Oracle® Database Express Edition
Installation Guide

21c for Linux x86-64

F32977-03

April 2024

ORACLE®

Oracle Database Express Edition Installation Guide, 21c for Linux x86-64

F32977-03

Primary Author: Prakash Jashnani

Contributing Authors: Prashanth Kannan

Contributors: Santanu Datta, Malai Stalin, Sudeepthi Siddavatam, Christopher Jones, Alex Keh

Contents

Preface

1   Installation Guide

2   Introduction

3   Requirements

4   Licensing Restrictions

5   Installing Oracle Database XE

ORACLE®

List of Tables

Preface

This guide explains how to install and configure Oracle Database Express Edition on Linux x86–64.

This guide also provides information about resources available to develop applications and how to remove the database software.

- Audience
- Documentation Accessibility
- Command Syntax
- Related Documents
- Conventions

Audience

This guide is intended primarily for application developers who are either developing applications or converting applications to run in the Oracle Database environment.

Oracle Database 21c Express Edition (Oracle Database XE) is a free version of the world's most capable relational database. Oracle Database XE is easy to install, easy to manage, and easy to develop with. With Oracle Database XE, you use an intuitive, browser-based interface to administer the database, create tables, views, and other database objects, import, export, and view table data, run queries and SQL scripts, and generate reports.

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.

Command Syntax

Refer to these command syntax conventions to understand command examples in this guide.

Convention	Description
`$`	Bourne or BASH shell prompt in a command example. Do not enter the prompt as part of the command.
`%`	C Shell prompt in a command example. Do not enter the prompt as part of the command.
`#`	Superuser (root) prompt in a command example. Do not enter the prompt as part of the command.
`monospace`	UNIX command syntax
backslash \	A backslash is the UNIX and Linux command continuation character. It is used in command examples that are too long to fit on a single line. Enter the command as displayed (with a backslash) or enter it on a single line without a backslash: `dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \  count=10000`
braces { }	Braces indicate required items: `.DEFINE {macro1}`
brackets [ ]	Brackets indicate optional items: `cvtcrt termname [outfile]`
ellipses ...	Ellipses indicate an arbitrary number of similar items: `CHKVAL fieldname value1 value2 ... valueN`
italic	Italic type indicates a variable. Substitute a value for the variable: `library_name`
vertical line |	A vertical line indicates a choice within braces or brackets: `FILE filesize [K|M]`

Related Documents

To help you with your development efforts, consult the books in the development category of the Oracle database documentation set at *Oracle Database Development*

Title	Content
2 Day Developer's Guide	Explains how to develop applications with Oracle Database.
2 Day + Java Developer's Guide	Provides a tutorial that shows you how to use Java and JDBC to connect to Oracle Database, and demonstrates how to develop a simple Java application that accesses and modifies data.

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

- *Oracle Database SQL Language Reference*
- *Oracle Database PL/SQL Language Reference*
- *Oracle Database PL/SQL Packages and Types Reference*
- *Oracle Database JSON Developer's Guide*
- *Oracle Database SODA for PL/SQL Developer's Guide*
- *Oracle Database Development Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Database SecureFiles and Large Objects Developer's Guide*
- *Oracle Database Object-Relational Developer's Guide*
- *Oracle Database Concepts*
- *Oracle Database Sample Schemas*

> ✎ **See Also:**
>
> - *Oracle Application Express Release 21.1*
> - *https://www.oracle.com/database/technologies/application-development.html*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	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.

1

Installation Guide

Welcome to *Oracle Database Express Edition Installation Guide for Linux x86-64*.

This guide covers the following topics:

- Introduction
- Requirements
- Licensing Restrictions
- Installing Oracle Database XE
- Connecting to Oracle Database XE
- Starting and Stopping Oracle Database
- Using Enterprise Manager Express
- Moving from Previous Versions of Oracle Database XE to XE 21c
- Migrating Data and Applications
- Deinstalling Oracle Database XE
- Reporting Security Vulnerabilities
- Globalization Support

2

Introduction

Oracle Database XE is a fully free edition of the Oracle Database.

Development Environments

Oracle Database XE supports the following development environments:

- **Java**: Develop and deploy modern database-bound Java Web applications (Servlets), modules (Microservices) or standalone Java frameworks using the Oracle JDBC Driver, the Universal Connection Pool (UCP), and the Database-embedded JVM (for in-place, server-side processing).

  Visit *https://www.oracle.com/database/technologies/appdev/jdbc.html* for more information.

- **C and C++** : Developers can use Oracle Call Interface (OCI) and Oracle C++ Call Interface (OCCI) to create high performance programs accessing Oracle Database XE. ODBC and the ODPI-C wrapper over OCI are also usable.

  Visit https://www.oracle.com/technetwork/database/features/oci/index-090945.html for more information.

- **.NET**: Visual Studio and .NET developers can use Oracle Data Provider for .NET (ODP.NET) and Oracle Developer Tools for Visual Studio (ODT) for full development life cycle support.

  Visit *https://www.oracle.com/database/technologies/appdev/dotnet.html* for more information

  For a walkthrough on creating a .NET application with Oracle Database XE:

  visit *https://www.oracle.com/tools/technologies/quickstart-dotnet-for-oracle-database.html*.

- **Oracle SQL Developer**: Oracle SQL Developer is a graphical version of SQL*Plus that gives database developers a convenient way to perform basic tasks. You can connect to any target Oracle Database XE schema using standard Oracle database authentication. Once connected, you can perform operations on objects in the database.

  Download and install Oracle SQL Developer from:

  https://www.oracle.com/database/sqldeveloper/

- **Oracle SQL Developer Web**: Included with Oracle REST Data Services, Oracle SQL Developer Web is the web-based version of Oracle SQL Developer that enables you to execute queries and scripts, create database objects, build data models, and monitor database activity.

- **Oracle Developer Tools for VS Code**: This free Visual Studio Code extension enables editing and execution of SQL and PL/SQL for Oracle Database and Oracle Autonomous Database.

