Oracle® R Enterprise

Installation and Administration Guide

Release 1.5.1
E72864-05
November 2017

Oracle R Enterprise Installation and Administration Guide, Release 1.5.1

E72864-05

Primary Authors: David McDermid, Qin Wang

Contributing Authors: Kathy L. Taylor

Contributors: Sherry Lamonica, Humberto Daniel Vazquez Blancarte, Mark Hornick

Contents

4 Installing Oracle R Enterprise Server

8 Administrative Tasks for Oracle R Enterprise

A A Sample Installation of Oracle R Enterprise

B R Package Installation Tips

Preface

This document explains how to install and administer Oracle R Enterprise Release 1.5.1.

Audience

This document is intended for anyone who is responsible for installing or administering Oracle R Enterprise. Installation of Oracle R Enterprise requires knowledge of R and knowledge of Oracle Database.

Related Documents

The Oracle R Enterprise documentation set includes the following:

- *Oracle R Enterprise Installation and Administration Guide* (this manual)
- *Oracle R Enterprise User's Guide*
- *Oracle R Enterprise Release Notes*

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.

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.

Convention	Meaning
`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 Oracle R Enterprise Installation and Administration Guide

This preface describes changes in *Oracle R Enterprise Installation and Administration Guide* for Oracle R Enterprise Release 1.5.1 and for earlier releases to Oracle R Enterprise Release 1.4.

Changes in this Guide for Release 1.5.1

This guide contains information about these installation changes for Oracle R Enterprise Release 1.5.1.

For information about other new features in Oracle R Enterprise Release 1.5.1, see *Oracle R Enterprise User's Guide*.

New Data Manipulation Package

The new package `OREdplyr` contains Oracle R Enterprise functions for data manipulation such as filtering for rows, selecting specific columns, re-ordering rows, adding new columns, and summarizing data.

New Graph Analysis Package

Beginning with Oracle Database 12*c*, Release 2 (12.2) , the `OAAgraph` package provides an R interface to the Oracle Spatial and Graph Parallel Graph Engine (PGX) for use with Oracle R Enterprise and database tables. The `OAAgraph` package and its supporting packages require separate downloads and installation on server and client systems.

> **See Also:**
>
> Installing OAAgraph Components (page 7-1)

Updated Supporting Packages

Updated supporting packages are `DBI` and `ROracle`.

The Oracle R Enterprise supporting packages are:

```
arules 1.1-9
Cairo 1.5-8
DBI 0.5
png 0.1-7
randomForest 4.6-10
ROracle 1.3-1
statmod 1.4-21
```

> **See Also:**
>
> Installing Oracle R Enterprise Client (page 6-1) for details about the supporting packages

R 3.3.0 Requirement

Oracle R Enterprise 1.5.1 requires R 3.3.0. As with earlier releases of Oracle R Enterprise, Oracle recommends that you use Oracle R Distribution.

Oracle R Distribution Installation

Beginning with Oracle R Distribution 3.3.0, you can install the Linux RPMs in a directory other than the default Linux `R_HOME`.

> **See Also:**
>
> Install Oracle R Distribution on Linux in a Non-Default R_HOME (page 8-1)

New RPM for Oracle R Distribution

This release has a new RPM R-core-extra-3.3.0-1.el6.x86_64.rpm.

R has always depended on several third party libraries, specifically, `zlib`, `bzip2`, `xz`, `pcre`, and `curl`. Prior to R-3.3.0, R depended on much older versions of these libraries, but, if they were not found on the system, bundled copies were included that were built on the fly.

R-3.3.0 depends on much newer versions of these libraries and no longer contains the bundled copies. This means that R 3.3.0 won't build against Linux 6 as is, because the native versions of these libraries are older than those that R-3.3.0 requires.

The R-core-extra RPM contains the required versions of these libraries and is provided as a convenience for users of Oracle Linux 6. Adding the location of the libraries in R-core-extra to `LD_LIBRARY_PATH` removes the need to built these libraries separately. Oracle Linux 7 introduces the required versions of these libraries, but the R-core-extra RPM is provided as a convenience if needed.

Changes in this Guide for Release 1.5

- **Oracle R Enterprise 1.5 includes a new supporting package:** `randomForest` It also has newer versions of some other supporting packages. The Oracle R Enterprise 1.5 supporting packages are:

```
arules 1.1-9
Cairo 1.5-8
DBI 0.3-1
png 0.1-7
randomForest 4.6-10
ROracle 1.2-1
statmod 1.4.21
```

- **R 3.2 requirement**

  Oracle R Enterprise 1.5 requires R-3.2.0. As with earlier releases of Oracle R Enterprise, Oracle recommends that you use Oracle R Distribution.

Changes in this Guide for Release 1.4.1

- **Installation and administration of Oracle R Enterprise Server are significantly enhanced.**

  A single script, `server.sh` or `server.bat`, performs all administrative operations that affect Oracle R Enterprise Server. You can run the script to install, upgrade, or uninstall the server or to create or configure Oracle R Enterprise users. The script can be run interactively or in batch or hybrid mode.

  > **See Also:**
  >
  > Installing Oracle R Enterprise Server (page 4-1) for details

- **The Multitenant Container Database (CDB) feature of Oracle Database 12*c* is supported**.

  You can install support for Oracle R Enterprise Server in a multitenant environment. Oracle R Enterprise Server must be installed in a pluggable database, not in the root database.

  > **See Also:**
  >
  > – *Oracle Database Concepts* for an introduction to multitenant architecture
  >
  > – *Oracle Database Administrator's Guide* for information about managing a multitenant environment

- **Oracle R Enterprise 1.4.1 includes two new supporting packages: `arules` and `statmod`**. The Oracle R Enterprise 1.4.1 supporting packages are:

  ```
  arules 1.1-3
  cairo 1.5-5
  DBI 0.2-7
  png 0.1-7
  ROracle 1.1-12
  statmod 1.4.20
  ```

> **See Also:**
>
> Installing Oracle R Enterprise Client (page 6-1) for details about the supporting packages

- **Oracle R Enterprise supports both R 3.0.1 and R 3.1.1.**

  Previously only R 3.0.1 was supported.

Changes in this Guide for Release 1.4

- **R 3.0.1 requirement**

  Oracle R Enterprise 1.4 requires R 3.0.1. As with earlier releases of Oracle R Enterprise, Oracle recommends that you use Oracle R Distribution.

  See Installing R for Oracle R Enterprise (page 3-1).

- **Oracle R Distribution supported on Microsoft Windows**

  Oracle R Distribution 3.0.1 is supported on 64-bit Windows in addition to the 64-bit Linux and UNIX platforms that were supported in earlier releases.

  See Installing Oracle R Distribution on Microsoft Windows (page 3-11).

- **Cairo package used for graphics display on the server**

  Oracle R Enterprise 1.4 uses Cairo to display graphics on an Oracle R Enterprise server. Cairo is an open source R package that creates high quality bitmap, vector, and display output.

  Cairo is bundled with the Oracle R Enterprise supporting packages. With Cairo, there is no longer a need to configure an X11 server on Oracle Solaris and AIX servers.

  See Table 6-2 (page 6-2)

- **New client packages**

  Oracle R Enterprise 1.4 includes two new client packages:

  – `OREcommon` — Common low-level functionality for Oracle R Enterprise

  – `OREembed` — Embedded R functionality for Oracle R Enterprise

  See Table 6-1 (page 6-2).

- **Enhancements to Oracle R Enterprise Server installation script**

  The Oracle R Enterprise Server 1.4 installation script includes these enhancements:

  – **Configuration mode**

    When the `configonly` flag is set, the Installer performs database configuration for Oracle R Enterprise but does not copy the Oracle R Enterprise libraries to `$ORACLE_HOME/lib` and does not install the Oracle R Enterprise client packages.

  – **Prompts for the RQSYS password and displays the default password**

    The default password is displayed so that the user can determine whether to accept the default password or specify a different password.

- **Enhancements to demo_user script**

  The `demo_user` script, which creates a database user for Oracle R Enterprise, can now enable an existing user for Oracle R Enterprise in addition to creating a new user.

  See Creating a Database User for Oracle R Enterprise (page 8-6).

- **Migration scripts**

  Oracle R Enterprise 1.4 includes scripts for importing and exporting Oracle R Enterprise data and schema objects from a source environment to a target environment. The source and target must have the same version of Oracle Database and Oracle R Enterprise.

1

Overview of Oracle R Enterprise Installation

This chapter introduces the Oracle R Enterprise installation process. This chapter contains the following topics:

1.1 Oracle R Enterprise Architecture

Oracle R Enterprise has a client/server architecture based on Oracle Database and Oracle Client.

R engines run on the server computer and on each client computer. Oracle R Enterprise supports three key capabilities:

- **R Transparency**

  Oracle R Enterprise packages on the client support R transparency, which enables Oracle tables to appear "transparently" as native R objects. Oracle R Enterprise packages provide transparent access to Oracle Database tables and views, enabling users to invoke standard R functions, which are translated into SQL transparently to the user.

- **Predictive Analytics and Machine Learning**

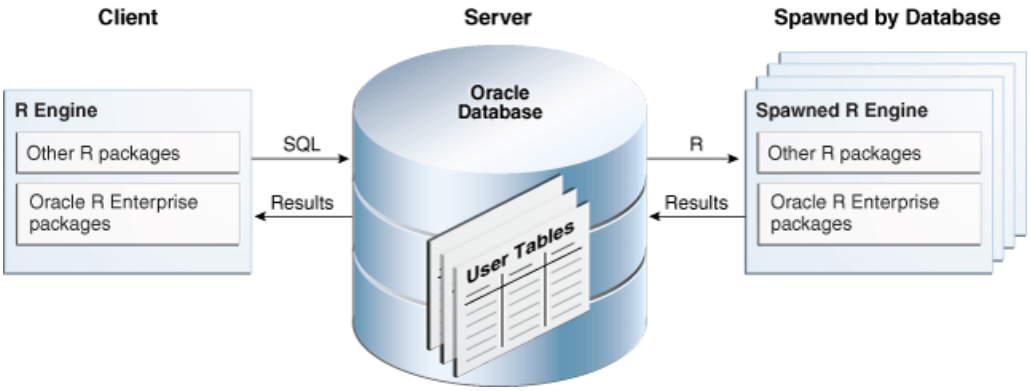
  Oracle R Enterprise supports a wide range of parallel and distributed algorithms supporting predictive analytics and machine learning. This enables both scalability and improved performance, while leveraging a convenient R interface to in-database and database server-side algorithms.

- **Embedded R Execution**

  Oracle R Enterprise packages, libraries, and R and SQL APIs on the server support the execution of R commands within SQL queries and PL/SQL statements. Embedded R is executed in spawned R engines that can run in parallel. With embedded R, you can execute R algorithms on very large data sets, and you can use database facilities like `DBMS_SCHEDULER` to schedule the execution of user-defined R functions for lights out processing.

Figure 1-1 Client/Server Architecture of Oracle R Enterprise

This figure illustrates the client/server architecture of Oracle R Enterprise.



The components of Oracle R Enterprise are:

- Client R engine
- Database server engine
- R engines spawned by Oracle Database

1.2 Client and Server Components of Oracle R Enterprise

- **Oracle R Enterprise Client Components**:
  - Oracle Database Client
  - Oracle R Enterprise packages and supporting packages
- **Oracle R Enterprise Server Components:**
  - Oracle Database with schema objects and shared libraries for supporting Oracle R Enterprise clients
  - Oracle R Enterprise packages and supporting packages

1.3 Oracle R Enterprise Installation Steps

These steps and this roadmap illustrate a typical Oracle R Enterprise installation.

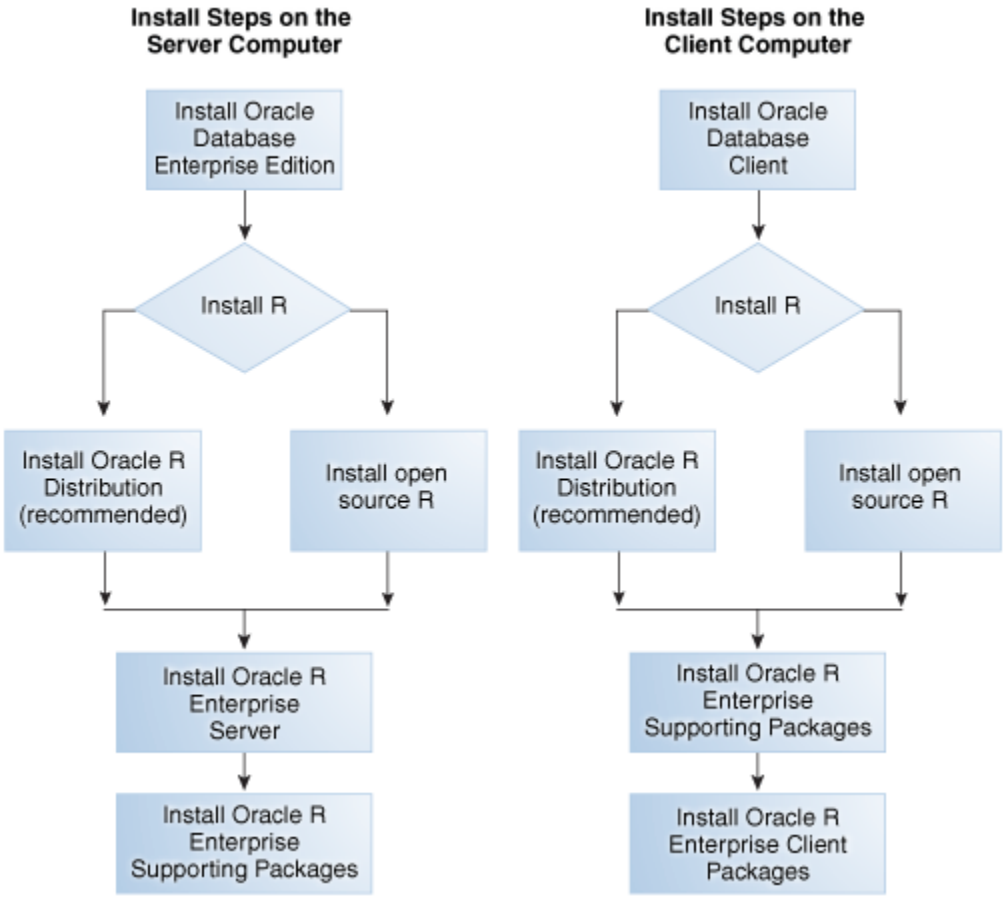
The Oracle R Enterprise Server installation script can install the supporting packages and create a database user along with the installation of Oracle R Enterprise Server.

> **Note:**
>
> If you intend to use both client and server components of Oracle R Enterprise on the computer that is hosting Oracle Database, then you do not need to perform a separate client installation. A local installation of Oracle Database Client is automatically included in the installation of Oracle Database.

Illustration of the Installation Steps

Figure 1-2 Oracle R Enterprise Client and Server Installation Steps

This figure illustrates the Oracle R Enterprise client and server installation steps.



Oracle R Enterprise Installation Roadmap

This roadmap provides the steps required to install and configure a typical Oracle R Enterprise environment. To install Oracle R Enterprise, do the following:

1. Verify that your system meets the basic requirements
2. Obtain the correct installation software
3. Perform and validate the installations

Table 1-1 Tasks for Installing Oracle R Enterprise

Task	Description	Documentation
1. Review the Oracle R Enterprise sample installation.	Review the steps for a typical installation of Oracle R Enterprise on a Linux server and a Windows client.	A Sample Installation of Oracle R Enterprise (page A-1)
2. Verify supported platforms and system requirements.	Use the Platform Requirements table and Server Support Matrix to verify your environment meets the requirements for installation.	Oracle R Enterprise System Requirements (page 1-5)
3. Identify installation users for Oracle R Distribution and Oracle R Enterprise.	The user executing the installation and configuration on your system requires sufficient permissions and privileges. For Oracle R Distribution, the installation user is root (Unix/Linux) or Administrator (Windows).	User Requirements (page 4-8) for Oracle R Enterprise Server
4. Download the product installers.	Oracle R Distribution is available from Oracle's public yum or Oracle's Open Source Software portal (OSS). Oracle Database and Oracle R Enterprise are available on Oracle's Technology Network (OTN).	Oracle Database Software Downloads Oracle R Distribution Downloads Oracle R Enterprise Downloads
5. Install and configure Oracle Database.	Oracle R Enterprise requires the 64-bit version of Oracle Database Enterprise Edition.	Installing and Configuring the Database for Oracle R Enterprise (page 2-1)
6. Install and configure R.	Oracle R Enterprise requires an installation of R on each node of the server and on each client computer that interacts with the server. Oracle R Distribution is recommended. The Oracle R Enterprise Server components must be installed on the database server.	Installing R for Oracle R Enterprise (page 3-1)
7. Install and configure Oracle R Enterprise Server.	Oracle R Enterprise includes several components on the server. Together these components enable an Oracle R Enterprise Client to interact with Oracle R Enterprise Server.	Installing Oracle R Enterprise Server (page 4-1) Installing Oracle R Enterprise on Exadata (page 5-1)
8. Install Open Source R packages on the Oracle R Enterprise server.	Embedded R execution with Oracle R Enterprise allows the use of CRAN or other third-party R packages in user-defined R functions executed on the Oracle Database server.	About R Package Installation for Oracle R Enterprise (page B-2)

Table 1-1 (Cont.) Tasks for Installing Oracle R Enterprise

Task	Description	Documentation
9. Install and configure the Oracle R Enterprise Client.	If a physical client is configured, then you must install the following Oracle R Enterprise components separately on each client computer: • R • Oracle Instant Client • Oracle R Enterprise Client packages • Oracle R Enterprise Client Supporting packages If you wish to run the Oracle R Enterprise Client through a web browser, then install RStudio Server on the database server (Linux only).	Installing Oracle R Enterprise Client (page 6-1) Installing RStudio (page C-1)
10. Install Open Source R packages on the Oracle R Enterprise client.	R packages installed on the Oracle R Enterprise server must also be installed on the Oracle R Enterprise client.	R Package Installation Basics (page B-1)
11. Optionally, install the Oracle R Enterprise `OAAgraph` package and its supporting packages.	The `OAAgraph` package provides access to the Oracle PGX high-performance graph analytics engine of Oracle Spatial and Graph, an Oracle Database option.	Installing OAAgraph Components (page 7-1)
12. Validate the Oracle R Enterprise Installation.	Test the Oracle R Enterprise installation by running example scripts on the Oracle R Enterprise server and client.	Validating Basic Oracle R Enterprise Functionality (page 4-11)

1.4 Oracle R Enterprise System Requirements

Oracle R Enterprise runs on 64-bit platforms only.

Both client and server components are supported on each of the platforms described in this topic.

Table 1-2 Oracle R Enterprise Platform Requirements

Operating System	Hardware Platform	Description
Linux x86-64	Intel and AMD	• 64-bit Oracle Linux Releases 6 and 7 • 64-bit Red Hat Enterprise Linux Releases 6 and 7 Oracle Linux may be running on Oracle Exadata Database Machine.

Table 1-2 (Cont.) Oracle R Enterprise Platform Requirements

Operating System	Hardware Platform	Description
Oracle Solaris on x86-64 (64-Bit) Oracle Solaris on SPARC-64 (64-Bit)	Intel and SPARC	• 64-bit Oracle Solaris 10 update 11 through Oracle Solaris 11 for both SPARC and x86-64 (Intel) platforms • Oracle SPARC SuperCluster • Oracle Solaris Studio (formerly Sun Studio) 12u3 or later Oracle Solaris may be running on Oracle Exadata Database Machine.
IBM AIX on POWER Systems (64-Bit)	IBM	64-bit IBM AIX 5.3 or higher
Microsoft Windows x64 (64-Bit)	Intel	64-bit Microsoft Windows

The following table shows the supported configurations of Oracle R Enterprise server components. Oracle recommends that you use Oracle R Distribution, Oracle's free distribution of R, with Oracle R Enterprise. You should install Oracle R Distribution before installing Oracle R Enterprise.

Table 1-3 Oracle R Enterprise Configuration Requirements and Server Support Matrix

Oracle R Enterprise Version	Open Source R or Oracle R Distribution	Oracle Database Release
1.5.1	3.3.0	11.2.0.4, 12.1.0.1, 12.1.0.2, 12.2.0.1
1.5	3.2.0	11.2.0.4, 12.1.0.1, 12.1.0.2
1.4.1	3.0.1, 3.1.1	11.2.0.3, 11.2.0.4, 12.1.0.1, 12.1.0.2
1.4	2.15.2, 2.15.3, 3.0.1	11.2.0.3, 11.2.0.4, 12.1.0.1
1.3.1	2.15.1, 2.15.2, 2.15.3	11.2.0.3, 11.2.0.4, 12.1.0.1
1.3	2.15.1	11.2.0.3, 11.2.0.4, 12.1.0.1
1.2	2.15.1	11.2.0.3, 11.2.0.4, 12.1.0.1
1.1	2.13.2	11.2.0.3, 11.2.0.4, 12.1.0.1
1.0	2.13.2	11.2.0.3, 11.2.0.4, 12.1.0.1

Note:

In Oracle Database Release 12.1.0.2, for some embedded R operations to be successful, both Oracle R Enterprise releases 1.4.1 and 1.5 require the database patch -- 20173897 WRONG RESULT OF GROUP BY FROM A TABLE RETURNED BY EXTPROC (Patch).

> **See Also:**
>
> - "Oracle R Distribution and Oracle R Enterprise (page 3-1)".

Verifying 64-Bit Architecture on Microsoft Windows

Oracle R Enterprise only runs on 64-bit operating systems. You can determine if your Windows system is 64-bit by following these steps:

- Windows 7 or Windows Vista:
  1. From Windows Control Panel, choose **System**.
  2. Verify that **System type** is **64-bit Operating System**.
- Windows XP:
  1. From the Start menu, choose **My Computer**.
  2. Click **Properties**.
  3. On the **System** tab, verify that the system is **x64 Edition**.

2

Installing and Configuring the Database for Oracle R Enterprise

This chapter explains how to install and configure Oracle Database to support Oracle R Enterprise Server. This chapter contains these topics:

2.1 Installing Oracle Database for Oracle R Enterprise

Lists the publications that have Oracle Database installation instructions for different platforms.

