-Specific Values describes these initialization parameters:

**Table 15-1 Initialization Parameters Without Windows-Specific Values**

Parameter	Description
AUDIT_FILE_DEST	Supported on Windows to write XML format audit files
DB_WRITER_PROCESSES	Supported, but typically unnecessary due to Windows asynchronous I/O capabilities
COMPATIBLE_NO_RECOVERY	Uses default value set in Oracle Database kernel (no Windows-specific value)
BACKGROUND_CORE_DUMP	Specifies whether Oracle Database includes SGA in core file for Oracle Database background processes
SHADOW_CORE_DUMP	Specifies whether Oracle Database includes SGA in core file for foreground (client) processes
CORE_DUMP_DEST	Specifies directory where Oracle Database dumps core files
CPU_COUNT	Oracle Database automatically sets value to the number of processors available for your Oracle Database instance
HI_SHARED_MEMORY_ADDRESS	Not applicable to Windows
SHARED_MEMORY_ADDRESS	Not applicable to Windows
LARGE_POOL_SIZE	Uses maximum value limited by available memory
LOG_BUFFER	Starter database uses value set in Oracle Database kernel (no Windows-specific value). The Custom database creation option of Database Configuration Assistant enables you to customize the value for this parameter.
SPIN_COUNT	Uses default value set in Oracle Database kernel (no Windows-specific value)

## Displaying Initialization Parameter Values

Learn how you can view Windows-specific parameter values.

To view Windows-specific parameter values, use an ASCII editor to open the initialization parameter file:

```
ORACLE_HOME\admin\db_name\pfile\init.ora
```

To display any parameter value whether set in the initialization parameter file or the Oracle Database kernel, enter the following command at the SQL\*Plus command prompt:

```
SQL> SHOW PARAMETER parameter_name
```

where *parameter\_name* is the name of a specific initialization parameter.

## About Unmodifiable Database Initialization Parameters

Check the initialization parameters in the Unmodifiable Database Initialization Parameters when creating a new database.

They cannot be modified after you have created the database.

**Table 15-2 Unmodifiable Database Initialization Parameters**

Parameter	Description
DB_BLOCK_SIZE	Specifies size in bytes of standard Oracle Database blocks.
DB_NAME	Specifies name of the database to be created. Database name is a string of eight characters or less. You cannot change the name of a database.

**Related Topics**

- [Postinstallation Database Creation on Windows](#)  
Learn how to create a database after installing Oracle Database, using either Oracle Database Configuration Assistant or command-line tools.

## About Calculating Database Limits

Use the size guidelines in this section to calculate Oracle Database limits.

**Table 15-3 Block Size Guidelines**

Type	Size
Maximum block size	16,384 bytes or 16 kilobytes (KB)
Minimum block size	2 kilobytes (KB)
Maximum blocks for each file	4,194,304 blocks
Maximum possible file size with 16 K sized blocks	64 Gigabytes (GB) (4,194,304 * 16,384) = 64 gigabytes (GB)

**Table 15-4 Maximum Number of Files for Each Database**

Block Size	Number of Files
2 KB	20,000
4 KB	40,000
8 KB	65,536
16 KB	65,536

**Table 15-5 Maximum File Sizes**

Type	Size
Maximum file size for a FAT file	4 GB
Maximum file size in NTFS	16 Exabytes (EB)
Maximum database size	65,536 * 64 GB equals approximately 4 Petabytes (PB)
Maximum control file size	20,000 blocks

# 16

## Configuration Parameters and the Registry

Learn how to use Windows Registry for various Oracle Database for Windows components. The recommended values and ranges for configuration parameters are listed.

### Note:

Windows Registry is referred to as registry.

- [About Configuration Parameters](#)  
Oracle Database for Windows uses configuration parameters to locate files and specify run-time parameters common to all Oracle products.
- [Registry Overview](#)  
Oracle Database for Windows stores its configuration information in a repository (the registry) that is organized in a tree format.
- [Registry Parameters Overview](#)  
Describes Oracle Database for Windows registry parameters for the following keys.
- [Overview of Oracle RAC Registry Parameters](#)  
Oracle RAC registry values are based on the clusterware.
- [Managing Registry Parameters with regedit](#)  
Learn how to manage registry parameters.

## About Configuration Parameters

Oracle Database for Windows uses configuration parameters to locate files and specify run-time parameters common to all Oracle products.

When an Oracle program or an application requires translation for a particular configuration variable, Oracle Database for Windows uses the associated parameter. All Oracle parameters are stored in the registry.

## Registry Overview

Oracle Database for Windows stores its configuration information in a repository (the registry) that is organized in a tree format.

The tree format consists of keys in the registry and parameter values for the keys. Keys and parameter values can be viewed and modified in Registry Editor.

Keys are folders that appear in the left pane of a Registry Editor window. A key contains subkeys or parameters.

 **Note:**

Although Registry Editor lets you view and modify registry keys and parameter values, you typically are not required to do so. In fact, you can render your system useless if you make incorrect changes. Therefore, only advanced users must edit the registry. Back up your system before making any changes in the registry.

Parameters in the Registry Editor appear as a string, consisting of three components:

- Parameter name
- Value class or type of entry
- Value itself

For example, parameter `ORACLE_SID` can have the following entry in the registry:

```
ORACLE_SID:reg_sz:orcl1
```

Value classes for Oracle Database for Windows parameters are:

- String value with a `REG_SZ`, `REG_EXPAND_SZ` (for an expandable string), or a `REG_MULTI_SZ` (for multiple strings) prefix to identify a parameter value entry as a data string
- Binary value with a `REG_BINARY`, `REG_DWORD` or `REG_QWORD` prefix to identify a value entry as a `dword` or `qword` (hexadecimal data) entry

Most Oracle Database for the Windows parameter values are string types. Use Oracle Universal Installer defaults when a type is not given.

## Registry Parameters Overview

Describes Oracle Database for Windows registry parameters for the following keys.

Other products, such as Oracle Enterprise Manager, have additional keys and parameters that are not described.

- [About HKEY\\_LOCAL\\_MACHINE\SOFTWARE\ORACLE\KEY\\_HOMENAME](#)  
Each time you install Oracle products into a new Oracle home on your computer, `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME` is created.
- [About HKEY\\_LOCAL\\_MACHINE\SOFTWARE\ORACLE](#)  
This subkey contains the following parameter:
- [About HKEY\\_LOCAL\\_MACHINE\SYSTEM\CurrentControlSet\Services](#)  
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet` contains the following keys:

### About HKEY\_LOCAL\_MACHINE\SOFTWARE\ORACLE\KEY\_HOMENAME

Each time you install Oracle products into a new Oracle home on your computer, `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME` is created.

This subkey contains parameter values for most Oracle products.

`HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME` includes the following parameters for an Oracle home directory. Depending on products you install, additional parameters can also be created. See your Windows development manuals.

- **MSHELP\_TOOLS**  
Specifies the location of Windows help files.
- **NLS\_LANG and Other Globalization Parameters**  
Specifies supported language, territory, and character set.
- **ORA\_CWD**  
Specifies current working directory.
- **ORA\_SID\_AUTOSTART**  
Starts Oracle Database when `OracleServiceSID` service is started.
- **ORA\_SID\_PFILE**  
Specifies full path to initialization parameter file.
- **ORA\_SID\_SHUTDOWN**  
When set to `true`, the default value, this parameter shuts down the instance of Oracle Database identified by `SID` when `OracleServiceSID` is stopped manually—using either the Control Panel or Net stop command.
- **ORA\_SID\_SHUTDOWN\_TIMEOUT**  
Sets maximum time (in seconds) to wait for shutdown to complete before the service for a particular `SID` stops.
- **ORA\_SID\_SHUTDOWNTYPE**  
Specifies mode in which Oracle Database is shut down when you stop `OracleServiceSID`.
- **ORA\_TZFILE**  
Specifies the location of time zone file.
- **ORACLE\_AFFINITY**  
Enables the scheduling of class threads on more than one processor group for systems with more than 64 CPUs.
- **ORACLE\_BASE**  
Specifies the top-level Oracle directory (for example, `C:\app\username\product\12.2.0`) that contains `ORACLE_HOME`, `admin`, and `oradata`.
- **ORACLE\_GROUP\_NAME**  
Specifies the name of the group containing icons of the Oracle products installed.
- **ORACLE\_HOME**  
Specifies Oracle home directory in which Oracle products are installed.
- **ORACLE\_HOME\_KEY**  
The `HKEY_LOCAL_MACHINE` location of Oracle parameters.
- **ORACLE\_HOME\_USER**  
A string type entry that holds the Oracle Home User value.
- **ORACLE\_HOMENAME**  
Specifies home name of Oracle home directory in which Oracle products are installed.
- **ORACLE\_PRRITY**  
Determines Windows scheduling priorities of the threads within the Oracle Database management system process.



- **ORACLE\_SID**  
Specifies the name of the Oracle Database instance on the host computer.
- **OSAUTH\_PREFIX\_DOMAIN**  
Enables user authentication.
- **RDBMS\_ARCHIVE**  
Specifies the location of backup database files.
- **RDBMS\_CONTROL**  
Specifies the location of backup database control files.
- **SQLPATH**  
Specifies the location of SQL scripts.



#### See Also:

*Oracle Database Installation Guide for Microsoft Windows* Appendix B, "Optimal Flexible Architecture" for details on the PATH variable and registry values when you are working with multiple Oracle homes.

## MSHELP\_TOOLS

Specifies the location of Windows help files.

The default value is:

```
ORACLE_HOME\mshelp
```

## NLS\_LANG and Other Globalization Parameters

Specifies supported language, territory, and character set.

This parameter specifies the language in which messages appear, the territory and its conventions for calculating week and day numbers, and the character set displayed. Oracle Universal Installer sets this value during installation based on the language setting of the operating system.

The default value for *NLS\_LANG*, if not set, is *AMERICAN\_AMERICA.US7ASCII*.

There are other globalization parameters that can be set along *NLS\_LANG* to override some values implicitly determined by *NLS\_LANG*. These parameters are:

```
NLS_DATE_FORMAT  
NLS_TIMESTAMP_FORMAT  
NLS_TIMESTAMP_TZ_FORMAT  
NLS_DATE_LANGUAGE  
NLS_NUMERIC_CHARACTERS  
NLS_CURRENCY  
NLS_ISO_CURRENCY  
NLS_DUAL_CURRENCY  
NLS_SORT
```

The following parameters can also be set along *NLS\_LANG* to determine globalization behavior that is independent from the value of *NLS\_LANG*:

NLS\_CALENDAR  
NLS\_COMP  
NLS\_NCHAR\_CONV\_EXCP  
NLS\_LENGTH\_SEMANTICS

 **Note:**

All globalization parameters set in the environment and Registry for a database client are ignored if *NLS\_LANG* is not set.

 **See Also:**

*Oracle Database Globalization Support Guide* for more information about *NLS\_LANG* and other globalization parameters

## ORA\_CWD

Specifies current working directory.

For example, if you set this parameter and then use ORADIM, a log file called *oradim.log* is created in this directory. This parameter must be manually set.

## ORA\_SID\_AUTOSTART

Starts Oracle Database when *OracleServiceSID* service is started.

The default value is *true*.

## ORA\_SID\_PFILE

Specifies full path to initialization parameter file.

The default value is *ORACLE\_BASE\admin\DB\_NAME\pfile\init.ora*

## ORA\_SID\_SHUTDOWN

When set to *true*, the default value, this parameter shuts down the instance of Oracle Database identified by *SID* when *OracleServiceSID* is stopped manually—using either the Control Panel or Net stop command.

## ORA\_SID\_SHUTDOWN\_TIMEOUT

Sets maximum time (in seconds) to wait for shutdown to complete before the service for a particular *SID* stops.

The default value is 90.

## ORA\_SID\_SHUTDOWNTYPE

Specifies mode in which Oracle Database is shut down when you stop OracleServiceSID.

Valid values are a (abort), i (immediate), and n (normal). The default value is i.

## ORA\_TZFILE

Specifies the location of time zone file.

Each file contains:

- Valid time zone names
- Offset from UTC
- Abbreviation for standard time
- Abbreviation for daylight savings time

In previous releases, the default value for ORA\_TZFILE was

```
ORACLE_BASE\ORACLE_HOME\oracore\zoneinfo\timezlrq.dat
```

Starting with Oracle Database 11g Release 2 (11.2), the default value is

```
ORACLE_HOME\oracore\zoneinfo\timezlrq_11.dat
```

The `timezone_version_number.dat` data files contain most commonly used time zones and are smaller for better database performance. The new default, `timezlrq_version_number.dat`, includes time zones not defined in the smaller file.



### See Also:

*Oracle Database Globalization Support Guide* for additional details about time zone files

## ORACLE\_AFFINITY

Enables the scheduling of class threads on more than one processor group for systems with more than 64 CPUs.

This parameter must be manually added. Oracle recommends consulting Oracle Support Services before changing this parameter. The format is:

```
namen:[[processorgroup0][processorgroup1][. .2][. .3],]{cpumask0[ cpumask1 cpumask2  
cpumask3] | ALL};  
name1:[[0][1][2][3],]{cpumask0[ cpumask1 cpumask2 cpumask3] | ALL};  
name2:[[0][1][2][3],]{cpumask0[ cpumask1 cpumask2 cpumask3] | ALL};
```

Where, `processorgroup` is an optional parameter designating Windows CPU group. On systems with 64+ logical CPUs, Windows divides all available CPUs into 4 groups (0,1,2,3) with each group containing no more than 64 logical CPUs. By default, a process utilizes single processor group. The `processorgroup` parameter enables

Oracle to use more than 64 logical CPUs. Refer to the specific hardware configuration to determine the valid processor groups.

 **Note:**

You should not use the `ORACLE_AFFINITY` parameter with multiple processor groups on a system with fewer than 64 logical cores. On production servers any system with fewer than 64 logical CPUs can have only one processor group.

Each `namen` setting must be the name of a background thread, `USER` for non-background (shadow) threads, or `DEF` for any thread type not handled specifically. Valid background thread names include `DBW0`, `LGWR`, `PMON`, `SMON`, `ARCH`, `RECO`, `CKPT`, `TRWR`, `J000` through `J999`, `P000` through `P481`, and any other name found in the `NAME` column of the `v$bgprocess` data dictionary view.

The `cpumask` sets the affinity mask of the Oracle Database process. Each affinity setting must be a valid affinity mask or its numeric equivalent for the corresponding thread name. Process affinity masks are used only when Oracle Services are first started. Each thread's affinity is set only when the individual thread is started (for example, at database startup time for the background threads).

Few examples, to use multiple processor groups in a system with 160 logical CPUs, `ORACLE_AFFINITY` registry key in `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME` may be defined as follows:

The following examples show how set the `ORACLE_AFFINITY` registry key in `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOMENAME` to use multiple processor groups in a system with 160 logical CPUs. In the following examples, it is assumed that: `USER`, `DEF` are thread class names; `0`, `1`, `2`, `3` are valid CPU groups in the system; and `4294967295` is a valid affinity mask in the corresponding CPU group.

- Affinitize `USER` (foreground) threads to all CPUs in `processorgroup1` or to all CPUs in `processorgroup2` or to all CPUs in `processorgroup3` while alternating between the processor groups for each new foreground thread. Also, affinitize `DEF` class threads to CPUs `0-31` in `processorgroup0`.

```
USER:123,ALL;DEF:0,4294967295;
```

- Affinitize `USER` class threads either to CPUs `0-19` in `processorgroup0` or to CPUs `16-31` in `processorgroup2`. Also, affinitize `DEF` class threads to CPUs `0-19` in `processorgroup1`.

```
USER:02,1048575 4294901760;DEF:1,1048575;
```

- Affinitize `USER` class threads to all the CPUs of all processor groups while alternating between the processor groups for each new foreground thread. Also, affinitize `DEF` class threads to CPUs `0-31` in all the processor groups while alternating between the processor groups for each new `DEF` class thread.

```
USER:0123,ALL;DEF:0123,4294967295;
```

- Affinitize `USER` class threads to CPUs `0-31` in `processorgroup0`, CPUs `0-19` in `processorgroup1` and CPUs `0-19` in `processorgroup2` while alternating between the processor groups for each new foreground thread.

```
USER:012,4294967295 1048575 1048575;
```

## ORACLE\_BASE

Specifies the top-level Oracle directory (for example, `C:\app\username\product\12.2.0`) that contains `ORACLE_HOME`, `admin`, and `oradata`.

The default is `ORACLE_BASE`.

## ORACLE\_GROUP\_NAME

Specifies the name of the group containing icons of the Oracle products installed.

The parameter is added to your registry when you first install Oracle products, even if Oracle Universal Installer does not create a program group for Oracle products you have installed (for example, if you have installed only Oracle Net software). The default value is `Oracle - HOMENAME`.

## ORACLE\_HOME

Specifies Oracle home directory in which Oracle products are installed.

This directory is immediately beneath the Oracle base directory in the Oracle directory hierarchy. The default value is the drive letter and name that you specify during installation.

## ORACLE\_HOME\_KEY

The `HKEY_LOCAL_MACHINE` location of Oracle parameters.

The default value is `software\oracle\HOMEID`.

## ORACLE\_HOME\_USER

A string type entry that holds the Oracle Home User value.

If Windows built-in account is used as the Oracle Home User, then the string holds `NT Authority\System` and the user is not supposed to specify it explicitly.

## ORACLE\_HOMENAME

Specifies home name of Oracle home directory in which Oracle products are installed.

The default value is the name that you specify during installation.

## ORACLE\_PRRITY

Determines Windows scheduling priorities of the threads within the Oracle Database management system process.

The format is:

```
name1:priority1;name2:priority2 . . .
```

The `name` class sets the `priority` class of the Oracle Database process. Threads can be assigned a priority either collectively or individually. The collective name `user` designates non-background (shadow) threads; the collective name `def` designates any

thread type not handled specifically. Valid individual background thread names include DBW0, LGWR, PMON, SMON, ARCH0, RECO, CKPT, TRWR, SNP0 through SNP9, and any other name found in the `NAME` column of the `v$bgprocess` data dictionary view.

The default value is `class:normal; def:normal`.

 **Note:**

`ORACLE_PRRITY` is not automatically created for you in the registry. When it is not defined in the registry, Windows default values are used for thread priorities.

## ORACLE\_SID

Specifies the name of the Oracle Database instance on the host computer.

The value of this parameter is the *SID* for the instance. The default value is specified by the entry in the Database Identification window of Oracle Universal Installer.

## OSAUTH\_PREFIX\_DOMAIN

Enables user authentication.

When it is set to `true`, it enables the server to differentiate between one username and another, whether they are local users, domain users, or domain users on another domain in your network. When it is set to `false`, the domain is ignored, and the local user becomes the default value of the operating system user returned to the server. The default value is `true`.

## RDBMS\_ARCHIVE

Specifies the location of backup database files.

The default value is `ORACLE_HOME\database\archive`.

## RDBMS\_CONTROL

Specifies the location of backup database control files.

The default value is `ORACLE_HOME\database`.

## SQLPATH

Specifies the location of SQL scripts.

The default value is `ORACLE_HOME\dfs`.

## About HKEY\_LOCAL\_MACHINE\SOFTWARE\ORACLE

This subkey contains the following parameter:

- [INST\\_LOC](#)  
Specifies the location of Oracle Universal Installer files.

## INST\_LOC

Specifies the location of Oracle Universal Installer files.

The default value is *System Drive*:\program files\oracle\inventory.