  Download and install Oracle Developer Tools for VS Code from the Visual Studio Code Marketplace:

  https://marketplace.visualstudio.com/items?itemName=Oracle.oracledevtools.

- **Oracle Application Express**: Oracle Application Express (APEX) is a rapid web application development tool for the Oracle database.

  Download and install Application Express from:

  http://www.oracle.com/technetwork/developer-tools/apex/overview/index.html

- **Oracle REST Data Services (ORDS)**: ORDS makes it easy to develop modern REST interfaces for relational data in the Oracle Database and the Oracle Database 21c JSON Document Store.

  Download and install ORDS from:

  https://www.oracle.com/database/technologies/appdev/rest.html

- **SODA** (Simple Oracle Document Access) APIs that let you develop NoSQL-style applications against collections of JSON documents. Native language SODA drivers are available for common languages.

  Visit *https://docs.oracle.com/en/database/oracle/simple-oracle-document-access/* for more information.

Scripting Languages

You can use **scripting languages** such as:

- **PHP**

  Access Oracle Database with the PHP OCI8 extension or the PDO_OCI Driver. PHP OCI8 and PDO_OCI are part of the PHP open source project.

  Visit *http://php.net/oci8* for more information on PHP OCI8 and *http://php.net/pdo_oci* for more information on PDO_OCI.

- **Python**

  The cx_Oracle Python extension module enables access to Oracle Database.

  Visit *https://oracle.github.io/python-cx_Oracle/* for more information about cx_Oracle Python extension module.

- **Node.js**

  The node-oracledb add-on for Node.js powers high performance Oracle Database applications.

  Visit *https://oracle.github.io/node-oracledb/* for more information about node-oracledb.

- **ROracle**

  ROracle is an open source R package supporting a DBI-compliant Oracle driver based on the high performance OCI library.

  Visit *http://cran.r-project.org/web/packages/ROracle/index.html* for more information about ROracle.

- **Ruby**

  Build Ruby and Ruby on Rails applications using the `ruby-oci8` driver or `JRuby` with the Oracle Enhanced Adapter for ActiveRecord.

  Visit:

  – *http://www.rubydoc.info/gems/ruby-oci8/* for information about the ruby-oci8.

  – *http://jruby.org/* for information about JRuby.

- – *https://github.com/rsim/oracle-enhanced* for information about Oracle Enhanced Adapter.

- • **Others**

  Accessing Oracle Database XE from other languages including Go is possible using community drivers based on OCI or ODPI-C.

  Download Open Source drivers from:

  http://www.oracle.com/technetwork/database/database-technologies/scripting-languages/index.html

Oracle Call Interface (OCI) Demonstration Programs

A set of OCI demonstration programs and their corresponding project files are available in the `$ORACLE_HOME/demo` subdirectory after an Oracle Database Express Edition (XE) installation.

You can run these OCI demonstration programs to familiarize yourself with the steps involved in developing OCI applications. The Oracle Database Express Edition (XE) does not support generating the client shared library. The build option in demo_rdbms.mk is not valid for XE. You can compile and link application and demo programs with the provided header files. Because the object (.o's) and archive (.a's) libraries are not available in the installed location, you cannot use genclntsh and genclntst.

Examples

You can download and install Oracle Database Examples in an existing Oracle home to view the product demonstrations.

See *Examples Installation Guide* for more information about products available on the Oracle examples media

Examples are also available from *https://github.com/oracle/oracle-db-examples*

Learning More About Oracle Database XE

For more information on Oracle Database XE, see the following:

- • Oracle Database XE home page on the Oracle Technology Network:

  Visit *https://www.oracle.com/database/technologies/appdev/xe.html*.

- • Oracle Database XE Discussion Forum:

  You can search the Oracle Database XE Discussion Forum for answers to problems already discussed and post new questions to the community for answers.

  Visit *https://community.oracle.com/tech/developers/categories/oracle_database_express_edition_xe*.

3

Requirements

You must have `root` user credentials to install Oracle Database XE.

The system must meet the following software requirements:

- System Requirements
- Swap Space Requirements
- Server Component Kernel Parameter Requirements

System Requirements

Table 3-1 provides system requirements for Oracle Database XE.

Table 3-1   Oracle Database XE Requirements

Requirement	Value
Operating system	*Oracle Database Installation Guide* , "Operating System Checklist for Oracle Database Installation on Linux" for the list of supported Linux distributions and the set of minimum requirements for each x86–64 Linux platforms
Network protocol	The following protocols are supported: • IPC • UDP • TCP/IP • TCP/IP with SSL
RAM	1 GB of RAM minimum; 2 GB of RAM is recommended
Disk space	10 GB minimum

Swap Space Requirements

Minimum swap space recommended for Oracle Database XE is 2 GB or twice the size of RAM, whichever is lesser.

Server Component Kernel Parameter Requirements

The Oracle Database XE preinstallation RPM checks your system for the following kernel parameter settings.

If the kernel parameter values of your system are less than the values listed in Table 3-2, then the preinstallation RPM sets the recommended minimum kernel parameter values for you.

The values set in the `/etc/sysctl.d/97-oracle-database-sysctl.conf` file persist on system restarts.

Table 3-2   Kernel Parameter Settings Required for Oracle Database XE

Kernel Parameter	Setting
semmsl	250
semmns	32000
semopm	100
semmni	128
shmmax	4398046511104
shmmni	4096
shmall	1073741824
file-max	6815744
aio-max-nr	1048576
ip_local_port_range	9000–65500
panic_on_oops	1
rmem_default	262144
rmem_max	4194304
wmem_default	262144
wmem_max	1048576

> ✎ **See Also:**
>
> *Installation Guide for Linux* for more information about changing kernel parameter values manually

4
Licensing Restrictions

For more information on licensing details, see *Oracle Database Database Licensing Information User Manual*

This section covers the following topics:

- Oracle Database XE CPU Limitations
- Oracle Database XE Installation and Execution Restrictions
- Oracle Database XE User Data Limitations
- Oracle Database XE RAM Limitation

Oracle Database XE CPU Limitations

Oracle Database XE limits itself automatically to two cores for processing. For example, on a computer with 2 dual-core CPUs (four cores), if a large number of database clients try to simultaneously execute CPU-intensive queries, then Oracle Database XE will process the queries at the rate of just two cores even if more CPU capacity is available.