Oracle R Enterprise requires the 64-bit version of Oracle Database Enterprise Edition. To install Oracle Database, follow the installation instructions for your platform:

- **Linux**
  - *Oracle Database Installation Guide for Linux*, Release 12.1

    Oracle Database Installation Guide for Linux, Release 11.2
- **Oracle Solaris**
  - *Oracle Database Installation Guide for Oracle Solaris*, Release 12.1
  - *Oracle Database Installation Guide for Oracle Solaris*, Release 11.2
- **IBM AIX**
  - *Oracle Database Installation Guide for IBM AIX on POWER Systems (64-Bit)*, Release 12.1
  - *Oracle Database Installation Guide for IBM AIX on POWER Systems (64-Bit)*, Release 11.2
- **Microsoft Windows**
  - *Oracle Database Installation Guide for Microsoft Windows*, Release 12.1
  - *Oracle Database Installation Guide for Microsoft Windows*, Release 11.2

> **Note:**
>
> You can install Oracle R Enterprise Server in a pluggable database (PDB) within a multitenant container database (CDB). The database may not be the root database.
>
> For information about managing a multitenant environment, see *Oracle Database Administrator's Guide*.

2.2 Configuring EXTPROC for Embedded R Execution

An external procedure is a procedure invoked from a program that is written in a different language. Oracle Database uses an external procedure agent named `extproc` to support external procedures. Oracle R Enterprise uses `extproc` to support embedded R execution.

2.2.1 About EXTPROC

When an application invokes an external procedure, Oracle Database starts an `extproc` agent. The application uses the network connection established by Oracle Database to pass instructions to the agent for executing the procedure. The agent loads a DLL or shared library, runs the external procedure, and passes back to the application any values returned by the external procedure.

2.2.2 About EXTPROC Configuration for Oracle R Enterprise

Oracle R Enterprise uses the default configuration of `extproc`.

The `extproc` agent is spawned directly by Oracle Database, and no configuration changes are required to either `listener.ora` or `tnsnames.ora`. If `extproc` is configured on the database listener, it overrides the default settings.

By default, `extproc` supports any external procedure call. If you want to allow only external procedure calls for Oracle R Enterprise, you can edit the `EXTPROC_DLLS` environment variable in `ORACLE_HOME/hs/admin/extproc.ora`.

The following statement on a Linux or UNIX system sets `EXTPROC_DLLS` to only execute external procedures for Oracle R Enterprise:

```
SET EXTPROC_DLLS=ONLY:$ORACLE_HOME/lib/ore.so
```

To allow `extproc` to service any external procedure, set `EXTPROC_DLLS` to `ANY` or simply leave it blank (the default).

Beginning in R-3.3.0, for Linux 6 it is necessary to set `LD_LIBRARY_PATH` in `$ORACLE_HOME/hs/admin/extproc.ora` to the location of the R-core-extra RPM so that these libraries are found by the Oracle process running `extproc`. For example, the default location of the R-core-extra RPM is `/usr/lib64/R/port/Linux-X64/lib`. The `extproc.ora` entry for that location should be:

```
SET LD_LIBRARY_PATH=/usr/lib64/R/port/Linux-X64/lib
```

In Linux 7, setting `LD_LIBRARY_PATH` is not necessary because the required versions of these libraries are native to Linux 7 systems.

> **Note:**
>
> A database reboot is needed for changes in `extproc.ora` to take effect.

2.2.3 Troubleshooting EXTPROC

Calling an Oracle R Enterprise embedded R function may result in an error if a database configuration problem exists.

If an attempt to call an Oracle R Enterprise embedded R function results in the following error, then the external procedure did not succeed:

```
ORA-28575: unable to open RPC connection to external procedure agent.
```

This error is often a database configuration problem. It may be caused by any of the following:

- The Oracle R Enterprise user has not been granted RQADMIN role.
- The Oracle listener is not running,
- The Oracle listener configuration is incorrect, which may occur if the default external procedure configuration (which is recommended) is not being used.
- Networking layer restrictions or issues exist.
- Restrictions on external procedure calls are in force.

3
Installing R for Oracle R Enterprise

This chapter explains how to install R for Oracle R Enterprise.

This chapter contains these topics:

3.1 About R and Oracle R Enterprise

Oracle R Enterprise requires an installation of R on the server computer and on each client computer that interacts with the server.

R is third-party, open source software. Open source R is governed by GNU General Public License (GPL) and not by Oracle licensing.

3.1.1 About ROracle

ROracle is an open source R package that enables interaction between R and an Oracle database. ROracle is maintained and supported by Oracle.

ROracle is one of the open source supporting packages that is used by Oracle R Enterprise. The supporting packages are introduced in Client and Server Components of Oracle R Enterprise (page 1-2) and described in Table 6-2 (page 6-2).

3.1.2 Oracle R Distribution and Oracle R Enterprise

Oracle recommends that you use **Oracle R Distribution**, Oracle's free distribution of R, with Oracle R Enterprise. Oracle R Distribution offers significant advantages for Oracle R Enterprise.

Why Oracle R Distribution?

- Oracle R Distribution simplifies the installation of R for Oracle R Enterprise.
- Oracle R Distribution is supported by Oracle for customers of Oracle Advanced Analytics, Oracle Linux, and Oracle Big Data Appliance.
- On Windows and Linux, Oracle R Distribution simplifies integration with the **Intel Math Kernel Library** (MKL). MKL greatly improves the performance of many

mathematical computations in R, including highly vectorized and threaded Linear Algebra, Fast Fourier Transforms (FFT), Vector Math, and Statistics functions. (See Configuring Oracle R Distribution to Use MKL on the Client (page 3-11).)

- On Oracle Solaris, Oracle R Distribution automatically uses **Sun Performance Library**. Like MKL for Linux and Windows, Sun Performance Library offers improved performance of many mathematical computations. Sun Performance Library is part of Oracle Solaris Studio.

3.1.3 Open Source R and Oracle R Enterprise

Although Oracle recommends that you use Oracle R Distribution whenever possible, you can use open source R with Oracle R Enterprise.

If you choose to use open source R, then you must build it from source. Use the following configuration parameters:

```
./configure  --with-lapack --with-ICU=no --enable-R-shlib
```

3.2 Installing Oracle R Distribution on Linux

You can install Oracle R Distribution on Oracle Linux and on Redhat Enterprise Linux.

Before you begin the installation, verify that your Linux version is supported by Oracle R Enterprise, as described in the table of platform requirements in Oracle R Enterprise System Requirements (page 1-5). You can use this command to verify the Linux version:

```
# uname -r
```

> **Note:**
>
> For Oracle Linux systems that have access to the internet, Oracle recommends installing Oracle R Distribution from the Oracle Linux Yum Server.

This topic contains these sections:

3.2.1 Installing Oracle R Distribution on Oracle Linux Using Yum

Oracle recommends using yum to install Oracle R Distribution.

Yum simplifies the installation of Oracle R Distribution by automatically resolving RPM dependencies. If you install the RPMs directly, then you must resolve dependencies manually.

To install Oracle R Distribution on Oracle Linux Using Yum:

1. Log in to the Linux server as root and change to the `/etc/yum.repos.d` directory:

   ```
   # cd /etc/yum.repos.d
   ```

2. List the contents of the directory to determine if the yum configuration file is present. The name of the configuration file is `public-yum-xxx.repo`, where *xxx* is `ol6` for Oracle Linux 6, or `ol7`, for Oracle Linux 7.

   If the yum configuration file is not present, then download it from Oracle public yum by executing the `wget` command for your Linux platform:

   ```
   # wget http://public-yum.oracle.com/public-yum-xxx.repo
   ```

3. Open `public-yum-xxx.repo` in a text editor and specify `enabled=1` for *xxx*`_latest` and *xxx*`_addons`, where *xxx* indicates the version of Linux, either `ol6`, or `ol7`:

   ```
   [xxx_latest]
   enabled=1

   [xxx_addons]
   enabled=1
   ```

   Also, for Oracle Linux 7 only:

   ```
   [ol7_optional_latest]
   enabled = 1
   ```

   The location of the Oracle R Distribution packages is specified in *xxx*`_addons`. The location of the dependencies for the Oracle R Distribution RPMs is specified in *xxx*`_latest`. For Oracle Linux 7 only, several dependencies are in `optional_latest`.

   The URLs for the Oracle R Distribution RPMs in the addons repository are shown in the example at the end of this topic..

> **Note:**
>
> If you are not using the most recent version of Oracle Linux and you want to install dependent packages that are specific to your version, then you must enable the corresponding Oracle Linux repository.
>
> For example, to enable the Oracle Linux 6 base repository instead of the latest repository, follow these steps:
>
> a. Open the yum configuration file for the earlier version of Oracle Linux in an editor.
>
> ```
> /etc/yum.repos.d/public-yum-el6.repo
> ```
>
> b. Locate the section for Oracle Linux 6.
>
> ```
> [ol6_base]
> ```
>
> c. Change `enabled=0` to `enabled=1`.
>
> The result looks like this:
>
> ```
> [ol6_base]
> name=Oracle Linux $releasever installation media copy ($basearch)
> baseurl=http://public-yum.oracle.com/repo/OracleLinux/OL6/
> base/$basearch/
> gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
> gpgcheck=1
> enabled=1
> ```

4. Execute the `yum install` command to install R. Specify the version number to install for *Rversion*. For example, to install R-3.3.0, use the command `yum install R-3.3.0`.

   ```
   # yum install R-Rversion
   ```

   To install the most recent version of R that is available on Oracle public yum:

   ```
   # yum install R.x86_64
   ```

   > **Note:**
   >
   > Do not assume that the most recent version of R on Oracle public yum is supported by your version of Oracle R Enterprise. Consult the table of configuration requirements and server support in Oracle R Enterprise System Requirements (page 1-5) to determine which version of R you should use.

5. On Linux 6, install the R-core-extra RPM:

   ```
   yum install R-core-extra
   ```

6. On Linux 6, set the `LD_LIBRARY_PATH` environment variable to the location of the R-core-extra RPM.

   For example, the default location of the R-core-extra RPM is `/usr/lib64/R/port/Linux-X64/lib`. The following command sets `LD_LIBRARY_PATH` to the default location:

   ```
   export LD_LIBRARY_PATH=/usr/lib64/R/port/Linux-X64/lib
   ```

On Linux 7, the required versions of these libraries are available natively so setting `LD_LIBRARY_PATH` is not required.

About the R-core-extra RPM

R has always depended on several third party libraries, specifically, `zlib`, `bzip2`, `xz`, `pcre`, and `curl`. Prior to R-3.3.0, R depended on much older versions of these libraries, but, if they were not found on the system, bundled copies were included that were built on the fly.

R-3.3.0 depends on much newer versions of these libraries and no longer contains the bundled copies. This means that R 3.3.0 won't build against Linux 6 as is, because the native versions of these libraries are older than those that R-3.3.0 requires.

The R-core-extra RPM contains the required versions of these libraries and is provided as a convenience for users of Oracle Linux 6. Adding the location of the libraries in R-core-extra to `LD_LIBRARY_PATH` removes the need to built these libraries separately. Oracle Linux 7 introduces the required versions of these libraries, but the R-core-extra RPM is provided as a convenience if needed.

Example 3-1 Oracle R Distribution RPMs in addons Repository

In the following URLs, *Rversion* represents the version of Oracle R Distribution. For example, replace *Rversion* with 3.3.0-1 for R 3.3.0.

Oracle Linux 6:

```
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-
Rversion.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-core-
Rversion.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-core-
extra-Rversion.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-devel-
Rversion.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-
Rversion.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-
devel-Rversion.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-
static-Rversion.el6.x86_64.rpm
```

Oracle Linux 7:

```
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-
Rversion.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-
Rversion.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-
extra-Rversion.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-devel-
Rversion.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-
Rversion.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-
devel-Rversion.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-
static-Rversion.el7.x86_64.rpm
```

3.2.2 Installing Oracle R Distribution on Oracle Linux Using RPMs

Oracle recommends that you use yum to install Oracle R Distribution, because yum automatically resolves RPM dependencies. However, if yum is not available due to lack of internet access, then you can install the RPMs directly and resolve the dependencies manually.

To download and install the RPMs, log in as root and execute this command for each RPM listed in the following sections:

```
rpm -Uvh rpm_name
```

3.2.2.1 Oracle R Distribution 3.3.0 RPMs for Oracle Linux 7

The Oracle R Distribution RPMs for Oracle Linux 7 are listed as follows:

```
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/
R-3.3.0-2.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-
core-3.3.0-2.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-
extra-3.3.0-2.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-
devel-3.3.0-2.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/
libRmath-3.3.0-2.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-
devel-3.3.0-2.el7.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-
static-3.3.0-2.el7.x86_64.rpm
```

3.2.2.2 Oracle R Distribution 3.3.0 RPMs for Oracle Linux 6

The Oracle R Distribution RPMs for Oracle Linux 6 are listed as follows:

```
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/
R-3.3.0-2.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-
core-3.3.0-2.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-core-
extra-3.3.0-2.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-
devel-3.3.0-2.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/
libRmath-3.3.0-2.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-
devel-3.3.0-2.el6.x86_64.rpm
http://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-
static-3.3.0-2.el6.x86_64.rpm
```

3.2.3 Installing Oracle R Distribution on Red Hat Enterprise Linux

The Oracle Linux RPMs can be installed on Red Hat Linux systems. However, if you want to rebuild the Oracle R Distribution RPMs on a Red Hat Linux system, follow these instructions.

> **Tip:**
>
> *Rversion* represents the version of Oracle R Distribution. Replace *Rversion* with the R version you want to build. For example, replace *Rversion* with `rpm -i /refresh/home/rpmbuild/RPMS/x86_64/R-core-3.3.0-2.el6.x86_64.rpm` for R-3.3.0.

To install Oracle R Distribution on Red Hat Enterprise Linux:

1. Create an RPM build directory structure:

   ```
   mkdir -p /rpmbuild/{BUILD,RPMS,SOURCES,SPECS,SRPMS}
   ```

2. Set up RPM tools to use your own build tree (to avoid `root`):

   ```
   echo '%_topdir %(echo $HOME)/rpmbuild' > /.rpmmacros
   ```

3. From Oracle public yum, download the source RPM (*Rversion*.el*x*.src.rpm where *x* is the Oracle Linux version you are using).

   For Red Hat Enterprise Linux 6:

   Oracle Linux 6 (x86_64) Addons

   Save the source RPM to the `rpmbuild/SRPMS` directory.

4. Rebuild Red Hat Enterprise Linux using `rpmbuild`.

   ```
   rpmbuild --rebuild /rpmbuild/SRPMS/R-Rversion.elx.src.rpm
   ```

   > **Note:**
   >
   > If any dependencies are missing, install them as root.

   The binary RPMs are built and saved under `/rpmbuild/RPMS`.

5. Log in as root and execute these commands to install R:

   ```
   # rpm -i path/rpmbuild/RPMS/R-Rversion.elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/R-core-Rversion.elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/libRmath-Rversion.elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/libRmath-devel-Rversion.elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/libRmath-static-Rversion.elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/R-devel-Rversion.elx.x86_64.rpm
   ```

   For example, this command installs R 3.3.0 on Red Hat Enterprise Linux x86-64 version 6, where the path to `rpmbuild` is `/refresh/home/`.

   ```
   rpm -i /refresh/home/rpmbuild/RPMS/x86_64/R-core-3.3.0-2.el6.x86_64.rpm
   ```

3.3 Installing Oracle R Distribution on Oracle Solaris

You can install Oracle R Distribution on Oracle Solaris on Intel and on SPARC platforms.

Before you begin the installation, verify that your Oracle Solaris version is supported by Oracle R Enterprise, as described in the table of platform requirements in Oracle R Enterprise System Requirements (page 1-5). You can use this command to verify the version of Oracle Solaris:

```
uname -r
```

To install Oracle R Distribution on Oracle Solaris:

1. Go to the Oracle Open Source Software Download page for Oracle R Distribution.
2. Download the files for your installation, where *Rversion* is the version you are installing and *sunstudioversion* is the version of Sun Studio. For R-3.3.0, *Rversion* is 3.3.0.0:
   - For x86 64-bit systems:

     ord-*Rversion*-sol10-x86-64-*sunstudioversion*.tar.gz
     ord-*Rversion*-supporting-sol10-x86-64-*sunstudioversion*.tar.gz
   - For SPARC 64-bit systems:

     ord-*Rversion*-sol10-sparc-64-*sunstudioversion*.tar.gz
     ord-*Rversion*-supporting-sol10-sparc-64-*sunstudioversion*.tar.gz
3. Uncompress the first file, either `sol110-x86-64` or `sol10-sparc`.
4. Run `install.sh` as root to install the Solaris PKG file for Oracle R Distribution, where *installation_path* is the path to the directory in which to install Oracle R Distribution.

   ```
   # install.sh installation_path
   ```

   If you do not specify an installation path, then the default path is used. The default path is:
   - For Solaris SPARC: `/usr/lib/sparcv9`
   - For Solaris Intel: `/usr/lib/amd64`
5. Uncompress the second file, either `supporting-sol10-x86-64` or `supporting-sol10-sparc`, to a local directory such as `$ORACLE_HOME/lib`. Add that directory to `$LD_LIBRARY_PATH`.

   These tar files contain the shared libraries for `libR.so`:
   - `libiconv.so.2`
   - `libncurses.so.5`
   - `libreadline.so.6`
   - `libsunperf.so`

   `libsunperf.so`, Sun Performance Library, and its dependent shared libraries are included in Oracle Solaris Studio.
6. Set environment variables as follows, where *installation_path* is the path to the directory in which to install Oracle R Distribution:

- For ksh:

```
# export R_HOME=installation_path/R
# export PATH=$R_HOME/bin:$PATH
# export LD_LIBRARY_PATH=$R_HOME/lib:$LD_LIBRARY_PATH
```

- For csh:

```
# setenv R_HOME=installation_path/R
# setenv PATH=$R_HOME/bin:$PATH
# setenv LD_LIBRARY_PATH=$R_HOME/lib:$LD_LIBRARY_PATH
```

7. Run the following command to verify that `libR.so` is picking up its shared library dependencies correctly from the local directory.

```
# ldd -r installation_path/R/lib/libR.so
```

8. Start R by typing `R` at the command prompt:

```
% R
```

3.4 Installing Oracle R Distribution on IBM AIX

Before installing Oracle R Distribution, verify that your version of IBM AIX is supported by Oracle R Enterprise, as described in the table of platform requirements in Oracle R Enterprise System Requirements (page 1-5). You can use this command to verify the version of IBM AIX:

```
uname -r
```

To install Oracle R Distribution on IBM AIX:

1. Go to the Oracle Open Source Software Download page for Oracle R Distribution.

2. Download the files for your installation, where *Rversion* is 3.3.0.0 for R-3.3.0:

```
ord.Rversion-aix.tar.gz
ord-supporting-aix-Rversion.tar.gz
```

3. Uncompress and untar `ord-supporting-aix-Rversion.tar.gz`:

```
$ gunzip ord-supporting-aix-Rversion.tar.gz  # get ord-supporting-aix-
Rversion.tar
$ tar -xvf ord-supporting-aix-Rversion.tar   # extract contents of .tar file
$ ls ord-supporting-aix-Rversion             # list of rpms

bash-4.2-5.aix5.1.ppc.rpm
bzip2-1.0.6-1.aix5.1.ppc.rpm
bzip2-devel-1.0.6-1.aix5.1.ppc.rpm
cairo-1.10.0-1.aix5.2.ppc.rpm
curl-7.28.1-1.aix5.1.ppc.rpm
curl-devel-7.28.1-1.aix5.1.ppc.rpm
expat-2.0.1-3.aix5.1.ppc.rpm
fontconfig-2.5.0-1.aix5.1.ppc.rpm
gettext-0.17-1.aix5.1.ppc.rpm
glib2-2.28.6-1.aix5.1.ppc.rpm
info-4.13a-2.aix5.1.ppc.rpm
libiconv-1.14-1.aix5.1.ppc.rpm
libidn-1.29-1.aix5.1.ppc.rpm
libidn-devel-1.29-1.aix5.1.ppc.rpm
libpng-1.5.9-1.aix5.1.ppc.rpm
libpng-devel-1.5.9-1.aix5.1.ppc.rpm
libssh2-1.4.3-2.aix5.1.ppc.rpm
```

```
libssh2-devel-1.4.3-2.aix5.1.ppc.rpm
openldap-2.4.23-0.3.aix5.1.ppc.rpm
openldap-devel-2.4.23-0.3.aix5.1.ppc.rpm
openssl-1.0.1p-1.aix5.1.ppc.rpm
openssl-devel-1.0.1p-1.aix5.1.ppc.rpm
pcre-8.35-1.aix5.1.ppc.rpm
pcre-devel-8.35-1.aix5.1.ppc.rpm
pixman-0.28.2-1.aix5.1.ppc.rpm
pkg-config-0.25-2.aix5.1.ppc.rpm
readline-6.2-3.aix5.1.ppc.rpm
readline-devel-6.2-3.aix5.1.ppc.rpm
texinfo-4.13a-2.aix5.1.ppc.rpm
xrender-0.9.1-3.aix5.2.ppc.rpm
xz-devel-5.0.7-1.aix5.1.ppc.rpm
xz-libs-5.0.7-1.aix5.1.ppc.rpm
zlib-1.2.6-1.aix5.1.ppc.rpm
zlib-devel-1.2.6-1.aix5.1.ppc.rpm
```

You can also download these RPMs from AIX Open Source Packages.

4. Install the RPMs as root using an `rpm` command:

```
$ cd /download_directory/ord-supporting-aix-Rversion
$ su
# rpm -i *.rpm
```

To upgrade existing dependencies, use:

```
# rpm -UF *.rpm
```

If you experience conflicts with dependencies, use:

```
# rpm -UF --nodeps *.rpm
```

5. Add `/opt/freeware/lib64` and `/opt/freeware/lib` to the `LIBPATH` environment variable:

   - For ksh:

     ```
     $ export LIBPATH=/opt/freeware/lib64:/opt/freeware/lib:$LIBPATH
     ```

   - For csh:

     ```
     $ setenv LIBPATH /opt/freeware/lib64:/opt/freeware/lib:$LIBPATH
     ```

   Ensure that `/opt/freeware/lib64` is before `/opt/freeware/lib` and `/opt/freeware/lib` is before `/usr/lib`.

6. Uncompress `ord-Rversion-aix.tar.gz` to get `ord-Rversion-aix.tar` `install.sh` and `uninstall.sh`:

```
$ gunzip ord.Rversion-aix.tar.gz
$ tar -xf ord.Rversion-aix.tar
```

7. Run `install.sh` as root to install the filesets in Oracle R Distribution:

   In the following command, *installation_path* is the desired location for the Oracle R Distribution that is different than `/` and *rte_package_name* is the Oracle R Distribution fileset to install.

```
$ sudo ./install.sh rte_package_name installation_path
```

The path `/opt/R` is used if you do not specify an installation path.

