Contents

Preface

Technology Rebrand vii
Audience vii
Related Documents vii
Documentation Accessibility vii
Conventions viii

Changes in Oracle Machine Learning for R Installation and Administration Guide

Changes in this Guide for Release 1.5.1 ix

1   Overview of Oracle Machine Learning for R Installation

Oracle Machine Learning for R Architecture 1-1
Client and Server Components of Oracle Machine Learning for R 1-2
Oracle Machine Learning for R Installation Steps 1-2
Oracle Machine Learning for R System Requirements 1-5

2   Install and Configure the Database for Oracle Machine Learning for R

Install Oracle Database for Oracle Machine Learning for R 2-1
Configure EXTPROC for Embedded R Execution 2-1
   About EXTPROC 2-1
   About EXTPROC Configuration for OML4R 2-2
   Troubleshooting EXTPROC 2-3

3   Install R for Oracle Machine Learning for R

About R and Oracle Machine Learning for R 3-1
   About ROracl 3-1
   Oracle R Distribution and OML4R 3-2
   Open Source R and OML4R 3-2
Install Oracle R Distribution on Linux 3-2
  Install Oracle R Distribution on Oracle Linux Using Yum 3-3
  Install Oracle R Distribution on Oracle Linux Using RPMs 3-6
    Oracle R Distribution 3.3.0 RPMs for Oracle Linux 7 3-6
    Oracle R Distribution 3.3.0 RPMs for Oracle Linux 6 3-6
  Install Oracle R Distribution on Red Hat Enterprise Linux 3-7
Install Oracle R Distribution on Oracle Solaris 3-8
Install Oracle R Distribution on IBM AIX 3-9
Install Oracle R Distribution on Microsoft Windows 3-12
Configure Oracle R Distribution to Use MKL on the Client 3-12
  Enable MKL Support for Oracle R Distribution on a Linux Client 3-12
    Modifying the Number of Threads for MKL on Linux 3-13
  Enable MKL Support for Oracle R Distribution on a Windows Client 3-13
    Modify the Number of Threads for MKL on Windows 3-14
Uninstall Oracle R Distribution 3-14
  Uninstall Oracle R Distribution on Windows 3-14
  Uninstall Oracle R Distribution on Linux 3-14
  Uninstalling Oracle R Distribution on Oracle Solaris 3-15
  Uninstall Oracle R Distribution on IBM AIX 3-15

4  Install Oracle Machine Learning for R Server

  About Oracle Machine Learning for R Server 4-1
    About the RQSYS Schema 4-2
    Security Best Practices for OML4R 4-2
  Oracle Machine Learning for R Server Requirements 4-3
    System Requirements 4-3
    Environment Variables 4-3
    User Requirements 4-4
      About Operating System Authentication 4-4
      Verify the Group Membership of Your User ID 4-5
  Install Oracle Machine Learning for R Server for Oracle Database 18c and Later 4-5
    Prepare to Use OML4R Server with R-3.6.1 or Later 4-6
    Install OML4R Server Using rqcfg.sql 4-7
  Install Oracle Machine Learning for R Server for Oracle Database 12c and Earlier 4-8
    About the Server Script 4-10
      Overview of Server Script Operations 4-11
      Server Script Syntax 4-11
      Server Script Examples 4-12
    Verify the OML4R Server Installation 4-13
5 Install Oracle Machine Learning for R on Exadata

About Oracle Machine Learning for R on Exadata 5-1
Install Oracle Machine Learning for R on Exadata Using DCLI 5-2
  Install Oracle R Distribution Across Exadata Compute Nodes Using DCLI 5-3
    DCLI Command Summary for Oracle R Distribution installation on Exadata 5-5
  Install OML4R Server Across Exadata Compute Nodes Using DCLI for 12c and Earlier 5-5
    DCLI Commands Summary for Oracle Machine Learning for R Server 5-10
  Install OML4R Server Across Exadata Compute Nodes Using DCLI for 18c and Later 5-7
Install Oracle Machine Learning for R for Oracle RAC Without DCLI 5-10