To take advantage of the full processing power of your computer, you can use a different Oracle Database Edition such as Oracle Database 21c Standard Edition 2 or Oracle Database 21c Enterprise Edition.

Oracle Database XE Installation and Execution Restrictions

Oracle Database XE restricts itself to only one installation per logical environment. The logical environment can either be a virtual host such as a VM or container, or a physical host. If more than one Oracle Database XE installation is attempted to be started in such a logical environment, an "ORA-00442: Oracle Database Express Edition (XE) single instance violation error" is raised and the database will not start. This does not affect any existing installation or new installations of Oracle Database 21c Standard Edition 2 or Oracle Database 21c Enterprise Edition. To run more than one Oracle Database instance or install more than one copy of the database software, upgrade to Oracle Database 21c Standard Edition 2 or Oracle Database 21c Enterprise Edition.

Oracle Database XE User Data Limitations

The maximum amount of user data in an Oracle Database XE database cannot exceed 12 GB. If the user data grows beyond this limit, then an `ORA-12592` error will appear. To use more than 12 GB of user data, upgrade to Oracle Database 21c Standard Edition 2 or Oracle Database 21c Enterprise Edition.

Oracle Database XE RAM Limitation

The maximum amount of RAM that an Oracle Database XE database uses cannot exceed 2 gigabytes, even if more is available.

To use more than 2 gigabytes of RAM, upgrade to Oracle Database 21c Standard Edition 2 or Oracle Database 21c Enterprise Edition.

For more information about managing memory, refer to *Oracle Database 2 Day DBA*.

5
Installing Oracle Database XE

You can install Oracle Database Express Edition using RPM packages.

An RPM-based installation performs preinstallation checks, extracts the database software, reassigns ownership of the extracted software to the preconfigured user and groups, maintains the Oracle inventory, and executes all root operations required to configure the Oracle Database software for a single-instance Oracle Database creation and configuration.

The RPM–based installation process detects when the minimum requirements for an installation are not met and prompts you to finish these minimum preinstallation requirements.

This section covers the following topics:

- Installing Oracle Database XE Using RPM Packages
- Performing a Silent Installation
- Setting the Oracle Database XE Environment Variables

Installing Oracle Database XE Using RPM Packages

Perform the following steps to install and configure Oracle Database XE using RPM packages.

Before attempting to install Oracle Database XE 21c, uninstall any existing Oracle Database XE or database with the SID `XE` from the target system.

An Oracle Database XE installation will consume around 9 gigabytes of disk space under `/opt`. If this disk partition does not have the required disk space available, you must add space or mount an alternative partition as `/opt/oracle`. This disk partition is the defined Oracle Base where the software and database will reside.

> **✎ Note:**
>
> The Oracle Database XE installation does not support usage of symbolic links (symlink) for that disk.

Installing Oracle Database XE RPM

1. Use `sudo` to log in as `root`.

   ```
   $ sudo -s
   ```

2. Run the Oracle Database Preinstallation RPM

   a. For Oracle Linux, the Database Preinstallation RPM is pulled automatically.

For Oracle Linux 8:

```
# dnf -y install oracle-database-preinstall-21c
```

For Oracle Linux 7:

```
# yum -y install oracle-database-preinstall-21c
```

b. For Red Hat Enterprise Linux distributions, download and install the Oracle Database Preinstallation RPM using the following commands:

For Red Hat Enterprise Linux 7

```
# curl -o oracle-database-preinstall-21c-1.0-1.el7.x86_64.rpm
https://yum.oracle.com/repo/OracleLinux/OL7/latest/x86_64/
getPackage/oracle-database-preinstall-21c-1.0-1.el7.x86_64.rpm
# yum -y localinstall oracle-database-
preinstall-21c-1.0-1.el7.x86_64.rpm
```

For Red Hat Enterprise Linux 8:

```
# curl -o oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
https://yum.oracle.com/repo/OracleLinux/OL8/appstream/x86_64/
getPackage/oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
# dnf -y localinstall oracle-database-
preinstall-21c-1.0-1.el8.x86_64.rpm
```

> **✎ Note:**
>
> • The Oracle Database Preinstallation RPM automatically creates Oracle installation owner and groups. It also sets up other kernel configuration settings required for Oracle installations. If you plan to use job-role separation, then create the extended set of database users and groups depending on your requirements.
>
> • Use the -y option if you want yum to skip the package confirmation prompt.
>
> • The yum command that is provided with Oracle Linux 8 is a symbolic link to the dnf command. The yum and dnf commands are completely interchangeable. However, the dnf command provides significant improvements in functionality and performance when compared to the yum command.
>
> • See, About DNF for more information about the dnf command.

3. Access the Oracle XE software download page:

https://www.oracle.com/database/technologies/xe-downloads.html

4. Download an appropriate rpm file required for performing an RPM-based installation to a directory of your choice:

   • For Oracle Linux 7: oracle-database-xe-21c-1.0-1.ol7.x86_64.rpm

- For Oracle Linux 8: `oracle-database-xe-21c-1.0-1.ol8.x86_64.rpm`

5. Install the database software.

    - For Oracle Linux 7:

      ```
      # yum -y localinstall oracle-database-xe-21c-1.0-1.ol7.x86_64.rpm
      ```

    - For Oracle Linux 8:

      ```
      # dnf -y localinstall oracle-database-xe-21c-1.0-1.ol8.x86_64.rpm
      ```

> **✎ Note:**
>
> Check the RPM log file to review the system configuration changes. For example, review this file for latest changes: `/var/log/oracle-database-preinstall-21c/results/orakernel.log`.