The options for *rte_package_name* are:

Option	Description
ORD	Install all of the filesets in Oracle R Distribution.
ORD.core	Installs only the ORE.core fileset.
ORD.devel	Installs only the ORE.devel fileset, which requires the ORE.core fileset.

The following command installs all of the filesets using the specified path:

```
$ sudo ./install.sh ORD /opt/R/3_3
```

The following commands install the ORD.core and ORE.devel filesets:

```
$ sudo ./install.sh ORD.core
$ sudo ./install.sh ORD.devel
```

8. Add *installation_path*/usr/bin to the PATH environment variable:
   - For ksh:

     ```
     $ export PATH=installation_path/usr/bin:$PATH
     ```

   - For csh:

     ```
     $ setenv PATH installation_path/usr/bin:$PATH
     ```

9. Run ldd to ensure that shared library dependencies were picked up correctly:

   ```
   $ ldd installation_path/usr/lib/R/bin/exec/R
   $ ldd installation_path/usr/lib/R/lib/libR.so (libiconv, libreadline)
   $ ldd installation_path/usr/lib/R/lib/libRlapack.so
   $ ldd installation_path/usr/lib/R/lib/libRblas.so
   ```

3.5 Installing Oracle R Distribution on Microsoft Windows

Before installing Oracle R Distribution, verify that your version of Microsoft Windows is supported by Oracle R Enterprise, as described in the table of platform requirements in Oracle R Enterprise System Requirements (page 1-5).

Follow these steps to install Oracle R Distribution on Windows:

1. Go to the Oracle Open Source Software Download page for Oracle R Distribution.
2. Select R Distribution for Windows 64 bit. Save the zip file on your computer.

   ORD-*Rversion*-win.zip

3. Unzip the file and extract the executable file.

   ORD-*Rversion*-win.exe

4. Double-click the executable file to start the installation of Oracle R Distribution.
5. Follow the instructions to complete the installation.

3.6 Configuring Oracle R Distribution to Use MKL on the Client

The instructions in this section explain how to configure Oracle R Distribution to use MKL on a Linux or Windows client. With this simple configuration step, Oracle R Distribution dynamically uses MKL if it is installed on your system.

This topic contains these sections:

3.6.1 Enabling MKL Support for Oracle R Distribution on a Linux Client

Follow these steps to enable MKL for Oracle R Distribution on a Linux Client.

1. Install MKL. You can download MKL from the Intel® Math Kernel Library website.

   Note: To install MKL on your computer, you must have an MKL license.

2. Add `libmkl_rt.so`, `$RHOME/lib`, and `$ORACLE_HOME/lib` to the `LD_LIBRARY_PATH` system environment variable. For example, in the Bash shell:

   ```
   export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:
                          /path_to/libmkl_rt.so:
                          ${RHOME}/lib:
                          ${ORACLE_HOME}/lib
   ```

3. Start R and execute the `Sys.BlasLapack` function:

   ```
   Sys.BlasLapack()
        $vendor
        [1] "Intel Math Kernel Library (Intel MKL)"
        $nthreads
        [1] -1
   ```

The returned value of `$vendor` indicates that MKL has replaced the `BLAS` and `LAPACK` that are native to R.

The returned value of `nthreads` indicates the number of threads to be used by MKL. By default all available threads are used (`$nthreads`= -1).

3.6.1.1 Modifying the Number of Threads for MKL on Linux

You can change the number of threads to be used by MKL by editing the system environment variable `MKL_NUM_THREADS`. For example, the following statement in the Bash shell, causes MKL to use 3 threads:

```
export MKL_NUM_THREADS=3
```

After setting `MKL_NUM_THREADS` to 3, the output of `Sys.BlasLapack` shows a value of 3 for `$nthreads`.

```
R> Sys.BlasLapack()
     $vendor
     [1] "Intel Math Kernel Library (Intel MKL)"
     $nthreads
     [1] 3
```

3.6.2 Enabling MKL Support for Oracle R Distribution on a Windows Client

Follow these steps to enable MKL for Oracle R Distribution on a Windows client (64-bit).

1. Install MKL. You can download MKL from the Intel® Math Kernel Library website:

Note: To install MKL on your computer, you must have an MKL license.

2. Add the location of `libOrdBlasLoader.dll` and `mkl_rt.dll` to the `PATH` system environment variable.For instructions, see.

> **Note:**
>
> In a typical installation of Oracle R Distribution, `libOrdBlasLoader.dll` is located in the R home directory:
>
> `C:\Program Files\R\R-version\bin\x64`
>
> In a full installation of MKL 11.1, `mkl_rt.dll` is located in the Intel MKL Composer XE directory:
>
> `C:\Program Files (x86)\Intel\Composer XE 2013 SP`

3. Start R and execute the `Sys.BlasLapack` function:

```
R> Sys.BlasLapack()
     $vendor
     [1] "Intel Math Kernel Library (Intel MKL)"
     $nthreads
     [1] -1
```

The returned value of `$vendor` indicates that MKL has replaced the `BLAS` and `LAPACK` that are native to R.

The returned value of `nthreads` indicates the number of threads to be used by MKL. By default all available threads are used (`$nthreads`= -1).

3.6.2.1 Modifying the Number of Threads for MKL on Windows

You can change the number of threads to be used by MKL by editing the system environment variable `MKL_NUM_THREADS`. If `MKL_NUM_THREADS` does not exist, then you must create it as described in:

After setting `MKL_NUM_THREADS` to 3, the output of `Sys.BlasLapack` shows a value of 3 for `$nthreads`.

```
R> Sys.BlasLapack()
     $vendor
     [1] "Intel Math Kernel Library (Intel MKL)"
     $nthreads
     [1] 3
```

3.7 Uninstalling Oracle R Distribution

To uninstall R, follow the instructions in the following sections:

3.7.1 Uninstalling Oracle R Distribution on Windows

Uninstall Oracle R Distribution just as you would uninstall any other Windows program, using **Programs and Features** in Windows Control Panel.

3.7.2 Uninstalling Oracle R Distribution on Linux

To uninstall Oracle R Distribution on Linux, log in as root and execute these commands in this order.

The example uninstalls R-3.3.0. To uninstall a different version of R, replace the R version in the example with the number of the version you want to uninstall.

Example 3-2 Linux Commands for Uninstalling Oracle R Distribution

Execute the `rpm -e` *`rpmname`* command, where *rpmname* is the name of the RPM you want to remove.

For example, to remove R-3.3.0 on Oracle Linux 6:

```
rpm -e R-3.3.0-2.el6
rpm -e R-devel
rpm -e R-core
rpm -e R-core-extra
rpm -e libRmath-devel
rpm -e libRmath
rpm -e libRmath-static
```

3.7.3 Uninstalling Oracle R Distribution on Oracle Solaris

Follow these instructions to uninstall Oracle R Distribution on Oracle Solaris.

To uninstall Oracle R Distribution on Oracle Solaris, follow the instructions in the readme on the Oracle R Distribution downloads page on the Oracle Technology Network.

The Oracle R Distribution installation directory on Oracle Solaris includes an uninstall script. Log in as root and run the script as follows, where *installation_path* is the path to the directory in which Oracle R Distribution is installed:

Example 3-3 Solaris Script for Uninstalling Oracle R Distribution

```
./uninstall.sh installation_path
```

> **Note:**
>
> If you do not specify the installation path, then the following default path is used:
>
> - For Solaris SPARC: `/usr/lib/sparcv9`
> - For Solaris Intel: `/usr/lib/amd64`

3.7.4 Uninstalling Oracle R Distribution on IBM AIX

Follow these instructions to uninstall Oracle R Distribution on IBM AIX.

To uninstall Oracle R Distribution on IBM AIX, follow the instructions in the readme on the Oracle R Distribution downloads page on the Oracle Technology Network.

Example 3-4 AIX Scripts for Uninstalling Oracle R Distribution

Run `uninstall.sh` as root to uninstall the filesets in Oracle R Distribution:

In the following command, *installation_path* is the location in which Oracle R Distribution is installed and *rte_package_name* is the Oracle R Distribution fileset to uninstall.

```
$ sudo ./uninstall.sh rte_package_name installation_path
```

The path `/opt/R` is used if you do not specify an installation path.

The options for *rte_package_name* are:

Option	Description
`ORD`	Uninstall all of the filesets in Oracle R Distribution.
`ORD.core`	Uninstalls only the `ORE.core` fileset.
`ORD.devel`	Uninstall only the `ORE.devel` fileset.

The following command uninstalls all of the filesets using the specified path:

```
$ sudo ./uninstall.sh ORD /opt/R/3_3
```

The following commands uninstall the `ORD.core` and `ORE.devel` filesets:

```
$ sudo ./install.sh ORD.core
$ sudo ./install.sh ORD.devel
```

4
Installing Oracle R Enterprise Server

This chapter explains how to install and administer Oracle R Enterprise Server. This chapter includes these topics:

4.1 About Oracle R Enterprise Server

Oracle R Enterprise includes components on the Oracle Database Enterprise Edition server that enable an Oracle R Enterprise client to interact with Oracle R Enterprise Server.

The components are:

- Oracle R Distribution or open source R
- Oracle R Enterprise Server
  - The RQSYS schema, described in About the RQSYS Schema (page 4-1)
  - Metadata and executable code in `sys`
  - Oracle R Enterprise Server libraries in `$ORACLE_HOME/lib` (Linux and UNIX) or `%ORACLE_HOME%\bin` (Windows)
  - Oracle R Enterprise R packages in `$ORACLE_HOME/R/library` (`%ORACLE_HOME%\R\library` on Windows)

    The Oracle R Enterprise packages and supporting packages on the server support embedded R execution. These same packages must be installed separately on each client computer. (See About the Oracle R Enterprise Packages (page 6-1)).

4.1.1 About the RQSYS Schema

The RQSYS schema is the system account for Oracle R Enterprise in Oracle Database.

It contains metadata, PL/SQL packages, and other executable code that is used internally by Oracle R Enterprise Server.

The Oracle R Enterprise Server installation process creates RQSYS as a locked account with an expired password. The `rqsys` user does not have the `CREATE SESSION` privilege.

4.1.2 Security Best Practices for Oracle R Enterprise

To minimize the risk of compromising the security of an Oracle R Enterprise Server in Oracle Database, Oracle recommends the following security best practices.

Oracle R Enterprise Server components in an Oracle Database instance include the locked and password-expired RQSYS schema, which contains and manages Oracle R Enterprise metadata. Users connect to Oracle R Enterprise Server through their database connection credentials. The RQADMIN role grants a user the privilege of creating R functions as scripts in the Oracle R Enterprise R script repository; those scripts can be executed using Oracle R Enterprise embedded R execution.

Oracle recommends the following security best practices.

- Do not unlock the RQSYS schema or enable its login.
- Grant the RQADMIN role only to database users who are responsible for creating and managing the R script repository.
- Create private R scripts and grant access to other users as needed. Global R scripts are visible to and can be executed by any Oracle R Enterprise user.
- Use parameters or the Oracle R Enterprise datastore to transfer data between embedded R execution scripts and Oracle Database. R scripts should not interact with the server file system or the network.
- Set the Oracle R Enterprise embedded R execution memory limit properly based upon the Oracle Database server resources and usage patterns. The default value is 2 GB per connection.
- Use the auto-connect feature (`connect=TRUE`) instead of providing explicit database credentials when connecting back to the Oracle Database server in an R script that uses embedded R execution.
- Do not allow unauthorized R packages or C libraries to be loaded on the Oracle Database server for use in embedded R execution.
- Load dependent shared libraries from the `$ORACLE_HOME/lib` directory to prevent the use of unauthorized libraries.

Related Topics

- About the RQADMIN Role (page 8-7)
  The `server` script installation process creates a database role called RQADMIN.
- Controlling Memory Used by Embedded R (page 8-11)

4.2 About the SERVER Script

A single script called `server` manages the installation and administration of Oracle R Enterprise Server. You can rerun the `server` script whenever you need to install, uninstall, upgrade, or configure server-side components of Oracle R Enterprise.

4.2.1 Overview of SERVER Operations

The `server` script supports the following operations:

- Installs Oracle R Enterprise Server
- Uninstalls Oracle R Enterprise Server
- Upgrades Oracle R Enterprise Server and migrates data from the earlier installation
- Installs the supporting packages, if they are available
- Creates or configures a database user, if one does not exist

> **Note:**
>
> You can use the `server` script to install the supporting packages and create users, or you can choose to perform these tasks separately, as described in the following sections:
>
> - Installing the Oracle R Enterprise Supporting Packages (page 6-8)
> - Creating a Database User for Oracle R Enterprise (page 8-6)

4.2.2 SERVER Syntax

The `server` script supports a set of command-line arguments that direct its activities. The script can be run in interactive mode, in batch mode, or in hybrid mode. If you run the script without arguments, it installs or upgrades Oracle R Enterprise Server in interactive mode; it attempts to install the supporting packages; and it creates or configures a database user.

The command-line arguments for the `server` script are described in the following table. The arguments for the script are the same for Linux, UNIX, and Windows. You can obtain a listing of the arguments with brief descriptions by executing the following on a Linux or UNIX system:

```
./server.sh -h
or
./server.sh --help
```

On a Windows system, you can obtain a listing of the arguments with brief descriptions by executing the following:

```
server.bat -h
or
server.bat --help
```

Table 4-1 SERVER Script Command-Line Arguments

Argument	Description
`-y`	Never prompt.

Table 4-1 (Cont.) SERVER Script Command-Line Arguments

Argument	Description
`-i`, `--install`	Install or upgrade Oracle R Enterprise Server. An installation or upgrade includes the following by default: • Installation of the supporting packages, if they are present. • Creation or configuration of a database user, if one does not exist.
`-u`, `--uninstall`	Uninstall Oracle R Enterprise Server: • When used with `--keep` (the default), the script removes the RQSYS metadata and PL/SQL packages from the database but retains the libraries and R packages under Oracle home (partial uninstall). • When used with `--full`, the script removes the libraries and R packages under Oracle home in addition to the RQSYS metadata and PL/SQL packages in the database. (full uninstall). See Uninstalling Oracle R Enterprise (page 8-4).
`-s`, `--setup-user`	Create or configure a database user for Oracle R Enterprise (the default).
`--keep`	When uninstalling Oracle R Enterprise Server, keep the R packages and libraries under Oracle home but remove the database objects. Allows Oracle R Enterprise support to be removed from a single database instance or pluggable database (PDB) without affecting other databases in Oracle home. See Performing a Partial Uninstall (page 8-5).
`--full`	When uninstalling Oracle R Enterprise Server, remove the R packages and libraries under Oracle home in addition to the database objects. See Performing a Full Uninstall (page 8-5).
`--no-supp`	When combined with `--install`, prevents installation of the supporting packages. By default the supporting packages are installed if they are available.
`--no-user`	When combined with `--install`, prevents the creation of an Oracle R Enterprise user. By default a user is created if one does not already exist.
`--admin`	Grant the `rqadmin` role to the Oracle R Enterprise user. By default, the `rqadmin` role is *not* granted. The `--admin` option should be used with caution. It is only available in batch mode. See About the RQADMIN Role (page 8-7).
`--sys PASSWORD`	`sys` password. The `sys` password is not required if the script is running under operating system authentication, as described in About Operating System Authentication (page 4-8).
`--pdb NAME`	The name of a pluggable database (PDB) in a multitenant container database (CDB). Multitenant architecture enables an Oracle database to function as a container database that includes zero, one, or many pluggable databases. For information about multitenant architecture, see *Oracle Database Concepts*.
`--perm PERM`	Permanent tablespace for RQSYS.

Table 4-1 (Cont.) SERVER Script Command-Line Arguments

Argument	Description
`--temp TEMP`	Temporary tablespace for RQSYS.
`--rqsys PASSWORD`	RQSYS password. See About the RQSYS Schema (page 4-1).
`--user-perm PERM`	Permanent tablespace for Oracle R Enterprise user.
`--user-temp TEMP`	Temporary tablespace for Oracle R Enterprise user.
`--pass PASSWORD`	Oracle R Enterprise user password.
`--user USER`	Oracle R Enterprise database user name.

4.2.3 SERVER Examples

This topic provides these examples:

- A Default Interactive Installation (page 4-5)
- A Default Batch Installation (page 4-6)
- User Configuration in Interactive Mode (page 4-6)
- User Configuration in Batch Mode (page 4-6)

4.2.3.1 A Default Interactive Installation

If your Linux or UNIX or Windows system meets the requirements specified in System Requirements (page 4-7), then this command performs a default, first-time installation of Oracle R Enterprise Server:

For Linux or UNIX:

```
./server.sh
```

For Windows:

```
server.bat
```

As shown in Example A-1 (page A-4), a default, interactive installation performs the following:

- Prints out information about the environment
- Prompts for the password and permanent and temporary tablespaces for `rqsys`
- Prompts whether to install the supporting packages. (Installs the supporting packages by default if they are available.)
- Prompts whether to create a user account for Oracle R Enterprise. (Creates a user by default if one does not exist.) When creating a user, prompts for the permanent and temporary tablespaces.

4.2.3.2 A Default Batch Installation

This example shows an installation like the one in A Default Interactive Installation (page 4-5), but specified to run in batch mode.

For Linux or UNIX:

```
./server.sh  -y  --install  --setup-user  --sys ORASYSPSWD,
          --perm SYSAUX  --temp TEMP  --rqsys RQSYSPSWD
          --user-perm USERS  --user-temp TEMP  --pass RQUSERPSWD  --user RQUSER
```

For Windows:

```
server.bat  -y  --install  --setup-user  --sys ORASYSPSWD,
          --perm SYSAUX  --temp TEMP  --rqsys RQSYSPSWD
          --user-perm USERS  --user-temp TEMP  --pass RQUSERPSWD  --user RQUSER
```

4.2.3.3 User Configuration in Interactive Mode

The `server` script automatically creates or configures a user if one does not already exist. If you supply the name of an existing user, the script configures it to support Oracle R Enterprise.

See Example 8-2 (page 8-6).

4.2.3.4 User Configuration in Batch Mode

This example shows how the `server` script could be run to grant the `rqadmin` role to the user created in A Default Batch Installation (page 4-6). The `--admin` argument is only available in batch mode.

For Linux or UNIX:

```
./server.sh  -y  --setup-user  --admin  --sys ORASYSPSWD -
          -pass RQUSERPSWD  --user RQUSER
```

For Windows:

```
server.bat  -y  --setup-user  --admin  --sys ORASYSPSWD -
          -pass RQUSERPSWD  --user RQUSER
```

See About the RQADMIN Role (page 8-7).

4.3 Oracle R Enterprise Server Requirements

Before installing Oracle R Enterprise Server, verify your system environment, and ensure that your user ID has the proper permissions.

You should also have installed the Oracle R Enterprise Server prerequisites: Oracle Database and Oracle R Distribution or open source R.

Related Topics

- Installing and Configuring the Database for Oracle R Enterprise (page 2-1)
- Installing R for Oracle R Enterprise (page 3-1)
  This chapter explains how to install R for Oracle R Enterprise.

4.3.1 System Requirements

- The operating system must conform to the requirements specified in Oracle R Enterprise System Requirements (page 1-5).
- Oracle Database must be installed and configured as described in Installing and Configuring the Database for Oracle R Enterprise (page 2-1).

> **Note:**
>
> You can install Oracle R Enterprise Server in a pluggable database (PDB) in a multitenant environment. See *Oracle Database Administrator's Guide*

- R must be installed as described in Installing R for Oracle R Enterprise (page 3-1).

4.3.2 Environment Variables

Table 4-2 Environment Variable Requirements for Oracle R Enterprise Server

Platform	Environment Variable Requirement
all	**$ORACLE_SID** must specify the service identifier (SID) of the database that will support Oracle R Enterprise. **$ORACLE_HOME** must specify the home directory of the database identified by `ORACLE_SID`. On Windows, you can find the value of Oracle home and the Oracle instance identifier in the Windows Registry. If more than one Oracle home or Oracle instance exist on this computer, then you can specify the required values in environment variables. See Creating and Modifying Environment Variables on Windows (page 8-8).
Linux	**$LD_LIBRARY_PATH** must include `$ORACLE_HOME/lib`. **$PATH** must include `$ORACLE_HOME/bin`.
Oracle Solaris	**$LD_LIBRARY_PATH** must include `$ORACLE_HOME/lib`. **$PATH** must include `$ORACLE_HOME/bin`.
IBM AIX	**$LIBPATH** must include `$ORACLE_HOME/lib`. **$PATH** must include `$ORACLE_HOME/bin`.
Microsoft Windows	**%PATH%** must include `%R_HOME%\bin\x64`. The default value of `%R_HOME%` is `C:\Program Files\R\R-3.2.0`. You can find the value of the R home directory in the Windows Registry. If more than one R home exist on this computer, then you can specify the required value in an environment variable. See Creating and Modifying Environment Variables on Windows (page 8-8).

4.3.3 User Requirements

The operating system user that installs Oracle R Enterprise Server must meet the requirements described in this section.

Table 4-3 User Requirements for Oracle R Enterprise Server Installer

Platform	User Requirement
Linux and UNIX	• Member of the `dba` group • Has write access to `$ORACLE_HOME/lib`
Microsoft Windows	• Administrator access • Member of the `ora_dba` group • Has write access to `%ORACLE_HOME%\bin`

4.3.3.1 About Operating System Authentication

The Oracle R Enterprise Server installation script uses **system authentication** to connect to the database identified by `ORACLE_HOME` and `ORACLE_SID`. System authentication is based on the operating system credentials of the user instead of the database credentials.

For example, on a Linux system, the Oracle R Enterprise installation script uses this statement to start SQL*Plus without a password:

```
$ORACLE_HOME/bin/sqlplus / as sysdba
```

Membership in a special operating system group enables system authentication for Oracle Database. The operating system group is created during installation of the database, and the identity of the installer is automatically assigned to the group. The generic name for the group is OSDBA. On Linux and UNIX, the name for OSDBA is `dba`. On Windows, the name for OSDBA is `ora_dba`.

The user that installs Oracle R Enterprise Server must belong to OSDBA.

4.3.3.2 Verifying the Group Membership of Your User ID

As described in "About Operating System Authentication (page 4-8)", the Linux or UNIX user ID that runs the Oracle R Enterprise Server installation script must belong to the `dba` group. Membership in the `dba` group is also required for running other Oracle R Enterprise scripts on the server. On Windows, the `dba` group is called `ora_dba`.