## About HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet contains the following keys:

- Control
- Enum
- HardwareProfiles
- Services

The first three are used by the operating system. You can edit only the *Services* subkey, which contains [Parameters for Oracle Database Services](#).

- [Parameters for Oracle Database Services](#)  
HKEY\_LOCAL\_MACHINE\SYSTEM\CURRENTCONTROLSET\SERVICES subkey contains additional subkeys that correspond to each Oracle Database service.

## Parameters for Oracle Database Services

HKEY\_LOCAL\_MACHINE\SYSTEM\CURRENTCONTROLSET\SERVICES subkey contains additional subkeys that correspond to each Oracle Database service.

Each service subkey contains the following parameters:

- *DisplayName* specifies the service name of the instance whose *SID* is *SID*. The default value is the name of the service. For example, *OracleServiceORCL1*, where *ORCL1* is the *SID*.
- *ImagePath* specifies the fully qualified path name of the executable started by the service and any command-line arguments passed into the executable at run time. The default value is the path to the executable file of the product.
- *ObjectName* specifies the logon user account and computer to which the service must log on. The default value is *LocalSystem*.

## Overview of Oracle RAC Registry Parameters

Oracle RAC registry values are based on the clusterware.

If you are not using the clusterware, then some of this information may not be applicable to your particular cluster environment.



### Note:

Oracle RAC is only supported on 64-bit Windows server operating systems.

- [About HKEY\\_LOCAL\\_MACHINE\SOFTWARE\ORACLE\OCR](#)  
This subkey contains the following values:

## About HKEY\_LOCAL\_MACHINE\SOFTWARE\ORACLE\OCR

This subkey contains the following values:

- `OCRROOT` points to the location of the Oracle Cluster Registry file
- `LOCAL_ONLY` which is set to `False` for a cluster installation and `True` for a single-instance database installation

## Managing Registry Parameters with regedit

Learn how to manage registry parameters.

### Note:

Do not edit your registry unless absolutely necessary. If an error occurs in your registry, then Oracle Database for Windows can stop functioning, and the registry itself can become unusable.

- [Modifying a Parameter Value with regedit](#)  
Use this procedure to modify a parameter value with regedit.
- [Adding a Registry Parameter with regedit](#)  
Use this procedure to add a registry parameter with regedit.

## Modifying a Parameter Value with regedit

Use this procedure to modify a parameter value with regedit.

To edit Oracle-related settings:

1. Start Registry Editor in one of the two ways:

- From the command prompt, enter:  

```
C:\> regedit
```
- From the **Start** menu, select **Run**, enter `regedit` in the **Open** field, and click **OK**.

The Registry Editor window appears.

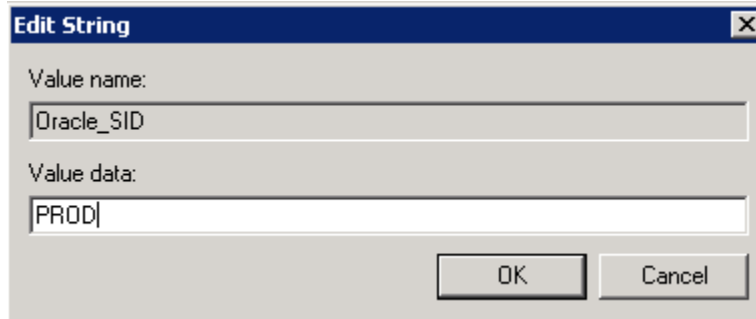
2. Navigate to the values you want to view or modify by double-clicking appropriate keys.

The left-hand side of the window shows the hierarchy of the registry keys, and the right-hand side of the window shows the various values associated with a key.

3. Double-click the parameter to change the value data to the new `SID`.

The Edit String dialog appears:





4. Make any necessary edits.
5. Click **OK**.
6. Choose **Exit** from the **Registry Editor** menu.

## Adding a Registry Parameter with regedit

Use this procedure to add a registry parameter with regedit.

To add a parameter to the registry:

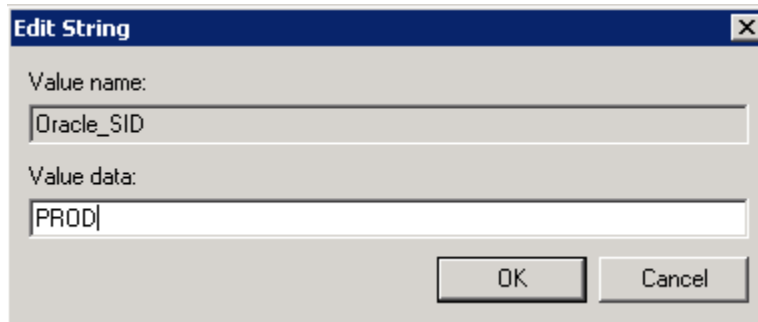
1. Start Registry Editor in one of the two ways:
  - From the command prompt, enter:  

```
C:\> regedit
```
  - From the **Start** menu, select **Run**, enter `regedit` in the **Open** field, and click **OK**.

The Registry Editor window appears.

2. Navigate to the registry key to which you want to add the new value.
3. Choose **New** from the **Edit** menu.
4. From the list, select the data type that you want to edit:
  - String Value
  - Binary value
  - DWORD (32-bit) Value
  - QWORD (64-bit) Value
  - Multi-String Value
  - Expandable String Value
5. A *New Value #1* string value name is created on the right pane of the Registry Editor window of the chosen data type. Example, `REG_EXPAND_SZ` and so on.
6. Right-click the parameter, select **Rename** and press **Enter** to rename it.
7. Double-click the parameter to change the value data to the new SID.
8. Click **OK**.

The Edit String dialog appears:



9. Type the value for the parameter.
10. Click **OK**.  
Registry Editor adds the parameter.
11. Choose **Exit** from the **Registry** menu.

# Developing Applications for Windows

Describes about the sources of information on developing applications for Windows and outlines a procedure for building and debugging external procedures.

- [About Finding Information on Application Development for Windows](#)  
Describes where to find information on developing applications specifically for Windows.
- [32-bit to 64-bit Application Migration](#)  
Use Oracle 64-bit components to create your applications.
- [About Building External Procedures](#)  
Describes how to create and use external procedures on Windows.
- [Overview of Multithreaded Agent Architecture](#)  
An agent process is started for each session to access a system at the same time leading to several thousand agent processes concurrently.
- [About Debugging External Procedures](#)  
Usually, when an external procedure fails, its C prototype is faulty.
- [About Accessing Text Files with UTL\\_FILE](#)  
Package `UTL_FILE` allows your PL/SQL programs to read and write the operating system text files.

## About Finding Information on Application Development for Windows

Describes where to find information on developing applications specifically for Windows.

These products are included on your Oracle Database Server media.

 **Note:**

Oracle Objects for OLE (OO4O), Oracle COM Automation on Windows, and Oracle Counters for Windows Performance Monitor are not supported on Oracle Database 19c for 64-bit and 32-bit Windows.

- [About Java Enhancements](#)  
Oracle Database includes an integrated Java Virtual Machine and JIT Compiler.
- [About ODP.NET](#)  
Oracle Data Provider for .NET (ODP.NET) is an implementation of a Microsoft ADO.NET data provider for Oracle Database.

- [About Oracle Developer Tools for Visual Studio](#)  
The Oracle Developer Tools for Visual Studio (ODT) is a tightly integrated "Add-in" for Microsoft Visual Studio.
- [About Oracle Providers for ASP.NET](#)  
Starting with .NET Framework 2.0, ASP.NET includes service providers that store the state in databases.
- [About XML Support](#)  
Oracle XML products include XML Developer's Kit (XDK) and Oracle XML SQL Utility.
- [About Support for Internet Applications](#)  
Oracle Database support for internet applications includes Oracle WebCenter Portal, which enables you to publish your data to the web, Oracle Web Tier (Oracle HTTP Server), and PL/SQL Embedded Gateway, which offers PL/SQL procedures stored in Oracle Database that can be started through browsers.
- [About Oracle Services for Microsoft Transaction Server](#)  
With Oracle Services for Microsoft Transaction Server, Oracle Database can be a resource manager in Microsoft Distributed Transaction Coordinator (DTC) transactions.
- [About Oracle ODBC Driver](#)  
Open Database Connectivity (ODBC) provides a common C programming interface for applications to access data from database management systems.
- [About Pro\\*C/C++ and Pro\\*COBOL Applications](#)  
Learn about Pro\*C/C++ and Pro\*COBOL applications:

**See Also:**

*Oracle Database Upgrade Guide* for a list of desupported features

## About Java Enhancements

Oracle Database includes an integrated Java Virtual Machine and JIT Compiler.

Oracle Database also provides Oracle Java Database Connectivity (JDBC) Drivers.

**Related Topics**

- *Oracle Database JDBC Developer's Guide*

## About ODP.NET

Oracle Data Provider for .NET (ODP.NET) is an implementation of a Microsoft ADO.NET data provider for Oracle Database.

ODP.NET uses Oracle native APIs to offer fast and reliable access to Oracle data and features from any .NET application. ODP.NET also uses and inherits classes and interfaces available in the Microsoft .NET Framework, .NET Standard, and .NET Core class libraries.

 **See Also:**

*Oracle Data Provider for .NET Developer's Guide for Microsoft Windows*

## About Oracle Developer Tools for Visual Studio

The Oracle Developer Tools for Visual Studio (ODT) is a tightly integrated "Add-in" for Microsoft Visual Studio.

ODT integrates with Visual Studio to make it easy to browse and edit Oracle schema objects using integrated visual designers and can automatically generate .NET code through a simple drag and drop. Developers can modify table data, execute Oracle SQL statements, edit and debug PL/SQL code, generate and edit SQL scripts, and develop and deploy .NET stored procedures. There are many more features included with these tools. For more information, visit the ODT web home at:

<http://www.oracle.com/technetwork/developer-tools/visual-studio/overview/index.html>

## About Oracle Providers for ASP.NET

Starting with .NET Framework 2.0, ASP.NET includes service providers that store the state in databases.

By storing this state in a database, applications can ensure high availability of data, while making the data equally available to all web servers.

 **See Also:**

*Oracle Providers for ASP.NET Developer's Guide for Microsoft Windows*

## About XML Support

Oracle XML products include XML Developer's Kit (XDK) and Oracle XML SQL Utility.

 **See Also:**

- *Oracle XML Developer's Kit Programmer's Guide*
- *Oracle XML DB Developer's Guide*
- *Oracle Database XML C API Reference*
- *Oracle Database XML C++ API Reference*
- *Oracle Database SQL Language Reference*
- *Oracle Database PL/SQL Packages and Types Reference*

## About Support for Internet Applications

Oracle Database support for internet applications includes Oracle WebCenter Portal, which enables you to publish your data to the web, Oracle Web Tier (Oracle HTTP Server), and PL/SQL Embedded Gateway, which offers PL/SQL procedures stored in Oracle Database that can be started through browsers.

### See Also:

- *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*
- *Oracle Fusion Middleware Tutorial for Oracle WebCenter Portal Developers*

### Note:

Oracle Portal is not supported on Oracle Database for Microsoft Windows (x64).

Oracle Database provides built-in mechanisms that address the requirements of the largest PHP, Ruby, Python Web, and Rich Internet Applications. The features include extreme connectivity, scalability, caching, nonintrusive performance acceleration, advanced security, and high-availability.

## About Oracle Services for Microsoft Transaction Server

With Oracle Services for Microsoft Transaction Server, Oracle Database can be a resource manager in Microsoft Distributed Transaction Coordinator (DTC) transactions.

Oracle Services for Microsoft Transaction Server act as a proxy for Oracle Database to the DTC.

### See Also:

- *Oracle Services for Microsoft Transaction Server Developer's Guide for Microsoft Windows*
- *Oracle Provider for OLE DB Developer's Guide for Microsoft Windows*

## About Oracle ODBC Driver

Open Database Connectivity (ODBC) provides a common C programming interface for applications to access data from database management systems.

Access to databases is managed by the ODBC Driver Manager. The driver manager provides the linkage between an ODBC application and an ODBC driver for a specific database management system.

Oracle ODBC Driver provides access to Oracle databases for applications written using the ODBC interface.

Oracle Technology Network has both Oracle home based ODBC driver and Instant Client enabled ODBC driver.

Download Instant Client enabled ODBC driver from:

<http://www.oracle.com/technetwork/database/features/instant-client/index.html>

## About Pro\*C/C++ and Pro\*COBOL Applications

Learn about Pro\*C/C++ and Pro\*COBOL applications:

### See Also:

- *Pro\*C/C++ Programmer's Guide*
- *Pro\*COBOL Programmer's Guide*
- *Oracle Call Interface Programmer's Guide*

## 32-bit to 64-bit Application Migration

Use Oracle 64-bit components to create your applications.

To migrate your business logic, refer to Migration Tips on the Microsoft Developer Network (MSDN) website at <http://msdn.microsoft.com/en-us/library/windows/desktop/aa384214.aspx>.

### Note:

NCHAR columns in user tables are not changed during the migration.



### See Also:

"Converting 32-bit Oracle Database to 64-bit Oracle Database" and "Post-Upgrade Tasks for Oracle Database" in *Oracle Database Upgrade Guide*

## About Building External Procedures

Describes how to create and use external procedures on Windows.

The following files are located in `ORACLE_HOME\rdbms\extproc`:

- **extern.c** is the code example shown in "Writing an External Procedure"  
**make.bat** is the batch file that builds the dynamic link library
- **extern.sql** automates the instructions described in "Registering an External Procedure" and "Executing an External Procedure"
- [External Procedures Overview](#)  
External procedures are functions written in a third-generation language (C, for example) and callable from within PL/SQL or SQL as if they were a PL/SQL routine or function.
- [Installing and Configuring Oracle Database and Oracle Net Services](#)  
Describes about the installation and configuration of Oracle Database and Oracle Net.
- [Writing an External Procedure](#)  
Using a third-generation programming language, you can write functions to be built into DLLs and started by `EXTPROC`.
- [Building a DLL](#)  
After writing your external procedures in a third-generation programming language, use the appropriate compiler and linker to build a DLL, making sure to export the external procedures as noted previously.
- [Registering an External Procedure](#)  
Once you have built a DLL containing your external procedures, you must register your external procedures with Oracle Database.
- [Restricting Library-Related Privileges to Trusted Users Only](#)  
The `CREATE LIBRARY`, `CREATE ANY LIBRARY`, `ALTER ANY LIBRARY`, and `EXECUTE ANY LIBRARY` privileges, and grants of `EXECUTE ON library_name` convey a great deal of power to users.
- [Executing an External Procedure](#)  
To run an external procedure, you must call the PL/SQL program unit (that is, the alias for the external function) that registered the external procedure.

## External Procedures Overview

External procedures are functions written in a third-generation language (C, for example) and callable from within PL/SQL or SQL as if they were a PL/SQL routine or function.

External procedures let you take advantage of strengths and capabilities of a third-generation programming language in a PL/SQL environment.



 **Note:**

Oracle Database also provides a special purpose interface, the call specification, that lets you call external procedures from other languages, as long as they are callable by C.

The main advantages of external procedures are:

- Performance, because some tasks are performed more efficiently in a third-generation language than in PL/SQL, which is better suited for SQL transaction processing
- Code re-usability, because dynamic link libraries (DLLs) can be called directly from PL/SQL programs on the server or in client tools

You can use external procedures to perform specific processes:

- Solving scientific and engineering problems
- Analyzing data
- Controlling real-time devices and processes

 **Note:**

Special security precautions are warranted when configuring a listener to handle external procedures.

To create and use an external procedure, perform the following steps:

1. [Installing and Configuring Oracle Database and Oracle Net Services](#)
2. [Writing an External Procedure](#)
3. [Building a DLL](#)
4. [Registering an External Procedure](#)
5. [Restricting Library-Related Privileges to Trusted Users Only](#)
6. [Executing an External Procedure](#)

 **Note:**

- You must have a C compiler and linker installed on your system to build DLLs.
- You can combine the instructions described in the fourth and fifth tasks into one SQL script that automates the task of registering and executing your external procedure. See `ORACLE_HOME\rdbms\extproc\extern.sql` for an example of a SQL script that combines these steps.



**See Also:**

"[Modifying Configuration of External Procedures for Higher Security](#)" and *Oracle Database Net Services Administrator's Guide*

## Installing and Configuring Oracle Database and Oracle Net Services

Describes about the installation and configuration of Oracle Database and Oracle Net.

- [Installing Oracle Database](#)  
Learn how to install Oracle Database on your Windows server.
- [Configuring Oracle Net Services](#)  
During database server installation, Oracle Net Configuration Assistant configures `listener.ora` and `tnsnames.ora` files for external procedure calls.

## Installing Oracle Database

Learn how to install Oracle Database on your Windows server.

Follow the steps in to install these products on your Windows server:

- Oracle Database Enterprise Edition, Oracle Database Standard Edition, or Oracle Database Personal Edition. Each type contains PL/SQL, from which external procedures are called, and the PL/SQL external procedure program (`EXTPROC`), which runs external procedures.
- Oracle Net Services
- Oracle Protocol Support



**See Also:**

*Oracle Database Installation Guide for Microsoft Windows*

## Configuring Oracle Net Services

During database server installation, Oracle Net Configuration Assistant configures `listener.ora` and `tnsnames.ora` files for external procedure calls.

When an application calls an external procedure, Oracle Net Listener starts an external procedure agent called `EXTPROC`. By default, the `extproc` process communicates directly through the server process. Using a network connection established by the listener, the application passes the following information to `EXTPROC`:

- DLL name
- External procedure name
- Parameters (if necessary)

`EXTPROC` then loads the DLL, runs the external procedure, and passes back any values returned by the external procedure.

If you overwrite default `listener.ora` and `tnsnames.ora` files, then you must manually configure the following files for the external procedure behavior described previously to occur:

- `ORACLE_HOME\network\admin\listener.ora`
- `ORACLE_HOME\network\admin\tnsnames.ora`

 **Note:**

Additional security may be required for the listener in a production environment.

 **See Also:**

*Oracle Database Net Services Administrator's Guide*

## Writing an External Procedure

Using a third-generation programming language, you can write functions to be built into DLLs and started by `EXTPROC`.

The following is a simple Microsoft Visual C++ example of an external procedure called `FIND_MAX`:

 **Note:**

Because external procedures are built into DLLs, they must be explicitly exported. In this example, the `DLEXPOR` storage class modifier exports the function `FIND_MAX` from a dynamic link library.

```
#include <windows.h>
#define NullValue -1
/*
   This function tests if x is at least as big as y.
*/
long __declspec(dllexport) find_max(long    x,
                                   short   x_indicator,
                                   long    y,
                                   short y_indicator,
                                   short *ret_indicator)
{
    /* It can be tricky to debug DLL's that are being called by a process
       that is spawned only when needed, as in this case.
       Therefore try using the DebugBreak(); command.
       This starts your debugger. Uncomment the line with DebugBreak();
       in it and you can step right into your code.
    */
    /* DebugBreak(); */
```

```
/* First check to see if you have any nulls. */
/* Just return a null if either x or y is null. */

if ( x_indicator==NullValue || y_indicator==NullValue) {
    *ret_indicator = NullValue;
    return(0);
} else {
    *ret_indicator = 0;          /* Signify that return value is not null. */
    if (x >= y) return x;
    else return y;
}
}
```

## Building a DLL

After writing your external procedures in a third-generation programming language, use the appropriate compiler and linker to build a DLL, making sure to export the external procedures as noted previously.