The installation of Oracle Database software is now complete.

Creating and Configuring an Oracle Database

The configuration script creates a container database (`XE`) with one pluggable database (`XEPDB1`) and configures the listener at the default port (1521) and Enterprise Manager Express on port 5500.

You can modify the configuration parameters by editing the `/etc/sysconfig/oracle-xe-21c.conf` file.

The parameters set in this file are explained in more details in the silent mode installation procedure: Performing a Silent Installation.
To create the Oracle XE database with the default settings, perform the following steps:

1. Execute as user `root` using `sudo`.

   ```
   $ sudo -s
   ```

2. Run the service configuration script:

   ```
   # /etc/init.d/oracle-xe-21c configure
   ```

   At the prompt, specify a password for the `SYS`, `SYSTEM`, and `PDBADMIN` administrative user accounts. Oracle recommends that the password entered should be at least 8 characters in length, contain at least 1 uppercase character, 1 lower case character and 1 digit [0-9].

> **✎ See Also:**
>
> The same password will be used for these accounts. The password should conform to the Oracle recommended standards. See Oracle Database Security Guide for more information about guidelines for securing passwords

After the configuration completes, the database and listener are started.

Configuration, Database Files and Logs Location

Table 5-1   Configuration, Database Files and Logs Location

File Name and Location	Purpose
`/opt/oracle`	Oracle Base. This is the root of the Oracle Database XE directory tree.
`/opt/oracle/product/21c/dbhomeXE`	Oracle Home. This home is where the Oracle Database XE is installed. It contains the directories of the Oracle Database XE executables and network files.
`/opt/oracle/oradata/XE`	Database files.
`/opt/oracle/diag` subdirectories	Diagnostic logs. The database alert log is `/opt/oracle/diag/rdbms/xe/XE/trace/alert_XE.log`
`/opt/oracle/cfgtoollogs/dbca/XE`	Database creation logs. The `XE.log` file contains the results of the database creation script execution.
`/etc/sysconfig/oracle-xe-21c.conf`	Configuration default parameters.
`/etc/init.d/oracle-xe—21c`	Configuration and services script.

> **Note:**
>
> If a host does not have any IP address other than loop back address assigned (typically in a docker or in an another container environment), Oracle Net Configuration Assistant (Oracle NETCA) may fail during the installation with the error `No valid IP Address returned for the host hostname` in the netca trace log. Please assign an IP address and retry the installation.

Performing a Silent Installation

You can install Oracle Express Edition using the silent mode. This mode can be used for embedded install of XE (with your application) or unattended operation.

To perform a silent installation, a password for the administrative accounts must be provided as a parameter to the script, or specified in the configuration file.

1. Create a wrapper shell script to perform the silent installation.

    It should contain commands similar to the following:

    For Oracle Linux 7:

    ```
    #!/bin/bash

    yum -y localinstall /downloads/oracle-database-xe-21c-1.0-1.ol7.x86_64.rpm
    > /xe_logs/XEsilentinstall.log 2>&1

    /etc/init.d/oracle-xe-21c configure >> /xe_logs/XEsilentinstall.log 2>&1
    ```

    For Oracle Linux 8

```
#!/bin/bash

yum -y localinstall /downloads/oracle-database-xe-21c-1.0-1.ol8.x86_64.rpm > /
xe_logs/XEsilentinstall.log 2>&1

/etc/init.d/oracle-xe-21c configure >> /xe_logs/XEsilentinstall.log 2>&1
```

Alternatively, you can pass on the password in the script, such as:

```
(echo "password"; echo "password";) | /etc/init.d/oracle-xe-21c configure
>> /xe_logs/XEsilentinstall.log 2>&1
```

Replace *password* with a password that is secure. The password entered should be at least 8 characters in length, contain at least 1 uppercase character, 1 lower case character and 1 digit [0-9].

2. Make the wrapper script executable.

```
chmod +x myscript.sh
```

3. Run as `root` using `sudo`.

```
sudo ./myscript.sh
```

XE Oracle Home is `/opt/oracle/product/21c/dbhomeXE` .

For details of the installation, review the `/xe_logs/XEsilentinstall.log` file.

The XE database is configured with default settings. It is not necessary to modify these parameters unless you have specific requirements. Make a copy of the configuration file `/etc/sysconfig/oracle-xe-21c.conf` before modifying it. Make your modifications after the RPM install and before configuring the database.

The provided configuration file `/etc/sysconfig/oracle-xe-21c.conf` sets the following:

- `LISTENER_PORT`: A valid listener numeric port value for the database listener. Do not specify for automatic port assignment.

- `EM_EXPRESS_PORT`: A valid port numeric value for Oracle Enterprise Manager (EM) Express listener. This is set to port 5500.

- `CHARSET`: Character set of the database. This is set to `AL32UTF8`.

- `DBFILE_DEST` Database file directory. By default, the database files are stored in the Oracle Base `/opt/oracle/oradata` subdirectory. You can also create your own database file directory. However, the permissions for this file path should be owned by `oracle user`.

- `SKIP_VALIDATIONS`: Skip validation for memory and disk space. Default: false.

> ⚠️ **Caution:**
>
> When you modify and save a file containing plain text password, ownership of the file should be given to the Oracle software installation owner only, and permissions on the file should be changed to 600. Oracle strongly recommends that database administrators or other administrators delete or secure such files containing plain text passwords when they are not in use.

> **Note:**
>
> The password should conform to the Oracle recommended standards. See
> *Oracle Database Security Guide* for more information about guidelines for
> securing passwords

You can find the database creation logs under the Oracle Base `/opt/oracle/cfgtoollogs/dbca/` subdirectory.

> **See Also:**
>
> Configuration, Database Files and Logs Location for a summary of important
> files and their locations

Setting the Oracle Database XE Environment Variables

After you have installed and configured Oracle Database XE, you must set the environment before using Oracle Database XE.