To determine the group membership of your Linux or UNIX user ID, type this command:

```
% groups
dba  othergroup
```

To determine the group membership of your Windows user ID:

1. Open Windows Control Panel.
2. Select **Users Accounts**.
3. Select **Manage User Accounts**.

4. On the Users tab of the User Accounts dialog, the name, domain, and group of each user account are listed. Verify that your user ID belongs to the group `ora_dba`.

4.4 Installing Oracle R Enterprise Server

Follow these steps to install Oracle R Enterprise Server.

> **Note:**
>
> To install Oracle R Enterprise Server without needing to respond to visual prompts, use a batch mode installation such as that described in "A Default Batch Installation (page 4-6)" and "User Configuration in Batch Mode (page 4-6)".

To install Oracle R Enterprise Server:

1. Ensure that your system satisfies the requirements specified in Oracle R Enterprise Server Requirements (page 4-6).
2. Create an installation directory for the Oracle R Enterprise server components. The directory can have any name. For example:

   ```
   /oreserver_install_dir
   ```

3. Download the Oracle R Enterprise Server installation files and supporting packages from the Oracle R Enterprise Downloads page on the Oracle Technology Network.
   a. Accept the license agreement and download the Oracle R Enterprise **Server** files for your platform to your installation directory.
   b. Accept the license agreement and download the Oracle R Enterprise **Supporting** packages for your platform to your installation directory.

   The installation directory now contains two zip files.

   ```
   ore-server-platform-arch-version.zip
   ore-supporting-platform-arch-version.zip
   ```

4. Unzip the files.

   ```
   unzip ore-server-platform-arch-version.zip
   unzip ore-supporting-platform-arch-version.zip
   ```

   The installation directory looks like this after you unzip both files:

   For Linux or UNIX:

   ```
   /oreserver_install_dir
        ore-server-platform-arch-version.zip
        ore-supporting-platform-arch-version.zip
        server.sh
        /server
        /supporting
   ```

   For Windows:

   ```
   \oreserver_install_dir
        ore-server-platform-arch-version.zip
   ```

```
ore-supporting-platform-arch-version.zip
server.bat
\server
\supporting
```

5. For Linux or UNIX, run `server.sh`. For Windows, run `server.bat`. The script performs a default, first-time installation of Oracle R Enterprise Server, as described in A Default Interactive Installation (page 4-5).

   For Linux or UNIX:

   ```
   ./server.sh
   ```

   For Windows:

   ```
   server.bat
   ```

> **Note:**
>
> Beginning in R-3.3.0, on Oracle Linux 6 system, it is necessary to set `LD_LIBRARY_PATH` in `$ORACLE_HOME/hs/admin/extproc.ora` to the location of the R-core-extra RPM so that these libraries are found by the Oracle process running `extproc`. On Linux systems, the default location of the R-core-extra RPM is `/usr/lib64/R/port/Linux-X64/lib`. In `extproc.ora`, enter:
>
> ```
> SET LD_LIBRARY_PATH=/usr/lib64/R/port/Linux-X64/lib
> ```
>
> For changes in `extproc.ora` to take effect, you must stop and restart the database.
>
> This procedure is not necessary on a Linux 7 system, as the required versions of the libraries provided by the R-core-extra RPM are available natively..

> **See Also:**
>
> Example A-1 (page A-4) for an example with output

About the R-core-extra RPM

R has always depended on several third party libraries, specifically, `zlib`, `bzip2`, `xz`, `pcre`, and `curl`. Prior to R-3.3.0, R depended on much older versions of these libraries, but, if they were not found on the system, bundled copies were included that were built on the fly.

R-3.3.0 depends on much newer versions of these libraries and no longer contains the bundled copies. This means that R 3.3.0 won't build against Linux 6 as is, because the native versions of these libraries are older than those that R-3.3.0 requires.

The R-core-extra RPM contains the required versions of these libraries and is provided as a convenience for users of Oracle Linux 6. Adding the location of the libraries in R-core-extra to `LD_LIBRARY_PATH` removes the need to built these libraries separately. Oracle Linux 7 introduces the required versions of these libraries, but the R-core-extra RPM is provided as a convenience if needed.

4.5 Verifying the Oracle R Enterprise Server Installation

The Oracle R Enterprise server installation script creates log files in the server subdirectory of the installation directory. Examine the log files to verify the success of the installation process.

The following commands on a Linux or UNIX system list the log files:

```
cd ./oreserver_install_dir/server
ls *.log
outcdb.log  rqconfig.log  rqdrop.log  rqgrant.log  rqinst.log  rqpdrp.log
rqproc.log  rquser.log
```

If there are problems with the installation and you are unable to resolve them, you can request help from My Oracle Support or from the Oracle R Enterprise Forum.

4.5.1 Validating Basic Oracle R Enterprise Functionality

After connecting to the database, test the basic functionality of Oracle R Enterprise with these commands.

First execute these commands from an R instance directly on the database server and then execute them from the Oracle R Enterprise client.

```
## Is the ORE client connected to the ORE server?
## The output of this command should be TRUE.
ore.is.connected()

## List the available database tables
ore.ls()

## Push an R dataframe to a database table
CARS <- ore.push(cars)
head(CARS)

## Run embedded R
ore.doEval(function(){library(ORE)})
```

4.5.2 Running the Oracle R Enterprise Example Scripts

To fully validate the success of the installation, run the suite of Oracle R Enterprise demo scripts.

In a successful installation, all demo scripts run to completion without errors.

The example scripts are located in `$ORACLE_HOME/R/library/ORE/demo`.

This R command provides a list of available examples:

```
demo(package="ORE")
```

These commands run two of the examples. The `aggregate` script tests the use of an R function on data that is resident in database memory; the `row_apply` script tests embedded R execution.

```
demo("aggregate", package="ORE")
demo("row_apply", package="ORE")
```

This command exits from R.

```
q()
```

4.6 Installing Oracle R Enterprise Server in a Multitenant Environment

You can install Oracle R Enterprise Server in one or more pluggable databases (PDBs) within a multitenant environment. Oracle R Enterprise Server must be installed in a pluggable database, not in the root database.

If you have more than one instance of Oracle R Enterprise Server installed in a Multitenant Container Database (CDB) and you want to uninstall one instance but retain the others, you can perform a partial uninstall as described in Performing a Partial Uninstall (page 8-5).

5

Installing Oracle R Enterprise on Exadata

This chapter explains how to install Oracle R Distribution and Oracle R Enterprise Server on Oracle Exadata Database Machine. This chapter includes these topics:

5.1 About Oracle R Enterprise on Exadata

Exadata is an ideal platform for Oracle R Enterprise.

The parallel resources of R computations in Oracle R Enterprise take advantage of the massively parallel grid infrastructure of Exadata.

To install Oracle R Enterprise on Exadata:

1. On *each* node:
   - Install Oracle R Distribution
   - Verify and configure the environment
   - Install Oracle R Enterprise Server and the supporting packages
2. On the *first* node only, create a user.

You can simplify the process of installing Oracle R Enterprise on Exadata by using the **Distributed Command Line Interface** (DCLI).

Related Topics

- Installing R for Oracle R Enterprise (page 3-1)
- Oracle R Enterprise Server Requirements (page 4-6)
- Installing Oracle R Enterprise Server (page 4-9)
- Creating a Database User for Oracle R Enterprise (page 8-6)
- Using DCLI to Install Oracle R Enterprise on Exadata (page 5-1)

5.2 Using DCLI to Install Oracle R Enterprise on Exadata

You can use DCLI to simplify the installation of Oracle R Enterprise on Exadata.

With DCLI, you can use a single command to install Oracle R Distribution and Oracle R Enterprise Server across multiple Exadata compute nodes. The following example shows the output of the DCLI help option, which explains the basic syntax of the utility.

> **See Also:**
>
> For more details about DCLI, go to the My Oracle Support website, log in with your Customer Support Identifier, and type `DCLI` in the search box.

Example 5-1 DCLI Help Option Output

```
$ dcli -h

Distributed Shell for Oracle Storage

This script executes commands on multiple cells in parallel threads.
The cells are referenced by their domain name or ip address.
Local files can be copied to cells and executed on cells.
This tool does not support interactive sessions with host applications.
Use of this tool assumes ssh is running on local host and cells.
The -k option should be used initially to perform key exchange with
cells.  User may be prompted to acknowledge cell authenticity, and
may be prompted for the remote user password.  This -k step is serialized
to prevent overlayed prompts.  After -k option is used once, then
subsequent commands to the same cells do not require -k and will not require
passwords for that user from the host.
Command output (stdout and stderr) is collected and displayed after the
copy and command execution has finished on all cells.
Options allow this command output to be abbreviated.

Return values:
 0 -- file or command was copied and executed successfully on all cells
 1 -- one or more cells could not be reached or remote execution returned
      non-zero status.
 2 -- An error prevented any command execution

Examples:
 dcli -g mycells -k
 dcli -c stsd2s2,stsd2s3 vmstat
 dcli -g mycells cellcli -e alter iormplan active
 dcli -g mycells -x reConfig.scl

usage: dcli [options] [command]

options:
 --version           show program's version number and exit
 -c CELLS            comma-separated list of cells
 -d DESTFILE         destination directory or file
 -f FILE             file to be copied
 -g GROUPFILE        file containing list of cells
 -h, --help          show help message and exit
 -k                  push ssh key to cell's authorized_keys file
 -l USERID           user to login as on remote cells (default: celladmin)
 -n                  abbreviate non-error output
 -r REGEXP           abbreviate output lines matching a regular expression
 -s SSHOPTIONS       string of options passed through to ssh
 --scp=SCPOPTIONS    string of options passed through to scp if different from
                     sshoptions
 --serial            serialize execution over the cells
 -t                  list target cells
 --unkey             drop keys from target cells' authorized_keys file
 -v                  print extra messages to stdout
```

```
--vmstat=VMSTATOPS  vmstat command options
-x EXECFILE         file to be copied and executed
```

5.3 Installing Oracle R Distribution Across Exadata Compute Nodes

This section explains how to run DCLI to install Oracle R Distribution across multiple Exadata Linux compute nodes.

The commands are summarized in DCLI Command Summary for Oracle R Distribution installation on Exadata (page 5-5).

> **Important:**
>
> Before beginning the installation, review the instructions for installing Oracle R Distribution in Installing R for Oracle R Enterprise (page 3-1).

To install Oracle R Distribution on Exadata using DCLI, follow these steps:

1. Configure the Exadata environment to enable automatic authentication for DCLI on each compute node.
   a. Generate an SSH public-private key for the root user. Execute the following command as root on any node:

      ```
      $ ssh-keygen -N '' -f /.ssh/id_dsa -t dsa
      ```

      This command generates public and private key files in the `.ssh` subdirectory of the home directory of the root user.

   b. In a text editor, create a file that contains the names of all the compute nodes in the rack. Specify each node name on a separate line. For example, the `nodes` file for a 2-node cluster could contain entries like the following:

      ```
      $ cat nodes
      exadb01
      exadb02
      ```

   c. Run the DCLI command with the `-k` option to establish SSH trust across all the nodes. The `-k` option causes DCLI to contact each node sequentially (not in parallel) and prompts you to enter the password for each node.

      ```
      $ dcli -t -g nodes -l root -k -s "\-o StrictHostkeyChecking=no"
      ```

      DCLI with `-k` establishes SSH Trust and User Equivalence. Subsequent DCLI commands will not prompt for passwords.

2. Request the file `ord-linux-x86_64-`*`Rversion`*`-`*`Exadataversion`*`.tar.gz`from Oracle Support , where *Rversion* is the version of Oracle R Distribution to install and *Exadataversion* is your Exadata version number. Install Oracle R Distribution using yum or, if an internet connection is unavailable, by installing the Oracle R Distribution RPMs manually.
   a. Log in to My Oracle Support.
   b. Click **Contact Us**.

c. Request permission to access this file:

   `ord-linux-x86_64-Rversion-Exadataversion.tar.gz`

d. When permission is granted, log in as root to any compute node and download the file.

3. Create a directory and replicate the downloaded file in this directory across all nodes. For example, the following commands create the directory `/home/oracle/ORD` and replicate the file `ord-linux-x86_64-Rversion-Exadataversion.tar.gz` in this directory.

```
$ dcli -t -g nodes -l root mkdir -p /home/oracle/ORD
$ dcli -t -g nodes -l root -f
        ord-linux-x86_64-Rversion-Exadataversion.tar.gz -d
        /home/oracle/ORD/ord-linux-x86_64-Rversion-Exadataversion.tar.gz
```

4. Uncompress and untar the file to replicate the dependent RPMs across all nodes.

```
$ dcli -t -g nodes -l root tar xvfz
        /home/oracle/ORD/ord-linux-x86_64-Rversion-Exadataversion.tar.gz
        -C /home/oracle/ORD
$ ls /home/oracle/ORD/ord-linux-x86_64-Rversion-Exadataversion.tar.gz
```

   Alternatively, you can download these RPMs from the Oracle public yum server. The locations of the RPMs are listed in "Installing Oracle R Distribution on Oracle Linux Using RPMs (page 3-6)".

5. To install the new RPMs and update existing RPMs across nodes, execute the following RPM command:

```
$ dcli -t -g nodes -l root rpm -i --force
        /home/oracle/ORD/ord-linux-x86_64-Rversion-Exadataversion/*.rpm
```

   The `--force` flag prevents errors from circular dependencies.

6. Verify the R installations on each node by first returning to the location where R is installed and then starting R.

```
$ dcli -g nodes -l oracle R RHOME
exadb01: /usr/lib64/R
exadb02: /usr/lib64/R
```

   The following command returns this output for each node.

```
$ dcli -g nodes -l oracle R --vanilla
...
exadb01: R is free software and comes with ABSOLUTELY NO WARRANTY.
exadb01: You are welcome to redistribute it under certain conditions.
exadb01: Type 'license()' or 'licence()' for distribution details.
exadb01:
exadb01: Natural language support but running in an English locale
exadb01:
exadb01: R is a collaborative project with many contributors.
exadb01: Type 'contributors()' for more information and
exadb01: 'citation()' on how to cite R or R packages in publications.
exadb01:
exadb01: Type 'demo()' for some demos, 'help()' for on-line help, or
exadb01: 'help.start()' for an HTML browser interface to help.
exadb01: Type 'q()' to quit R.
exadb01:
exadb01: You are using Oracle's distribution of R. Please contact
```

```
exadb01: Oracle Support for any problems you encounter with this
exadb01: distribution.
```

5.3.1 DCLI Command Summary for Oracle R Distribution installation on Exadata

The DCLI commands used to install Oracle R Distribution on a Linux Exadata system are listed in the following example.

Replace `version` with the version number of the Oracle R Distribution that you are using.

Example 5-2 DCLI Command Summary for Oracle R Distribution

```
ssh-keygen -N " -f ~/.ssh/id_dsa -t dsa
vi nodes # enter node names
dcli -t -g nodes -l root -k -s "\-o StrictHostkeyChecking=no"
dcli -t -g nodes -l root mkdir -p /home/oracle/ORD
dcli -t -g nodes -l root -f ord-linux-x86_64-version.tar.gz -d
          /home/oracle/ORD/ord-linux-x86_64-version.tar.gz
dcli -t -g nodes -l root tar xvfz /home/oracle/ORD
          /ord-linux-x86_64-version.tar.gz -C /home/oracle/ORD
dcli -t -g nodes -l root rpm -i --force
          /home/oracle/ORD/ord-linux-x86_64-version/*.rpm
dcli -g nodes -l root R RHOME
dcli -g nodes -l root R --vanilla
```

5.4 Installing Oracle R Enterprise Server Across Exadata Compute Nodes

This section explains how to run DCLI to install Oracle R Enterprise Server across multiple Exadata Linux compute nodes.

The commands are summarized in DCLI Commands Summary for Oracle R Enterprise Server (page 5-7).

> **Important:**
>
> Before beginning the installation, review the instructions for installing Oracle R Enterprise Server in Installing Oracle R Enterprise Server (page 4-1).

To install Oracle R Enterprise Server on Exadata using DCLI, follow these steps:

1. Ensure that the `ORACLE_HOME`, `ORACLE_SID`, `R_HOME`, `PATH`, and `LD_LIBRARY_PATH` environment variables are properly set on each node, and are defined in the same shell where the DCLI script will run. For example, you could specify values like the following in a bashrc file:

   ```
   export ORACLE_HOME=/hostname/app/oracle/product/release_number/dbhome_1
   export ORACLE_SID=ORCL
   export R_HOME=/usr/lib64/R
   export PATH=$PATH:$R_HOME/bin:$ORACLE_HOME/bin
   ```

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib:$RHOME_lib:$R_HOME/port/
Linux-X64/lib
```

2. Go to the Oracle R Enterprise page on the Oracle Technology Network.

   Select **Oracle R Enterprise Downloads**. On the Downloads page, select Oracle R Enterprise Server and the Supporting Packages for Linux. The following files are downloaded for Oracle R Enterprise, where *version* is the Oracle R Enterprise, release number.

   ```
   ore-server-linux-x86-64-version.zip
   ore-supporting-linux-x86-64-version.zip
   ```

3. Log in as root, and copy the installers for Oracle R Enterprise Server and the supporting packages across nodes. For example:

   ```
   $ dcli -g nodes -l oracle mkdir -p /home/oracle/ORE
   $ dcli -g nodes -l oracle -f ore-server-linux-x86-64-version.zip -d
        /home/oracle/ORE/ore-server-linux-x86-64-version.zip
   $ dcli -g nodes -l oracle -f ore-supporting-linux-x86-64-version.zip -d
        /home/oracle/ORE/ore-supporting-linux-x86-64-version.zip
   ```

4. Unzip the Oracle R Enterprise Server bundle on each node:

   ```
   $ dcli -t -g nodes -l oracle unzip
        /home/oracle/ORE/ore-server-linux-x86-64-version.zip -d
        /my_destination_directory/
   ```

5. Unzip the supporting packages on each node:

   ```
   $ dcli -t -g nodes -l oracle unzip
        /home/oracle/ORE/ore-supporting-linux-x86-64-version.zip -d
        /my_destination_directory/
   ```

6. Install Oracle R Enterprise server components:

   ```
   $ dcli -t -g nodes -l oracle "cd /my_destination_directory; ./server.sh -y
         --admin --sys syspassword --perm permtablespace
         --temp temptablespace --rqsys rqsyspassword
         --user-perm usertablespace --user-temp usertemptablespace
         --pass rquserpassword --user RQUSER"
   ```

   > **Note:**
   >
   > The `server` script creates a user for Oracle R Enterprise. By default, the script does *not* grant the `rqadmin` role to the user.
   >
   > Any Oracle R Enterprise user can execute embedded R, but only those with the `rqadmin` role can create and drop the R scripts in the database. Use caution when granting the `rqadmin` role.
   >
   > For more information about the role, see About the RQADMIN Role (page 8-7).

7. Verify Oracle R Enterprise loads.

   ```
   > library(ORE)
   Loading required package: OREbase
   Attaching package: OREbase
   The following objects are masked from âpackage:baseâ:
       cbind, data.frame, eval, interaction, order, paste, pmax, pmin,
       rbind, table
   ```

```
Loading required package: OREembed
Loading required package: OREstats
Loading required package: MASS
Loading required package: OREgraphics
Loading required package: OREeda
Loading required package: OREmodels
Loading required package: OREdm
Loading required package: lattice
Loading required package: OREpredict
Loading required package: ORExml
```

5.4.1 DCLI Commands Summary for Oracle R Enterprise Server

The DCLI commands used to install Oracle R Enterprise Server and the supporting packages on a Linux Exadata system are listed in the following example.

Example 5-3 DCLI Command Summary for Oracle R Enterprise Server

```
dcli -g nodes -l oracle mkdir -p /home/oracle/ORE
dcli -g nodes -l oracle -f ore-server-linux-x86-64-version.zip -d
     /home/oracle/ORE/ore-server-linux-x86-64-version.zip
dcli -g nodes -l oracle -f ore-supporting-linux-x86-64-version.zip -d
     /home/oracle/ORE/ore-supporting-linux-x86-64-version.zip
dcli -t -g nodes -l oracle unzip
     /home/oracle/ORE/ore-server-linux-x86-64-version.zip -d
     /home/oracle/ORE/
dcli -t -g nodes -l oracle /home/oracle/ORE/server.sh
sqlplus / as sysdba
grant RQADMIN to rquser;
exit;
dcli -t -g nodes -l oracle ORE -e "library(ORE)"
```

Related Topics

- Security Best Practices for Oracle R Enterprise (page 4-2)
  To minimize the risk of compromising the security of an Oracle R Enterprise Server in Oracle Database, Oracle recommends the following security best practices.

6
Installing Oracle R Enterprise Client

This chapter explains how to install Oracle R Enterprise Client. This chapter includes these topics:

6.1 About Oracle R Enterprise Client

Oracle R Enterprise includes several components that must be installed separately on each client computer.

Components of Oracle R Enterprise Client

- R (See Installing R for Oracle R Enterprise (page 3-1))
- Oracle Database Client Software
- Oracle R Enterprise packages
- Oracle R Enterprise supporting packages

The Oracle R Enterprise client components can be installed in any order:

6.1.1 About Oracle Database Client Software

ROracle, one of the supporting packages used by Oracle R Enterprise, requires an installation of Oracle Database client software to enable communication between an R client and an Oracle database. The Database client can be either Oracle Database Client or Oracle Database Instant Client:

- **Oracle Database Client** is distributed with Oracle Database and is based in the Oracle home of the database.
- **Oracle Database Instant Client** is a free, standalone implementation of Oracle Database Client. Oracle Instant Client is not based in an Oracle home directory and requires less disk space than Oracle Database Client.

6.1.2 About the Oracle R Enterprise Packages

The Oracle R Enterprise packages are a set of Oracle proprietary packages that support Oracle R Enterprise.

These packages are required on each client computer and on the server computer that hosts Oracle R Enterprise Server. On the server, the Oracle R Enterprise packages are installed automatically by the Oracle R Enterprise Server installation script.

> **Note:**
>
> The version of the Oracle R Enterprise packages on the client must match the version of the Oracle R Enterprise packages on the server.

Table 6-1 Oracle R Enterprise Packages

Package Name	Description
`ORE`	The top-level package for Oracle R Enterprise.
`OREbase`	Corresponds to the open source R `base` package.
`OREcommon`	Contains common low-level functionality for Oracle R Enterprise.
`OREdm`	Exposes Oracle Data Mining algorithms through R.
`OREdplyr`	Transparently implements `dplyr` data manipulation functions for `ore.frame` and `ore.numeric` objects.
`OREeda`	Contains functions for exploratory data analysis.
`OREembed`	Supports embedded R.
`OREgraphics`	Corresponds to the open source R `graphics` package.
`OREmodels`	Contains functions for advanced analytical modeling.
`OREpredict`	Enables scoring data in Oracle Database using R models.
`OREstats`	Corresponds to the open source R `stats` package.
`ORExml`	Supports XML translation between R and Oracle Database.

6.1.3 About the Oracle R Enterprise Supporting Packages

The supporting packages are a set of open source packages that support the Oracle R Enterprise packages.

Table 6-2 Oracle R Enterprise Supporting Packages

Package Name	Description
`arules`	Provides the infrastructure for representing, manipulating, and analyzing transactional data and patterns (frequent itemsets and association rules).
`Cairo`	Supports graphic rendering on Oracle R Enterprise server.
`DBI`	A database interface definition for communication between R and Oracle Database.
`png`	Supports the reading and writing of PNG images for Oracle R Enterprise objects.
`randomForest`	Provides classification and regression based on a forest of trees using random inputs.
`ROracle`	Oracle Database interface for R-based OCI.

Table 6-2 (Cont.) Oracle R Enterprise Supporting Packages

Package Name	Description
`statmod`	Provides statistical modeling functions, including growth curve comparisons, limiting dilution analysis, mixed linear models, heteroscedastic regression, Tweedie family generalized linear models, the inverse-Gaussian distribution and Gauss quadrature.

6.2 Installing Oracle Database Instant Client

Oracle R Enterprise requires Oracle Database client software.

Oracle Instant Client is suitable for most configurations of Oracle R Enterprise.

This topic includes these sections:

- Installing Oracle Database Instant Client on Windows (page 6-3)
- Installing Oracle Database Instant Client on Linux or UNIX (page 6-4)

Related Topics

- About Oracle Database Client Software (page 6-1)

6.2.1 Installing Oracle Database Instant Client on Windows

To Install Oracle Instant Client on Windows:

1. Create an installation directory for the Oracle R Enterprise client components. For example:

   ```
   c:\oreclient_install_dir
   ```

2. Go to the Oracle Database Instant Client page on the Oracle Technology Network.
3. Select **See Instant Client downloads for all platforms**.
4. On the Instant Client Downloads page, select **Instant Client for Microsoft Windows (x64)**.
5. On the Instant Client Downloads for Microsoft Windows (x64) page, accept the license agreement.
6. Choose **Instant Client Package - Basic** for your version of Oracle Database.
7. Save the file in the installation directory that you created in Step 1. For example:

   ```
   c:\oreclient_install_dir\instantclient-basic-windows.x64-12.1.0.2.0.zip
   ```

8. Unzip the file. The files are extracted into a subdirectory called `instantclient_`*`version`*, where *`version`* is your version of Oracle Database. For example:

   ```
   c:\oreclient_install_dir\instantclient_12_1
   ```

9. Return to the Instant Client Downloads for Microsoft Windows (x64) page.
10. Accept the license agreement and select **Instant Client Package - SDK** for your version of Oracle Database.
11. Save the file in the installation directory that you created in Step 1.