See your compiler and linker documentation for instructions on building a DLL and exporting its functions.

You can build the external procedure `FIND_MAX`, created in "Writing an External Procedure", into a DLL called `extern.dll` by going to `ORACLE_HOME\rdbms\extproc` and typing `make`. After building the DLL, you can move it to any directory on your system.

The default behavior of `EXTPROC` is to load DLLs only from `ORACLE_HOME\bin` or `ORACLE_HOME\lib`. To load DLLs from other directories, you must set environment variable `EXTPROC_DLLS` to a colon (:) separated list (semicolon-separated on Windows systems) of the DLL names qualified with their complete paths. The preferred way to set this environment variable is through the `ENVS` parameter in `listener.ora`.



### See Also:

*Oracle Database Development Guide* for more information on `EXTPROC`

## Registering an External Procedure

Once you have built a DLL containing your external procedures, you must register your external procedures with Oracle Database.

Now, you can configure the `EXTPROC` process to be authenticated through a `CREDENTIAL` for better security.

Oracle Database supports two new extensions to the `CREATE LIBRARY` command. This includes a `CREDENTIAL` clause and a `DIRECTORY` object option. The `CREDENTIAL` clause defines the user the `EXTPROC` runs as while the `DIRECTORY` object option specifies the directory where the DLL can be located.

To create a PL/SQL library to map to the DLL:

1. Set environment variable `EXTPROC_DLLS` in the `ENVS` parameter in `listener.ora`. For example:

```

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME=PLSExtProc)
      (ENVS=EXTPROC_DLLS=C:\app\oracle\product\19.0.0\dbhome_1\rdbms\extproc
\extern.dll)
      (ORACLE_HOME=C:\app\oracle\product\19.0.0\dbhome_1)
      (PROGRAM=extproc)
    )
  )

```

2. Start SQL\*Plus:

```
C:\> sqlplus
```

3. Connect to the database with appropriate username and password.

4. Create the PL/SQL library using the CREATE LIBRARY command:

```

DBMS_CREDENTIAL.CREATE_CREDENTIAL(...);
CREATE DIRECTORY DLL_LOC as ...;
CREATE LIBRARY externProcedures as 'extern.dll' in DLL_LOC credential
the_credential;

```

where `the_credential` is the name chosen during the `DBMS_CREDENTIAL.CREATE_CREDENTIAL` invocation

```

SQL> CREATE LIBRARY externProcedures AS 'C:\app\oracle\product
\19.0.0\dbhome_1\rdbms\ extproc\extern.dll';

```

where `externProcedures` is an alias library (essentially a schema object in the database), and

```
C:\app\oracle\product\19.0.0\dbhome_1\rdbms\extproc\extern.dll
```

is the path to the Windows operating system `dll` `extern.dll`. This example uses `C:\app\oracle\product\19.0.0` as your Oracle base and `dbhome_1` as your Oracle home.

 **Note:**

The DBA must grant the `EXECUTE` privilege on the PL/SQL library to users who want to call the library's external procedure from PL/SQL or SQL. Separate `EXECUTE` privilege on credential and directory object extensions are required for them to function properly.

5. Create a PL/SQL program unit specification.

Do this by writing a PL/SQL subprogram that uses the `EXTERNAL` clause instead of declarations and a `BEGIN...END` block. The `EXTERNAL` clause is the interface between PL/SQL and the external procedure. The `EXTERNAL` clause identifies the following information about the external procedure:

- Name
- DLL alias
- Programming language in which it was written
- Calling standard (defaults to C if omitted)

In the following example, `externProcedures` is a DLL alias. You need the `EXECUTE` privilege for this library. The external procedure to call is `find_max`. If enclosed in double quotation marks, it becomes case-sensitive. The `LANGUAGE` term specifies the language in which the external procedure was written.

```
CREATE OR REPLACE FUNCTION PLS_MAX(  
  x BINARY_INTEGER,  
  y BINARY_INTEGER)  
RETURN BINARY_INTEGER AS EXTERNAL  
  LIBRARY externProcedures  
  NAME "find_max"  
  LANGUAGE C  
  PARAMETERS (  
    x long,                -- stores value of x  
    x_INDICATOR short,    -- used to determine if x is a NULL value  
    y long,                -- stores value of y  
    y_INDICATOR short,    -- used to determine if y is a NULL value  
    RETURN_INDICATOR short ); -- need to pass pointer to return value's  
                                -- indicator variable to determine if NULL  
-- This means that my function is defined as:  
-- long max(long x, short x_indicator,  
-- long y, short y_indicator, short * ret_indicator)
```

## Restricting Library-Related Privileges to Trusted Users Only

The `CREATE LIBRARY`, `CREATE ANY LIBRARY`, `ALTER ANY LIBRARY`, and `EXECUTE ANY LIBRARY` privileges, and grants of `EXECUTE ON library_name` convey a great deal of power to users.

If you plan to create PL/SQL interfaces to libraries, only grant the `EXECUTE` privilege to the PL/SQL interface. Do not grant `EXECUTE` on the underlying library. You must have the `EXECUTE` privilege on a library to create the PL/SQL interface to it. However, users have this privilege implicitly on libraries that they create in their own schemas. Explicit grants of `EXECUTE ON library_name` are rarely required. Only make an explicit grant of these privileges to trusted users, and never to the `PUBLIC` role.

## Executing an External Procedure

To run an external procedure, you must call the PL/SQL program unit (that is, the alias for the external function) that registered the external procedure.

These calls can appear in any of the following:

- Anonymous blocks
- Standalone and packaged subprograms
- Methods of an object type
- Database triggers
- SQL statements (calls to packaged functions only)

In “Registering an External Procedure”, PL/SQL function `PLS_MAX` registered external procedure `find_max`. Follow these steps to run `find_max`:

1. Call PL/SQL function `PLS_MAX` from a PL/SQL routine named `UseIt`:

```
SET SERVER OUTPUT ON  
CREATE OR REPLACE PROCEDURE UseIt AS
```

```
a integer;  
b integer;  
c integer;  
  
BEGIN  
  
a := 1;  
b := 2;  
c := PLS_MAX(a,b);  
dbms_output.put_line('The maximum of '||a||' and '||b||' is '||c);  
  
END;
```

## 2. Run the routine:

```
SQL> EXECUTE UseIt;
```

### Related Topics

- [Registering an External Procedure](#)  
Once you have built a DLL containing your external procedures, you must register your external procedures with Oracle Database.

## Overview of Multithreaded Agent Architecture

An agent process is started for each session to access a system at the same time leading to several thousand agent processes concurrently.

The agent processes operation regardless of whether each individual agent process is currently active. Agent processes and open connections can consume a disproportionate amount of system resources. This problem is addressed by using multithreaded agent architecture.

The multithreaded agent architecture uses a pool of shared agent threads. The tasks requested by the user sessions are put in a queue and are picked up by the first available multithreaded agent thread. Because only a small percentage of user connections are active at a given moment, using a multithreaded architecture allows for more efficient use of system resources.

### See Also:

- *Oracle Database Development Guide*
- *Oracle Database Heterogeneous Connectivity User's Guide*

## About Debugging External Procedures

Usually, when an external procedure fails, its C prototype is faulty.

That is, the prototype does not match the one generated internally by PL/SQL. This can happen if you specify an incompatible C data type. For example, to pass an OUT parameter of type REAL, you must specify `float *`. Specifying `float`, `double *`, or any other C data type results in a mismatch.

In such cases, you get a `lost RPC connection to external procedure agent error`, which means that agent `extproc` terminated abnormally because the external procedure caused a core dump.

- [Using Package DEBUG\\_EXTPROC](#)  
To help you debug external procedures, PL/SQL provides the utility package `DEBUG_EXTPROC`.

 **See Also:**

*Oracle Database Data Cartridge Developer's Guide*, for information on how to avoid errors when declaring C prototype parameters

## Using Package DEBUG\_EXTPROC

To help you debug external procedures, PL/SQL provides the utility package `DEBUG_EXTPROC`.

To install the package, run the script `dbgextp.sql`, which you can find in the PL/SQL demo directory.

To use the package, follow instructions in `dbgextp.sql`. Your Oracle Database account must have `EXECUTE` privileges on the package and `CREATE LIBRARY` privileges.

To debug external procedures:

1. From Windows Task Manager, in the Processes dialog, select **ExtProc.exe**.
2. Right click, and select **Debug**.
3. Click **OK** in the message window.

If you have built your DLL in a debug fashion with Microsoft Visual C++, then Visual C++ is activated.

4. In the Visual C++ window, select **Edit > Breakpoints**.

Use the breakpoint identified in `dbgextp.sql` in the PL/SQL demo directory.

 **See Also:**

- `ORACLE_HOME\rdbms\extproc\readme.doc` (explains how to run the sample and provides debugging advice)
- *Oracle Database PL/SQL Language Reference*
- *Oracle Database Development Guide* for more information about "Calling External Procedures"
- *Oracle Database Data Cartridge Developer's Guide*

## About Accessing Text Files with UTL\_FILE

Package `UTL_FILE` allows your PL/SQL programs to read and write the operating system text files.

It provides a restricted version of the standard operating system stream file I/O, including open, put, get, and close operations. When you want to read or write a text



file, you call the function `fopen`, which returns a file handle for use in subsequent procedure calls. For example, the procedure `put_line` writes a text string and line terminator to an open file, and the procedure `get_line` reads a line of text from an open file into an output buffer.

`FSEEK`, a `UTL_FILE` subprogram, adjusts the file pointer forward or backward within the file by the number of bytes specified. In order for `UTL_FILE.FSEEK` to work correctly, the lines in the file must have the platform-specific line terminator characters. On Windows platform, the correct line terminator characters are `<CR><LF>`.

 **See Also:**

- *Oracle Database PL/SQL Packages and Types Reference* for more information about `UTL_FILE`
- *Oracle Database Development Guide* for information about "Retrieving HTTP URL Contents from PL/SQL"

# A

## Storing Tablespaces on Raw Partitions

Learn how to configure your system to store data files for a tablespace on raw partitions.

### Note:

Oracle RAC requires additional configuration tools.

- [Raw Partition Overview](#)  
Data files for tablespaces can be stored on a file system or on raw partitions. A raw partition is a portion of a physical disk that is accessed at the lowest possible level.
- [Configuring Disks for Oracle Automatic Storage Management](#)  
To use Oracle Automatic Storage Management with direct attached storage (DAS) or storage area network (SAN) storage, the disks must be stamped with a header by `asmtool` or `asmtoolg` (GUI version).

### See Also:

*Oracle Real Application Clusters Administration and Deployment Guide* for information about creating logical partitions and assigning symbolic links. Do not use this appendix to create partitions for Oracle RAC.

## Raw Partition Overview

Data files for tablespaces can be stored on a file system or on raw partitions. A raw partition is a portion of a physical disk that is accessed at the lowest possible level.

Input/output (I/O) to a raw partition offers approximately a 5% to 10% performance improvement over I/O to a partition with a file system on it.

- [About Physical Disk](#)  
A physical disk represents the entire disk and points to the following:
- [About Logical Partition](#)  
Logical partitions point to drives other than `\Device\Harddiskx\Partition0`.
- [About Physical Disk and Logical Partition Considerations](#)  
Consider the following when deciding which raw partition to use:
- [About Compatibility Issues](#)  
You can create logical partitions, but define physical disk convention names for them. For example:

## About Physical Disk

A physical disk represents the entire disk and points to the following:

```
\Device\Harddiskx\Partition0
```

Symbolic link name `\\.\PhysicalDrivex` is automatically defined by Windows for every hard disk in the computer. For example, a computer with three hard disks have the following symbolic links:

```
\\.\PhysicalDrive0  
\\.\PhysicalDrive1  
\\.\PhysicalDrive2
```

Internally, these names expand to the following:

```
\\.\PhysicalDrive0 =\Device\Harddisk0\Partition0  
\\.\PhysicalDrive1 =\Device\Harddisk1\Partition0  
\\.\PhysicalDrive2 =\Device\Harddisk2\Partition0
```

Partition0 is special, because it represents the entire physical disk regardless of any partitioning scheme on that disk. Windows writes a signature on the first block of all disks it recognizes. To avoid overwriting that block, Oracle Database skips the first block of a physical raw partition that is used for an Oracle Database data file.

 **Note:**

Although you can use physical disks, Oracle recommends that you use logical partitions.

## About Logical Partition

Logical partitions point to drives other than `\Device\Harddiskx\Partition0`.

They are initially assigned names with drive letters (`\\.\drive_letter:`) and typically re-assigned symbolic link names (`\\.\symbolic link name`). For example, `\\.\D:` may be assigned a symbolic link name of `\\.\ACCOUNTING_1`. Regardless of whether a drive letter or symbolic link name is used, logical partitions are defined to represent a specific partition in a disk rather than the entire disk. Internally, these names can expand to:

```
\\.\D:= \Device\Harddisk2\Partition1  
\\.\ACCOUNTING_1= \Device\Harddisk3\Partition2
```

 **Note:**

Oracle Database does not skip the first block of a logical raw partition used for an Oracle Database data file.

## About Physical Disk and Logical Partition Considerations

Consider the following when deciding which raw partition to use:

- Physical disks are automatically defined by Windows to represent the entire disk, and must never be defined by the user.
- Logical partitions must be defined by the user to represent a specific partition in a disk. These partitions must be logical partitions or drives contained in an extended partition. They must never be defined as `Partition0`.
- Using an entire disk (`Partition0`) for an Oracle Database data file and using a partition that occupies the entire disk for an Oracle Database data file are not the same thing. Even when a partition occupies the entire disk, there is still a small space on the disk that is not part of the partition.
- If you are using an entire disk for an Oracle Database data file (`Partition0`), then use the predefined physical raw names that Windows provides.
- If you are using a specific partition and it occupies the entire disk, then use a logical partition.
- If you are using a specific partition created with Windows disk-management tools, then define and use a symbolic link name rather than a logical partition number (even if it occupies the entire disk).

 **Note:**

For both physical and logical raw conventions, use `OCOPY` to transfer the contents of a raw partition to a standard file system for backup purposes.

## About Compatibility Issues

You can create logical partitions, but define physical disk convention names for them. For example:

```
\\.\PhysicalDriveACCOUNTING_1 = \Device\Harddisk2\Partition1  
\\.\PhysicalDriveACCOUNTING_2 = \Device\Harddisk3\Partition1
```

Oracle Database then handles data files using the physical disk convention even though it really is a logical partition. This does not cause any data corruption or loss as long as you continue to use physical disk naming conventions. Oracle recommends that you convert to the logical partition at your earliest convenience.

You can also create logical names representing `Partition0`, but this is definitely not recommended. For example:

```
\\.\ACCOUNTING_1 = \Device\Harddisk1\Partition0
```

This poses severe problems, because Disk Management typically writes a signature into the first block of every disk, and consequently may overwrite a portion of the data file header. It can also cause data loss. Never use `Partition0` with the logical partition convention.

Physical and logical partition conventions are not compatible with one another because of the extra block that is skipped for physical raw conventions. This also means you cannot simply use OCOPI to copy from a physical disk to a logical partition, because contents of these partitions are incompatible.

To convert from a physical convention to a logical convention, you must:

1. Perform a full database export to a (local) file system.
2. Create logical partitions and define logical names for these partitions.
3. Recreate the database by using the new logical partitions.
4. Perform the full database import to the newly-created database.

If your database installation uses physical disk conventions with logical partitions, Oracle recommends converting to the logical partition conventions at your earliest convenience, using the preceding steps.



#### See Also:

Your operating system documentation for information about creating extended and logical partitions

## Configuring Disks for Oracle Automatic Storage Management

To use Oracle Automatic Storage Management with direct attached storage (DAS) or storage area network (SAN) storage, the disks must be stamped with a header by `asmttool` or `asmttoolg` (GUI version).

Each DAS or SAN disk must have a partition table. Oracle recommends creating exactly one partition for each disk containing the entire disk. Use Microsoft Computer Management or the command-line tool `diskpart` to create the partition. Once the partitions have been created, run `asmttoolg` or `asmttool`. These tools associate meaningful, persistent names with disks to facilitate using those disks with Oracle Automatic Storage Management. Oracle Automatic Storage Management uses disk strings to more easily operate on groups of disks at once, so the names created by `asmttool` make this easier than using Windows drive letters. All disk names created by `asmttool` begin with the prefix `ORCLDISK` for identification purposes.

Oracle Automatic Storage Management uses the value of initialization parameter `ASM_DISKSTRING` as its search path when it discovers disks. The default value of `ASM_DISKSTRING` is `\\.\ORCLDISK $n$` . If you want a different search path, then you must specify a different value for this parameter.

 **See Also:**

- *Oracle Database Installation Guide for Microsoft Windows "Step 3: Manually Configuring Disks for Oracle Automatic Storage Management"* for instructions on using `asmtool` or `asmtoolg`
- *Oracle Automatic Storage Management Administrator's Guide* for information about "Initialization Parameter Files for an Oracle ASM Instance"

# B

## Oracle Net Services Configuration on Windows

Learn about Oracle Net Services configuration for Windows.

- [About Configuring Oracle Database to Communicate with Oracle ASM](#)  
Oracle Databases that use Oracle Automatic Storage Management (Oracle ASM) and the databases that are managed by Oracle Grid infrastructure must use Windows native authentication, which is enabled by default.
- [About Modifying Oracle Net Services Registry Parameters and Subkeys](#)  
The registry contains entries for Oracle Net Services parameters and subkeys.
- [About Listener Requirements](#)  
In Oracle Database, the listener is set to start automatically at system restart.
- [Overview of Optional Configuration Parameters](#)  
You can use the Windows parameters listed here with Oracle Net Services.
- [Overview of Advanced Network Configuration](#)  
Describes the advanced configuration procedures specifically for Oracle Net Services on Windows operating systems.



### See Also:

*Oracle Database Net Services Administrator's Guide*

## About Configuring Oracle Database to Communicate with Oracle ASM

Oracle Databases that use Oracle Automatic Storage Management (Oracle ASM) and the databases that are managed by Oracle Grid infrastructure must use Windows native authentication, which is enabled by default.

To ensure that it is, check that the `sqlnet.ora` file, by default located in `ORACLE_HOME\network\admin`, has NTS enabled. For example:

```
sqlnet.authentication_services=(NTS)
```



### See Also:

[About Windows Authentication Protocols](#)

## About Modifying Oracle Net Services Registry Parameters and Subkeys

The registry contains entries for Oracle Net Services parameters and subkeys.

To successfully add or modify Oracle Net Services configuration parameters, you must understand where they are located and the rules that apply to them.

- [About Oracle Net Service Subkeys](#)  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services contains subkeys that correspond to services.

### About Oracle Net Service Subkeys

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services contains subkeys that correspond to services.

Depending on what is installed, your Oracle Net Services consist of all or a subset of the following:

- OracleHOMENAMEClientCache
- OracleHOMENAMECMAAdmin
- OracleHOMENAMECMan
- OracleHOMENAMETNSListener

Each service subkey contains the parameters shown in Service Subkey Parameters.

**Table B-1 Service Subkey Parameters**

Parameter	Description
DisplayName	Specifies service name.
ImagePath	Specifies fully qualified path name of the executable started by service and any command line arguments passed to executable at run time.
ObjectName	Specifies logon user account and computer to which service must log on.

### About Listener Requirements

In Oracle Database, the listener is set to start automatically at system restart.