The oraenv and coraenv scripts can be used to set your environment variables.

For example, to set your environment variables in Bourne, Bash, or Korn shell without being prompted by the script:

```
$ export ORACLE_SID=XE
$ export ORAENV_ASK=NO
$ . /opt/oracle/product/21c/dbhomeXE/bin/oraenv

ORACLE_HOME = [] ? /opt/oracle/product/21c/dbhomeXE
The Oracle base has been set to /opt/oracle
```

For C shell:

```
$ setenv ORACLE_SID XE
$ setenv ORACLEENV_ASK NO
$ source /opt/oracle/product/21c/dbhomeXE/bin/coraenv
```

See *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems* for more information about setting a common environment.

6

Connecting to Oracle Database XE

Connecting Locally using OS Authentication

When you install Oracle Database XE, the oracle user is granted SYSDBA privileges. You can use the following commands to connect to the database.

```
cd <oracle_home>/bin
sqlplus / as sysdba
```

These commands connect you to the root container `CDB$ROOT` of the multitenant database (CDB) as database user `SYS`. This method of connecting to the database works even if the Net Services listener is not running.

The Net Services Listener and Default Services

The Net Services database listener for your XE database allows you to connect to the database over TCP/IP from the same machine or other machines on the network. The configuration of the Listener can be viewed using the following commands run from the command prompt:

```
cd <oracle_home>/bin
lsnrctl status

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=dbhost.example.com)
(PORT=1521)))
STATUS of the LISTENER
-----------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 21.0.0.0.0 - Production
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Default Service           XE
Listener Parameter File   /opt/oracle/homes/OraDB21Home1/network/admin/listener.ora
Listener Log File         /opt/oracle/diag/tnslsnr/dbhost/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=dbhost.example.com)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=dbhost.example.com)(PORT=5500))
(Security=(my_wallet_directory=/opt/oracle/admin/XE/xdb_wallet))(Presentation=HTTP)
(Session=RAW))
Services Summary...
Service "77f81bd10c818208e053410cc40aef5a" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
Service "XE" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
Service "XEXDB" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
Service "xepdb1" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
The command completed successfully
```

The output from the `lsnrctl` command shows values of a number of important parameters:

- the port the listener listens on
- the list of services registered with the listener
- the port EM Express listens on
- the name of the configuration file used by the listener
- the name of the log file

You specify a service when connecting to the database through the listener. The default services created by Oracle Database XE are XE and `XEPDB1`. The XE service connects you to the root container of the database (`CDB$ROOT`) and the `XEPDB1` service connects you to the default pluggable database (`XEPDB1`) created at the installation time. For each new pluggable database (PDB) in the database, there will be a new default service created with the same name as the PDB.

> **Note:**
>
> If the Oracle Database XE instance is shut down, the `lsnrctl status` command does not show any services you can connect to.

Connecting to Oracle Database Using Easy Connect Naming Method

You can connect to the database using the following Easy Connect strings:

- Multitenant container database: *host*[*:port*]
- Pluggable database: *host*[*:port*]/*service_name*

`XEPDB1` is the service name defined for the first PDB created by default. You can replace `XEPDB1` with the name of another PDB you want to connect to.

Specifying the port is optional when the listener is setup with the default port `1521`. You must specify the port number if you use another port.

Connection strings for local connections were provided on the final screen of the install. If you are connecting from a remote computer, you need to provide the hostname (where XE is installed) instead of localhost.

Please note that, Net Services database listener must be running on the database host on the specified port for the connections to succeed.

For example, you can connect to the root container of the database from a client computer with SQL*Plus using the following commands:

```
cd <oracle_home>/bin
sqlplus system@dbhost.example.com:1521
```

You can connect to the default pluggable database `XEPDB1` using the following commands:

```
cd <oracle_home>/bin
sqlplus system@dbhost.example.com:1521/XEPDB1
```

Replace `dbhost.example.com` with your database host name. If needed, replace `1521` with the port number the listener listens on. You can replace `XEPDB1` with the name of another PDB you want to connect to.

To shorten connect strings or to avoid hardcoding host name and port in application code and DBA scripts, you can define an alias for the connect string in the configuration file `orabasehome/network/admin/tnsnames.ora` on database clients. See Configuring the Local Naming Method for more details.

> **See Also:**
>
> - *2 Day DBA* for more information about the listener and network configuration
>
> - *Oracle Database Development Guide* for more information about connection strategies for database applications
>
> - Oracle Database Net Services Administrator's Guide and Oracle Database Net Services Reference for more information about Net Services and their configuration.
>
> - Oracle Multitenant Administrator's Guide for more information about containers and connecting to a container in a CDB.

7
Starting and Stopping Oracle Database

You can start and stop the database manually, set it to automatically after the system shuts down and starts, or using Enterprise Manager.

Shutting Down and Starting Up Using SQL*Plus

You can shut down and start the database using SQL*Plus.

To shutdown the database, login to the oracle user with its environment variables set for access to the XE database, and issue the following SQL*Plus command:

```
$ sqlplus / as sysdba
SQL> SHUTDOWN IMMEDIATE
```

To start the database, issue the commands:

```
SQL> STARTUP
SQL> ALTER PLUGGABLE DATABASE ALL OPEN;
```

Automating Shutdown and Startup

Oracle recommends that you configure the system to automatically start Oracle Database when the system starts, and to automatically shut it down when the system shuts down. Automating database shutdown guards against incorrect database shutdown.

To automate the startup and shutdown of the listener and database, execute the following commands as `root`:

```
$ sudo -s
```

For Oracle Linux 7:

```
# systemctl daemon-reload
# systemctl enable oracle-xe-21c
```

For Oracle Linux 8:

```
# systemctl daemon-reload
# systemctl enable oracle-xe-21c
```

Shutting Down and Starting Up Using the Configuration Services Script

> **Note:**
>
> The XE configuration script shows an incorrect listener status message. When the database listener is stopped or down, the script might show the following listener status message :
>
> ```
> /etc/init.d/oracle-xe-21c status
> Status of the Oracle XE 21c service:
>
> LISTENER status: NOT CONFIGURED
> XE Database status: STOPPED
> ```
>
> This is an incorrect listener status message. It does not mean that the script in incorrectly configured.