```
c:\oreclient_install_dir\instantclient-sdk-windows.x64-12.1.0.2.0.zip
```

12. Unzip the file. The files are extracted into the `instantclient_version` subdirectory.
13. Add the full path of the Instant Client to the environment variables `OCI_LIB64` and `PATH`. The following steps set the variables to the path used in this example, `c:\myoreclient\instantclient_12_1`:
    a. In Windows Control Panel, choose **System**, then click **Advanced system settings**.
    b. On the **Advanced** tab, click **Environment Variables**.
    c. Under **System variables**, create `OCI_LIB64` if it does not already exist. Set the value of `OCI_LIB64` to `c:\oreclient\instantclient_12_1`.
    d. Under **System variables**, edit `PATH` to include `c:\oreclient\instantclient_12_1`.

6.2.2 Installing Oracle Database Instant Client on Linux or UNIX

You can install Oracle Database Instant Client from a zip file on Linux or UNIX systems.

On Linux, you can also install from RPMs.

This topic includes these sections:

6.2.2.1 Installing Oracle Instant Client from a Zip File

1. Create an installation directory for the Oracle R Enterprise client components. For example:

   ```
   mkdir oreclient_install_dir
   ```

2. Go to the Oracle Database Instant Client page on the Oracle Technology Network:
3. Select **See Instant Client downloads for all platforms**. On the Instant Client Downloads page, select the Instant Client for your platform.
4. Accept the license agreement and select the **Instant Client Package - Basic** RPM for your version of Oracle Database.
5. Save the file in the installation directory that you created in Step 1. For example:

   ```
   \oreclient_install_dir\instantclient-basic-linux.x64-12.1.0.2.0.zip
   ```

6. Unzip the file. The files are extracted into a subdirectory called `instantclient_version`, where *version* is your version of Oracle Database. For example:

   ```
   unzip instantclient-basic-linux.x64-12.1.0.2.0.zip
   ls
     instantclient_12_1/
     instantclient-basic-linux.x64-12.1.0.2.0.zip
   ```

7. Return to the Oracle Database Instant Client page for your platform.
8. Select the Instant Client for your platform.

9. On the Instant Client Downloads page for your platform, accept the license agreement and select **Instant Client Package - SDK** for your version of Oracle Database.
10. Save the file in the installation directory that you created in Step 1. For example:

```
\oreclient_install_dir\instantclient-sdk-linux.x64-12.1.0.2.0.zip
```

11. Unzip the file. The contents are extracted into the `instantclient_version` subdirectory.

```
unzip instantclient-sdk-linux.x64-12.1.0.2.0.zip
ls
  /instantclient_12_1
  instantclient-basic-linux.x64-12.1.0.2.0.zip
  instantclient-sdk-linux.x64-12.1.0.2.0.zip
cd instantclinet_12_1
ls
  /help
  /sdk
  /vc10
  /vc11
```

6.2.2.2 Installing Oracle Instant Client on Linux from RPMs

1. Create an installation directory for the Oracle R Enterprise client components. For example:

```
mkdir oreclient_install_dir
```

2. Go to the Oracle Database Instant Client page on the Oracle Technology Network:
3. Choose **See Instant Client downloads for all platforms**.
4. On the Instant Client Downloads page, choose **Instant Client for Linux x86-64**.
5. On the Instant Client Downloads page for Linux, accept the license agreement and select the RPM for **Instant Client Package - Basic**.
6. As the root user, install the RPM:

```
rpm -i oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
```

7. Return to the Instant Client Downloads page for Linux x86-64.
8. Accept the license agreement and download the RPM for **Instant Client Package - SDK** for your version of Oracle Database. As root, install the RPM:

```
rpm -i oracle-instantclient12.1-sdk-12.1.0.2.0-1.x86_64.rpm
```

9. The RPMs place the files in standard locations that the ROracle configuration script can find. For example, Oracle Instant Client 12.1 is installed in `/usr/lib/oracle/12.1/client64/lib`.
10. After installing Oracle Instant Client, add the path of the Oracle Instant Client libraries to `LD_LIBRARY_PATH`. For example:

```
export LD_LIBRARY_PATH=/usr/lib/oracle/12.1/client64/lib:$LD_LIBRARY_PATH
```

6.3 Installing the Oracle R Enterprise Packages

Install the Oracle R Enterprise packages on each client computer.

The Oracle R Enterprise packages are automatically included in the installation on the server.

This topic includes these sections:

6.3.1 Installing the Oracle R Enterprise Packages on Windows

1. Download the Oracle R Enterprise packages from the Oracle R Enterprise Downloads page on the Oracle Technology Network.
2. Accept the license agreement and select the Oracle R Enterprise packages for your platform. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:

   ```
   c:\oreclient_install_dir\ore-client-win-x86_64-1.5.1.zip
   ```

   NOTE: Choose the same installation directory for all Oracle R Enterprise client components.
3. Unzip the file. The contents are extracted into the `client` subdirectory:

   The resulting installation directory, shown in the example at the end of this section, contains Oracle Instant Client and the Oracle R Enterprise packages.
4. Choose one of the following methods to install the Oracle R Enterprise packages on Windows:
   - **Install from the R Console**
     a. Start R x64 from the Windows Start menu.
     b. Execute this R command for each zip file in the `client` directory:

        ```
        install.packages("oreclient_install_dir/client/client_package_name.zip',
        repos=NULL)
        ```

        Each successful package installation produces this message in the R console:

        ```
        package 'package_name' successfully unpacked and MD5 sums checked
        ```
   - **Install from the R GUI**
     a. Start R x64 from the Windows Start menu.
     b. Select **Packages** from the **RGui (64-bit)** menu bar.
     c. From the **Packages** menu, select **Install package(s) from local zip files**.
     d. Change to the `client` directory.
     e. Select all the files in the directory.
     f. Click **Open**.

        Each package installation produces this message in the R console:

        ```
        package 'package_name' successfully unpacked and MD5 sums checked
        ```
   - **Install from the Windows command prompt**
     a. Start R x64 from the Windows Start menu.

b. Open a Windows command window.

c. Change directory to the `client` directory and type these commands:

```
R CMD INSTALL OREbase_1.5.1.zip
R CMD INSTALL OREcommon_1.5.1.zip
R CMD INSTALL OREstats_1.5.1.zip
R CMD INSTALL OREgraphcs_1.5.1.zip
R CMD INSTALL OREeda_1.5.1.zip
R CMD INSTALL OREembed_1.5.1.zip
R CMD INSTALL ORExml_1.5.1.zip
R CMD INSTALL OREdm_1.5.1.zip
R CMD INSTALL OREdplyr_1.5.1.zip
R CMD INSTALL OREmodels_1.5.1.zip
R CMD INSTALL OREpredict_1.5.1.zip
R CMD INSTALL ORE_1.5.1.zip
```

Each package installation generates this message:

```
package 'package_name' successfully unpacked and MD5 sums checked
```

Example 6-1 Client Installation Directory Containing Client Packages and Instant Client

```
c:\oreclient_install_dir
     \client
          \ORE_1.5.1.zip
          \OREbase_1.5.1.zip
          \OREcommon_1.5.1.zip
          \OREdm_1.5.1.zip
          \OREdplyr_1.5.1.zip
          \OREeda_1.5.1.zip
          \OREembed_1.5.1.zip
          \OREgraphics_1.5.1.zip
          \OREmodels_1.5.1.zip
          \OREpredict_1.5.1.zip
          \OREstats_1.5.1.zip
          \ORExml_1.5.1.zip
     \instantclient_12_1
instantclient-basic-linux.x64-12.1.0.2.0.zip
instantclient-sdk-linux.x64-12.1.0.2.0.zip
ore-client-win-x86_64-1.5.1.zip
```

6.3.2 Installing the Oracle R Enterprise Packages on Linux or UNIX

1. Download the Oracle R Enterprise packages from the Oracle R Enterprise Downloads page on the Oracle Technology Network.
2. Accept the license agreement and select the Oracle R Enterprise packages for your platform. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:

   ```
   /oreclient_install_dir/ore-client-platform-arch-version.zip
   ```

   NOTE: Choose the same installation directory for all Oracle R Enterprise client components.
3. Unzip the file:

   ```
   % unzip ore-client-platform-arch-version.zip
   ```

When you unzip the file, the `/client` directory is created and these files are extracted.

```
/client/ORE_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREbase_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREcommon_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREdm_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREdplyr_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREeda_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREembed_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREgraphics_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREmodels_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREpredict_version_R_arch-unknown-platform-gnu.tar.gz
/client/OREstats_version_R_arch-unknown-platform-gnu.tar.gz
/client/ORExml_version_R_arch-unknown-platform-gnu.tar.gz
```

4. Change to */oreclient_install_dir*/client.
5. Execute the following commands:

```
R CMD INSTALL ORE_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREbase_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREcommon_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREdm_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREdplyr_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREeda_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREembed_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREgraphics_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREmodels_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREpredict_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL OREstats_version_R_arch-unknown-platform-gnu.tar.gz
R CMD INSTALL ORExml_version_R_arch-unknown-platform-gnu.tar.gz
```

6.4 Installing the Oracle R Enterprise Supporting Packages

Install the Oracle R Enterprise supporting packages on each client computer and on the server that hosts Oracle R Enterprise Server.

This topic includes these sections:

- Installing the Supporting Packages on Windows (page 6-8)
- Installing the Supporting Packages on Linux or UNIX (page 6-10)

6.4.1 Installing the Supporting Packages on Windows

1. Download the supporting packages from the Oracle R Enterprise Downloads page on the Oracle Technology Network.
2. Accept the license agreement and select the **Supporting** packages for your platform. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:

   ```
   c:\oreclient_install_dir\ore-supporting-win-x86_64-1.5.1.zip
   ```

   NOTE: Choose the same installation directory for all Oracle R Enterprise client components.
3. Unzip the file. The contents are extracted into the `supporting` subdirectory:

The resulting installation directory, shown in the example at the end of this section, contains all the client components: Oracle Instant Client, Oracle R Enterprise packages, and Oracle R Enterprise supporting packages.

4. Choose one of the following methods to install the supporting packages on Windows:

   - **Install from the R Console**

     a. Start R x64 from the Windows Start menu.

     b. Execute this R command for each zip file in the `client` directory:

     ```
     install.packages("oreclient_install_dir/support/
     support_package_name.zip', repos=NULL)
     ```

     Each successful package installation produces this message in the R console:

     ```
     package 'package_name' successfully unpacked and MD5 sums checked
     ```

   - **Install from the R GUI**

     a. Start R x64 from the Windows Start menu.

     b. Select **Packages** from the **RGui (64-bit)** menu bar.

     c. From the **Packages** menu, select **Install package(s) from local zip files**.

     d. Change to the `support` directory.

     e. Select all the files in the directory.

     f. Click **Open**.

     Each package installation produces this message in the R console:

     ```
     package 'package_name' successfully unpacked and MD5 sums checked
     ```

   - **Install from the Windows command prompt**

     a. Start R x64 from the Windows Start menu.

     b. Open a Windows command window.

     c. Change directory to the `client` directory and type these commands:

     ```
     R CMD INSTALL ROracle_1.3-1.zip
     R CMD INSTALL DBI_0.5.zip
     R CMD INSTALL png_0.1-7.zip
     R CMD INSTALL Cairo_1.5-8.zip
     R CMD INSTALL arules_1.1-9.zip
     R CMD INSTALL statmod_1.4.21.zip
     R CMD INSTALL randomForest_4.6-10.zip
     ```

     Each package installation generates this message:

     ```
     package 'package_name' successfully unpacked and MD5 sums checked
     ```

Example 6-2 Client Installation Directory Containing All Client Components

```
c:\oreclient_install_dir
     \client
     \instantclient_version
     \supporting
         \arules_version.zip
         \Cairo_version.zip
         \DBI_version.zip
```

```
        \png_version.zip
        \ROracle_version.zip
        \statmod_version.zip
        \randomForest_version.zip
instantclient-basic-linux.x64-version.zip
instantclient-sdk-linux.x64-version.zip
ore-client-win-x86_64-version.zip
ore-supporting-win-x86_64-version.zip
```

6.4.2 Installing the Supporting Packages on Linux or UNIX

1. Download the Oracle R Enterprise packages from the Oracle R Enterprise Downloads page on the Oracle Technology Network.
2. Accept the license agreement and select the **Supporting** packages for your platform. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:

   ```
   /oreclient_install_dir/ore-supporting-platform-arch-1.5.1.zip
   ```

   NOTE: Choose the same installation directory for all Oracle R Enterprise client components.
3. Unzip the file:

   ```
   % unzip ore-supporting-platform-arch-1.5.1.zip
   ```

   When you unzip the file, the `/supporting` directory is created and these files are extracted.

   ```
   /supporting/arules_1.1-9_R_arch-unknown-platform.tar.gz
   /supporting/Cairo_1.5-8_R_arch-unknown-platform.tar.gz
   /supporting/DBI_0.5_R_arch-unknown-platform.tar.gz
   /supporting/png_0.1-7_R_arch-unknown-platform.tar.gz
   /supporting/ROracle_1.3-1_R_arch-unknown-platform.tar.gz
   /supporting/statmod_1.4.21_R_arch-unknown-platform.tar.gz
   /supporting/randomForest_4.6-10_R_arch-unknown-platform.tar.gz
   ```
4. Change to */oreclient_install_dir*`/supporting`
5. Execute the following commands:

   ```
   R CMD INSTALL ROracle_1.3-1_R_arch-unknown-platform.tar.gz
   R CMD INSTALL DBI_0.5_R_arch-unknown-platform.tar.gz
   R CMD INSTALL png_0.1-7_R_arch-unknown-platform.tar.gz
   R CMD INSTALL Cairo_1.5-8_R_arch-unknown-platform.tar.gz
   R CMD INSTALL arules_1.1-9_R_arch-unknown-platform.tar.gz
   R CMD INSTALL statmod_1.4.21_R_arch-unknown-platform.tar.gz
   R CMD INSTALL randomForest_4.6-10_R_arch-unknown-platform.tar.gz
   ```

6.5 Connecting Oracle R Enterprise Client to Oracle R Enterprise Server

To connect Oracle R Enterprise Client to Oracle R Enterprise Server, start R using the ORE script:

```
% ORE
R> library(ORE)
```

The following examples connect as user `RQUSER` with password `RQUSERpsw`:

- For a remote database, specify the Oracle Database service identifier (SID), the host name, and the port for the connection.

```
ore.connect(user="RQUSER", sid="orcl", host="SVR3", password="RQUSERpsw",
            port=1521, all=TRUE)
```

> **Note:**
>
> To avoid specifying the password and other connection details in embedded R scripts, you can use Oracle Wallet. See Creating an Oracle Wallet for an Oracle R Enterprise Connection (page 8-11).

- For a local database, specify the connection as follows:

```
ore.connect("RQUSER", password="RQUSERpsw", conn_string="", all=TRUE)
```

7

Installing OAAgraph Components

The `OAAgraph` and supporting packages for use with Oracle R Enterprise require separate downloads and installation on server and client systems.

You may download the `OAAgraph` package and its supporting packages from the OAAgraph Downloads website. Reference documentation for the `OAAgraph` functions is available at *Oracle R Enterprise Graph Analytics Functions*.

This chapter includes these topics:

7.1 Install and Configure PGX Server Components

To use `OAAgraph` package functions with Oracle R Enterprise 1.5.1, follow these instructions to install and configure the PGX server in Oracle Database 12*c*, Release 2 (12.2).

Prerequisites

The following are prerequisites for using the OAAgraph package with Oracle R Enterprise:

- Oracle Database 12*c*, Release 2 (12.2), Enterprise Edition, with the Oracle Spatial and Graph option installed.

  For multi-tenant installations, install the Oracle Spatial and Graph in the CDB first and then the PDBs. Oracle recommends using the`catcon.pl` script first in the CDB and then in the PDBs.

- The database initialization parameter `MAX_STRING_SIZE` set to `EXTENDED`.

Apply the PGQL Patch

PGQL (Property Graph Query Language) is a SQL-like query language for graphs. Apply Patch Number 25640325 to your Oracle Database 12.2 installation, which adds support for PGQL to the Oracle Spatial and Graph option of the database. On the My Oracle Support website, search for patch number 25640325. Download the patch, extract the contents of the zip file, and then follow the instructions in the README file.

Configure the PGX Server

After applying the patch, navigate to the `$ORACLE_HOME/md/property_graph/pgx` directory and edit the file `conf/server.conf`. The following is a minimal configuration:

```
{
  "port": 7007,
  "enable_tls": false
}
```

> **Caution:**
>
> The `"enable_tls": false` line disables the Secure Sockets Layer (SSL) and Transport Layer Security (TLS). You might want to disable socket security for testing the connection; however Oracle strongly recommends that you turn on SSL/TLS for any production deployment.

Configure the Environment to Use Oracle JDK 8

Make sure the `java` binary on your `PATH` points to a recent version of Oracle JDK 8. If you do not have the Oracle JDK 8 installed on your system, you can use the one that is bundled with the database. For example:

```
$ export ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1
$ export JAVA_HOME=$ORACLE_HOME/jdk
$ export PATH=$JAVA_HOME/bin:$PATH
$ java -version
java version "1.8.0_91"
Java(TM) SE Runtime Environment (build 1.8.0_91-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.91-b14, mixed mode)
```

Verify the Spatial and Graph Option

To verify that the Spatial and Graph option is installed, do the following:

1. In SQL*Plus, log in to the database.
2. Run the `create_pg` command.

   ```
   exec opg_apis.create_pg('testgraph', 2, 4, 'SYSAUX', '');
   ```

   If the command succeeds, then the option is installed.

If the command fails, see *Oracle Big Data Spatial and Graph User's Guide and Reference* for installation instructions.

Related Topics

- Tutorial: Configure TLS/SSL security certificates
- PGX Server Configuration Guide
- PGX Engine and Runtime Configuration Guide
- Java SE Development Kit 8 Downloads

7.2 Start the PGX Server

To use `OAAgraph` package functions with Oracle R Enterprise 1.5.1, follow these instructions to start the PGX server in Oracle Database 12*c*, Release 2 (12.2).

1. Navigate to the `$ORACLE_HOME/md/property_graph/pgx` directory.
2. Run the `bin/start-server` script:

   ```
   ./bin/start-server
   ```

The server is up and listening on port 7007 after you see a message such as the following:

```
INFO: Starting ProtocolHandler ["http-nio-7007"]
```

The launcher script displays a few warning messages that JAR files cannot be found. This is a known issue and the messages can safely be ignored.

> **Note:**
>
> The script does not return by default. To stop the server, press **Ctrl** —**C** or close the terminal.

7.3 Install OAAgraph and Supporting Client Packages

Prerequisites for and instructions on installing the `OAAgraph` and supporting client packages.

The `OAAgraph` package translates R function calls into analytic requests to a PGX server in Oracle Database 12*c* Release.2.

The prerequisites for installing the `OAAgraph` and the supporting client packages are:

- A Linux or Microsoft Windows x86 system
- Open Source R or Oracle R Distribution 3.3.0
- Oracle R Enterprise 1.5.1 or Oracle R Enterprise 1.5
- Oracle JDK 8 or later

  Ensure that the following are correct:

  - The `JAVA_HOME` environment variable points to the Oracle JDK 8 installation directory
  - The java binary on your `PATH` environment variable points to the correct executable

In the following steps, the command examples are for a Linux system.

To install the `OAAgraph` package and the client supporting packages, do the following:

1. Download the `OAAgraph-supporting.zip` file.
2. Extract the contents of the file.

   ```
   unzip OAAgraph-supporting.zip -d /tmp/supporting
   ```

3. Install the supporting package.

   ```
   R CMD INSTALL /tmp/supporting/rJava_0.9-9-LGPL.tar.gz
   ```

4. Install the `OAAgraph` package.

   On Linux:

   ```
   R CMD INSTALL pgx-2.4.2-oaa-client.tgz
   ```

   On Windows:

   ```
   R CMD INSTALL pgx-2.4.2-oaa-client.tgz --multiarch
   ```

5. Configure the `rJava` library that is part of the supporting package.

   ```
   sudo -E R CMD javareconf
   ```

8

Administrative Tasks for Oracle R Enterprise

This chapter describes administrative tasks for maintaining and optimizing Oracle R Enterprise.

This chapter contains these topics:

8.1 Install Oracle R Distribution on Linux in a Non-Default R_HOME

Beginning with Oracle R Distribution 3.3.0, the Linux RPMs can be installed to a directory other than the default Linux `R_HOME`, `/usr/lib64/R`.

The procedure in the following example installs the Oracle R Distribution 3.3.0 RPMs to a non-default location and still allows the user to invoke the previously installed version, R-3.2.0.

The example installs the RPMs into the directory `/opt/R330`. It installs the following RPMs:

```
R-3.3.0-2.el6.x86_64.rpm
R-core-3.3.0-2.el6.x86_64.rpm
R-core-extra-3.3.0-2.el6.x86_64.rpm
R-devel-3.3.0-2.el6.x86_64.rpm
libRmath-3.3.0-2.el6.x86_64.rpm
libRmath-devel-3.3.0-2.el6.x86_64.rpm
libRmath-static-3.3.0-2.el6.x86_64.rpm
```

1. From the directory that contains the RPMs, install the Oracle R Distribution 3.3.0 RPMs to a non-default location using the `--prefix` flag:

   ```
   # rpm -i *.rpm --prefix=/opt/R330
   ```

2. Set `R_HOME` to the R-3.3.0 location and add `$R_HOME/bin` to `PATH`:

```
# export R_HOME=/opt/R330/lib64/R

# export PATH=$R_HOME/bin:$PATH
```

3. Invoke the newly installed R-3.3.0. If you receive the following error, then add `$R_HOME/port/Linux-X64/lib` to `LD_LIBRARY_PATH` so R recognizes the correct versions of the pcre, zlib, xz, and bzip libraries:

```
# R

/opt/R330/lib64/R/bin/exec/R: error while loading shared libraries:
libpcre.so.1: cannot open shared object file: No such file or directory

# export LD_LIBRARY_PATH=$R_HOME/port/Linux-X64/lib:$LD_LIBRARY_PATH

# R

Oracle Distribution of R version 3.3.0  (--) -- "Supposedly Educational"
Copyright (C)  The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)...
```

If you still want to use the previous version of R, rename the default R executable `/usr/bin/R` to the old R version; for example, `/usr/bin/R-3.2.0`:

```
# mv /usr/bin/R /usr/bin/R-3.2.0
```

Now you can invoke R 3.2.0:

```
# R-3.2.0
Oracle Distribution of R version 3.2.0  (--) -- "Full of Ingredients"
Copyright (C)  The R Foundation for Statistical Computing
Platform: x86_64-unknown-linux-gnu (64-bit)...
```

8.2 Upgrading Oracle R Enterprise

You can upgrade Oracle R Enterprise to the current release from any previous release by reinstalling the product.

> **Note:**
>
> Upgrade from Oracle R Enterprise 1.1 is not supported on IBM AIX. To upgrade Oracle R Enterprise 1.1 on IBM AIX, first uninstall Oracle R Enterprise 1.1 (including R) and then download and install the later version.

To upgrade Oracle R Enterprise and migrate your data:

1. Ensure that you have the version of R that is required for the new version of Oracle R Enterprise.

   See the table of configuration requirements and server support in Oracle R Enterprise System Requirements (page 1-5) for the R requirement.

2. To upgrade Oracle R Enterprise Server, run the `server` script to perform an installation:

```
./server.sh
```

When the script detects an earlier version of Oracle R Enterprise Server, it asks if you want to upgrade. Type `Yes` to start the upgrade. (Type `No` to aborts the process.)

See About the SERVER Script (page 4-2) for details.

.

3. To upgrade Oracle R Enterprise Client, re-install the Oracle R Enterprise packages and supporting packages. You do not need to uninstall the current packages before installing the new packages.

   See Installing Oracle R Enterprise Client (page 6-1) for instructions.

8.3 Migrating Oracle R Enterprise Data

Oracle R Enterprise Server includes migration scripts that you can run to migrate the RQSYS schema and Oracle R Enterprise user data from a source database to a target database

The source and target must have the same version of the Oracle Database and of Oracle R Enterprise Server.

To locate the scripts, navigate to the `server` directory and change to the `migration` subdirectory.

/oreserver_install_dir/server/migration

The `migration` subdirectory contains a README and the following subdirectories:

- `exp` — contains the script `ore_srcexport.pl` for exporting the RQSYS schema and all Oracle R Enterprise user data to a dump file.
- `imp` — contains the script `ore_destimport.pl` for importing the RQSYS schema and all Oracle R Enterprise user data from the dump file created by `ore_srcexport.pl`.
- `oreuser` — contains scripts for exporting and importing data for a specific Oracle R Enterprise user.

Instructions for running the migration scripts are provided in the README.

8.4 Migrating Oracle R Enterprise After a Database Upgrade

After upgrading your Oracle Database, you must migrate your Oracle R Enterprise Server components to the new `ORACLE_HOME`.

If you do not migrate the Oracle R Enterprise Server components to the new `ORACLE_HOME`, then running an R function using Oracle R Enterprise embedded R execution results in errors such as:

```
ORA-28578: protocol error during callback from an external procedure
```

The components of Oracle R Enterprise Server are:

- The Oracle Database schema RQSYS and schema-related objects
- Oracle Database shared libraries for supporting Oracle R Enterprise clients
- Oracle R Enterprise packages and supporting packages installed on the Oracle Database server

After a database upgrade, you must migrate the RQSYS schema and dependent database components to the new ORACLE_HOME. The Oracle R Enterprise packages must also be installed to the new database location.

You can easily do this by running the Oracle R Enterprise Server installation script against the new ORACLE_HOME. Doing so creates a new path to the ORACLE_HOME in the Oracle R Enterprise metadata.

If the Oracle R Enterprise user already exists in the upgraded database, then use the --no-user flag when running the server script.

The following steps illustrate migrating Oracle R Enterprise 1.5.0 from an initial database installation to a new database after a database upgrade. Oracle Database was upgraded from Release 12.1.0.2 to Release 12.2.0.1. Oracle R Distribution and Oracle R Enterprise are not upgraded, only migrated to the new ORACLE_HOME.

1. Before migrating the Oracle R Enterprise components, back up the RQSYS schema and Oracle R Enterprise user schema.
2. Run the Oracle R Enterprise Server installation script against the new ORACLE_HOME.

   ```
   $ ./server.sh --no-user
   ```

3. As the sysdba user, verify that the Oracle R Enterprise configuration script is pointing to the new ORACLE_HOME:

   ```
   SQL> SELECT * FROM sys.rq_config;

   NAME                                                    VALUE
   ------------------------------------------------------------
   R_HOME       /usr/lib64/R
   R_LIBS_USER  /u01/app/oracle/product/12.2.0.1/dbhome_1/R/library
   VERSION      1.5
   ...
   ```

4. As the sysdba user, verify that the Oracle R Enterprise dependent libraries ore.so and librqe.so are in the new ORACLE_HOME:

   ```
   SQL> select library_name, file_spec from all_libraries where owner = 'RQSYS';

   LIBRARY_NAME                                              FILE_SPEC
   ------------  ---------------------------------------------------
   RQ$LIB        /u01/app/oracle/product/12.2.0.1/dbhome_1/lib/ore.so
   RQELIB        /u01/app/oracle/product/12.2.0.1/dbhome_1/lib/librqe.so
   ```

5. Finally, test the Oracle R Enterprise installation against the upgraded ORACLE_HOME by running Oracle R Enterprise demonstration programs.

Related Topics

- Running the Oracle R Enterprise Example Scripts (page 4-11)
  To fully validate the success of the installation, run the suite of Oracle R Enterprise demo scripts.

8.5 Uninstalling Oracle R Enterprise

This topic contains these sections:

Related Topics

- Uninstalling Oracle R Distribution (page 3-13)

8.5.1 Uninstalling Oracle R Enterprise Server

To uninstall Oracle R Enterprise Server, run the `server` script with the `--uninstall` option.

You can perform either a full or a partial uninstall. A partial uninstall is performed by default.

8.5.1.1 Performing a Partial Uninstall

A partial uninstall removes the RQSYS metadata and PL/SQL packages from the database but leaves the libraries and R packages that support Oracle R Enterprise Server in Oracle home.

If Oracle R Enterprise Server support is installed in more than one database instance in the same Oracle home, or if it is installed in a pluggable database (PDB), then a partial uninstall removes Oracle R Enterprise Server support from the specified database without affecting the other databases. The `server` script performs a partial uninstall by default.

> **Note:**
>
> If you accidentally perform a full uninstall for one of the instances or PDBs that share support for Oracle R Enterprise Server, then the other shared instances or PDBs will no longer support Oracle R Enterprise Server. You can easily restore Oracle R Enterprise Server support in Oracle home by rerunning the `server` script to perform an installation in one of the shared instances or PDBs.

If you run the `server` script with the `-u` option, then a partial uninstall is performed. You can specify the `--keep` option to explicitly request a partial uninstall. The following commands all perform a partial uninstall of Oracle R Enterprise Server:

```
./server.sh  --uninstall
./server.sh  -u
./server.sh  -u --keep
./server.sh  --uninstall  --keep
```

8.5.1.2 Performing a Full Uninstall

A full uninstall removes the RQSYS schema metadata and PL/SQL code from the database and removes all Oracle R Enterprise Server libraries and R packages from Oracle home.

The following commands each perform a full uninstall of Oracle R Enterprise Server:

```
./server.sh --uninstall --full
./server.sh -u  -full
```

> **Note:**
>
> If you accidentally perform a full uninstall in a shared Oracle home, then rerun the `server` script to reinstall Oracle R Enterprise Server support. See Performing a Partial Uninstall (page 8-5) for details.

8.5.2 Uninstalling Oracle R Enterprise Client

To uninstall the Oracle R Enterprise packages and supporting packages, start R and type the commands listed in Example 8-1 (page 8-6).

Example 8-1 R Commands for Uninstalling Oracle R Enterprise Packages

```
remove.packages("ORE")
remove.packages("ORExml")
remove.packages("OREeda")
remove.packages("OREcommon")
remove.packages("OREdplyr")
remove.packages("OREembed")
remove.packages("OREgraphics")
remove.packages("OREstats")
remove.packages("OREbase")
remove.packages("ROracle")
remove.packages("DBI")
remove.packages("Cairo")
remove.packages("png")
remove.packages("OREdm")
remove.packages("OREpredict")
remove.packages("arules")
remove.packages("statmod")
remove.packages("randomForest")
```

8.6 Installing Additional R Packages on Linux or UNIX

On Linux and UNIX platforms, the Oracle R Enterprise Server installation provides the `ORE` script, which you can run from the operating system prompt to install additional R packages. The `ORE` script is a wrapper for the R installation command: `R CMD INSTALL`.

By default, R packages are installed in `/usr/lib64/R/library`. The `ORE` script, however, installs R packages in a subdirectory under `$ORACLE_HOME/R/library`.

To execute the script:

```
ORE CMD INSTALL R_package_name
```

8.7 Creating a Database User for Oracle R Enterprise

The `server` script installation process automatically creates or configures a user for Oracle R Enterprise if one does not already exist.

Example 8-2 Creating an Oracle R Enterprise User

```
./server.sh
.
.
```

```
Choosing ORE user
  ORE user to use [list]:
```

Press **Enter** to display a list of available users.

```
BI
HR
IX
OE
SCOTT
SH
  ORE user to use [list]: ruser2
.
.
```

If you choose a user that exists, the script configures the user to support Oracle R Enterprise. If you specify a user that does not already exist, the script creates the user.

Example 8-3 Creating an Oracle R Enterprise User in SQL*Plus

You can create an Oracle R Enterprise user in SQL*Plus by following these steps:

1. Log in with system privileges:

   ```
   SQLPLUS / AS SYSDBA
   ```

2. Execute a statement like the following to create the user:

   ```
   CREATE USER ore_username IDENTIFIED BY password
   DEFAULT TABLESPACE default_tablepace_name
   TEMPORARY TABLESPACE temp_tablespace_name
   QUOTA UNLIMITED ON default_tablespace_name;
   ```

   > **See Also:**
   >
   > *Oracle Database SQL Language Reference* for details about creating a user

3. Grant the required privileges:

   ```
   GRANT CREATE SESSION,
         CREATE TABLE,
         CREATE VIEW,
         CREATE PROCEDURE,
         CREATE MINING MODEL
   TO ore_username;
   ```

   > **See Also:**
   >
   > *Oracle Database SQL Language Reference* for details about granting privileges to a user

8.7.1 About the RQADMIN Role

The `server` script installation process creates a database role called RQADMIN.

When the RQADMIN role is granted to an Oracle R Enterprise user, the user can create and drop R scripts for embedded R execution. By default, the `server` script does *not* grant the RQADMIN role to the Oracle R Enterprise user.

> **Note:**
>
> Any Oracle R Enterprise user can execute embedded R, but only Oracle R Enterprise users with the RQADMIN role can create and drop the R scripts.

As shown in User Configuration in Batch Mode (page 4-6), you can run the `server` script with the `--admin` option to grant the RQADMIN role to an Oracle R Enterprise user. The `--admin` option is only available when you run the script in batch mode.

If you choose to grant the RQADMIN role in SQL*Plus, then log in with system privileges and execute a statement like the following:

```
SQLPLUS / AS SYSDBA
GRANT RQADMIN TO ore_username;
```

> **Caution:**
>
> Use caution when granting the RQADMIN role. Only users that require Oracle R Enterprise administrative privileges should have this role.

Related Topics

- Security Best Practices for Oracle R Enterprise (page 4-2)
  To minimize the risk of compromising the security of an Oracle R Enterprise Server in Oracle Database, Oracle recommends the following security best practices.

8.8 Creating and Modifying Environment Variables on Windows

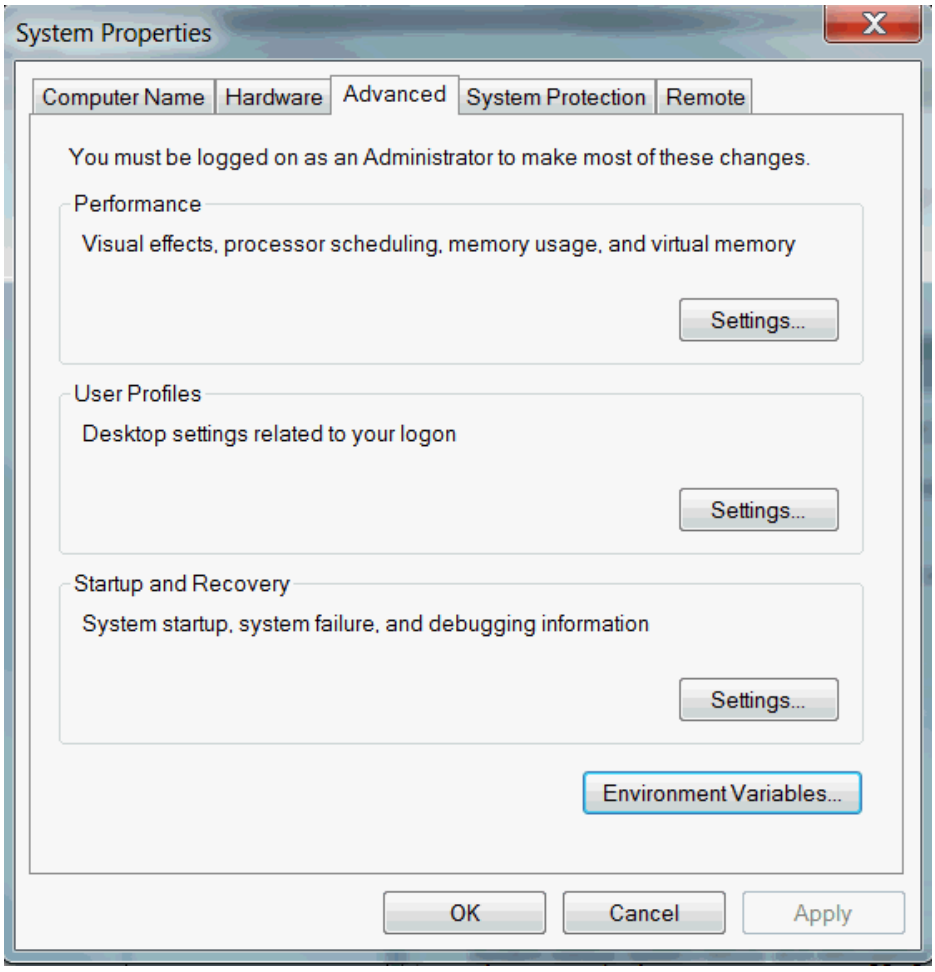
If the `PATH`, `ORACLE_SID`, and `ORACLE_HOME` environment variables do not exist, you must create them.

Assign the values specified in Figure 8-2 (page 8-10). On Windows systems, you must be an administrator to create or modify environment variables.

To create or modify environment variables on Windows:

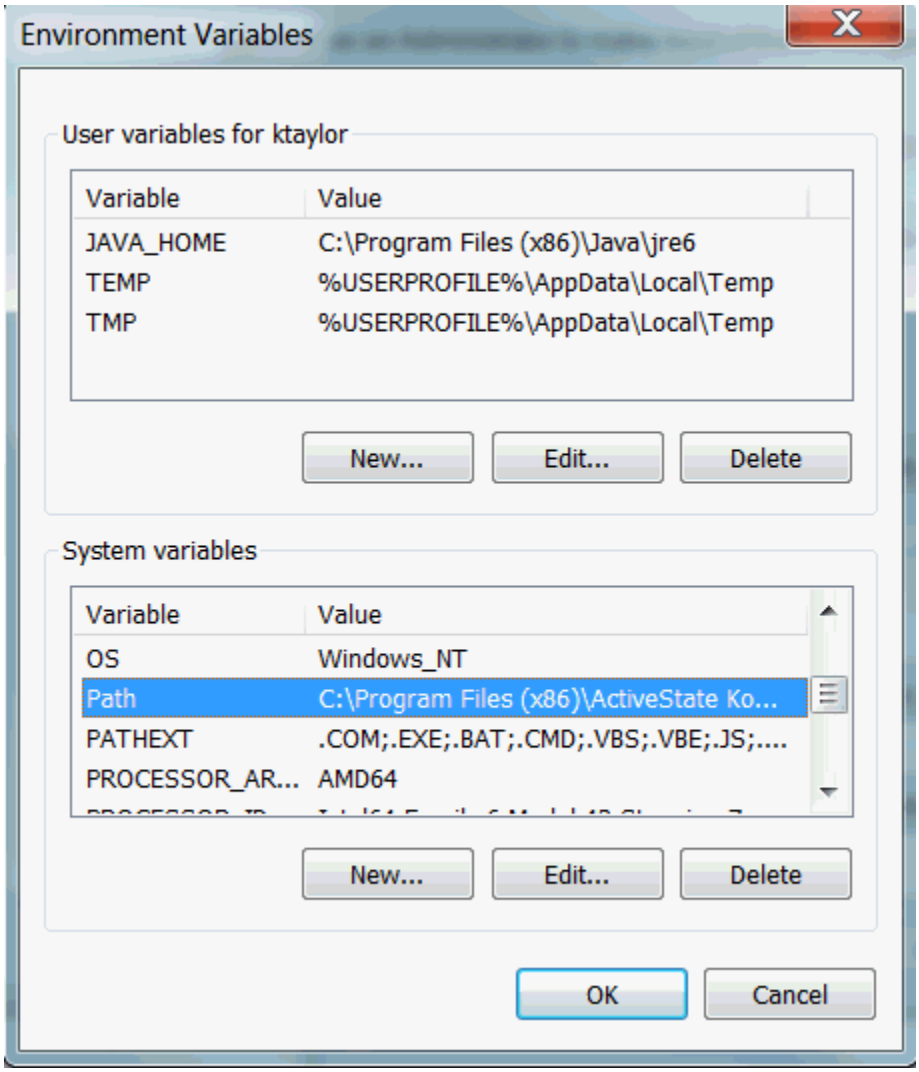
1. Right-click the **Computer** icon and choose **Properties**, or in Windows Control Panel, choose **System**.
2. Choose **Advanced system settings**.

Figure 8-1 Advanced System Settings in Windows



3. On the Advanced tab, click **Environment Variables**.

Figure 8-2 Environment Variables Dialog in Windows



4. Click **New** to create a new environment variable. Click **Edit** to modify an existing environment variable.

> **Note:**
>
> The graphical user interface for creating environment variables may vary slightly, depending on your version of Windows.

8.9 Creating an Oracle Wallet for an Oracle R Enterprise Connection

An Oracle wallet is a password-protected container for storing security credentials in Oracle Database.

Wallets provide a secure mechanism for specifying connection details in embedded R scripts.

To create a wallet for an Oracle R Enterprise connection:

1. Start Oracle Wallet Manager:
   - (Linux and UNIX) At the command line, enter `owm`.
   - (Windows) Select **Start**, **Programs**, **Oracle-HOME_NAME**, I**ntegrated Management Tools, Wallet Manager.**
2. Follow the instructions in your Oracle Database documentation to create the wallet:
   - For Oracle Database 12.1, see "Using Oracle Wallet Manager" in *Oracle Database Enterprise User Security Administrator's Guide*:

     `http://www.oracle.com/pls/topic/lookup?ctx=db121&id=DBIMI160`
   - For Oracle Database 11.2, see "Using Oracle Wallet Manager" in *Oracle Database Advanced Security Administrator's Guide*:

     `http://www.oracle.com/pls/topic/lookup?ctx=db112&id=ASOAG160`
3. Locate the connection string for the Oracle R Enterprise database in `tnsnames.ora`. For example:

   ```
   mydb_test =
         (DESCRIPTION =
            (ADDRESS =
               (PROTOCOL = TCP)
               (HOST = server23)
               (PORT = 1521)
            )
            (CONNECT_DATA = (sid=ORCL))
         )
   ```
4. Specify the connection information in the wallet. Follow the instructions in the Oracle Database security documentation referenced in Step 2.
5. After you configure the wallet, you can connect to the Oracle R Enterprise server database by simply specifying the connection identifier. For example:

   ```
   ore.connect(conn_string = "mydb_test", all = TRUE)
   ```

   To learn more about `ore.connect`, use the R help command:

   ```
   help(ore.connect)
   ```

8.10 Controlling Memory Used by Embedded R

You can control the memory used by embedded R execution by limiting the heap memory (vector and cons in R terminology) that is automatically managed by the R `gc`

mechanism. To limit the size of heap memory in the database, use the `sys.rqconfigset` utility. The keyword arguments for `sys.rqconfigset` are described in the following table.

Table 8-1 SYS.RQCONFIGSET Keyword Arguments

Keyword	Default	Description
MIN_VSIZE	32M	Minimum R vector heap memory
MAX_VSIZE	4G	Maximum R vector heap memory
MIN_NSIZE	1M	Minimum number of R cons cells
MAX_NSIZE	20M	Maximum number of R cons cells

Example 8-4 SQL Commands for Controlling Memory Used by Embedded R

```
-- Set the minimum R vector heap memory to 20M
EXEC sys.rqconfigset('MIN_VSIZE', '20M');

-- Set the maximum R vector heap memory to 100M
EXEC sys.rqconfigset('MAX_VSIZE', '100M')

-- Set the minimum number of R cons cells to 500x1024
EXEC sys.rqconfigset('MIN_NSIZE', '500K');

-- Set the maximum number of R cons cells to 10x10x1024
EXEC sys.rqconfigset('MAX_NSIZE', '10M');

-- Set maximum vector heap memory and maximum cons cells to unlimited
EXEC sys.rqconfigset('MAX_VSIZE', NULL);
EXEC sys.rqconfigset('MAX_NSIZE', NULL);
```

Note:

The `sys.rqconfigset` procedure does not control the C type memory that may be allocated by Calloc, Realloc, calloc, or malloc. Such C type memory is mainly created to hold temporary values used by R functions that are implemented in C. Under normal circumstances, C type memory is limited in size and does not significantly affect the memory usage of R.

The `sys.rqconfigset` procedure edits settings in a configuration table called `sys.rq_config`. You can view the contents of this table to verify various environment settings for Oracle R Enterprise. Among the settings stored in `sys.rq_config` are the memory limits for embedded R. If necessary, you can modify these memory limits, however in most cases you should not modify the values in `sys.rq_config`.

The following query shows sample values stored in `sys.rq_config`.