You can verify the listener status by choosing the Windows **Control Panel**, then **Administrative Tools**, and then viewing the status of **Services**.

Oracle usually recommends that you only have a single net listener service running on a Windows computer at any one time. This single listener can support multiple databases. If you must have two different net listener services running on a Windows computer at the same time, ensure that they are configured to listen on different TCP/IP port numbers.

If the same IP address and port are used for different listeners, then you expect that the second and subsequent listeners fail to bind, instead, Windows allows them all to



listen on the same IP address and port, resulting in an unexpected behavior of the listeners. This is a suspected Windows operating system problem with TCP/IP and has been reported to Microsoft.

- [Running Oracle Net Services](#)  
Oracle Net Services such as Oracle Listener, CMADMIN, and CMAN Proxy Listener run under the specified Oracle Home User account (Windows User Account) specified during the installation.

## Running Oracle Net Services

Oracle Net Services such as Oracle Listener, CMADMIN, and CMAN Proxy Listener run under the specified Oracle Home User account (Windows User Account) specified during the installation.

Oracle recommends that you specify the standard Windows User Account (not an Administrator) as the Oracle Home User. Oracle Universal Installer also has an option to create a new Windows User Account with limited privileges. When the Windows built-in account is used as the Oracle Home User, then Oracle Listener service runs under an high-privileged Local System Account (LSA) for database installation. For CMAN installation, if a Windows built-in account is used as the Oracle Home User, then CMADMIN and CMAN Proxy Listener runs under a low privileged LocalService Account.

### Note:

The Isnrctl start command or CMAN proxy start command may prompt for password when relevant service does not exist. This happens for the first time for a given alias. Once the service is created, all subsequent start commands do not prompt for password. However, there is no prompt for password if you select Use Built-in Account as Oracle Home User.

### See Also:

- *Oracle Database Net Services Administrator's Guide* for more information about User Accounts and Security
- *Oracle Database Net Services Reference* for more information about START

## Overview of Optional Configuration Parameters

You can use the Windows parameters listed here with Oracle Net Services.

Oracle Net Service first checks for the parameters as environment variables, and uses the values defined. If environment variables are not defined, then Oracle Net Services searches for these parameters in the registry.

- [About LOCAL Parameter](#)  
You can use the parameter `LOCAL` to connect to Oracle Database without specifying a connect identifier in the connect string.
- [About TNS\\_ADMIN Parameter](#)  
You can add the parameter `TNS_ADMIN` to change the directory path of Oracle Net Services configuration files from the default location of `ORACLE_HOME\network\admin`.
- [About USE\\_SHARED\\_SOCKET Parameter](#)  
You can set the parameter `USE_SHARED_SOCKET` to `true` to enable the use of shared sockets.

## About LOCAL Parameter

You can use the parameter `LOCAL` to connect to Oracle Database without specifying a connect identifier in the connect string.

The value of the parameter `LOCAL` is any connect identifier, such as a net service name. For example, if the parameter `LOCAL` is specified as `finance`, you can connect to a database from SQL\*Plus with:

```
SQL> CONNECT SMITH  
Enter password: password
```

rather than

```
SQL> CONNECT SMITH@finance  
Enter password: password
```

Oracle Net checks if `LOCAL` is defined as an environment variable or as a parameter in the registry, and uses `finance` as the service name. If it exists, then Oracle Net connects.

## About TNS\_ADMIN Parameter

You can add the parameter `TNS_ADMIN` to change the directory path of Oracle Net Services configuration files from the default location of `ORACLE_HOME\network\admin`.

For example, if you set `TNS_ADMIN` to `ORACLE_HOME\test\admin`, then the configuration files are used from `ORACLE_HOME\test\admin`.

## About USE\_SHARED\_SOCKET Parameter

You can set the parameter `USE_SHARED_SOCKET` to `true` to enable the use of shared sockets.

If this parameter is set to `true`, the network listener passes the socket descriptor for client connections to the database thread. As a result, the client does not establish a new connection to the database thread and database connection time improves. Also, all database connections share the port number used by the network listener, which can be useful if you are setting up third-party proxy servers.

This parameter only works in the dedicated server mode in a TCP/IP environment. If this parameter is set, you cannot use the Oracle database listener to spawn Oracle Database. To spawn a dedicated server for an instance of Oracle Database not

associated with the same Oracle home as the listener and have shared socket enabled, you must also set parameter `USE_SHARED_SOCKET` for both Oracle homes.

## Overview of Advanced Network Configuration

Describes the advanced configuration procedures specifically for Oracle Net Services on Windows operating systems.

- [About Configuring Authentication Method](#)
- [About Configuring Security for Named Pipes Protocol](#)
- [Modifying Configuration of External Procedures for Higher Security](#)

### About Configuring Authentication Method

Oracle Net Services provides authentication methods for Windows operating systems using Windows Native Authentication.

### About Configuring Security for Named Pipes Protocol

The network listener service may be unable to open the Named Pipes created by Oracle Names unless service `OracleHOMENAME_TNSListener` has a valid user name and password associated with it.

#### See Also:

Your operating system documentation for instructions on setting up network listener permissions

### Modifying Configuration of External Procedures for Higher Security

This section supplements the generic information provided in *Oracle Database Net Services Administrator's Guide* to configure a listener on Windows operating systems to exclusively handle external procedures. For a higher level of security, you are instructed to start the listener for external procedures from a user account with lower privileges than the `oracle` user. For Windows operating systems, this requires that you change the user account from `LocalSystem` to a local, unprivileged user for the `OracleHOMENAME_TNSListenerextproc_listener_name` service.

#### Note:

The following instructions assume that you have performed steps 1 through 5 in the section "Modifying Configuration of External Procedures for Higher Security" and the procedure described in Default Configuration for External Procedures section in *Oracle Database Net Services Administrator's Guide*.

To change the listener account:

1. Create a new user account and grant it Log on as a Service privilege.

 **Note:**

Ensure that this user account does not have general access to files owned by `oracle`. Specifically, this user must not have permission to read or write to database files or to the Oracle Database server address space. In addition, this user must have read access to the `listener.ora` file, but must not have the write access.

2. Stop service `OracleHOMENAME_TNSListener_extproc_listener_name`.

If the `OracleHOMENAME_TNSListener_extproc_listener_name` service does not exist, issue the following command from the command prompt:

```
lsnrctl start extproc_listener_name
```

This creates the `OracleHOMENAME_TNSListener_extproc_listener_name` service. When you return to the list of services, stop this service before proceeding to the next step of this procedure.

3. Select `OracleHOMENAME_TNSListener_extproc_listener_name` service in the Services dialog and then display the properties of the service.
4. Select **This Account** and enter the user name and password.
5. Start the listener by clicking **Start**. You must start the listener in this way because you cannot use the Listener Control utility to start the listener running as an unprivileged local user.

 **Note:**

You can also use NET START `OracleHOMENAME_TNSListener_extproc_listener_name` to start the listener from the command prompt. Running the listener with lower privileges prevents you from using Listener Control utility SET commands to alter the configuration of this listener in file `listener.ora`. You can perform other administrative tasks on this listener with the Listener Control utility, including stopping the listener. Oracle recommends that you complete `listener.ora` file configuration before running the listener.

 **See Also:**

- Your operating system documentation for instructions on accessing the Services dialog and stopping services
- *Oracle Database Net Services Administrator's Guide*

# C

## Running Windows Services

Oracle Database 19c supports Windows services to run under low-privileged, non-administrative accounts such as the LocalService, or an authenticated Windows User Account instead of the high-privileged Local System Account (LSA) for better security.

- [About Windows Services for Oracle Database](#)  
ORADIM creates Oracle Database service, Oracle VSS Writer service, and Oracle Scheduler service to run under the Oracle Home User account.

### About Windows Services for Oracle Database

ORADIM creates Oracle Database service, Oracle VSS Writer service, and Oracle Scheduler service to run under the Oracle Home User account.

Oracle Home User is the standard Windows User Account (not an Administrator), specified during installation, that runs most of the Windows services required by Oracle for Oracle home.

If this Oracle Home User is a Windows Local User Account or a Windows Domain User Account, then ORADIM prompts for a password for that account and accepts the same through `stdin`.

All Oracle administration tools that create Windows services have been modified to prompt for the password of Oracle Home User when the Oracle Home User is a Windows Local User Account or a Windows Domain User Account, and the password for Oracle Home User is not stored in the Oracle Wallet.

- [About Running Windows Services in Oracle Home](#)  
Depending on the type of database installation and user account used as the Oracle Home User, Windows services run under low-privileged, non-administrative accounts such as a LocalService, or an authenticated Windows User Account, or as a high-privileged Local System Account (LSA) in Oracle home.
- [Additional Privileges Required by Oracle Database Services](#)  
Certain functions performed by the Oracle Database service require additional privileges.
- [Granting Additional Operating System Privileges Manually](#)  
To grant an operating system privilege to a specific user, perform the following steps:

#### Related Topics

- [About Creating and Starting an Oracle Database Service](#)  
Learn how to create and start an Oracle Database service.
- [About Ways to Manage Oracle Database Services](#)  
Learn how to manage the services that Oracle Database installs on your computer.

## About Running Windows Services in Oracle Home

Depending on the type of database installation and user account used as the Oracle Home User, Windows services run under low-privileged, non-administrative accounts such as a LocalService, or an authenticated Windows User Account, or as a high-privileged Local System Account (LSA) in Oracle home.

**Table C-1 Running Windows Services**

Type of Installation	Oracle Home User	Windows Service User for the Services
Oracle Database Server	Windows User Account	Windows User Account
Oracle Database Server	Built-in Account	Local System Account
Oracle Database Client	Windows User Account	Windows User Account
Oracle Database Client	Built-in Account	LocalService
Oracle Grid Infrastructure (with the Grid Infrastructure Management Repository)	Windows User Account	Grid Listeners using LocalService Database services using Windows User Account <sup>1</sup> Clusterware services using Local System Account
Oracle Grid Infrastructure (without the Grid Infrastructure Management Repository)	Built-in Account	Grid Listeners using LocalService Clusterware services using Local System Account

<sup>1</sup> Clusterware requires administrative privileges so it always uses Local System Account to run Windows services.

## Additional Privileges Required by Oracle Database Services

Certain functions performed by the Oracle Database service require additional privileges.

Oracle Universal Installer and other Oracle tools automatically grant the following privileges to the Windows services SID of the respective services during the creation of these services:

- **SeIncreaseBasePriorityPrivilege:** A process requires this privilege to change the priority of its threads. This privilege is granted to Windows service SIDs of Oracle Automatic Storage Management (Oracle ASM) or Oracle Database services.
- **SeBackupPrivilege:** This privilege is required to perform backup operations. It is granted to the Windows service SIDs of Oracle VSS Writer service.
- **SeBatchLogonRight:** This privilege is required for an account to log on using the batch logon type. It is granted to the Windows service SIDs of Oracle Scheduler service.

To enable Oracle Database to use Large Pages or working set features, the following additional operating system privileges must be manually granted by the operating

system administrator to either the Oracle Home User or to the Windows service `SIDS` of the specified Oracle Database service during the creation of these services.

Oracle recommends granting privileges to the Windows service `SID` of Oracle Database service instead of the Oracle Home User. The Windows service `SID` of the database service follows this syntax, `NT AUTHORITY\OracleServiceSID`.

- `SeLockMemoryPrivilege`: This privilege is required to lock pages in memory. Oracle Database requires this privilege to use Large Pages.
- `SeIncreaseQuotaPrivilege`: This privilege is required to change the memory quota for a process. This is needed while setting the `max` and `min` working set sizes for the database.

#### Related Topics

- [Overview of Large Page Support](#)  
Large page support is a feature of Oracle Database.

## Granting Additional Operating System Privileges Manually

To grant an operating system privilege to a specific user, perform the following steps:

1. From the **Start** menu, select **Control Panel**.
2. Double-click **Administrative Tools**.
3. Double-click **Local Security Policy**.
4. In the left pane of the Local Security Policy window, expand **Local Policies** and select **User Rights Assignment**.
5. In the right pane of the Local Security Policy window, double-click the relevant user privilege. For example, select **Adjust memory quotas for a process** to change the memory quota for a process or select **Lock pages in memory** to use Large Pages.
6. Click **Add User or Group**.
7. Enter the Oracle Home User name in **Enter the object names to select** field and click **Check Names**.
8. Click **OK** to close the Select Users, Computers, Service Accounts, or Groups dialog box.
9. Click **OK** to close the Properties window for the privilege.

# D

## Error Messages on Windows

Learn about the various error messages, causes, and corrective actions that are specific to the operation of Oracle Database for Windows. You can also learn about the database connection issues.

### Note:

The `ora.hlp` file, which was shipped in the previous releases, is no longer available.

- [ORA-09275: CONNECT INTERNAL No Longer Supported](#)
- [ORA-15252 to ORA-15266: User Replacement Failure on Windows](#)  
The following error messages are displayed when the SQL statement or the ASMCMD command for user replacement fails on Windows:
- [ORA-15301 to ORA-15302: Failure to Modify Ownership, Group, and Permission of Opened Files](#)  
Review the error messages that are displayed when trying to modify the ownership, group membership, or permission of a file fails while changing a file's owner.
- [OSD-04000 to OSD-04599: Windows-Specific Oracle Database Messages](#)  
Error messages in this section are Oracle Database operating system-dependent (OSD) messages displayed in response to an error condition in Windows.
- [DIM-00000 to DIM-00228: ORADIM Command Syntax Errors](#)  
ORADIM is a command-line tool for starting and stopping database instances that is only available on Oracle Database for Windows.
- [Database Connection Issues](#)  
Review database connection issues.

### See Also:

*Oracle Database Error Messages* for information about error messages.

## ORA-09275: CONNECT INTERNAL No Longer Supported

### **ORA-09275**

Connect internal is not a valid DBA connection

Cause: `CONNECT INTERNAL` is no longer supported for DBA connections.



Action: If NTS is enabled, you can connect to the database as `CONNECT / AS SYSDBA` or `CONNECT / AS SYSOPER`. If NTS is not enabled, you can connect as `CONNECT SYS AS SYSDBA`. You can also connect as an existing user with the appropriate password.

## ORA-15252 to ORA-15266: User Replacement Failure on Windows

The following error messages are displayed when the SQL statement or the ASMCMD command for user replacement fails on Windows:

### **ORA-15252**

user name '%s' does not exist in OS user dictionary

Cause: The specified user name was not a valid operating system user.

Action: Specify a valid operating system user.

### **ORA-15260**

permission denied on ASM disk group

Cause: An attempt was made to perform an operation on an ASM disk group without the appropriate privileges.

Action: Ask the ASM administrator to perform the operation or grant the required privileges.

### **ORA-15261**

user '%s' already exists in disk group '%s'

Cause: The specified UID already existed in the disk group.

Action: Specify a different UID.

### **ORA-15262**

user '%s' does not exist in disk group '%s'

Cause: The specified UID did not exist in the disk group.

Action: Specify a user with a valid UID.

### **ORA-15263**

user name '%s' exceeds limit of %s characters

Cause: The length of the specified user name exceeded the maximum limit.

Action: Reduce the length of the user name to a value within the limit.

### **ORA-15264**

Operating system function returned error %s

Cause: An operating system error occurred.

Action: Correct the operating system error and retry the operation.

**ORA-15265**

user identification number not in range of [%s,%s]

Cause: The operating system user identification number exceeded the maximum value.

Action: Change the user identification number to a value within the accepted range.

**ORA-15266**

user identification number '%s' is not allowed

Cause: This user identification number is used internally by ASM.

Action: Please choose a different user identification number.

## ORA-15301 to ORA-15302: Failure to Modify Ownership, Group, and Permission of Opened Files

Review the error messages that are displayed when trying to modify the ownership, group membership, or permission of a file fails while changing a file's owner.

The following error messages are displayed:

**ORA-15301**

cannot change %s of the open file '%s'

Cause: A SET OWNERSHIP or SET PERMISSION command was attempted on an open file.

Action: Close the file and retry the SQL command.

**ORA-15302**

active use of files owned by user '%s' precludes its drop

Cause: A DROP USER command specified a user owning files that were in use.

Action: Close all files owned by this user.

## OSD-04000 to OSD-04599: Windows-Specific Oracle Database Messages

Error messages in this section are Oracle Database operating system-dependent (OSD) messages displayed in response to an error condition in Windows.

Each message in this section triggers an Oracle Database error message.

Error messages appear first in summary tables consisting of error numbers and the corresponding error message. Following the tables is a more detailed discussion of errors, including causes and corrective actions.

---

<b>File I/O Errors:</b>	<b>OSD-04000 to OSD-04099</b>
4000	Logical block size mismatch

---

---

<b>File I/O Errors:</b>	<b>OSD-04000 to OSD-04099</b>
4001	Invalid logical block size
4002	Unable to open file
4003	Unable to read file header block
4004	Invalid file header
4005	SetFilePointer() failure, unable to read from file
4006	ReadFile() failure, unable to read from file
4007	Truncated read
4008	WriteFile() failure, unable to write to file
4009	Truncated write
4010	<create> option specified, file already exists
4011	GetFileInformationByHandle() failure, unable to obtain file info
4012	File size mismatch
4013	Unable to read line from file
4014	Unable to close file
4015	An asynchronous I/O request returned an error
4016	Error queuing an asynchronous I/O request
4017	Unable to open the specified RAW device
4018	Unable to access the specified directory or device
4019	Unable to set file pointer
4020	Unable to set eof file marker
4021	Unable to read file
4022	Unable to write file
4023	SleepEx() failure, unable to Sleep
4024	Unable to delete file
4025	Invalid question asked
4026	Invalid parameter passed

---

<b>Memory Errors:</b>	<b>OSD-04100 to OSD-04199</b>
4100	malloc() failure, unable to allocate memory
4101	Invalid SGA: SGA not initialized
4102	Unable to open/create file for shared memory object
4103	Unable to attach to SGA: SGA does not exist
4104	Unable to map shared memory (SGA) into the address space
4105	Shared memory (SGA) mapped to wrong address
4106	Unable to allocate memory with VirtualAlloc
4107	Unable to deallocate memory with VirtualFree
4108	Unable to protect memory with VirtualProtect

---

<b>Process Errors:</b>	<b>OSD-04200 to OSD-04299</b>
4200	Unable to begin another thread
4201	No pid structure supplied to spdcr()
4202	DosSetPriority() failure, unable to set process priority
4203	DosKillProcess() failure, unable to kill process
4204	Invalid pid
4205	CreateProcess() failure, unable to spawn process
4207	Invalid priority specified in CONFIG parameter ORACLE_PRIORITY
4208	OpenProcess() failure, unable to open process handle
4209	Incorrect or unknown background image name given to spdcr()
4210	Timeout waiting for thread semaphore
4211	Thread information not found
4212	Maximum number of ORACLE threads reached
4213	ORACLE thread unable to DuplicateHandle()
4214	ORACLE thread unable to CreateEvent()
4215	Bad function code supplied to sstthreadop
4216	Unable to find file handle for that thread
4217	Unable to retrieve system username for current user
4218	Cannot post thread
4219	Bad thread list semaphore
4221	Target thread is currently busy
4222	Unable to get the threads context
4223	Unable to set the threads context
4224	Unable to suspend the target thread
4225	Unable to resume the target thread
<b>Loader Errors:</b>	<b>OSD-04300 to OSD-04399</b>
4300	Unable to read complete record from data file
4301	Record size too large
4302	Invalid record type, load options, or both
<b>Semaphore Errors:</b>	<b>OSD-04400 to OSD-04499</b>
4400	Unable to acquire internal semaphore for process
4401	WaitForSingleObject() failure, unable to obtain semaphore
<b>Miscellaneous Errors:</b>	<b>OSD-04500 to OSD-04599</b>
4500	Illegal option specified
4501	Internal buffer overflow
4502	Translations nested too deep
4503	Text contains no translatable elements

---

**Miscellaneous Errors: OSD-04500 to OSD-04599**

---

4505	stdin not responding
4506	Unable to spawn process through system()
4510	Operating system roles are not supported
4511	Unable to get date and time from the operating system
4512	Unable to translate the 'USERNAME' config.ora variable on server
4513	'remote_os_authent' init.ora variable not set to true
4514	The Windows Group name is too long for internal buffer
4515	This command is not implemented at this time

---

- [File I/O Errors: OSD-04000 to OSD-04099](#)  
Review File I/O errors.
- [Memory Errors: OSD-04100 to OSD-04199](#)  
Review memory errors.
- [Process Errors: OSD-04200 to OSD-04299](#)  
Review process errors.
- [Loader Errors: OSD-04300 to OSD-04399](#)  
Review loader errors.
- [Semaphore Errors: OSD-04400 to OSD-04499](#)  
Review Semaphore errors.
- [Miscellaneous Errors: OSD-04500 to OSD-04599](#)  
Review miscellaneous errors.