6 Install Oracle Machine Learning for R Client

About OML4R Client 6-1
  About Oracle Database Client Software 6-1
  About the OML4R Packages 6-2
  About the OML4R Supporting Packages 6-2
Install Oracle Database Instant Client 6-3
  Installing Oracle Database Instant Client on Windows 6-3
  Install Oracle Database Instant Client on Linux or UNIX 6-4
    Install Oracle Instant Client from a Zip File 6-4
    Install Oracle Instant Client on Linux from RPMs 6-5
Install the Oracle Machine Learning for R Packages 6-6
  Install the OML4R Packages on Windows 6-6
  Install the OML4R Packages on Linux or UNIX 6-7
Install the OML4R Supporting Packages 6-8
  Install the Supporting Packages on Windows 6-8
  Install the Supporting Packages on Linux or UNIX 6-10
Connect OML4R Client to OML4R Server 6-12

7 Administrative Tasks for Oracle Machine Learning for R

Install Oracle R Distribution on Linux in a Non-Default R_HOME 7-1
Upgrade Oracle Machine Learning for R 7-2
Migrate Oracle Machine Learning for R Data 7-3
Migrate Oracle Machine Learning for R After a Database Upgrade 7-3
Uninstall Oracle Machine Learning for R 7-4
  Uninstall OML4R Server from Oracle Database 18c or Later 7-5
  Uninstall OML4R Server from Oracle Database 12c and Earlier 7-5
    Performing a Partial Uninstall 7-6
Performing a Full Uninstall 7-6
Uninstall OML4R Client 7-7
Install Additional R Packages on Linux or UNIX 7-7
Create a Database User for Oracle Machine Learning for R 7-7
About the RQADMIN Role 7-8
Create and Modify Environment Variables on Windows 7-9
Create an Oracle Wallet for an Oracle Machine Learning for R Connection 7-12
Control Memory Used by Embedded R 7-14

A Sample Installation of Oracle Machine Learning for R

A Sample Installation of Oracle Machine Learning for R Sample Installation Environment A-1
Install Oracle Machine Learning for R on the Server A-2
Verify the Environment A-2
Install Oracle R Distribution A-3
Install Oracle Machine Learning for R Server A-4
Install Oracle Machine Learning for R on the Client A-6
Install Oracle R Distribution on the Windows Client A-6
Install Oracle Instant Client A-7
Install the Oracle Machine Learning for R Packages A-8
Install the Oracle Machine Learning for R Supporting Packages A-9
Verifying the Oracle Machine Learning for R Installation A-10

R Package Installation Tips

R Package Installation Basics B-1
Set the R Repository B-2
About R Package Installation for Oracle Machine Learning for R B-2
About CRAN Task Views B-3

Installing RStudio

About RStudio C-1
Install RStudio Server C-1
Install RStudio Desktop C-2

Oracle R Distribution Packages

Index
Preface

This document explains how to install and administer Oracle Machine Learning for R (OML4R) Release 1.5.1.

Technology Rebrand

Oracle R Enterprise is now Oracle Machine Learning for R (OML4R).

Oracle has rebranded the suite of products and components that support machine learning with Oracle Database and Big Data. This technology is now known as Oracle Machine Learning (OML).

The OML application programming interface for R, previously under the name Oracle R Enterprise, is now named Oracle Machine Learning for R (OML4R). The package, class, and function names are not rebranded. They remain ORE, OREbase, ore.frame, ore.connect, and so on.

The OML application programming interfaces for SQL include PL/SQL packages, SQL functions, and data dictionary views. Using these APIs is described in publications, previously under the name Oracle Data Mining, that are now named Oracle Machine Learning for SQL (OML4SQL). The PL/SQL package and database view names are not rebranded. They remain DBMS_DATA_MINING, ALL_MINING_MODELS, and so on.

The Oracle R Advanced Analytics for Hadoop (ORAAH) technology is now Oracle Machine Learning for Spark (OML4Spark).

For more information, see Oracle Machine Learning.

Audience

This document is intended for anyone who is responsible for installing or administering Oracle Machine Learning for R.

Installation of OML4R requires knowledge of R and knowledge of Oracle Database.

Related Documents

The Oracle Machine Learning for R documentation set includes the following publications.