Run the following commands as `root` using `sudo`.

```
$ sudo -s
```

Oracle Linux 7:

Run the following command to start the listener and database:

```
# systemctl start oracle-xe-21c
```

Run the following command to stop the database and the listener:

```
# systemctl stop oracle-xe-21c
```

Run the following command to stop and start the listener and database:

```
# systemctl restart oracle-xe-21c
```

Oracle Linux 8:

Run the following command to start the listener and database:

```
# systemctl start oracle-xe-21c
```

Run the following command to stop the database and the listener:

```
# systemctl stop oracle-xe-21c
```

Run the following command to stop and start the listener and database:

```
# systemctl restart oracle-xe-21c
```

Related Topics

- Shutting Down and Starting Up Using SQL*Plus
- Altering the Open Mode of a PDB Using STARTUP and SHUTDOWN

ORACLE®

8

Using Enterprise Manager Express

You can leverage all the functionality and ease of use of Enterprise Manager (EM) to manage your XE database.

To connect to EM Express, go to `https://localhost:5500/em` from the browser on the system.

Port `5500` is the default port assigned during database creation. During install, we automatically choose next port (for example, `5501`) if `5500` is already used. Then you must provide that port.

Making Oracle Database EM Express Available to Remote Clients

After you install Oracle Database XE, EM Express is only available from the local server, it cannot be accessed remotely.

To make EM Express available to remote clients, start SQL*Plus and log in as `SYSTEM` and execute this procedure:

```
$ sqlplus system
Enter password: SYSTEM_password
SQL> EXEC DBMS_XDB.SETLISTENERLOCALACCESS(FALSE);
```

9

Moving from Previous Versions of Oracle Database XE to XE 21c

This section explains how to export and import data between Oracle Database XE 18c and XE 21c.

Oracle Database XE 11g Release 2 (11.2) users need to first upgrade to XE 18c and then upgrade from XE 18c to XE 21c using the information provided in this section.

For information on upgrading from Oracle Database XE 11g Release 2 (11.2) to XE 18c, see *Exporting and Importing Data between Oracle Database XE 11.2 and 18c*.

Upgrading Oracle Database 18c XE PDB to 21c XE

Unplug the PDB from Oracle Database 18c XE

Upgrade Instructions

> **Note:**
>
> - You cannot use Oracle Database Upgrade Assistant (Oracle DBUA) to perform the upgrade.
> - You cannot use Oracle Database Configuration Assistant (Oracle DBCA) to plug 18c XE to 21c XE database.

1. Install Oracle Database 18c XE.

2. As the `root` user, create a directory to place the 18c XE `oradata` files.

   ```
   # mkdir /opt/xepdbxml
   # chown -R oracle:oinstall /opt/xepdbxml
   # chmod -R 777 /opt/xepdbxml
   ```

3. From Oracle Database 18c XE, connect to the database as SYSDBA.

   ```
   SQL> CONNECT / AS SYSDBA
   SQL> ALTER SESSION SET CONTAINER=CDB$ROOT;
   ```

   The session is altered.

4. Close the PDB you want to upgrade.

   For example, use the following command to close the PDB `XEPDB1`.

   ```
   SQL> ALTER PLUGGABLE DATABASE XEPDB1 CLOSE;
   ```

The pluggable database is closed.

5. Unplug the Oracle Database 18c XE PDB using the following SQL command syntax:

```
SQL> ALTER PLUGGABLE DATABASE XEPDB1 UNPLUG INTO '/opt/xepdbxml/
XEPDB1.xml';
```

In the above example, the pdb name is `XEPDB1` and the location of the PDB XML file is `/opt/xepdbxml/XEPDB1.xml`.

The `Pluggable database altered` response is displayed when the command completes.

6. Drop the closed PDB and keep the data files.

```
SQL> DROP PLUGGABLE DATABASE XEPDB1 KEEP DATAFILES;
```

7. Copy all the 18c XE `XEPDB1 oradata` files from `/opt/oracle/oradata/XE/XEPDB1` to `/opt/xepdbxml`:

```
$ cd /opt/oracle/oradata/XE/XEPDB1
$ ls
sysaux01.dbf  system01.dbf  temp01.dbf  undotbs01.dbf  users01.dbf


$ cp *.dbf /opt/xepdbxml/
$ cd /opt/xepdbxml/
$ ls
sysaux01.dbf  system01.dbf  temp01.dbf  undotbs01.dbf  users01.dbf
```

8. Update all the `dbf` files in the `/opt/xepdbxml/XEPDB1.xml` file with the new `oradata` file paths.

For example, in the `XEPDB1.xml` file, replace `/opt/oracle/oradata/XE/XEPDB1/system01.dbf` with `/opt/xepdbxml/system01.dbf`. Similarly, update the remaining `.dbf` files.

9. Deinstall Oracle Database 18c XE using the command: `/etc/init.d/oracle-xe-18c delete`.

10. Remove the 18c XE package:

```
$ rpm -e oracle-database-xe-18c-1.0-1.x86_64
```

Plugging in the Oracle Database 18c XE PDB to 21c XE CDB

1. Install the latest version of Oracle Database 21c XE.

For more information, see Installing Oracle Database XE.

2. Connect to the 21c XE CDB as SYSDBA, and plug in the PDB from the 18c XE CDB to the 21c XE CDB using the following command:

```
SQL> CONNECT / AS SYSDBA
SQL> CREATE PLUGGABLE DATABASE XEPDB2 USING '/opt/xepdbxml/XEPDB1.xml'
NOCOPY TEMPFILE REUSE;
```

A new pluggable database, `XEPDB2` is created.