## File I/O Errors: OSD-04000 to OSD-04099

Review File I/O errors.

### **OSD-04000**

Logical block size mismatch

Cause: Database block size specified in the initialization parameter file does not match block size of actual database files.

Action: Use matching logical block sizes.

### **OSD-04001**

Invalid logical block size

Cause: Logical block size is not a multiple of 512 bytes, or it is too large.

Action: Change the value of `DB_BLOCK_SIZE` in the initialization parameter file.

### **OSD-04002**

Unable to open file

Cause: Specified path or file name is invalid, or destination device is full. This error can also be caused by insufficient Windows file handles.

Action: Make sure path and file exist, and device has free space. If this fails, then increase number of Windows file handles.

**OSD-04003**

Unable to read file header block

Cause: Media has been damaged.

Action: Recover file, if necessary, and verify that Windows is functioning correctly.

**OSD-04004**

Invalid file header

Cause: File is damaged.

Action: Recover file.

**OSD-04005**

SetFilePointer() failure, unable to read from file

Cause: Unexpected return from Windows system service SetFilePointer().

Action: Check operating system error code and operating system documentation.

**OSD-04006**

ReadFile() failure, unable to read from file

Cause: Unexpected return from Windows system service ReadFile().

Action: Check operating system error code and operating system documentation.

**OSD-04007**

Truncated read

Cause: System encountered an unexpected end-of-file, which is due to damaged media.

Action: Verify that file is not damaged.

**OSD-04008**

WriteFile() failure, unable to write to file

Cause: Unexpected return from Windows system service WriteFile().

Action: Check operating system error code and operating system documentation.

**OSD-04009**

Truncated write

Cause: Destination device is full, or media is damaged.

Action: Verify that device has free space and that file is not damaged.

**OSD-04010**

<create> option specified, file already exists

Cause: File you attempted to create already exists.

Action: Delete existing file or use `REUSE` option in SQL statement.

**OSD-04011**

GetFileInformationByHandle() failure, unable to obtain file info

Cause: Unexpected return from Windows system service GetFileInformationByHandle().

Action: Check operating system error code and operating system documentation.

**OSD-04012**

File size mismatch

Cause: File to be re-used is either too large or too small.

Action: Specify correct file size or delete existing file.

**OSD-04013**

Unable to read line from file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check operating system error code (if available) and operating system documentation. If no operating system error code is presented, then verify that media is not damaged.

**OSD-04014**

Unable to close file

Cause: Media has been damaged.

Action: Recover file, if necessary, and verify that Windows is functioning correctly.

**OSD-04015**

Asynchronous I/O request returned an error

Cause: Unexpected return from Windows system service.

Action: Check operating system error code and operating system documentation.

**OSD-04016**

Error queuing an asynchronous I/O request

Cause: Unexpected return from Windows system service.

Action: Check operating system error code and operating system documentation.

**OSD-04017**

Unable to open the specified RAW device

Cause: An invalid path or file name was specified, or device is full.

Action: Make sure file exists and device is not full; verify that operating system is functioning correctly.

**OSD-04018**

Unable to access the specified directory or device

Cause: An invalid path name was specified.

Action: Make sure directory or device exists and is accessible.

**OSD-04019**

Unable to set file pointer

Cause: This error is caused by an operating system error or by damaged media.

Action: Check operating system error code (if available) and operating system documentation. If no operating system error code is presented, then verify that media is not damaged.

**OSD-04020**

Unable to set eof file marker

Cause: This error is caused by an operating system error or by damaged media.

Action: Check operating system error code (if available) and operating system documentation. If no operating system error code is presented, then verify that media is not damaged.

**OSD-04021**

Unable to read file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check operating system error code (if available) and operating system documentation. If no operating system error code is presented, then verify that media is not damaged.

**OSD-04022**

Unable to write file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check operating system error code (if available) and operating system documentation. If no operating system error code is presented, then verify that media is not damaged.

**OSD-04023**

SleepEx() failure, unable to Sleep

Cause: Unexpected return from Windows system service.

Action: Check operating system error code and operating system documentation.

**OSD-04024**

Unable to delete file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check operating system error code (if available) and operating system documentation. If no operating system error code is presented, then verify that media is not damaged.



**OSD-04025**

Invalid question asked

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04026**

Invalid parameter passed

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

## Memory Errors: OSD-04100 to OSD-04199

Review memory errors.

**OSD-04100**

Malloc() failure, unable to allocate memory

Cause: Program is out of memory.

Action: Shut down all unnecessary processes or install more memory in the computer.

**OSD-04101**

Invalid SGA: SGA not initialized

Cause: System Global Area (SGA) has been allocated but not initialized.

Action: Wait until STARTUP has completed before attempting to connect.

**OSD-04102**

Unable to open/create file for shared memory object

Cause: Unexpected return from Windows system service CreateFile().

Action: Check operating system error code and operating system documentation.

**OSD-04103**

Unable to attach to SGA: SGA does not exist

Cause: SGA does not exist.

Action: Start up an Oracle Database instance.

**OSD-04104**

Unable to map shared memory (SGA) into the address space

Cause: Unexpected return from Windows system service MapViewOfFileEx().

Action: Check operating system error code and operating system documentation.

**OSD-04105**

Shared memory (SGA) mapped to wrong address

Cause: Unexpected return from Windows system service MapViewOfFileEx().

Action: Check operating system error code and operating system documentation.

**OSD-04106**

Unable to allocate memory with VirtualAlloc

Cause: Program is out of memory.

Action: Shut down all unnecessary processes or install more memory in the computer.

**OSD-04107**

Unable to deallocate memory with VirtualFree

Cause: Unexpected return from Windows system service VirtualFree().

Action: Check operating system error code and operating system documentation.

**OSD-04108**

Unable to protect memory with VirtualProtect

Cause: Unexpected return from Windows system service VirtualProtect().

Action: Check operating system error code and operating system documentation.

## Process Errors: OSD-04200 to OSD-04299

Review process errors.

**OSD-04200**

Unable to begin another thread

Cause: Program has run out of system resources.

Action: Shut down all unnecessary processes; install more memory in the computer.

**OSD-04201**

No pid structure supplied to spdcr()

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04202**

DosSetPriority() failure, unable to set process priority

Cause: Unexpected return from Windows system service DosSetPriority().

Action: Check operating system error code and operating system documentation.

**OSD-04203**

DosKillProcess() failure, unable to kill process

Cause: Unexpected return from Windows system service DosKillProcess().

Action: Check operating system error code and operating system documentation.

**OSD-04204**

Invalid pid

Cause: Process ID not recognized by system; process previously terminated.

Action: Verify that process ID is correct and that process is active.

**OSD-04205**

CreateProcess() failure, unable to spawn process

Cause: Unexpected return from Windows system service CreateProcess().

Action: Check operating system error code and operating system documentation.

**OSD-04207**

Invalid priority specified in CONFIG parameter *ORACLE\_PRIORITY*

Cause: Priority specified is invalid or out of range.

Action: Specify a valid setting for *ORACLE\_PRIORITY*.

**OSD-04208**

OpenProcess() failure, unable to open process handle

Cause: Unexpected return from Windows system service OpenProcess().

Action: Check operating system error code and operating system documentation.

**OSD-04209**

Incorrect or unknown background image name given to spdcrc()

Cause: Unexpected background name given to spdcrc().

Action: Contact Oracle Support Services.

**OSD-04210**

Timeout waiting for thread semaphore

Cause: An Oracle Database thread died holding the semaphore.

Action: Restart Oracle Database instance.

**OSD-04211**

Thread information not found

Cause: An Oracle Database thread died without deleting its information.

Action: Restart Oracle Database instance.

**OSD-04212**

Maximum number of Oracle threads reached

Cause: Maximum number of Oracle Database threads for the instance is reached.

Action: Wait until some connections exit before trying again.

**OSD-04213**

Oracle thread unable to DuplicateHandle()

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04214**

Oracle thread unable to CreateEvent()

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04215**

Bad function code supplied to sstheadop

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04216**

Unable to find file handle for that thread

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04217**

Unable to retrieve system username for current user

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04218**

Cannot post thread

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04219**

Bad thread list semaphore

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

**OSD-04221**

Target thread is currently busy

Cause: Target thread is processing an Oracle Database utility command.

Action: Wait and resend command.

**OSD-04222**

Unable to get the threads context

Cause: Check operating system error code.

Action: Remedy operating system error.

**OSD-04223**

Unable to set the threads context

Cause: Check operating system error code.

Action: Remedy operating system error.

**OSD-04224**

Unable to suspend the target thread

Cause: Check operating system error code.

Action: Remedy operating system error.

**OSD-04225**

Unable to resume the target thread

Cause: Check operating system error code.

Action: Remedy operating system error.

## Loader Errors: OSD-04300 to OSD-04399

Review loader errors.

**OSD-04300**

Unable to read complete record from the datafile

Cause: Datafile ended in the middle of a record. This error occurs when loading files with a fixed record length.

Action: Verify that the datafile is of correct length and contains complete records.

**OSD-04301**

Record size too large

Cause: Specified record size is too large to load.

Action: Reduce record size and reload data.

**OSD-04302**

Invalid record type, load options, or both

Cause: Control file's Windows file processing options string contains an invalid option or keyword.

Action: Set Windows file processing options string to an acceptable value.

## Semaphore Errors: OSD-04400 to OSD-04499

Review Semaphore errors.

### **OSD-04400**

Unable to acquire internal semaphore for process

Cause: Oracle Database has exceeded the maximum number of connections.

Action: Delete any unused connections and try again.

### **OSD-04401**

WaitForSingleObject() failure, unable to obtain semaphore

Cause: Unexpected return from Windows system service WaitForSingleObject().

Action: Check operating system error code and operating system documentation.

## Miscellaneous Errors: OSD-04500 to OSD-04599

Review miscellaneous errors.

### **OSD-04500**

Illegal option specified

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

### **OSD-04501**

Internal buffer overflow

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

### **OSD-04502**

Translations nested too deep

Cause: Program encountered too many intermediate translations while attempting to translate a configuration variable.

Action: Simplify values of configuration parameters to include fewer intermediate translations.

### **OSD-04503**

Text contains no translatable elements

Cause: Program cannot recognize variables in text to be translated.

Action: Check and, if necessary, correct text to be translated.

### **OSD-04505**

stdin not responding

Cause: System cannot receive input from standard input stream.

Action: Verify that process has access to an input device.

**OSD-04506**

Unable to spawn process through system()

Cause: System is out of memory or executable is invalid.

Action: Shut down unnecessary processes; install more memory in the computer.  
Verify name of executable.

**OSD-04510**

Operating system roles are not supported

Cause: An attempt was made to use an operating system role.

Action: Only use roles that were created 'IDENTIFIED BY *PASSWORD*' as opposed to 'IDENTIFIED EXTERNALLY'.

**OSD-04511**

Unable to get date and time from the operating system

Cause: Unexpected return from GetLocalTime() call.

Action: Verify that system time is correct on the computer.

**OSD-04512**

Unable to translate the '*USERNAME*' config.ora variable on server

Cause: '*USERNAME*' configuration parameter variable on host is not properly set.

Action: Verify '*USERNAME*' variable is set.

**OSD-04513**

'remote\_os\_authent' init.ora variable not set to TRUE

Cause: For remote operating system logon to function, '*REMOTE\_OS\_AUTHENT*' parameter must be set to TRUE.

Action: Shut down and start up instance with '*REMOTE\_OS\_AUTHENT* = TRUE' in initialization parameter file.

**OSD-04514**

The Windows Group name is too long for internal buffer

Cause: Windows Group name is too long.

Action: Use a shorter Windows group name.

## DIM-00000 to DIM-00228: ORADIM Command Syntax Errors

ORADIM is a command-line tool for starting and stopping database instances that is only available on Oracle Database for Windows.

It is not available on any other platform.

Oradim Errors	DIM-0000 to DIM-00228
00000	ORADIM completed with no errors
00001	ORADIM: <command> [options]. Refer to manual.
00002	The specified command was invalid.
00003	An argument is missing for the parameter
00004	SID or service name was not specified.
00005	SID with more than 64 characters specified.
00006	Missing SID
00007	Missing or invalid -STARTMODE parameter. Valid -STARTMODE parameter is AUTO or MANUAL.
00008	A valid service name is OracleService appended with a SID
00009	SID name is mandatory.
00010	SYSTEM\CurrentControlSet\Services\OracleService key does not exist.
00011	The specified service does not exist.
00012	A PFILE is necessary for AUTOSTART option.
00013	Service start mode could not be set in the registry.
00014	Cannot open the Windows Service Control Manager.
00015	Cannot start already-running ORACLE - shut it down first.
00016	Missing or invalid -SHUTTYPE parameter. A valid -SHUTTYPE parameter is SRVC or INST.
00017	Instance shutdown mode must be one of the following: a for abort, i for immediate or n for normal.
00018	Failed to stop Oracle Service.
00019	Create Service Error.
00020	A service for this name exists.
00021	Registry open failed.
00040	Invalid option for the -NEW command.
00041	Invalid option for the -EDIT command.
00042	Invalid option for the -DELETE command.
00043	Invalid option for the -STARTUP command.
00044	Invalid option for the -SHUTDOWN command.
00045	Internal error in ORADIM.
00046	Invalid Pfile.
00050	Instance deleted.
00051	Instance created.
00075	Failed to control service.
00076	Failed to delete service.
00077	Failed to change service configuration.
00078	Failed to start service.
0090	SID name is invalid.
0092	Unable to determine Oracle service user.



Oradim Errors	DIM-0000 to DIM-00228
0093	Invalid option for the -ACL command.
0094	Failed to change ACLs on the object.
00200	Enter one of the following commands:
00201	Create an instance by specifying the following options:
00202	-NEW -SID sid   -ASMSID sid   -MGMTDBSID sid   -IOSSID sid   -APXSID sid
00203	-SRVC svc   -ASMSRVC svc   -MGMTDBSRVC svc   -IOSSRVC svc
00204	-APXSRVC svc [-SYSPWD pass] [-STARTMODE auto manual]
00205	[-SRVCSTART system demand] [-PFILE file   -SPFILE]
00206	[-SHUTMODE normal immediate abort] [-TIMEOUT secs] [-RUNAS osusr/ospass]
00207	Edit an instance by specifying the following options:
00208	-EDIT -SID sid   -ASMSID sid   -MGMTDBSID sid   -IOSSID sid   -APXSID sid
00209	[-SYSPWD pass] [-STARTMODE auto manual] [-SRVCSTART system demand]
00210	[-PFILE file   -SPFILE] [-SHUTMODE normal immediate abort]
00211	[-SHUTTYPE svc inst] [-RUNAS osusr/ospass]
00212	Delete instances by specifying the following options:
00213	-DELETE -SID sid   -ASMSID sid   -MGMTDBSID sid   -IOSSID sid
00214	-APXSID sid  -SRVC svc   -ASMSRVC svc   -MGMTDBSRVC svc
00215	-IOSSRVC svc   -APXSRVC svc
00216	Startup services and instance by specifying the following options:
00217	-STARTUP -SID sid   -ASMSID sid   -MGMTDBSID sid   -IOSSID sid
00218	-APXSID sid [-SYSPWD pass] [-STARTTYPE svc inst svc,inst]
00219	[-PFILE filename   -SPFILE]
00220	Shutdown service and instance by specifying the following options:
00221	-SHUTDOWN -SID sid   -ASMSID sid   -MGMTDBSID sid   -IOSSID sid
00222	-APXSID sid [-SYSPWD pass] [-SHUTTYPE svc inst svc,inst]
00223	[-SHUTMODE normal immediate abort]
00224	Manipulate ACLs by specifying the following options:
00225	-ACL -setperm -addperm -removeperm dbfiles diag registry
00226	-USER username -OBJTYPE file dir registry -OBJPATH object-path
00227	-RECURSE true false [-HOST hostname]
00228	Query for help by specifying the following parameters: -?   -h   -help

**DIM-00000**

ORADIM completed with no errors.

Cause: The specified operation completed successfully.

Action: None.

**DIM-00001**

ORADIM: <command> [options]. Refer to manual.

Cause: The specified options were invalid or no arguments were supplied.

Action: Usage: ORADIM <command> [options]

**DIM-00002**

The specified command was invalid.

Cause: Valid commands are: -DELETE, -EDIT, -NEW, -STARTUP, and -SHUTDOWN.

Action: Use valid command.

**DIM-00003**

An argument is missing for the parameter.

Cause: Missing or invalid argument.

Action: Use a valid argument and run the program again.

**DIM-00004**

SID or service name was not specified.

Cause: Either a SID or service name is mandatory.

Action: Enter a valid *SID* of 64 characters and retry.

**DIM-00005**

SID with more than 64 characters specified.

Cause: SID with more than 64 characters specified.

Action: Change SID to 64 unique characters and ensure that there is no other service with this name.

**DIM-00006**

Missing SID.

Cause: SID was not specified in the arguments.

Action: Specify a SID.

**DIM-00007**

Missing or invalid -STARTMODE parameter. Valid -STARTMODE parameter is AUTO or MANUAL.

Cause: An argument for STARTMODE is missing.

Action: Enter a valid start mode and retry.

**DIM-00008**

A valid service name is OracleService appended with a SID

Cause: The Oracle service name specified is invalid.

Action: Correct the name of service and retry.

**DIM-00009**

SID name is mandatory.

Cause: SID was not specified.

Action: Enter the SID and retry.

**DIM-00010**

SYSTEM\CurrentControlSet\Services\OracleService key does not exist.

Cause: Specified registry key was not found.

Action: Try reinstalling. If the problem persists, contact Oracle Support Services.

**DIM-00011**

The specified service does not exist.

Cause: An attempt to edit a service failed.

Action: Make sure the service exists or user has enough privileges.

**DIM-00012**

A PFILE is necessary for AUTOSTART option.

Cause: A parameter file {PFILE} was not specified.

Action: Specify a parameter file.

**DIM-00013**

Service start mode could not be set in the registry.

Cause: The start mode entry in the registry for the service could not be set.

Action: Check if the user has privileges to modify registry.