```
SQL> SELECT * FROM sys.rq_config;

NAME                      VALUE
------------------------- ------------------------------------------------------
R_HOME                    /usr/lib64/R
R_LIBS_USER               /dbhome_1/R/library
VERSION                   1.5.1
MIN_VSIZE                 32M
```

```
MAX_VSIZE               4G
MIN_NSIZE               2M
MAX_NSIZE               20M
```

A

A Sample Installation of Oracle R Enterprise

This appendix presents the steps in a typical installation of Oracle R Enterprise on a Linux server and a Windows client. This appendix contains these topics:

> **Note:**
>
> This appendix describes an initial installation of Oracle R Enterprise. If Oracle R Enterprise components already exist on your client or server, refer to Upgrading Oracle R Enterprise (page 8-2).

A.1 About the Oracle R Enterprise Sample Installation Environment

About the server computer:

- The server is running Oracle Linux 6.
- The server has access to the internet and to Oracle public yum.
- Oracle Database Enterprise Edition 12.1.0.2 is installed on the server.
- Environment variables:
  - `$ORACLE_SID` specifies the identifier (SID) of the database.
  - `$ORACLE_HOME` specifies the home directory of the database.
  - `$LD_LIBRARY_PATH` includes `$ORACLE_HOME/lib`.
  - `$PATH` includes `$ORACLE_HOME/bin`.
- The Linux user ID of the installer:
  - Has sudo rights or root access for installing Oracle R Distribution.
  - Is a member of the dba group for installing and using Oracle R Enterprise.
  - Has write access to `$ORACLE_HOME/lib`.

About the client computer:

- The client is running 64-bit Windows.

- The client has access to the internet.

A.2 Installing Oracle R Enterprise on the Server

To install Oracle R Enterprise on the server computer, first verify that Oracle Database is installed and that the environment is configured as specified in About the Oracle R Enterprise Sample Installation Environment (page A-1). Next, complete these steps in the specified order:

1. Verify the Environment (page A-2)
2. Install Oracle R Distribution (page A-3)
3. Install Oracle R Enterprise Server (page A-3)

A.2.1 Verify the Environment

Table A-1 Checklist for Oracle R Enterprise Server Requirements

Question	Sample Answer
What is the Linux version?	`% cat /etc/redhat-release` `Enterprise Linux Server release 6.4`
Do you have access to the internet?	Start a browser

Can you log in as root?	`% sudo -su`
	`Password: ---------`
	`#`
	`# exit`
	`%`

Is Oracle Database installed?	`% SQLPLUS / as sysdba`
	`Copyright (c) 1982, 2017, Oracle.  All rights reserved.`
	`Connected to: Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bitProduction`
	`With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options`
	`> exit`
	`%`

What is the value of `$ORACLE_HOME`?	`% echo $ORACLE_HOME` `/myhome/product/12.1.0.2/dbhome_1`
What is the value of `$ORACLE_SID`?	`% echo $ORACLE_SID` `orcl`
Does `$LD_LIBRARY_PATH` include `$ORACLE_HOME/lib`?	`% echo $LD_LIBRARY_PATH` `/myhome/product/12.1.0.2/dbhome_1/lib:....`
Does `$PATH` include `$ORACLE_HOME/bin`?	`% echo $PATH` `/myhome/product/12.1.0.2/dbhome_1/bin:.......`

Table A-1 (Cont.) Checklist for Oracle R Enterprise Server Requirements

Question	Sample Answer
Are you a member of the dba group?	`% groups` `g102 dba`

Can you write to `$ORACLE_HOME/lib`?	`% ls -ld $ORACLE_HOME/lib`
	`drwxr-xr-x 3 myuser g102 12288 Jan 27 15:31`
	`/myhome/product/12.1.0.2/dbhome_1/lib/ ...`

A.2.2 Install Oracle R Distribution

To install Oracle R Distribution on the server from Oracle public yum, follow these steps:

1. Log in as root and change to `/etc/yum.repos.d`:

   ```
   cd /etc/yum.repos.d
   ```

2. List the contents of the directory to determine if the yum configuration file is present. The yum configuration file for Oracle Linux 6 is called `public-yum-el6.repo`.

   If `public-yum-el6.repo` is not present, then execute the following command to download it from Oracle public yum:

   ```
   wget http://public-yum.oracle.com/public-yum-el6.repo
   ```

3. Open `public-yum-el6.repo` in a text editor and specify `enabled=1` for `latest` and `addons`:

   ```
   [el6_latest]
   enabled=1

   [el6_addons]
   enabled=1
   ```

4. Install Oracle R Distribution 3.3 by executing these commands:

   ```
   yum install R-3.3.0
   yum install R-core-extra
   ```

5. Set `LD_LIBRARY_PATH` to the location of the files installed by the R-core-extra RPM:

6. Exit the root user.

   ```
   exit
   ```

A.2.3 Install Oracle R Enterprise Server

Oracle R Enterprise Server includes the RQSYS schema in Oracle Database and Oracle R Enterprise packages and shared libraries.

To install Oracle R Enterprise Server:

1. Verify the environment according to Table A-1 (page A-2).

2. Create an installation directory for the Oracle R Enterprise server components. The directory can have any name. For example:

   ```
   /myhome/myoreserver/
   ```

3. Download the Oracle R Enterprise Server installation files and supporting packages from the Oracle R Enterprise Downloads page on the Oracle Technology Network.

   a. Accept the license agreement and download the Oracle R Enterprise **Server** files for your platform to your installation directory.

   b. Accept the license agreement and download the Oracle R Enterprise **Supporting** packages for your platform to your installation directory.

   The installation directory now contains two zip files.

   ```
   ore-server-linux-x86-64-1.5.1.zip
   ore-supporting-linux-x86-64-1.5.1.zip
   ```

4. Unzip the files.

   ```
   unzip ore-server-linux-x86-64-1.5.1.zip
   unzip ore-supporting-linux-x86-64-1.5.1.zip
   ```

   The installation directory looks like this after you unzip both files:

   ```
   /myhome/myoreserver
        ore-server-linux-x86-64-1.5.1.zip
        ore-supporting-linux-x86-64-1.5.1.zip
        server.sh
        /server
        /supporting
   ```

5. Run `server.sh` to perform a default installation of Oracle R Enterprise Server as shown in the following example. In this example, the script runs interactively. User input is shown in bold.

> **Note:**
>
> When the script displays `[list]` in a prompt, you can press **Enter** to obtain a list of available items for your choice.

Example A-1 A Default, First-Time Installation of Oracle R Enterprise Server

```
hcearwigger@myserver> ./server.sh

Oracle R Enterprise 1.5.1 Server.

Copyright (c) 2012, 2017 Oracle and/or its affiliates. All rights reserved.

Checking platform .................. Pass
Checking R ......................... Pass
Checking R libraries ............... Pass
Checking ORACLE_HOME ............... Pass
Checking ORACLE_SID ................ Pass
Checking sqlplus ................... Pass
Checking ORACLE instance ........... Pass
Checking CDB/PDB ................... Pass
Checking ORE ....................... Pass
```

```
Choosing RQSYS tablespaces
  PERMANENT tablespace to use for RQSYS [list]:
EXAMPLE
SYSAUX
SYSTEM
USERS
  PERMANENT tablespace to use for RQSYS [list]: SYSAUX
  TEMPORARY tablespace to use for RQSYS [list]:
TEMP
  TEMPORARY tablespace to use for RQSYS [list]: TEMP
Choosing RQSYS password
  Password to use for RQSYS: XXXXXXX

Choosing ORE user
  ORE user to use [list]:
BI
HR
IX
OE
SCOTT
SH
  ORE user to use [list]: ruser2
Choosing RUSER2 tablespaces
  PERMANENT tablespace to use for RUSER2 [list]: USERS
  TEMPORARY tablespace to use for RUSER2 [list]: TEMP
Choosing RUSER2 password
  Password to use for RUSER2:


  Current configuration
  R Version ..................... Oracle Distribution of R version 3.3.0  (--)
  R_HOME ........................ /usr/lib64/R
  R_LIBS_USER ................... /product/12.1.0.2/dbhome_1/R/library
  ORACLE_HOME ................... /product/12.1.0.2/dbhome_1
  ORACLE_SID .................... orcl

  Existing R Version ............ None
  Existing R_HOME ............... None
  Existing ORE data ............. None
  Existing ORE code ............. None
  Existing ORE libraries ......... None

  RQSYS PERMANENT tablespace ..... SYSAUX
  RQSYS TEMPORARY tablespace ..... TEMP

  ORE user type .................. New
  ORE user name .................. RUSER2
  ORE user PERMANENT tablespace ...USERS
  ORE user TEMPORARY tablespace .. TEMP
  Grant RQADMIN role ............. No

  Operation ...................... Install/Upgrade/Setup

Proceed? [yes] y

Removing R libraries ............... Pass
Installing R libraries ............. Pass
Installing ORE libraries ........... Pass
Installing RQSYS data .............. Pass
Configuring ORE .................... Pass
Installing RQSYS code .............. Pass
```

```
Installing ORE packages ............ Pass
Creating ORE script ................ Pass
Installing migration scripts ....... Pass
Installing supporting packages ..... Pass
Creating ORE user .................. Pass
Granting ORE privileges ............ Pass

Done
```

A.3 Installing Oracle R Enterprise on the Client

To install Oracle R Enterprise on the client computer, first verify that the Windows environment meets the requirements.

The requirements are specified in About the Oracle R Enterprise Sample Installation Environment (page A-1). Next, complete these steps:

To install Oracle R Enterprise on the Client:

1. Install Oracle R Distribution on the Windows Client (page A-6)
2. Install Oracle Instant Client (page A-7)
3. Install the Oracle R Enterprise Packages (page A-8)
4. Install the Oracle R Enterprise Supporting Packages (page A-9)

A.3.1 Install Oracle R Distribution on the Windows Client

Before installing Oracle R Distribution, verify that your version of Microsoft Windows is supported by Oracle R Enterprise and that you have access to the internet.

> **See Also:**
>
> - Oracle R Enterprise System Requirements (page 1-5)
> - Verifying 64-Bit Architecture on Microsoft Windows (page 1-7)

To install Oracle R Distribution on Windows:

1. Go to the Oracle R Distribution downloadspage.
2. Under **R 3.3.0 Downloads**, select **R Distribution** for **Windows 64 bit**. Save the file on your computer.

   `ORD-3.3.0-win.zip`
3. When you unzip the file, the executable file is extracted.

   `ORD-3.3.0-win.exe`
4. Double-click the executable file to start the installation of Oracle R Distribution.
5. Follow the instructions to complete the installation.

A.3.2 Install Oracle Instant Client

Oracle R Enterprise requires Oracle Database Client.

Instead of installing the full Database Client, which must be installed in an Oracle home directory, you can install Oracle Instant Client.

To download and install Oracle Instant Client:

1. Create an installation directory for the Oracle R Enterprise client components. The directory can have any name. For example:

   ```
   c:\myoreclient
   ```

2. Navigate to the Oracle Database Instant Client page on the Oracle Technology Network.
3. Select **See Instant Client Downloads**.
4. On the Instant Client Downloads page, select **Instant Client for Microsoft Windows (x64)**.
5. Accept the license agreement.
6. Under **Version 12.1.0.2.0**, select **Instant Client Package - Basic** for Oracle Database 12.1.
7. Save the file in the installation directory that you created in Step 1. For example, if you choose the basic package, the following file is downloaded:

   ```
   c:\myoreclient\instantclient-basic-windows.x64-12.1.0.2.0.zip
   ```

8. Unzip the file.

   When you unzip the file, the `instantclient_12_1` subdirectory is created. The contents of the installation directory are shown as follows:

   ```
   myoreclient
     instantclient_12_1
       vc10
       vc11
       vc12
   ```

9. Return to the Instant Client download page for your platform.
10. Accept the license agreement and select **Instant Client Package - SDK**. Save the file in the directory that you created in Step 1.

    ```
    c:\myoreclient\instantclient-sdk-windows.x64-12.1.0.2.0.zip
    ```

11. Unzip the file.

    When you unzip the file, the `sdk` subdirectory is created. The contents of the installation directory are shown as follows:

    ```
    myoreclient
      instantclient_12_1
        help
        sdk
        vc10
        vc11
        vc12
    ```

12. Add the full path of the Instant Client to the environment variables `OCI_LIB64` and `PATH`. The following steps set the variables to the path used in this example, `c:\myoreclient\instantclient_12_1`:
    a. In Windows Control Panel, choose **System**.
    b. Click **Advanced systems settings**.
    c. On the **Advanced** tab, click **Environment Variables**.
    d. Under **System variables**, create `OCI_LIB64` if it does not already exist. Set the value of `OCI_LIB64` to `c:\oreclient\instantclient_12_1`.
    e. Under **System variables**, edit `PATH` to include `c:\oreclient\instantclient_12_1`.

> **Note:**
>
> The graphical user interface for creating environment variables may vary slightly, depending on your version of Windows.
>
> To be able to load the ROracle package, you must first add the full path of the Oracle Instant Client to the PATH and the OCI_LIB64 environment variables. For troubleshooting tips, refer to the Troubleshooting section in the ROracle INSTALL file on CRAN at ROracle INSTALL.

A.3.3 Install the Oracle R Enterprise Packages

Follow these steps to download and install the Oracle R Enterprise packages:

To download the Oracle R Enterprise packages:

1. Navigate to the Oracle R Enterprise Downloads page on the Oracle Technology Network:

   ```
   Oracle R Enterprise Downloads
   ```

2. Accept the License Agreement.
3. Select the **Client** packages for Windows. Save the file in the installation directory that you created in Install Oracle Instant Client (page A-7).

   ```
   c:\myoreclient\ore-client-win-x86_64-1.5.1.zip
   ```

4. Unzip the file.

   When you unzip the file, the `client` subdirectory is created. The contents of the installation directory are shown as follows:

   ```
   ORE_1.5.1.zip
   OREbase_1.5.1.zip
   OREcommon_1.5.1.zip
   OREdm_1.5.1.zip
   OREdplyr_1.5.1.zip
   OREeda_1.5.1.zip
   OREembed_1.5.1.zip
   OREgraphics_1.5.1.zip
   OREmodels_1.5.1.zip
   OREpredict_1.5.1.zip
   ```

```
OREstats_1.5.1.zip
ORExml_1.5.1.zip
```

To install the Oracle R Enterprise packages from the R Console:

1. Start R from the Windows Start menu. If you have installed both 32 and 64-bit R, be sure to choose 64-bit R.
2. In the R Console window, install the packages as follows:

```
install.packages("c:/myoreclient/client/ORE_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREbase_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREcommon_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREdm_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREdplyr_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREeda_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREembed_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREgraphics_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREmodels_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREpredict_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/OREstats_1.5.1.zip", repos=NULL)
install.packages("c:/myoreclient/client/ORExml_1.5.1.zip", repos=NULL)
```

Each successful package installation produces this message in the R console:

```
package 'package_name' successfully unpacked and MD5 sums checked
```

A.3.4 Install the Oracle R Enterprise Supporting Packages

Follow these steps to download and install the Oracle R Enterprise supporting packages:

To download the Oracle R Enterprise supporting packages:

1. Navigate to the Oracle R Enterprise Downloads page on the Oracle Technology Network:

   ```
   Oracle R Enterprise Downloads
   ```

2. Accept the License Agreement and select the **Supporting** packages for Windows. Save the file in the installation directory that you created in Install Oracle Instant Client (page A-7).

   ```
   c:\myoreclient\ore-supporting-win-x86_64-1.5.1.zip
   ```

3. Unzip the file.

   When you unzip the file, the `supporting` subdirectory is created. The contents of the installation directory are shown as follows:

   ```
   arules_1.1-9.zip
   Cairo_1.5-8.zip
   DBI_0.5.zip
   png_0.1-7.zip
   randomForest_4.6-10.zip
   ROracle_1.3-1.zip
   statmod_1.4.21.zip
   ```

To install the supporting packages from the R Console:

1. Start R from the Windows Start menu. If you have installed both 32 and 64-bit R, be sure to choose 64-bit R.

The R Console window is displayed.

2. Install the packages as follows:

```
install.packages("c:/myoreclient/supporting/ROracle_1.3-1.zip", repos=NULL)
install.packages("c:/myoreclient/supporting/DBI_0.5.zip", repos=NULL)
install.packages("c:/myoreclient/supporting/png_0.1-7.zip", repos=NULL)
install.packages("c:/myoreclient/supporting/Cairo_1.5-8.zip", repos=NULL)
install.packages("c:/myoreclient/supporting/arules_1.1-9.zip", repos=NULL)
install.packages("c:/myoreclient/supporting/randomForest_4.6-10.zip", repos=NULL)
install.packages("c:/myoreclient/supporting/statmod_1.4.21.zip", repos=NULL)
```

Each successful package installation produces this message in the R console:

```
package 'package_name' successfully unpacked and MD5 sums checked
```

A.4 Verifying the Oracle R Enterprise Installation

To verify that the basic functionality of Oracle R Enterprise is working, establish a connection to Oracle R Enterprise Server, execute several basic commands, and run some of the Oracle R Enterprise demo programs.

> **Note:**
>
> To start and use Oracle R Enterprise, your user ID must have the privileges required for Oracle R Enterprise installation. See User Requirements (page 4-8) for details.

Example A-2 Connecting to Oracle R Enterprise Server

To connect the Oracle R Enterprise Client to Oracle R Enterprise Server:

1. Select **R x64 3.3.0** from the Windows Start menu.

   The R Console is displayed.

2. Type this command to start Oracle R Enterprise:

```
> library(ORE)
```

3. Type this command to connect to the Oracle R Enterprise server. The following example connects user `rquser` to the database `orcl` on the server host `serv1` using port 1521:

```
>  ore.connect(user="rquser", sid="orcl", host="serv1", password="rquserpsw",
              port=1521, all=TRUE)
Loading required package: ROracle
Loading required package: DBI
```

4. Execute `ore.is.connected` to validate the connection. If the connection is successful, the command returns `TRUE`:

```
> ore.is.connected()
[1] TRUE
```

Example A-3 Listing the Database Tables Accessible to RQUSER

The `ore.ls` command lists the data sets that are available to the current user. For example, if `TABLE1` and `TABLE2` exist in the `rquser` schema:

```
> ore.ls()
[1] "TABLE1" "TABLE2"
```

Example A-4 Pushing an R Data Frame to a Database Table

The `ore.push` command pushes an R data frame to a database table or a database table to an R data frame. For example:

```
> cars <- ore.push(cars)
```

Example A-5 Executing an Embedded R Function

The `ore.doEval` command schedules execution of the specified function in the database-embedded R engine and returns the results.