3. Open the PDB in `UPGRADE` mode.

```
SQL> ALTER PLUGGABLE DATABASE XEPDB2 OPEN UPGRADE;
```

4. Upgrade the PDB using the `Parallel Upgrade Utility` command.

```
$ORACLE_HOME/perl/bin/perl  $ORACLE_HOME/rdbms/admin/catctl.pl -u sys -
d $ORACLE_HOME/rdbms/admin -c XEPDB2 -l $ORACLE_BASE -t catupgrd.sql
```

You can view the upgrade summary report at `/opt/oracle/upg_summary.log`. The PDB is now in the `MOUNTED` state.

5. Open the PDB to execute post-upgrade fixups.

```
SQL> ALTER PLUGGABLE DATABASE XEPDB2 OPEN;
```

6. As the `oracle` user, create the `utlrp` directory under `/opt/xepdbxml`. Use the `catcon.pl` utility to run `utlrp.sql` from the `$ORACLE_HOME/rdbms/admin` directory.

```
$ mkdir /opt/xepdbxml/utlrp
$ $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -
d $ORACLE_HOME/rdbms/admin -n 1 -l /opt/xepdbxml/utlrp -e -b utlrp -c
'XEPDB2' utlrp.sql
```

The script recompiles `INVALID` objects in the database.

Performing a Sanity for the New PDB

```
SQL> alter session set container=XEPDB2;
Session altered.

SQL> show con_name;
CON_NAME
------------------------------
XEPDB2
SQL> select count(*) , ' Invalid Objects' from all_objects where status =
'INVALID';
  COUNT(*) 'INVALIDOBJECTS'
---------- ----------------
         0  Invalid Objects
```

Related Topics

• Upgrade instructions for APEX users moving from 18c XE to 21c XE

10

Migrating Data and Applications

You can migrate data and applications in and out of your XE database using various methods.

Oracle recommends you use 11g Release 2 and higher to export your data. For information on upgrading from Oracle Database XE 11g Release 2 (11.2) to XE 18c, see *Exporting and Importing Data between Oracle Database XE 11.2 and 18c*.

Migrating from Oracle Database XE 21c to Oracle Database Enterprise Edition 21c is supported via the unplug, plug scenario only. You can unplug a PDB from XE 21c database and plug it into another Enterpise Edition 21c database of the same release. Migrating from Oracle Database XE 21c to any of the other Oracle Database 21c offerings is not supported.

You cannot directly upgrade from Oracle Database XE 11.2 to Oracle Database XE 21c. To upgrade to XE 21c from XE 11.2, first upgrade to XE 18c, then upgrade to 21c.

Procedure to Unplug a PDB from 21c XE and Plug into 21c EE

A Pluggable Database (PDB) created in Oracle Database XE 21c can be migrated to another Multitenant Container Database (CDB) Enterprise Edition (EE) 21c.

The standard plug compatibility requirements apply. See *Oracle Multitenant Administrator's Guide* for more information about general prerequisites for PDB creation.

For example, you want to unplug your xepdb1 out of the XE 21c XE CDB and plug it in another 21c EE CDB.

- Unplug xepdb1 from XE database and create the `/tmp/xepdb1.xml` metadata file using this command :

  ```
  ALTER PLUGGABLE DATABASE xepdb1 UNPLUG INTO '/tmp/xepdb1.xml';
  ```

- Login to your EE environment. The xepdb1 metadata file should be available from that system.

- Run the DBMS_PDB.CHECK_PLUG_COMPATIBILITY function to determine whether the unplugged PDB is compatible with the CDB.

  ```
  SET SERVEROUTPUT ON
  DECLARE
    compatible CONSTANT VARCHAR2(3) :=
      CASE DBMS_PDB.CHECK_PLUG_COMPATIBILITY(
             pdb_descr_file => '/tmp/xepdb1.xml',
             pdb_name       => 'XEPDB1')
      WHEN TRUE THEN 'YES'
      ELSE 'NO'
  END;
  BEGIN
    DBMS_OUTPUT.PUT_LINE(compatible);
  END;
  /
  ```

- If the PDB is compatible with the CDB, you can plug xepdb1 into the 21c EE database using this command.

```
CREATE PLUGGABLE DATABASE NEWXEPDB USING '/tmp/xepdb1.xml'
NOCOPY
TEMPFILE REUSE;
```

This example assumes :

– The XML file accurately describes the current locations of the files. Therefore, the SOURCE_FILE_NAME_CONVERT clause or SOURCE_FILE_DIRECTORY clause is not required.

– The files are in the correct location. Therefore, NOCOPY is included.

– Storage limits are not required for the PDB. Therefore, the STORAGE clause is not required.

– A file with the same name as the temp file specified in the XML file exists in the target location. Therefore, the TEMPFILE REUSE clause is required.

See *Oracle Multitenant Administrator's Guide* for more information about plugging in an unplugged PDB.

• Run catalog, catproc and utrlp on the plugged in NEWXEPDB.

```
sqlplus / as sysdba

SQL> SHOW PDBS;
CON_ID CON_NAME          OPEN MODE   RESTRICTED
------ ---------------- ---------- ----------
2      PDB$SEED          READ ONLY  NO
3      ORCLPDB           READ WRITE NO
4      NEWXEPDB          READ WRITE YES

SQL> ALTER SESSION SET CONTAINER=newxepdb;
SQL> SPOOL catalog.log
SQL> @$ORACLE_HOME/rdbms/admin/catalog
SQL> SPOOL OFF;

SQL> SPOOL catproc.log
SQL> @$ORACLE_HOME/rdbms/admin/catproc
SQL> SPOOL OFF;

SQL> SPOOL utlrp.log
SQL> @$ORACLE_HOME/rdbms/admin/utlrp
SQL> SPOOL OFF;

SELECT COMP_NAME, STATUS, VERSION FROM DBA_REGISTRY;

COMP_NAME                            STATUS       VERSION
------------------------------------ ------------ ------------
Oracle Database Catalog Views        VALID        21.0.0.0.0
Oracle Database Packages and Types   VALID        21.0.0.0.0
Oracle Real Application Clusters     OPTION OFF   21.0.0.0.0
JServer JAVA Virtual Machine         VALID        21.0.0.0.0
Oracle XDK                           VALID        21.0.0.0.0
Oracle Database Java Packages        VALID        21.0.0.0.0
OLAP Analytic Workspace              VALID        21.0.0.0.0
Oracle XML Database                  VALID        21.0.0.0.0
Oracle Workspace Manager             VALID        21.0.0.0.0
Oracle Text                          VALID        21.0.0.0.0
Oracle Multimedia                    VALID        21.0.0.0.0
Spatial                              VALID        21.0.0.0.0
Oracle OLAP API                      VALID        21.0.0.0.0
```