**DIM-00014**

Cannot open the Windows Service Control Manager.

Cause: The Service Control Manager could not be opened.

Action: Check for user privileges.

**DIM-00015**

Cannot start already-running ORACLE - shut it down first.

Cause: The instance is already started.

Action: Stop the database before restarting.

**DIM-00016**

Missing or invalid -SHUTTYPE parameter. A valid -SHUTTYPE parameter is SRVC or INST.

Cause: An option for SHUTTYPE was missing or invalid.

Action: Enter parameter to shut down the instance or the service and retry.

**DIM-00017**

Instance shutdown mode must be one of the following: a for abort, i for immediate or n for normal.

Cause: Invalid option to shut down the instance was specified.

Action: Enter the correct mode and retry.

**DIM-00018**

Failed to stop Oracle Service.

Cause: An attempt to stop the service failed.

Action: Retry, check for user privileges.

**DIM-00019**

Create service error.

Cause: Service could not be created.

Action: Check for user privileges and retry.

**DIM-00020**

A service for this name exists.

Cause: An attempt was made to create a service name when it already existed.

Action: Retry with a different service name or SID.

**DIM-00021**

Registry open failed

Cause: An attempt to open the registry failed.

Action: Check for user privileges and retry the operation.

**DIM-00040**

Invalid option for the -NEW command.

Cause: One or more arguments for creating new service is invalid.

Action: Specify required option and retry.

**DIM-00041**

Invalid option for the -EDIT command.

Cause: One or more arguments for editing existing service is invalid.

Action: Specify required option and retry.

**DIM-00042**

Invalid option for the -DELETE command.

Cause: One or more arguments for deleting service is invalid.

Action: Specify required option and retry.

**DIM-00043**

Invalid option for the -STARTUP command.

Cause: One or more arguments for starting the instance is invalid.

Action: Specify required option and retry.

**DIM-00044**

Invalid option for the -SHUTDOWN command.

Cause: One or more arguments for shutting down the instance is invalid.

Action: Specify required option and retry.

**DIM-00045**

Internal error in ORADIM

Cause: Unknown.

Action: Contact Oracle Support Services.

**DIM-00046**

Invalid Pfile.

Cause: The parameter file name is invalid.

Action: Check that the path name is correct.

**DIM-00050**

Instance deleted

Cause: The request for deleting instance was successful.

Action: None.

**DIM-00051**

Instance created.

Cause: The request for creating new instance was successful.

Action: None.

**DIM-00075**

Failed to control service.

Cause: An attempt to control the service failed.

Action: Check additional error, ensure that user has enough privileges.

**DIM-00076**

Failed to delete service.

Cause: The request for service deletion failed.

Action: Check additional error, ensure that user has enough privileges.

**DIM-00077**

Failed to change service configuration.

Cause: An attempt to change configuration failed.

Action: Check additional error, ensure that user has enough privileges.

**DIM-00078**

Failed to start service.

Cause: The request to start service failed.

Action: Check additional error, ensure that user has enough privileges.

**DIM-00090**

SID name is invalid.

Cause: An invalid SID name was provided.

Action: Enter a valid SID name with no more than 64 characters (alphanumeric) and retry.

**DIM-00092**

Unable to determine Oracle service user.

Cause: This is an internal Error. The Oracle service user could not be determined for this Oracle home.

Action: Contact Oracle Support Services.

**DIM-00093**

Invalid option for the -ACL command.

Cause: One or more options for setting ACLs was invalid.

Action: Specify the required option and retry.

**DIM-00094**

Failed to change ACLs on the object.

Cause: An attempt to change the ACLs on the object failed.

Action: Check additional errors. Make sure that the user has enough privileges.

**DIM-00200**

Enter one of the following commands:

Cause: Unknown.

Action: None.

**DIM-00201**

Create an instance by specifying the following options:

Cause: Unknown.

Action: None.

**DIM-00202**

-NEW -SID sid | -ASMSID sid | -MGMTDBSID sid | -IOSSID sid | -APXSID sid|

Cause: Unknown.

Action: None.

**DIM-00203**

-SRVC srvc | -ASMSRVC srvc | -MGMTDBSRVC srvc | -IOSSRVC srvc |

Cause: Unknown.

Action: None.

**DIM-00204**

-APXSRVC srvc [-SYSPWD pass] [-STARTMODE auto|manual]

Cause: Unknown.

Action: None.

**DIM-00205**

[-SRVCSTART system|demand] [-PFILE file | -SPFILE]

Cause: Unknown.

Action: None.

**DIM-00206**

[-SHUTMODE normal|immediate|abort] [-TIMEOUT secs] [-RUNAS osusr/ospass]

Cause: Unknown.

Action: None.

**DIM-00207**

Edit an instance by specifying the following options:

Cause: Unknown.

Action: None.

**DIM-00208**

-EDIT -SID sid | -ASMSID sid | -MGMTDBSID sid | -IOSSID sid | -APXSID sid

Cause: Unknown.

Action: None.

**DIM-00209**

[-SYSPWD pass] [-STARTMODE auto|manual] [-SRVCSTART system|demand]

Cause: Unknown.

Action: None.

**DIM-00210**

[-PFILE file | -SPFILE] [-SHUTMODE normal|immediate|abort]

Cause: Unknown.

Action: None.

**DIM-00211**

[-SHUTTYPE svc|inst] [-RUNAS osusr/ospass]

Cause: Unknown.

Action: None.

**DIM-00212**

Delete instances by specifying the following options:

Cause: Unknown.

Action: None.

**DIM-00213**

-DELETE -SID sid | -ASMSID sid | -MGMTDBSID sid | -IOSSID sid |

Cause: Unknown.

Action: None.

**DIM-00214**

-APXSID sid| -SRVC svc | -ASMSRVC svc | -MGMTDBSRVC svc |

Cause: Unknown.

Action: None.

**DIM-00215**

-IOSSRVC svc | -APXSRVC svc

Cause: Unknown.

Action: None.

**DIM-00216**

Startup services and instance by specifying the following options:

Cause: Unknown.

Action: None.

**DIM-00217**

-STARTUP -SID sid | -ASMSID sid | -MGMTDBSID sid | -IOSSID sid

Cause: Unknown.

Action: None.



**DIM-00218**

| -APXSID sid [-SYSPWD pass] [-STARTTYPE svc|inst|svc,inst]

Cause: Unknown.

Action: None.

**DIM-00219**

[-PFILE filename | -SPFILE]

Cause: Unknown.

Action: None.

**DIM-00220**

Shutdown service and instance by specifying the following options:

Cause: Unknown.

Action: None.

**DIM-00221**

-SHUTDOWN -SID sid | -ASMSID sid | -MGMTDBSID sid | -LOSSID sid |

Cause: Unknown.

Action: None.

**DIM-00222**

-APXSID sid [-SYSPWD pass] [-SHUTTYPE svc|inst|svc,inst]

Cause: Unknown.

Action: None.

**DIM-00223**

[-SHUTMODE normal|immediate|abort]

Cause: Unknown.

Action: None.

**DIM-00224**

Manipulate ACLs by specifying the following options:

Cause: Unknown.

Action: None.

**DIM-00225**

-ACL -setperm|-addperm|-removeperm dbfiles|diag|registry

Cause: Unknown.

Action: None.

**DIM-00226**

-USER username -OBJTYPE file|dir|registry -OBJPATH object-path

Cause: Unknown.

Action: None.

**DIM-00227**

-RECURSE true|false [-HOST hostname]

Cause: Unknown.

Action: None.

**DIM-00228**

Query for help by specifying the following parameters: -? | -h | -help

Cause: None.

Action: None.

## Database Connection Issues

Review database connection issues.

The following are the common Oracle Database connection error codes, their causes, and suggested remedies.

**TNS-12203**

TNS: unable to connect to destination

Cause: OracleServiceSID service, OracleHOMENAMEListener service, or both are not running.

Action: Ensure that both services are started.

**ORA-12560**

TNS: lost contact

Cause: OracleServiceSID service, OracleHOMENAMEListener service, or both are not running. You receive this error if you attempt to use any Oracle Database utilities, such as SQL\*Plus. This error is analogous to the following Oracle7 error: ORA-09352: Windows 32-bit Two-Task driver unable to spawn new ORACLE task.

Action: Ensure that both services are started.

**ORA-28575**

unable to open RPC connection to external procedure agent

Cause: tnsnames.ora and listener.ora files have not been correctly configured to use external procedures.

Action: Reconfigure services.

**ORA-06512**

at "APPLICATIONS.OSEXEC", line 0

Cause: `tnsnames.ora` and `listener.ora` files have not been correctly configured to use external procedures.

Action: Reconfigure services.

**ORA-06512**

at "APPLICATIONS.TEST", line 4

Cause: `tnsnames.ora` and `listener.ora` files have not been correctly configured to use external procedures.

Action: Reconfigure services.

**ORA-06512**

at line 2

Cause: `tnsnames.ora` and `listener.ora` files have not been correctly configured to use external procedures.

Action: Reconfigure services.

**ORA-01031 and LCC-00161**

Both codes appear at startup

Cause: Parameter file (`init.ora`) or Windows services are damaged. These errors usually appear when the parameter file cannot be read by Oracle Database at database startup.

Action: Delete and re-create the SID and services. Make sure you are logged on as the user Administrator, or a user within the Windows Administrator's Group with full administrative rights. At the command prompt, enter: `oradim -delete -sid sid` where `sid` is the name of your database (for example, `orcl`). Re-create the SID and services by entering: `oradim -new -sid sid -startmode auto -pfile full_path_to_init.ora`

# E

## Oracle Database Differences on Windows and UNIX

Learn about the differences between Oracle Database on Windows and UNIX. For Oracle Database developers and database administrators moving from a UNIX platform to Windows, this information can be helpful in understanding Windows features that are relevant to Oracle Database.

- [Automatic Startup and Shutdown](#)  
On UNIX, several files and scripts in different directories are used to start an instance automatically.
- [Background Processing and Batch Jobs](#)  
UNIX provides sophisticated control mechanisms for background processing and batch jobs.
- [Diagnostic and Tuning Utilities](#)  
On UNIX, utilities such as `sar` and `vmstat` are used to monitor Oracle Database background and shadow processes.
- [Direct Writes to Disk](#)  
On both UNIX and Windows platforms, bypassing the file system buffer cache ensures data is written to disk.
- [Dynamic Link Libraries \(DLLs\)](#)  
Shared libraries on UNIX are similar to shared DLLs on Windows.
- [Hot Backups](#)  
A (manual) hot backup is equivalent to backing up a tablespace that is in an offline backup mode.
- [Initialization Parameters: Multiple Database Writers](#)  
On UNIX, you can specify many database writer process with initialization parameter `DB_WRITERS`.
- [Installation Accounts and Groups](#)  
UNIX uses the concept of a DBA group. The `root` account cannot be used to install Oracle Database.
- [Oracle Database Installation](#)  
The following manual setup tasks, all required on UNIX, are *not* required on Windows:
- [Memory Resources](#)  
The resources provided by the UNIX default kernels are often inadequate for a medium or large instance of Oracle Database.
- [Microsoft Transaction Server](#)  
Windows coordinates distributed transactions through the Microsoft Distributed Transaction Coordinator (DTC), one of the components of Microsoft Transaction Server.

- [Multiple Oracle Homes and OFA](#)  
The goal of OFA is to place all Oracle Database software under one `ORACLE_HOME` directory and to spread database files across different physical drives as databases increase in size.
- [Oracle Home User and Oracle User](#)  
On Linux and UNIX systems, you must create and use a software owner user account (`oracle`), and this user must belong to the Oracle Inventory group (`oinstall`) and also must be a member of the appropriate OSDBA, OSOPER, OSBACKUPDBA, OSDGDBA, and OSKMDBA groups.
- [Raw Partitions](#)  
Data files for tablespaces may be stored on a file system or on raw partitions.
- [Windows Services](#)  
Windows services are similar to UNIX daemons.

## Automatic Startup and Shutdown

On UNIX, several files and scripts in different directories are used to start an instance automatically.

Other scripts are run on computer shutdown, allowing applications such as Oracle Database to shut down cleanly.

For automatic startup on Windows, set registry parameter `ORA_SID_AUTOSTART` to `true` using an Oracle Database tool such as ORADIM. Enter the following with parameters at the command prompt:

```
C:\> oradim options
```

To start the listener automatically, set services startup type to automatic.

For automatic shutdown on Windows, set registry parameters `ORA_SHUTDOWN` and `ORA_SID_SHUTDOWN` to stop the relevant `OracleServiceSID` and shut down. Set registry parameter `ORA_SID_SHUTDOWNTYPE` to control shutdown mode (default is `i`, or `immediate`).



### See Also:

- [Administering a Database on Windows](#)
- *Oracle Database 2 Day DBA*

## Background Processing and Batch Jobs

UNIX provides sophisticated control mechanisms for background processing and batch jobs.

For similar functionality on Windows, use the AT command or a GUI version in the Microsoft Resource Kit.

## Diagnostic and Tuning Utilities

On UNIX, utilities such as `sar` and `vmstat` are used to monitor Oracle Database background and shadow processes.

These utilities are not integrated with Oracle Database.

Performance utilities available on Windows include Task Manager, Control Panel, Event Viewer, and Microsoft Management Console.

Oracle Database is integrated with several of these tools. For example:

- Event Viewer displays system alert messages, including Oracle Database startup/shutdown messages and audit trail.
- Task Manager on Windows displays currently running processes and their resource usage, similar to the UNIX `ps -ef` command or HP OpenVMS `SHOW SYSTEM`. But Task Manager is easier to interpret and the columns can be customized.



### See Also:

- [Database Tools on Windows](#)
- [Monitoring a Database on Windows](#)

## Direct Writes to Disk

On both UNIX and Windows platforms, bypassing the file system buffer cache ensures data is written to disk.

On UNIX, Oracle Database uses the `O_SYNC` flag to bypass the file system buffer cache. The flag name depends on the UNIX port.

On Windows, Oracle Database bypasses the file system buffer cache completely.

## Dynamic Link Libraries (DLLs)

Shared libraries on UNIX are similar to shared DLLs on Windows.

Object files and archive libraries are linked to generate the Oracle Database executables. Relinking is necessary after certain operations, such as installation of a patch.

On Windows, Oracle Database DLLs form part of the executable at run time and are therefore smaller. DLLs can be shared between multiple executables. Relinking by the user is not supported, but the executable images can be modified using ORASTACK.

Modifying executable images on Windows reduces the chances of running out of virtual memory when using a large SGA or when supporting thousands of connections. However, Oracle recommends doing this only under the guidance of Oracle Support Services.

## Hot Backups

A (manual) hot backup is equivalent to backing up a tablespace that is in an offline backup mode.

Backup strategy on UNIX is as follows: put the tablespace into backup mode, copy the files to the backup location, and bring the tablespace out of backup mode.

Windows supports the same backup strategy, but you cannot copy files in use with the usual Windows utilities. Use OCOPI to copy open database files to another disk location. Then use a utility to copy the files to tape.

## Initialization Parameters: Multiple Database Writers

On UNIX, you can specify many database writer process with initialization parameter `DB_WRITERS`.

Multiple database writers can help, for example, when a UNIX port does not support asynchronous I/O.

`DB_WRITERS` is supported but typically unnecessary on Windows, which has its own asynchronous I/O capabilities.



### See Also:

[Oracle Database Specifications for Windows](#)

## Installation Accounts and Groups

UNIX uses the concept of a DBA group. The `root` account cannot be used to install Oracle Database.

A separate Oracle Database account must be created manually.

On Windows, Oracle Database must be installed by a Windows username in the Administrators group. The user name is automatically added to the Windows local group `ORA_DBA`, which receives the `SYSDBA` privilege. This allows the user to log in to the database using `CONNECT / AS SYSDBA` and not be prompted for a password.

You can also create an `ORA_OPER` group to grant `SYSOPER` privileges to the other Windows users.

Password files are located in the `ORACLE_HOME\database` directory and are named `pwdSID.ora`, where `SID` identifies the Oracle Database instance.

### Related Topics

- [Administering a Database on Windows](#)

 **See Also:**

*Oracle Database Installation Guide for Microsoft Windows* about "Operating System Groups Created During Oracle Database Installation"

## Oracle Database Installation

The following manual setup tasks, all required on UNIX, are *not* required on Windows:

- Set environment variables
- Create a DBA group for database administrators
- Create a group for users running Oracle Universal Installer
- Create an account dedicated to installing and upgrading Oracle Database components

 **See Also:**

*Oracle Database Installation Guide for Microsoft Windows*

## Memory Resources

The resources provided by the UNIX default kernels are often inadequate for a medium or large instance of Oracle Database.

The maximum size of a shared memory segment (`SHMMAX`) and maximum number of semaphores available (`SEMMNS`) may be too low for Oracle Database recommendations.

On Windows, fewer resources are needed for interprocess communication (IPC), because the Oracle Database relational database management system is thread-based and not process-based. These resources, including shared memory and semaphores, are not adjustable by the user.

## Microsoft Transaction Server

Windows coordinates distributed transactions through the Microsoft Distributed Transaction Coordinator (DTC), one of the components of Microsoft Transaction Server.

With Oracle Services for Microsoft Transaction Server, you can develop and deploy distributed transaction applications using .NET, COM, or COM+ and Oracle Database can be a resource manager in DTC transactions.

Microsoft Transaction Server is a Windows component that does not run on any other operating systems. However, Oracle Databases on any operating system can participate in Microsoft DTC transactions on Windows.



 **See Also:**

*Oracle Services for Microsoft Transaction Server Developer's Guide for Microsoft Windows*

## Multiple Oracle Homes and OFA

The goal of OFA is to place all Oracle Database software under one `ORACLE_HOME` directory and to spread database files across different physical drives as databases increase in size.

OFA is implemented on Windows and UNIX in the same way, and main subdirectory and file names are the same on both operating systems. Windows and UNIX differ, however, in their OFA directory tree top-level names and in the way variables are set.

On UNIX, `ORACLE_BASE` is associated with a user's environment. `ORACLE_HOME` and `ORACLE_SID` must be set in system or user login scripts. Symbolic links are supported. Although everything seems to be in one directory on the same hard drive, files may be on different hard drives if they are symbolically linked or have that directory as a mount point.

On Windows, `ORACLE_BASE` is defined in the registry (for example, in `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0`). `ORACLE_HOME` and `ORACLE_SID` are variables defined in the registry. Symbolic links like those on UNIX are not supported.

 **See Also:**

*Oracle Database Installation Guide for Microsoft Windows* in the section Appendix B, "Optimal Flexible Architecture"

## Oracle Home User and Oracle User

On Linux and UNIX systems, you must create and use a software owner user account (`oracle`), and this user must belong to the Oracle Inventory group (`oinstall`) and also must be a member of the appropriate OSDBA, OSOPER, OSBACKUPDBA, OSDGDBA, and OSKMDBA groups.

On Windows, you use an existing Windows User Account or create a new standard Windows User Account (not an administrator account) as the Oracle Home User. The various Oracle services on Windows run with the privileges of the Oracle Home User. This user is automatically added to various groups as required.

### Related Topics

- [Supporting Oracle Home User on Windows](#)  
Oracle Database supports the use of Oracle Home User, specified at the time of Oracle Database installation. Oracle Home User is used to run the Windows services for the Oracle home.

## Raw Partitions

Data files for tablespaces may be stored on a file system or on raw partitions.

A raw partition is a portion of a physical disk that is accessed at the lowest possible level.