```
> ore.doEval(function() { 123 })
[1] 123
```

Example A-6 Listing the Oracle R Enterprise Demo Scripts

The Oracle R Enterprise demo scripts are located in `$ORACLE_HOME/R/library/ORE/demo`. The `demo` command provides a list of available demos:

```
> demo(package="ORE")

Demos in package 'ORE':

aggregate                   Aggregation
analysis                    Basic analysis & data processing operations
basic                       Basic connectivity to database
binning                     Binning logic
columnfns                   Column functions
cor                         Correlation matrix
crosstab                    Frequency cross tabulations
datastore                   DataStore operations
datetime                    Date/Time operations
derived                     Handling of derived columns
distributions               Distribution, density, and quantile functions
do_eval                     Embedded R processing
esm                         Exponential smoothing method
freqanalysis                Frequency cross tabulations
glm                         Generalized Linear Models
graphics                    Demonstrates visual analysis
group_apply                 Embedded R processing by group
hypothesis                  Hyphothesis testing functions
matrix                      Matrix related operations
nulls                       Handling of NULL in SQL vs. NA in R
odm_ai                      Oracle Data Mining: attribute importance
odm_ar                      Oracle Data Mining: association rules
odm_dt                      Oracle Data Mining: decision trees
odm_em                      Oracle Data Mining: expectation maximization (12.2)
odm_esa                     Oracle Data Mining: explicit semantic analysis (12.2)
odm_glm                     Oracle Data Mining: generalized linear models
odm_kmeans                  Oracle Data Mining: enhanced k-means clustering
odm_nb                      Oracle Data Mining: naive Bayes classification
odm_nmf                     Oracle Data Mining: non-negative matrix factorization
odm_oc                      Oracle Data Mining: o-cluster
odm_partition               Oracle Data Mining: partition model (12.2)
odm_ralg                    Oracle Data Mining: extensible R algorithm (12.2)
odm_svd                     Oracle Data Mining: singular value decomposition (12.2
odm_svm                     Oracle Data Mining: support vector machines
ore_dplyr                   Data manipulation similar to dplyr
```

```
pca                     Principal Component Analysis
push_pull               RDBMS <-> R data transfer
randomForest            Random forest model
rank                    Attributed-based ranking of observations
reg                     Ordinary least squares linear regression
row_apply               Embedded R processing by row chunks
sampling                Random row sampling and partitioning of an ore.frame
script                  Create, list, load, drop, grant, and revoke R scripts
sql_like                Mapping of R to SQL commands
stepwise                Stepwise OLS linear regression
summary                 Summary functionality
table_apply             Embedded R processing of entire table
```

A.4.1 Executing Oracle R Enterprise Demo Scripts

You can further verify the success of the installation by running some of the Oracle R Enterprise demo scripts. If a script runs to completion without errors, then the demo is successful.

Example A-7 Executing the aggregate Demo

This example shows the `aggregate` demo with partial output.

```
> demo("aggregate", package="ORE")

        demo(aggregate)
        ---- ~~~~~~~~~

Type  <Return>   to start :

> #
> #     O R A C L E  R  E N T E R P R I S E  S A M P L E  L I B R A R Y
> #
> #     Name: aggregate.R
> #     Description: Demonstrates aggregations
> #     See also summary.R
> #
> #
> #
>
> ## Set page width
> options(width = 80)

> # Push the built-in iris data frame to the database
> IRIS_TABLE <- ore.push(iris)

> # Display the class of IRIS_TABLE
> class(IRIS_TABLE)
[1] "ore.frame"
attr(,"package")
[1] "OREbase"

> # Select count(Petal.Length) group by species
> x = aggregate(IRIS_TABLE$Petal.Length,
+               by = list(species = IRIS_TABLE$Species),
+               FUN = length)

> class(x)
[1] "ore.frame"
attr(,"package")
[1] "OREbase"
```

```
.
.
.
.
```

Example A-8 Executing the row_apply Demo

This example shows the `row_apply` demo with partial output.

```
> demo("row_apply", package="ORE")

        demo(row_apply)
        ---- ~~~~~~~~~

Type  <Return>   to start :

> #
> #     O R A C L E  R  E N T E R P R I S E  S A M P L E  L I B R A R Y
> #
> #     Name: row_apply.R
> #     Description: Execute R code on each row
> #
> #
>
> ## Set page width
> options(width = 80)

> # Push the built-in iris data frame to the database
> IRIS_TABLE <- ore.push(iris)

> # Display the class of IRIS_TABLE
> class(IRIS_TABLE)
[1] "ore.frame"
attr(,"package")
[1] "OREbase"

> # Apply given R function to each row
> ore.rowApply(IRIS_TABLE,
+              function(dat) {
+                  # Any R code goes here. Operates on one row of IRIS_TABLE at
+                  # a time
+                  cbind(dat, dat$Petal.Length)
+              })
$`1`
  Sepal.Length Sepal.Width Petal.Length Petal.Width   Species dat$Petal.Length
1          6.4         2.8          5.6         2.1 virginica              5.6

$`2`
  Sepal.Length Sepal.Width Petal.Length Petal.Width   Species dat$Petal.Length
1          7.2           3          5.8         1.6 virginica              5.8

$`3`
  Sepal.Length Sepal.Width Petal.Length Petal.Width   Species dat$Petal.Length
1          7.4         2.8          6.1         1.9 virginica              6.1

$`4`
  Sepal.Length Sepal.Width Petal.Length Petal.Width   Species dat$Petal.Length
1          7.9         3.8          6.4           2 virginica              6.4

$`5`
  Sepal.Length Sepal.Width Petal.Length Petal.Width   Species dat$Petal.Length
```

```
1          6.4         2.8          5.6         2.2 virginica              5.6

$`6`
.
.
.
.
```

Example A-9 Executing the cor Demo

This example shows the `cor` demo with partial output.

```
> demo ("cor")

        demo(cor)
        ---- ~~~

Type  <Return>   to start :

> #
> #     O R A C L E  R  E N T E R P R I S E  S A M P L E  L I B R A R Y
> #
> #     Name: cor.R
> #     Description: Correlation matrix
> #
> #
> #
>
> ## Set page width
> options(width = 80)

> # Push the built-in iris data frame to the database
> IRIS_TABLE <- ore.push(iris)

> # Display the class of IRIS_TABLE
> class(IRIS_TABLE)
[1] "ore.frame"
attr(,"package")
[1] "OREbase"

> # Remove non numeric columns
> iris_numeric = IRIS_TABLE[, c("Sepal.Length", "Sepal.Width",
+                               "Petal.Length", "Petal.Width")]

> # Pearson's correlation matrix
> cor(iris_numeric, use = "all.obs")
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000  -0.1175698    0.8717538   0.8179411
Sepal.Width    -0.1175698   1.0000000   -0.4284401  -0.3661259
Petal.Length    0.8717538  -0.4284401    1.0000000   0.9628654
Petal.Width     0.8179411  -0.3661259    0.9628654   1.0000000
.
.
.
.

Warning messages:
1: ORE object has no unique key - using random order
2: ORE object has no unique key - using random order
3: ORE object has no unique key - using random order
4: ORE object has no unique key - using random order
```

Example A-10 Executing the stepwise Demo

This example shows the `stepwise` demo with partial output.

```
> demo("stepwise")

        demo(stepwise)
        ---- ~~~~~~~~

Type  <Return>   to start :

> #
> #     O R A C L E  R  E N T E R P R I S E  S A M P L E  L I B R A R Y
> #
> #     Name: stepwise.R
> #     Description: STEPWISE Multivariate Regression
> #
> #
> #
>
> ## Set page width
> options(width = 80)

> # Push the built-in iris data frame to the database
> IRIS_TABLE <- ore.push(iris)

> # Display the class of IRIS_TABLE
> class(IRIS_TABLE)
[1] "ore.frame"
attr(,"package")
[1] "OREbase"

> # Let us first project out the non numeric columns
> IRIS_TABLE = IRIS_TABLE[, c("Sepal.Length", "Sepal.Width",
+                             "Petal.Length", "Petal.Width")]

> # Predict Sepal.Length based on the other 3 numeric columns
> # Do it stepwise
> model = ore.lm(Sepal.Length ~ ., data = IRIS_TABLE)

> model

Call:
ore.lm(formula = Sepal.Length ~ ., data = IRIS_TABLE)

Coefficients:
 (Intercept)   Sepal.Width  Petal.Length   Petal.Width
      1.8560        0.6508        0.7091       -0.5565
.
.
.
```

B

R Package Installation Tips

This appendix introduces some of the mechanics involved in working with R packages. If you are tasked with installing, uninstalling, or upgrading Oracle R Enterprise but you do not have extensive experience working with R packages, then you may find the information in this appendix helpful.

This appendix contains these topics:

B.1 R Package Installation Basics

You can install R packages from the R command line or from your system's command line.

R package installation basics are outlined in Chapter 6 of the *R Installation and Administration Guide*. The following example installs a package on Oracle Linux using Oracle R Distribution. It installs the `arules` package as root so that packages are installed in the default R system-wide location where all users can access it, `/usr/lib64/R/library`.

Within R, using the `install.packages` function always attempts to install the latest version of the requested package available on CRAN:

```
R> install.packages("arules")
```

If the `arules` package depends upon other packages that are not already installed locally, the R installer automatically downloads and installs those required packages. This is a huge benefit that frees users from the task of identifying and resolving those dependencies.

You can also install R from the shell command line. This is useful for some packages when an internet connection is not available or for installing packages not uploaded to CRAN. To install packages this way, first locate the package on CRAN and then download the package source to your local machine. For example:

```
$ wget http://cran.r-project.org/src/contrib/arules_1.1-9.tar.gz
```

Then, install the package using the command `R CMD INSTALL`:

```
$ R CMD INSTALL arules_1.1-9.tar.gz
```

A major difference between installing R packages using the R package installer at the R command line and shell command line is that package dependencies must be resolved manually at the shell command line. Package dependencies are listed in the

Depends section of the package's CRAN site. If dependencies are not identified and installed prior to the package's installation, you will see an error similar to:

```
ERROR: dependency 'xxx' is not available for package 'yyy'
```

As a best practice and to save time, always refer to the package's CRAN site to understand the package dependencies prior to attempting an installation.

If you don't run R as root, you won't have permission to write packages into the default system-wide location and you will be prompted to create a personal library accessible by your userid. You can accept the personal library path chosen by R, or specify the library location by passing parameters to the `install.packages` function. For example, to create an R package repository in your home directory:

```
R> install.packages("arules", lib="/home/username/Rpackages")
```

or

```
$ R CMD INSTALL arules_1.1-9.tar.gz --library=/home/username/Rpackages
```

Refer to the `install.packages` help file in R or execute `R CMD INSTALL --help` at the shell command line for a full list of command line options.

To set the library location and avoid having to specify this at every package install, simply create the R startup environment file `.Renviron` in your home area if it does not already exist, and add the following piece of code to it:

```
R_LIBS_USER = "/home/username/Rpackages"
```

B.2 Setting the R Repository

Each time you install an R package from the R command line, you are asked which CRAN mirror, or server, R should use. To set the repository and avoid having to specify this during every package installation, create the R startup command file .Rprofile in your home directory and add the following R code to it:

```
cat("Setting Seattle repository")
r = getOption("repos")
r["CRAN"] = "http://cran.fhcrc.org/"
options(repos = r)
rm(r)
```

This code snippet sets the R package repository to the Seattle CRAN mirror at the start of each R session

B.3 About R Package Installation for Oracle R Enterprise

Embedded R execution with Oracle R Enterprise allows the use of CRAN or other third-party R packages in user-defined R functions executed on the Oracle Database server. The steps for installing and configuring packages for use with Oracle R Enterprise are the same as for open source R. The database-side R engine just needs to know where to find the R packages.

The Oracle R Enterprise installation is performed by user oracle, which typically does not have write permission to the default site-wide library, /usr/lib64/R/library. On Linux and UNIX platforms, the Oracle R Enterprise Server installation provides the ORE

script, which is executed from the operating system shell to install R packages and to start R. The ORE script is a wrapper for the default R script, a shell wrapper for the R executable. It can be used to start R, run batch scripts, and build or install R packages. Unlike the default R script, the ORE script installs packages to a location writable by user oracle and accessible by all ORE users - $ORACLE_HOME/R/library.

To install a package on the database server so that it can be used by any R user and for use in embedded R execution, an Oracle DBA would typically download the package source from CRAN using wget. If the package depends on any packages that are not in the R distribution in use, download the sources for those packages, also.

For a single Oracle Database instance, replace the R script with ORE to install the packages in the same location as the Oracle R Enterprise packages.

```
$ wget http://cran.r-project.org/src/contrib/arules_1.1-9.tar.gz
$ ORE CMD INSTALL arules_1.1-9.tar.gz
```

Behind the scenes, the ORE script performs the equivalent of setting R_LIBS_USER to the value of $ORACLE_HOME/R/library, and all R packages installed with the ORE script are installed to this location. For installing a package on multiple database servers, such as those in an Oracle Real Application Clusters (Oracle RAC) or a multinode Oracle Exadata Database Machine environment, use the ORE script in conjunction with the Exadata Distributed Command Line Interface (DCLI) utility.

```
$ dcli -g nodes -l oracle ORE CMD INSTALL arules_1.1-9.tar.gz
```

The DCLI -g flag designates a file containing a list of nodes to install on, and the -l flag specifies the user id to use when executing the commands. For more information on using DCLI with Oracle R Enterprise, see Chapter 5 in the Oracle R Enterprise Installation Guide.

If you are using an Oracle R Enterprise client, install the package the same as any R package, bearing in mind that you must install the same version of the package on both the client and server machines to avoid incompatibilities.

B.4 About CRAN Task Views

CRAN also maintains a set of Task Views that identify packages associated with a particular task or methodology. Task Views are helpful in guiding users through the huge set of available R packages. They are actively maintained by volunteers who include detailed annotations for routines and packages. If you find one of the task views is a perfect match, you can install every package in that view using the ctv package - an R package for automating package installation.

To use the ctv package to install a task view, first, install and load the ctv package.

```
R> install.packages("ctv")

R> library(ctv)
```

Then query the names of the available task views and install the view you choose.

```
R> available.views()

R> install.views("TimeSeries")
```

Using and Managing Packages

To use a package, start up R and load packages one at a time with the library command.

Load the arules package in your R session.

R> library(arules)

Verify the version of arules installed.

To use a package, start up R and load packages one at a time with the library command.

Load the arules package in your R session.

R> library(arules)

Verify the version of arules installed.

R> packageVersion("arules")

[1] '1.1.9'

Verify the version of arules installed on the database server using embedded R execution.

R> ore.doEval(function() packageVersion("arules"))

View the help file for the apropos function in the arules

package

R> ?apropos

Over time, your package repository will contain more and more packages, especially if you are using the system-wide repository where others are adding additional packages. It's good to know the entire set of R packages accessible in your environment. To list all available packages in your local R session, use the installed.packages command:

R> myLocalPackages <- row.names(installed.packages())

R> myLocalPackages

C

Installing RStudio

This appendix provides tips for installing RStudio Server for use with Oracle R Enterprise on Linux. This appendix includes these topics:

C.1 About RStudio

RStudio is a free, open source Integrated Development Environment (IDE) for R. RStudio is available under GNU Affero General Public License (AGPL). You can use RStudio with Oracle R Enterprise, however RStudio is not included with Oracle R Enterprise. If you want to use RStudio, you must install and license it separately.

C.2 Installing RStudio Server

RStudio Server is a Linux application that provides a web-based interface to R on a server.

To install RStudio Server for use with Oracle R Enterprise:

1. Go to the RStudio website and navigate to the RStudio Server Download page. Download the server to your Linux system and follow the installation instructions.
2. Edit the configuration file `rserver.conf`. Supply the values of `RHOME` and `ORACLE_HOME`.

   ```
   sudo vi /etc/rstudio/rserver.conf
       rsession-ld-library-path=RHOME/lib:ORACLE_HOME/lib
   ```

   Note: The default value of `RHOME` on Linux is `/usr/lib64/R`.
3. Edit the configuration file `.Renviron`. Supply the values of `ORACLE_HOME`, `ORACLE_HOSTNAME`, and `ORACLE_SID`. For example, using the BASH shell:

   ```
   cd /home/oracle
   sudo vi .Renviron
       ORACLE_HOME=ORACLE_HOME
       ORACLE_HOSTNAME=ORACLE_HOSTNAME
       ORACLE_SID=ORACLE_SID

   export ORACLE_HOME
   export ORACLE_HOSTNAME
   export ORACLE_SID
   ```
4. Refer to the instructions for configuring the server. Return to the RStudio Server Download page, then navigate to the Configuring the Server article in the RStudio documentation.

5. To access the Oracle R Distribution 3.3.0 help within RStudio, modify as sudo or root the file /`usr/lib/rstudio-server/R/modules/SessionHelp.R`.

   Update `httpdPortIsFunction` to the following:

```
.rs.addFunction( "httpdPortIsFunction", function() {
   getRversion() >= "3.3"
})
```

C.3 Installing RStudio Desktop

RStudio Desktop is an IDE for standalone machines.

To install RStudio Desktop:

- 1. Install R.
  2. Go to the RStudio website, navigate to the RStudio Desktop Download page, and download RStudio Desktop.
  3. Run the installer and follow the prompts.
  4. Click the desktop icon to initialize RStudio.
  5. To access the Oracle R Distribution 3.2.0 help within RStudio, modify as Administrator the file *`RStudio Home Directory`*`\R\modules\SessionHelp.R`. In the following example, RStudio is installed in the `Program Files` folder on the C drive:

     ```
     C:\Program Files\RStudio\R\modules\SessionHelp.R
     ```

     Update `httpdPortIsFunction` to the following:

     ```
     .rs.addFunction( "httpdPortIsFunction", function() {
        getRversion() >= "3.2"
     })
     ```

D

Oracle R Distribution Packages

The table in this section lists the packages in Oracle R Distribution that are used by Oracle R Enterprise.

> **See Also:**
>
> - Table 6-1 (page 6-2) for a list of the packages in Oracle R Enterprise
> - Table 6-2 (page 6-2) for a list of the open source packages that ship with Oracle R Enterprise

Table D-1 Oracle R Distribution Packages Used by Oracle R Enterprise

Package Name	Package Description
`base`	The R Base Package
`boot`	Bootstrap Functions (originally by Angelo Canty for S)
`class`	Functions for Classification
`cluster`	Cluster Analysis Extended Rousseeuw et al
`codetools`	Code Analysis Tools for R
`compiler`	The R Compiler Package
`datasets`	The R Datasets Package
`foreign`	Read Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, dBase
`graphics`	The R Graphics Package
`grDevices`	The R Graphics Devices and Support for Colours and Fonts
`grid`	The Grid Graphics Package
`KernSmooth`	Functions for kernel smoothing for Wand & Jones (1995)
`lattice`	Lattice Graphics
`MASS`	Support Functions and Datasets for Venables and Ripley's MASS
`Matrix`	Sparse and Dense Matrix Classes and Methods
`methods`	Formal Methods and Classes
`mgcv`	GAMs with GCV/AIC/REML smoothness estimation and GAMMs by PQL
`nlme`	Linear and Nonlinear Mixed Effects Models
`nnet`	Feed-forward Neural Networks and Multinomial Log-Linear Models
`parallel`	Support for parallel computation, including random-number generation
`RFO`	Classification based on a forest of trees using random inputs
`rpart`	Recursive Partitioning
`spatial`	Functions for Kriging and Point Pattern Analysis

Table D-1 (Cont.) Oracle R Distribution Packages Used by Oracle R Enterprise

Package Name	Package Description
splines	Regression Spline Functions and Classes
stats	The R Stats Package
stats4	Statistical Functions using S4 Classes
survival	Survival analysis, including penalised likelihood.
tcltk	Tcl/Tk Interface
tools	Tools for Package Development
translation	Bindings for the Google Translate API v2
utils	The R Utils Package

E

License Information for Oracle R Enterprise

This appendix contains licensing information for third-party and open source products that are used in combination with Oracle R Enterprise. Licensing information for Oracle R Enterprise is in *Oracle Database Licensing Information*.

This appendix contains these topics:

E.1 Licensing for Open Source R

R is an open source language and environment that is governed by GPL2 and not under the terms of the Oracle license agreement.

R was initially written by Robert Gentleman and Ross Ihaka of the Statistics Department of the University of Auckland.

Since mid-1997 there has been a core group with write access to the R source, currently consisting of:

Douglas Bates
John Chambers
Peter Dalgaard
Seth Falcon
Robert Gentleman
Kurt Hornik
Stefano Iacus
Ross Ihaka
Friedrich Leisch
Uwe Ligges
Thomas Lumley
Martin Maechler
Duncan Murdoch
Paul Murrell
Martyn Plummer
Brian Ripley
Deepayan Sarkar
Duncan Temple Lang
Luke Tierney
Simon Urbanek

plus Heiner Schwarte up to October 1999 and Guido Masarotto up to June 2003.

For more information go to `http://www.r-project.org`.

Current R-core members can be contacted via email to `R-project.org` with name made up by replacing spaces by dots in the name listed above.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License version 2 as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

E.1.1 GNU GENERAL PUBLIC LICENSE Version 2

June 1991

Copyright © 1989, 1991 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of itin new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is

not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.

This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

E.1.2 Code derived from software contributed to Berkeley by Guido van Rossum

Copyright © 1989, 1993, The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A

PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

E.1.3 FIG: Facility for Interactive Generation of figures

Copyright © 1985-1988 by Supoj Sutanthavibul
Parts Copyright © 1989-2002 by Brian V. Smith
Parts Copyright © 1991 by Paul King
Parts Copyright © 1992 by James Tough
Parts Copyright © 1998 by Georg Stemmer
Parts Copyright © 1995 by C. Blanc and C. Schlick

Any party obtaining a copy of these files is granted, free of charge, a full and unrestricted irrevocable, world-wide, paid up, royalty-free, nonexclusive right and license to deal in this software and documentation files (the "Software"), including without limitation the rights to use, copy, modify, merge, publish and/or distribute copies of the Software, and to permit persons who receive copies from any such party to do so, with the only requirement being that this copyright notice remain intact.

E.1.4 unzip.h -- IO for uncompress .zip files using zlib

Version 1.01e, February 12th, 2005

Copyright © 1998-2005 Gilles Vollant

This unzip package allow extract file from .ZIP file, compatible with PKZip 2.04g WinZip, InfoZip tools and compatible.

Multi volume ZipFile (span) are not supported.

Encryption compatible with pkzip 2.04g only supported

Old compressions used by old PKZip 1.x are not supported

I WAIT FEEDBACK at mail info@winimage.com

Visit also `http://www.winimage.com/zLibDll/unzip.htm` for evolution

Condition of use and distribution are the same than zlib:

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an

acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

E.2 Licensing for Oracle R Distribution

The licensing for Oracle R Distribution is the same as the licensing for open source R described in Licensing for Open Source R (page E-1).

E.3 Licensing for ROracle

ROracle is an open source package for R that allows R users to interact with an Oracle database. ROracle was developed by an individual called David James. Oracle has taken over new development, maintenance, and all upgrade activity on this package.

ROracle is licensed under LGPL v.2 or later and not under the terms of your Oracle license agreement. For more information see:

http://cran.cnr.berkeley.edu/web/packages/ROracle/ROracle.pdf

E.3.1 GNU Lesser General Public License Version 2.1

February 1999

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and

use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example,

permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a. The modified work must itself be a software library.

b. You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c. You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d. If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place

satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you

(whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT

NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the library's name and an idea of what it does.

Copyright © year name of author

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

signature of Ty Coon, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

Index

P

R

S

U

Y