ORACLE®

```
Oracle Label Security              VALID        21.0.0.0.0
Oracle Database Vault              VALID        21.0.0.0.0

SQL> SELECT OWNER, OBJECT_NAME FROM DBA_INVALID_OBJECTS WHERE STATUS = 'INVALID';

no rows selected
```

11

Deinstalling Oracle Database XE

When you deinstall Oracle Database XE, all components, including data files, the database, and the software, are removed.

If you want to save your data files but remove the Oracle Database XE software and database, then first export the data before you deinstall.

Because the deinstallation process removes all files from the directory in which Oracle Database XE is installed, back up any files from the directory (if needed) before you deinstall. The database will no longer be operational after deinstallation.

Execute this procedure as `root` or with **root** privileges.

```
$ sudo -s
```

Run the following commands to deinstall Oracle Database XE:

- This deletes all the Oracle Database XE data files, the listener and configuration files. After this operation, only logs and the Oracle Home software will be present.

  ```
  # /etc/init.d/oracle-xe-21c delete
  ```

- This removes the software. After this operation, some content under Oracle Base `/opt/oracle` will remain and can be deleted manually.

  ```
  # yum remove oracle-database-xe-21c
  ```

- Optional: If you only installed Oracle Database XE on the system and have no further Oracle Database software installed, you can also remove the Oracle Database Preinstall RPM:

  - `# yum remove oracle-database-preinstall-21c-1.0-0.3.el7.x86_64`

  - `# yum remove oracle-database-preinstall-21c-1.0-0.3.el8.x86_64`

- You can delete the downloaded RPM files, for example:

  ```
  # rm oracle-database-preinstall-21c-1.0-1.el7.x86_64.rpm
  # rm oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
  # rm oracle-database-xe-21c-1.0-1.x86_64.rpm
  ```

12
Reporting Security Vulnerabilities

If you find any security vulnerabilities with Oracle Database XE, then send a description of the problem to Oracle at the following e-mail address:

secalert_us@oracle.com

Include the following information in your e-mail:

- A complete description of the problem you encountered
- The version of Oracle Database XE you were using
- The platform on which you were running Oracle Database XE
- Any scripts or examples that may be helpful in tracking down the security problem

13
Globalization Support

Oracle Database XE is configured by default to be able to process character data in all supported languages simultaneously:

- The database is created with the Unicode `AL32UTF8` character set. `AL32UTF8` is the recommended database character set suitable for storing data in practically any language. Multiple languages can be mixed even in a single character value. While not a recommended option, you can modify the `CHARSET` parameter in the `/etc/sysconfig/oracle-xe-21c.conf` configuration file to any other supported database character set before running `/etc/init.d/oracle-xe-21c configure`.

  Supported database character sets are listed in tables A-4 and A-6 in Appendix A of the *Oracle Database Globalization Support Guide*. Character sets from Table A-4 are preferred over character sets from Table A-6 because they contain more comprehensive character repertoires.

- The Oracle Database Express Edition (XE) supports the same globalization features that Oracle Database Enterprise Edition (EE) provides.

Setting Language and Locale Preferences for Client Connections

Configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set.

You must configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set. If your applications do not have their own specific methods to configure locale preferences, then the method you use to configure an Oracle database client connection depends on the access API you use to connect to the database. Check your application documentation, before you configure locale preferences for your applications.

For applications that connect to Oracle Databases using Oracle Call Interface (OCI) use NLS_LANG and other client settings with names that start with `NLS_` to set the locale conventions and client character set for Oracle Database sessions. It is important that you set the character set part of the NLS_LANG value properly. The character set you set must correspond to the character set used by your I/O devices, which in case of Microsoft Windows is either the ANSI Code Page (for GUI applications), such as WE8MSWIN1252, or the OEM Code Page (for Console mode applications), such as US8PC437. By doing this, the OCI API is notified about the character set of data that it receives from the application. OCI can then convert this data correctly to and from the database character set.

NLS_LANG and the other NLS settings can be specified either as environment variables or as Windows Registry settings. Environment variable values take precedence over Registry values.

Oracle Universal Installer sets a default value for the NLS_LANG setting in Registry when it creates a new Oracle home on Microsoft Windows. The NLS_LANG value is based on the

language of the Windows user interface, which is the language of Windows menu items and dialog box labels.

> **⚠ Caution:**
>
> Failure to set the client character set correctly can cause data loss.

Java applications that connect to Oracle Databases by using Oracle JDBC do not use NLS_LANG. Instead, Oracle JDBC maps the default locale of the Java VM in which the application runs to the Oracle Database language and territory settings. Oracle JDBC then configures the connected database session using these settings. Because Java works internally in Unicode, the client character set is always set to Unicode. Unless an application explicitly changes it, the default locale of the Java VM is set based on the locale of the user operating system on which the Java VM runs. Check your Java VM documentation for information about configuring the Java VM default locale.

> **✎ Note:**
>
> In 3-tier architecture deployments, application servers that are database clients can have settings in their configuration files that specify the NLS_LANG value or the Java VM locale. Check the documentation accompanying these servers.

> **✎ See Also:**
>
> *Oracle Database Globalization Support Guide* for more information about configuring user locale preferences