UNIX supports raw partitions (logical drives). There is no limitation on the number of disk drives.

Windows is limited to using drive letters A-Z, but creating raw partitions lets you bypass the disk drive limitation and divide disks into smaller sections.

Use Windows disk management tools to create an extended partition on a physical drive. An extended partition points to raw space on the disk that can be assigned multiple logical partitions for database files.

An extended partition avoids the four-partition limit on Windows by allowing you to define large numbers of logical partitions to accommodate applications using Oracle Database. Logical partitions can then be given symbolic link names to free up drive letters.

On supported Windows operating systems, create primary partitions and logical drives in extended partitions by selecting the **New Simple Volume** option. To create a raw device, select **Do not assign a drive letter or drive path**. To mount the raw device, assign and remove a drive letter. Do not use spanned volumes or striped volumes. These options convert the volume to a dynamic disk. Oracle Automatic Storage Management does not support dynamic disks.

 **Note:**

Oracle RAC is only supported on 64-bit Windows server operating systems.

## Windows Services

Windows services are similar to UNIX daemons.

Oracle Database registers a database instance as a service (`OracleServiceSID`). Services start background processes.

To connect to and use an Oracle Database instance, an Oracle Database service is created during database creation and associated with Oracle Database. Once a service is created with Oracle Database, the service can run even while no user is logged on.

From the **Start** menu, select **Control Panel**, then select **Administrative Tools**, and then select **Services** to access the Services dialog.



**See Also:**

[Administering a Database on Windows](#)

# Index

## A

---

### Access Control Lists (ACL)

- adding and deleting security group members, [14-18](#)
- available security groups, [14-15](#)
- setting NTFS ACLs manually, [5-13](#)
- setting on Net Service directory objects, [14-17](#)

### accessing Active Directory, [14-2](#)

### ACLs

- adding and deleting security group members, [14-18](#)
- available security groups, [14-15](#)
- setting NTFS ACLs manually, [5-13](#)
- setting on Net Service directory objects, [14-17](#)

### Active Directory

- adding and deleting security group members, [14-18](#)
- automatic discovery of directory servers, [14-3](#)
- how Oracle directory objects appear, [14-5](#)
- integration with Oracle objects, [14-3](#)
- managing Access Control Lists, [14-15](#)
- managing security groups, [14-15](#)
- testing connectivity with SQL\*Plus, [14-4](#)
- testing database connectivity, [14-4](#)
- user interface extensions, [14-4](#)

### adding and deleting users

- Net Service Objects, [14-18](#)
- OracleDBCreators, [14-18](#)
- OracleNetAdmins, [14-18](#)

### adding executables, [5-4](#)

- firewall exceptions for Oracle Clusterware and Oracle ASM, [5-5](#)
- firewall exceptions for Oracle Database, [5-4](#)
- firewall exceptions for Oracle Database Examples, [5-4](#)
- firewall exceptions for Oracle Gateways, [5-5](#)
- firewall exceptions for other Oracle Products, [5-6](#)

### administering external roles, [11-1](#)

- administration tools prompting for password
  - Domain User Account, [2-4](#)

### administration tools prompting for password (*continued*)

- Local User Account, [2-4](#)
- password not stored in Oracle Wallet, [2-4](#)

### administrator

- starting ASMCA tool
  - member of ORA\_ASMADMIN, [2-4](#)
- starting DBCA tool
  - member of ORA\_DBA, [2-4](#)
- starting DBUA tool
  - member of ORA\_ASMADMIN, [2-4](#)

### Administrators group

- running configuration tools, [2-5](#)

### advanced replication

- about, [5-18](#)
- configuring
  - adding and modifying initialization parameters, [5-19](#)
  - checking tablespace, [5-19](#)
  - monitoring data dictionary tables, [5-20](#)

### alert logs

- monitoring a database, [7-1](#)
- using, [7-6](#)

### archiving mode

- controlling, [2-9](#), [6-11](#)
- custom database, [2-9](#), [6-11](#)
- starter database, [2-9](#), [6-11](#)

### archiving procedures

- for redo log files, [2-9](#), [6-11](#)

### archiving redo log files, [2-9](#), [6-11](#)

### audit trail

- managing, [7-4](#)
- operating system, [2-12](#)

### AUDIT\_FILE\_DEST, [7-4](#), [15-4](#)

### Authenticated Users, [5-8](#)

- permissions for Oracle Database Client
  - Oracle home, [5-11](#)
- permissions for Oracle Database Oracle home, [5-11](#)
- permissions for Oracle Grid Infrastructure
  - Oracle home, [5-10](#)

### Authenticated Users group, [5-10](#)

### authentication

- automatically enabling during installation, [10-4](#)
- enhancements, [10-3](#)

authentication (*continued*)  
 OSAUTH\_PREFIX\_DOMAIN parameter,  
 11-4  
 overview, 10-1, 14-10  
 using a password file, 6-7  
 using Windows native authentication  
 methods, 14-10  
 using Windows native methods, 10-1  
 auto-starting Oracle Database services, 6-3  
 using Control Panel, 6-3  
 using Oracle Administration Assistant, 6-3

## B

---

backing up database  
 32-bit Oracle Database, 4-23  
 about, 5-7, 9-6  
 in archivelog mode  
 component-based, 5-7, 9-6, 9-9  
 volume-based, 5-7, 9-6, 9-9  
 in noarchivelog mode  
 component-based, 9-10  
 volume-based, 9-10  
 new database, 4-14  
 blocks for each file, maximum, 15-6

## C

---

command-line tool  
 ORADIM, 4-15  
 commands  
 CREATE DATABASE, 4-10  
 CREATE LIBRARY, 17-10  
 NET START, 4-11  
 REGEDT32, 4-13  
 SET INSTANCE, 11-8  
 SET ROLE, 11-10  
 configuration parameters  
 defined, 16-1  
 LOCAL, B-4  
 registry, defined, 16-1  
 TNS\_ADMIN, B-4  
 USE\_SHARED\_SOCKET, B-4  
 configuring  
 advanced replication, 5-19  
 adding and modifying initialization  
 parameters, 5-19  
 monitoring data dictionary tables, 5-20  
 Named Pipes Protocol Adapter, B-5  
 Oracle Multimedia, 5-16  
 Oracle Spatial and Graph, 5-18  
 automatically, 5-18  
 Oracle Text, 5-17  
 Windows firewall exceptions, 5-3  
 Windows firewall postinstallation, 5-6

configuring Oracle Net Services for external  
 procedures, 17-8  
 configuring Oracle Text  
 using Database Configuration Assistant, 5-17  
 using DBCA, 5-17  
 CONNECT /AS SYSDBA  
 connecting without a password, 10-4  
 using, 6-4  
 connecting  
 LOCAL parameter, B-4  
 to a database, 6-4  
 to active directory using Windows login  
 credentials, 14-4  
 CREATE LIBRARY command, 17-10  
 creating  
 an Oracle Context, 14-8  
 external operating system users, 11-3  
 external roles manually, 11-9  
 Oracle Schema objects, 14-6  
 ORACLE\_SID parameter, 4-14  
 custom database  
 archiving mode, 2-9, 6-11  
 noarchiving mode, 2-9, 6-11

## D

---

data dictionary tables, 5-20  
 Data Pump Export, starting, 2-7  
 Data Pump Import, starting, 2-7  
 database administrator (DBA) privileges  
 for ASM, 11-6  
 for databases, 11-6  
 database connection error messages, D-27  
 database monitoring  
 with alert logs, 7-6  
 with trace files, 7-6  
 database operator privileges  
 for databases, 11-6  
 database tools  
 operating system compatibility, 2-1  
 running with administrator privileges, 2-5  
 running with Windows User Account Control,  
 2-5  
 starting ASMCA, 2-6  
 starting DBCA, 2-6, 4-2  
 starting from the command line, 2-7  
 starting from the Start Menu, 2-6  
 starting in multiple Oracle Homes, 2-4  
 starting Microsoft ODBC Administration, 2-6  
 starting NetCA, 2-7, 14-8  
 starting Oracle Directory Manager, 2-6  
 starting Oracle Net Manager, 2-7  
 starting Oracle Wallet Manager, 2-7  
 databases  
 backing up, 4-14

- databases (*continued*)
    - connecting to, [6-4](#)
    - creating manually, [4-3](#)
    - deleting, [4-7](#)
    - exporting, [4-5](#)
    - importing, [4-12](#)
    - monitoring, [7-1](#)
    - naming conventions, [4-1](#)
    - password encryption, [6-11](#)
    - shutting down, [6-4](#), [6-5](#)
    - starting, [6-4](#)
  - DBCA prompts for password
    - Domain User Account, [4-2](#)
    - Local User Account, [4-2](#)
    - password not stored in Oracle Wallet, [4-2](#)
  - debugging external procedures, [17-13](#)
  - deleting database files, [4-7](#)
  - developing applications for Windows, [17-1](#)
  - directory naming software requirements, [14-9](#)
  - directory servers
    - automatic discovery of directory servers, [14-3](#)
    - features integrated with Oracle Database 11g, [14-3](#)
    - how Oracle directory objects display in Active Directory, [14-5](#)
    - managing Access Control Lists, [14-15](#)
    - user interface extensions, [14-4](#)
  - DLLs
    - compared to UNIX shared libraries, [E-3](#)
    - Oracle Real Application Clusters, [16-10](#)
  - dnfs\_batch\_size parameter
    - default value is 4096, [1-7](#)
    - recommended setting, [1-7](#)
    - to control the number of queued asynchronous I/Os, [1-7](#)
  - duplicating a database
    - creating a nonstandby database from shadow copies, [9-16](#)
    - creating a standby database from shadow copies, [9-17](#)
- ## E
- 
- encrypting, database passwords, [6-11](#)
  - enhanced security, [5-14](#), [5-15](#)
  - enhancing Oracle directory object type descriptions, [14-4](#)
  - error messages
    - DIM-00000 to DIM-00228, [D-16](#)
    - ORA-01102, [4-1](#)
    - ORA-09275, [D-1](#)
    - ORA-12560, [D-27](#)
    - ORA-15252 to ORA-15266, [D-2](#)
    - ORA-15301 to ORA-15302, [D-3](#)
  - error messages (*continued*)
    - OSD-04000 to OSD-04099, [D-6](#)
    - OSD-04100 to OSD-04199, [D-10](#)
    - OSD-04200 to OSD-04299, [D-11](#)
    - OSD-04300 to OSD-04399, [D-14](#)
    - OSD-04400 to OSD-04499, [D-15](#)
    - OSD-04500 to OSD-04599, [D-15](#)
  - Event Viewer
    - defined, [2-12](#)
    - for monitoring a database, [7-1](#)
    - integration with Oracle Database, [2-12](#)
    - logging operating system audit trail, [2-12](#)
    - managing, [7-4](#)
    - reading, [7-4](#)
    - starting, [2-10](#)
  - EXECUTE privileges, on a PL/SQL library, [17-11](#)
  - Export parameter mode, [4-5](#)
  - Export Wizard, [2-2](#)
  - exporting
    - databases, [4-5](#)
    - interactive mode, [4-5](#)
    - parameter mode, [4-5](#)
    - preferred tools, [2-8](#)
  - extended partition, [A-1](#)
  - EXTERNAL clause, [17-11](#)
  - external operating system users
    - administering, [11-1](#)
    - authentication, [11-3](#)
    - authentication on client computer, [11-5](#)
    - creating, [11-3](#)
    - migrating manually, [11-12](#)
  - external procedures
    - advantages, [17-7](#)
    - creating, [17-6](#)
    - creating a PL/SQL library, [17-10](#)
    - debugging, [17-13](#)
    - executing, [17-12](#)
    - EXTERNAL clause, [17-11](#)
    - granting EXECUTE privileges, [17-11](#)
    - registering with Oracle Database, [17-10](#)
    - using, [17-6](#)
    - using EXTPROC, [17-8](#)
    - writing, [17-9](#)
  - external roles
    - administering, [11-1](#)
    - authorization on client computer, [11-11](#)
    - authorization on Oracle database server, [11-10](#)
    - creating manually, [11-9](#)
  - EXTPROC agent
    - authentication using CREATE LIBRARY extension
      - CREDENTIAL clause, [17-10](#)
      - DIRECTORY object, [17-10](#)
    - example, [17-9](#)

EXTPROC agent (*continued*)

- explained, [17-8](#)
- responsibilities, [17-8](#)

## F

- failure to modify ownership, group, and permission of open files, [D-3](#)
- features supporting large user population
  - Oracle Database Shared Server Process, [1-7](#)
  - Oracle Net multiplexing and connection pooling, [1-7](#)
  - Oracle RAC, [1-7](#)
- file I/O enhancements, [1-6](#)
- file permissions, [5-14](#), [5-15](#)
- files
  - maximum number for each database, [15-6](#)
  - maximum size possible, [15-6](#)
  - sample init.ora, [15-3](#)
  - trace, [7-6](#)
- finding information on application development for Windows, [17-1](#)
- FSEEK line terminators, [17-14](#)

## H

- hiding password file
  - using command prompt, [6-9](#)
  - using Windows Explorer, [6-9](#)

## I

- Import Wizard, [2-2](#)
- importing
  - databases, [4-12](#)
  - parameter mode, [4-12](#)
  - preferred tools, [2-2](#)
- initialization parameter file
  - defined, [15-1](#)
  - displaying values, [15-5](#)
  - editing, [15-2](#)
  - location, [15-2](#)
  - operating system specific, [15-4](#)
  - unmodifiable, [15-5](#)
  - using Advanced Replication Support, [5-19](#)
- initialization parameters
  - OS\_ROLES, [10-3](#)
  - path in registry, [16-5](#)
- instances
  - modifying, [4-20](#)
  - Oracle Database, [1-4](#)
  - running multiple instances, [6-7](#)
- integration with Windows
  - Oracle Fail Safe, [1-8](#)

integration with Windows (*continued*)

- Oracle PKI, [1-8](#)
- Oracle Services for MTS, [1-9](#)

## L

- large page support
  - enabling, [8-3](#)
  - overview, [8-2](#)
  - running as user, [8-2](#)
- listener requirements, [B-2](#)
- LOCAL networking parameter, [B-4](#)

## M

- manually migrating external operating system users, [11-12](#)
- maximum file size of control files, [15-6](#)
- memory usage, [8-6](#)
- Microsoft Active Directory, [14-1](#)
- Microsoft Certificate Services, [13-2](#)
- Microsoft Certificate Stores, [13-2](#)
- Microsoft Management Console (MMC)
  - defined, [2-12](#)
  - integration with Oracle Database, [2-12](#)
  - starting, [2-10](#)
- migrating
  - Oracle Database 11g or earlier, [4-24](#)
  - Oracle Database 12c from Oracle Database 11g, [4-24](#)
- Migration Utility tool, [2-2](#)
- MMC
  - See Microsoft Management Console
- modifying executable images, [1-6](#)
- monitoring
  - alert logs, [7-1](#)
  - Event Viewer, [7-1](#)
  - Management Pack, [7-1](#)
  - trace files, [7-1](#)
- monitoring data dictionary tables, [5-20](#)
- multiple instances, running, [6-7](#)
- multithreaded agent architecture, [17-13](#)

## N

- Named Pipes Protocol Adapter, [B-5](#)
- Named Pipes Protocol Adapter with an Oracle Names Server, [B-5](#)
- naming conventions for multiple Oracle homes, [6-2](#)
- Net Service Objects security group, [14-16](#)
- networking parameters
  - LOCAL, [B-4](#)
  - TNS\_ADMIN, [B-4](#)
  - USE\_SHARED\_SOCKET, [B-4](#)

- noarchiving mode
    - custom database, [2-9, 6-11](#)
  - NTFS file system permission setting, [5-9](#)
  - NTLM (NT Lan Manager)
    - authenticating Windows domain users, [10-2](#)
    - authenticating Windows local users, [10-2](#)
    - deprecation, [10-2](#)
  - NTS
    - See Windows native authentication
- ## O
- 
- OPER privileges, [11-6](#)
  - operating system authentication
    - automatically enabling during installation, [10-4](#)
    - connecting as SYSDBA without a password, [10-4](#)
    - OSAUTH\_PREFIX\_DOMAIN parameter, [11-4](#)
  - operating systems
    - audit trail, [2-12](#)
    - authentication overview, [10-1, 14-10](#)
  - ORA\_DBA local group, [10-4](#)
  - Oracle ASM Configuration Assistant (ASMCA), [2-5](#)
  - Oracle ASM File Access Control
    - managing, [1-3, 1-4](#)
  - Oracle Automatic Storage Management (Oracle ASM)
    - about, [1-1](#)
    - configuring disks, [A-4](#)
  - Oracle Automatic Storage Management Configuration Assistant (ASMCA), [2-6](#)
  - Oracle Database
    - connecting remotely using SYSDBA
      - privileges, [6-10](#)
    - connecting to, [6-4](#)
    - password encryption, [6-11](#)
    - shutting down, [6-4, 6-5](#)
    - specifications, [15-6](#)
    - starting, [6-4](#)
    - verifying remotely, [6-11](#)
  - Oracle Database Configuration Assistant (DBCA), [2-5](#)
    - preferred tools, [2-2](#)
    - registering a database object in a directory server, [14-4](#)
    - starting, [2-6](#)
  - Oracle Database services
    - auto-starting, [6-3](#)
      - using Control Panel, [6-3](#)
      - using Oracle Administration Assistant, [6-3](#)
  - Oracle Database services (*continued*)
    - naming conventions for multiple Oracle homes, [6-2](#)
  - Oracle VSS Writer
    - command-line syntaxes, [9-5](#)
    - installing and uninstalling, [9-5](#)
    - integrating with third-party requester applications, [9-15, 9-16](#)
    - options, [9-5](#)
  - privileges
    - SeBackupPrivilege, [C-2](#)
    - SeBatchLogonRight, [C-2](#)
    - SeIncreaseBasePriorityPrivilege, [C-2](#)
    - SeLockMemoryPrivilege, [8-3](#)
  - run under
    - LocalService Account, [3-1](#)
    - NetworkService Account, [3-1](#)
    - Windows User Account, [3-1](#)
  - shutting down a database, [6-5](#)
  - starting, [6-2](#)
    - using command prompt, [6-2](#)
    - using Control Panel, [6-2](#)
    - using Oracle Administration Assistant, [6-2](#)
  - stopping, [6-3](#)
    - using command prompt, [6-3](#)
    - using Control Panel, [6-3](#)
    - using Oracle Administration Assistant, [6-3](#)
- Oracle Database Upgrade Assistant (DBUA), [2-5](#)
- Oracle Enterprise Manager Console
  - preferred tools, [2-2](#)
- Oracle Enterprise Manager Database Management Pack, [7-1](#)
- Oracle Home User
  - comparison with Linux/UNIX Oracle User, [E-6](#)
  - permissions, [5-10](#)
- Oracle Home User Control tool
  - command-line tool, [2-10](#)
  - updates password of Oracle Home User, [2-10](#)
- Oracle Installation User
  - permissions, [5-10](#)
- Oracle Locale Builder, [2-7](#)
- Oracle Managed Files, [4-11](#)
- Oracle Multimedia
  - about, [5-16](#)
  - configuring, [5-16](#)
  - enabling Oracle Database to store, manage, and retrieve images, [5-16](#)
- Oracle Net Configuration Assistant
  - configuring Oracle software with a directory server, [14-3, 14-4](#)
  - creating Oracle Context, [14-6](#)



- Oracle Net Configuration Assistant (*continued*)
    - creating Oracle schema object, [14-6](#)
  - Oracle Net Configuration Assistant (NetCA), [2-5](#), [2-7](#)
  - Oracle Net multiplexing and connection pooling, [1-7](#)
  - Oracle Net Services
    - advanced configuration, [B-5](#)
    - running CMADMIN, [B-3](#)
    - running CMAN, [B-3](#)
    - running Oracle Listener, [B-3](#)
  - Oracle Public Key Infrastructure, [13-1](#)
  - Oracle RAC, [1-7](#), [16-10](#)
  - Oracle Real Application Clusters, [16-10](#)
    - allows multiple server computers to access the same database files, [1-7](#)
    - increases the number of user connections, [1-7](#)
    - registry values, [16-10](#)
      - See also Oracle RAC
  - Oracle Scheduler, [5-15](#)
  - Oracle Spatial and Graph configuring, [5-18](#)
  - Oracle Text
    - about, [5-17](#)
    - configuring, [5-17](#)
    - enables text queries through SQL and PL/SQL, [5-17](#)
  - Oracle VSS writer
    - command-line syntaxes, [9-5](#)
    - component-based backup, [5-7](#), [9-6](#), [9-7](#)
    - installing and uninstalling, [9-5](#)
    - options, [9-5](#)
    - volume-based backup, [5-7](#), [9-6](#)
  - Oracle VSS Writer
    - shadow copies
      - component-based, [9-3](#)
      - volume-based, [9-3](#)
  - Oracle Wallet Manager
    - about, [12-2](#)
    - starting, [2-7](#)
  - Oracle Wallets, [12-1](#)
    - creating, [4-10](#)
    - for Oracle Database Services, [4-10](#)
    - storing in the registry, [12-1](#)
    - storing private keys and trust points, [12-1](#)
  - ORACLE\_SID, [4-13](#), [6-7](#)
  - OracleDBCreator security group, [14-16](#)
  - OracleHOMENAMEClientCache, [B-2](#)
  - OracleHOMENAMECMAAdmin, [B-2](#)
  - OracleHOMENAMECMan, [B-2](#)
  - OracleHOMENAMETNSListener, [B-2](#)
  - OracleHOMENAMETNSListener service, [B-5](#)
  - OracleNetAdmins security group, [14-16](#)
  - ORADIM
    - accepts operating system user name and password if no /ospass option after osusr, [4-15](#), [C-1](#)
    - command syntax errors, [D-16](#)
    - creates Oracle Database service, [4-15](#), [C-1](#)
    - creates Oracle Scheduler service, [4-15](#), [C-1](#)
    - creates Oracle VSS Writer service, [4-15](#), [C-1](#)
    - moving or copying password files, [6-9](#)
    - preferred tools, [2-2](#)
    - starting, [2-8](#)
    - using operating system user name and password, [2-3](#)
  - ORAPWD
    - creating password files, [6-7](#)
    - starting, [2-8](#)
  - OS\_AUTHENT\_PREFIX parameter
    - case-insensitive, [11-3](#)
    - using, [11-3](#)
  - OS\_ROLES parameter
    - using with external roles, [10-3](#)
  - OSAUTH\_PREFIX\_DOMAIN, [11-4](#)
  - OSAUTH\_PREFIX\_DOMAIN parameter, [11-4](#)
- ## P
- 
- parameter mode
    - Export, [4-5](#)
    - Import, [4-12](#)
  - parameters
    - AUDIT\_FILE\_DEST, [7-4](#)
    - INST\_LOC, [16-10](#)
    - LOCAL, [B-4](#)
    - MSHELP\_TOOLS, [16-4](#)
    - NLS\_LANG and Other Globalization Parameters, [16-4](#)
    - ORA\_AFFINITY, [16-6](#)
    - ORA\_CWD, [16-5](#)
    - ORA\_HOMENAME, [16-8](#)
    - ORA\_SID\_AUTOSTART, [16-5](#)
    - ORA\_SID\_PFILE, [16-5](#)
    - ORA\_SID\_SHUTDOWN, [16-5](#)
    - ORA\_TZFILE, [16-6](#)
    - ORACLE\_BASE, [16-8](#)
    - ORACLE\_GROUP\_NAME, [16-8](#)
    - ORACLE\_HOME, [16-8](#)
    - ORACLE\_HOME\_KEY, [16-8](#)
    - ORACLE\_HOME\_USER, [16-8](#)
    - ORACLE\_PRIORITY, [16-8](#)
    - ORACLE\_SID, [4-13](#), [6-7](#), [16-9](#)
    - OSAUTH\_PREFIX\_DOMAIN, [11-4](#), [16-9](#)
    - RDBMS\_ARCHIVE, [16-9](#)
    - RDBMS\_CONTROL, [16-9](#)
    - REMOTE\_LOGIN\_PASSWORDFILE, [6-7](#)
    - SGA\_MAX\_SIZE, [15-4](#)

- parameters (*continued*)
    - SQLPATH, [16-9](#)
    - TNS\_ADMIN, [B-4](#)
    - USE\_SHARED\_SOCKET, [B-4](#)
  - partitions
    - extended, [A-1](#)
    - logical partition, [A-2](#)
    - physical disk, [A-2](#)
    - raw, [A-1](#)
  - password
    - encryption, [6-11](#)
    - not needed with SYSDBA, [10-4](#)
    - utility, [6-7](#)
  - password file
    - authenticating database administrators, [6-7](#)
    - creating, [6-7](#)
    - hiding, [6-7](#)
      - using command prompt, [6-9](#)
      - using Windows Explorer, [6-9](#)
    - viewing
      - using command prompt, [6-9](#)
      - using Windows Explorer, [6-9](#)
  - permissions
    - Administrators, [5-10](#)
    - Oracle Home User, [5-10](#)
    - Oracle Installation User, [5-10](#)
    - SYSTEM, [5-10](#)
  - PhysicalDrive, [A-2](#)
  - PL/SQL Embedded Gateway, [17-4](#)
  - postinstallation
    - setting NTFS file system permissions, [5-9](#)
    - setting NTFS File System security, [5-14](#)
    - setting permissions for Windows Registry Entries, [5-13](#)
    - setting permissions for Windows Service Entries, [5-13](#)
    - setting Windows registry security, [5-14](#)
  - preferred tools
    - Backup Wizard, [2-2](#)
    - Load Wizard, [2-2](#)
    - OCOPY, [2-2](#)
    - Recovery Manager, [2-2](#)
    - Recovery Wizard, [2-2](#)
    - SQL\*Loader, [2-2](#)
  - privileges, [2-5](#)
  - PWDSID.ORA file, [6-7](#)
- Q**
- 
- querying background processes, [1-6](#)
- R**
- 
- raw partition
    - considerations, [A-3](#)
  - raw partition (*continued*)
    - defined, [A-1](#)
    - logical partition, [A-2](#)
    - overview, [A-1](#)
    - physical disk, [A-2](#)
  - Recovery Manager
    - preferred tools, [2-2](#)
    - starting, [2-8](#)
  - registering an external procedure, [17-10](#)
  - registry
    - adding parameters, [16-12](#)
    - and Oracle Real Application Clusters, [16-10](#)
    - configuration parameters, defined, [16-1](#)
    - editor, [16-1](#)
    - editor, starting, [2-10](#)
    - INST\_LOC, [16-10](#)
    - keys, defined, [16-1](#)
    - modifying values, [16-11](#)
    - MSHELP\_TOOLS, [16-4](#)
    - NLS\_LANG and Other Globalization Parameters, [16-4](#)
    - ORA\_AFFINITY, [16-6](#)
    - ORA\_CWD, [16-5](#)
    - ORA\_HOMENAME, [16-8](#)
    - ORA\_SID\_AUTOSTART, [16-5](#)
    - ORA\_SID\_PFILE, [16-5](#)
    - ORA\_SID\_SHUTDOWN, [16-5](#)
    - ORA\_SID\_SHUTDOWN\_TIMEOUT, [16-5](#)
    - ORA\_SID\_SHUTDOWN\_TYPE, [16-6](#)
    - ORA\_TZFILE, [16-6](#)
    - ORACLE\_BASE, [16-8](#)
    - ORACLE\_GROUP\_NAME, [16-8](#)
    - ORACLE\_HOME, [16-8](#)
    - ORACLE\_HOME\_KEY, [16-8](#)
    - ORACLE\_HOME\_USER, [16-8](#)
    - ORACLE\_PRIORITY, [16-8](#)
    - ORACLE\_SID, [16-9](#)
    - OracleHOMENAMEClientCache, [B-2](#)
    - OracleHOMENAMECMAAdmin, [B-2](#)
    - OracleHOMENAMECMan, [B-2](#)
    - OracleHOMENAMETNSListener, [B-2](#)
    - OSAUTH\_PREFIX\_DOMAIN, [11-4](#), [16-9](#)
    - RDBMS\_ARCHIVE, [16-9](#)
    - RDBMS\_CONTROL, [16-9](#)
    - REG\_BINARY, [16-2](#), [16-12](#)
    - REG\_DWORD, [16-2](#), [16-12](#)
    - REG\_EXPAND\_SZ, [16-2](#), [16-12](#)
    - REG\_MULTI\_SZ, [16-2](#), [16-12](#)
    - REG\_QWORD, [16-2](#), [16-12](#)
    - REG\_SZ, [16-2](#), [16-12](#)
    - REGEDT32, [16-11](#), [16-12](#)
    - setting security, [5-14](#)
    - SQLPATH, [16-9](#)
    - update ORACLE\_SID, [4-13](#)
  - registry keys, [16-1](#)

registry parameters for storing an Oracle Wallet, [12-1](#)

REMOTE\_LOGIN\_PASSWORDFILE, [6-7](#)

resetting passwords for default accounts, [5-8](#)

resolving database connection issue

- OEM failure, [4-16](#)
- OID failure, [4-16](#)
- startup mode set to automatic, [4-16](#)

Restoring and Recovering a Database

- archivelog mode
  - performing disaster recovery, [9-14](#)
  - recovering all tablespaces, [9-14](#)
  - recovering from the loss of all multiplexed control files, [9-13](#)
  - recovering tablespaces or datafiles, [9-13](#)
  - restoring server parameter file, [9-13](#)
- noarchivelog mode
  - restoring component-based backups, [9-14](#)
  - restoring volume-based backups, [9-14](#)

role authorization

- description, [10-3](#)
- method enhancements, [10-3](#)

running SYSASM

- authentication on Oracle database server, [11-8](#)

running SYSDBA

- authentication on Oracle database server, [11-8](#)

running SYSOPER

- authentication on Oracle database server, [11-8](#)

running tools with Windows User Account Control, [2-5](#)

running windows services, [C-2](#)

## S

---

services, [6-2](#)

- auto-starting, [6-3](#)
- shutting down a database, [6-5](#)
- starting, [6-2](#)
- stopping, [6-3](#)

SET ORACLE\_SID=SID, [6-7](#)

SET ROLE command, [11-10](#)

setting file permissions

- Database Upgrade Assistant, [5-15](#)
- Oracle Database Configuration Assistant, [5-15](#)
- Oracle Universal Installer, [5-15](#)

setting registry parameters

- for starting Oracle Database, [6-5](#)
- for stopping Oracle Database, [6-5](#)

setting registry parameters (*continued*)

- optional
  - ORA\_SID\_SHUTDOWN\_TIMEOUT, [6-6](#), [16-5](#)
  - ORA\_SID\_SHUTDOWNTYPE, [6-6](#), [16-6](#)
  - ORA\_SHUTDOWN, [6-5](#)
  - ORA\_SID\_AUTOSTART, [6-5](#)
  - ORA\_SID\_PFILE, [6-5](#)
  - ORA\_SID\_SHUTDOWN, [6-5](#)
- Shared Server Process, [1-7](#)
- shutting down databases, [6-4](#), [6-5](#)
- SQL\*Loader
  - control file conventions, [2-14](#)
  - preferred tools, [2-2](#)
  - starting, [2-8](#)
- SQL\*Plus
  - connecting to a database through Active Directory, [14-4](#)
  - preferred tools, [2-2](#)
  - shutting down the database, [6-4](#)
  - starting, [2-7](#), [2-8](#), [6-4](#)
  - starting the database, [6-4](#)
- SQL\*Plus Worksheet
  - preferred tools, [2-2](#)
- sqlnet.ora file
  - and Windows native authentication, [11-8](#), [11-10](#)
  - location of, [11-8](#), [11-10](#)
- starting
  - Oracle Database, [6-5](#)
  - Oracle Database services, [6-2](#)
    - using command prompt, [6-2](#)
    - using Control Panel, [6-2](#), [6-5](#)
    - using Oracle Administration Assistant, [6-2](#)
  - SQL\*Plus, [6-4](#)
  - TKPROF, [2-9](#)
- starting an Oracle Database instance, [4-10](#)
- starting Oracle Database, [6-5](#)
- stopping
  - Oracle Database, [6-5](#)
  - Oracle Database services, [6-3](#)
    - using command prompt, [6-3](#)
    - using Control Panel, [6-3](#), [6-5](#)
    - using Oracle Administration Assistant, [6-3](#)
- stopping Oracle Database, [6-5](#)
- storing an Oracle Wallet, [12-1](#)
- SYSDBA privileges
  - connecting without a password, [10-4](#)
  - member of
    - ORA\_DBA, [10-4](#)
    - ORA\_HOMENAME\_DBA, [10-4](#)
- SYSTEM user
  - permissions, [5-10](#)

---

**T**

Task Manager  
 starting, [2-10](#)  
 using, [2-13](#)

testing connectivity  
 from client computers, [14-11](#)  
 from Microsoft tools, [14-13](#)

thread-based architecture, [1-4](#)

threads  
 and Microsoft Management Console, [2-12](#)  
 and multiple Oracle Database instances, [1-4](#)  
 and ORASTACK, [1-6](#)  
 and shared server process, [1-7](#)  
 defined, [1-4](#)  
 optional and required, [1-4](#)  
 process errors, [D-5](#)  
 processor affinity, [16-6](#)  
 scheduling priority, [16-8](#)

TNS\_ADMIN networking parameter, [B-4](#)

tools, starting  
 asmtool, [2-7](#)  
 asmtoolg, [2-7](#)  
 Data Pump Export, [2-7](#)  
 Data Pump Import, [2-7](#)  
 Database Configuration Assistant, [2-6](#)  
 DBVERIFY, [2-7](#)  
 Event Viewer, [2-10](#)  
 Export, [2-8](#)  
 Microsoft Management Console, [2-10](#)  
 Microsoft ODBC Administration, [2-6](#)  
 Oracle Automatic Storage Management Configuration Assistant (ASMCA), [2-6](#)  
 Oracle Directory Manager, [2-6](#)  
 Oracle Locale Builder, [2-7](#)  
 Oracle Net Configuration Assistant, [2-7](#)  
 Oracle Net Manager, [2-7](#)  
 Oracle Wallet Manager, [2-7](#)  
 ORADIM, [2-8](#)  
 ORAPWD, [2-8](#)  
 Recovery Manager, [2-8](#)  
 Registry Editor, [2-10](#)  
 SQL\*Loader, [2-8](#)  
 SQL\*Plus, [2-7](#), [2-8](#)  
 Task Manager, [2-10](#)  
 TKPROF, [2-9](#)

trace files  
 for monitoring a database, [7-1](#)  
 using, [7-6](#)

troubleshooting  
 ORA-12560 error, [D-27](#)  
 ORA-28575 error, [D-27](#)  
 TNS-12203 error, [D-27](#)  
 Windows firewall exceptions, [5-8](#)

tuning Windows Server operating system, [8-2](#)

---

**U**

UNIX and Windows, Oracle Database differences, [E-1](#)

USE\_SHARED\_SOCKET networking parameter, [B-4](#)

user authentication  
 description, [10-3](#)  
 enhancement methods, [10-3](#)

user group privileges  
 ORA\_ASMADMIN, [11-6](#)  
 ORA\_ASMDBA, [11-6](#)  
 ORA\_ASMOPER, [11-6](#)  
 ORA\_DBA, [11-6](#)  
 ORA\_HOMENAME\_DBA, [11-6](#)  
 ORA\_HOMENAME\_OPER, [11-6](#)  
 ORA\_HOMENAME\_SYSBACKUP, [11-6](#)  
 ORA\_HOMENAME\_SYSDG, [11-6](#)  
 ORA\_HOMENAME\_SYSKM, [11-6](#)  
 ORA\_OPER, [11-6](#)

user replacement failure on Windows, [D-2](#)

using Oracle Administration Assistant  
 start Oracle Database, [6-5](#)  
 stop Oracle Database, [6-5](#)

using ORADIM  
 creating an Oracle Database instance, [4-17](#)  
 modifying an instance, [4-20](#)  
 starting an Oracle Database instance, [4-18](#)  
 starting services, [4-18](#)

using UTL\_FILE, [17-14](#)

using VSS  
 database backup and recovery  
 concepts, [9-2](#)  
 purpose, [9-2](#)  
 scope, [9-2](#)  
 steps, [9-4](#)

---

**V**

viewing password file  
 using command prompt, [6-9](#)  
 using Windows Explorer, [6-9](#)

Volume Shadow Copy Service (VSS), [9-2](#)

VSS  
 Oracle VSS Writer, [9-2](#)  
 backup, [9-4](#)  
 VSS provider, [9-2](#)  
 VSS requester, [9-2](#)

---

**W**

Wallet Resource Locator, [13-3](#)

- Windows
  - and UNIX, Oracle Database differences, [E-1](#)
- Windows 32-bit operating system features
  - large user populations, [1-7](#)
  - Oracle PKI, [1-8](#)
- Windows domains
  - administering external users and roles, [11-1](#)
  - basic features, [10-3](#)
- Windows firewall exceptions
  - configuring, [5-3](#)
  - troubleshooting, [5-8](#)
- Windows firewall postinstallation, [5-6](#)
- Windows local groups, [10-4](#)
- Windows local groups with DBA privileges, [11-10](#)
- Windows native authentication, [10-2](#)
  - benefits, [10-1](#), [14-10](#)
  - enhancements, [10-3](#)
  - installation of, [10-1](#), [14-10](#)
  - methods and use of, [10-1](#), [14-10](#)
  - overview, [10-1](#), [14-10](#)
  - role authorization enhancements, [10-3](#)
  - setting the sqlnet.ora file, [11-8](#), [11-10](#)
  - user and role requirements, [10-3](#)
  - user authentication enhancements, [10-3](#)
- Windows tuning
  - applying latest service packs, [8-9](#)
  - closing unnecessary foreground applications, [8-12](#)
- Windows tuning (*continued*)
  - configuring server to be an application server, [8-6](#)
  - disabling unnecessary services, [8-6](#)
  - foreground applications, [8-5](#)
  - multiple striped volumes for sequential and random access, [8-11](#)
  - multiplex windows server virtual memory paging file, [8-11](#)
  - overview, [8-2](#)
  - removing unused network protocols, [8-7](#)
  - resetting the network protocol bind order, [8-8](#)
  - setting the order of multiple network interface cards, [8-8](#)
  - using hardware and operating system striping, [8-9](#)
- Windows utility tool
  - ORADIM, [4-15](#)
- Windows-specific
  - audit trail, [7-4](#)
  - initialization parameter file, [15-1](#)
  - parameter file location, [15-2](#)
  - parameter filename and location, [15-1](#)
  - password filename and location, [6-7](#)
  - role syntax, [11-10](#)
  - trace file names, [7-6](#)