- Oracle Machine Learning for R Release Notes
- Oracle Machine Learning for R Licensing Information User Manual
- Oracle Machine Learning for R User’s Guide

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.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><code>monospace</code></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
Changes in Oracle Machine Learning for R Installation and Administration Guide

Changes for OML4R Release 1.5.1.

Oracle has rebranded the suite of products and components that support machine learning with Oracle Database and Big Data. This technology is now known as Oracle Machine Learning (OML).

The OML application programming interface for R, previously under the name Oracle R Enterprise, is now named Oracle Machine Learning for R (OML4R). The package, class, and function names are not rebranded. They remain ORE, OREbase, ore.frame, ore.connect, and so on.

Changes in this Guide for Release 1.5.1

Installation changes for Oracle Machine Learning for R Release 1.5.1.

For information about other new features in OML4R Release 1.5.1, see Changes in This Release for Oracle Machine Learning for R in Oracle Machine Learning for R User’s Guide.

Installation Script for OML4R Server

Beginning with Oracle Database 18c, the OML4R Server installation script, rqcfg.sql, is part of the database. For more information, see Install Oracle Machine Learning for R Server for Oracle Database 18c and Later.

Data Manipulation Package

The package OREdplyr contains OML4R functions for data manipulation such as filtering for rows, selecting specific columns, re-ordering rows, adding new columns, and summarizing data.

Updated Supporting Packages

Updated supporting packages are DBI and ROracle.

The OML4R 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
R-3.3.0 or R-3.6.1 Requirement

OML4R 1.5.1 requires R-3.3.0 or R-3.6.1. As with earlier releases of OML4R, Oracle recommends that you use Oracle R Distribution.

Note:

R-3.6.1 has an OML4R 1.5.1 binary built under R-3.6.1 and is not compatible with the OML4R 1.5.1 built under R-3.3.0.

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.

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 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 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 build 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.

See Also:

Install R for Oracle Machine Learning for R for information about installing R and Oracle R Distribution
Overview of Oracle Machine Learning for R

This chapter introduces the OML4R installation process. This chapter contains the following topics:

Oracle Machine Learning for R Architecture

OML4R has a client/server architecture based on Oracle Database and Oracle Client.

R engines run on the server computer and on each client computer. OML4R supports three key capabilities:

• **R Transparency**
  OML4R packages on the client support R transparency, which enables Oracle tables to appear “transparently” as native R objects. OML4R 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 for in-database execution.

• **Predictive Analytics and Machine Learning**
  OML4R 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**
  OML4R packages, libraries, and R and SQL APIs on the server support the execution of user-defined R functions within SQL queries and PL/SQL statements. Embedded R execution spawns R engines that can run in parallel, for data-parallel and task-parallel execution. With embedded R execution, you can execute user-defined R functions, possibly leveraging third-party packages. With facilities like the `DBMS_SCHEDULER` database package, you can schedule the execution of user-defined R functions for lights-out processing.

Figure 1-1  Client/Server Architecture of OML4R

This figure illustrates the client/server architecture of OML4R.
Client and Server Components of Oracle Machine Learning for R

Lists the client and server components of OML4R.

• OML4R Client Components:
  – Oracle Database Client
  – OML4R packages and supporting packages

• OML4R Server Components:
  – Oracle Database with schema objects and shared libraries for supporting OML4R clients
  – OML4R packages and supporting packages

Oracle Machine Learning for R Installation Steps

These steps and this roadmap illustrate a typical OML4R installation.

For Oracle Database 18c or later, use the rqcfg.sql script to install OML4R Server, as described in Install Oracle Machine Learning for R Server for Oracle Database 18c and Later.

For Oracle Database 12c or earlier, use the OML4R Server script to install OML4R Server, as described in Install Oracle Machine Learning for R Server for Oracle Database 12c and Earlier. That installation script can install the supporting packages and create a database user along with the installation of OML4R Server.

Note:

If you intend to use both client and server components of OML4R 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  OML4R Client and Server Installation Steps

This figure illustrates the OML4R client and server installation steps.
OML4R Installation Roadmap

This roadmap provides the steps required to install and configure a typical OML4R environment. To install OML4R, 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 OML4R

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Review the OML4R sample installation.</td>
<td>Review the steps for a typical installation of OML4R on a Linux server and a Windows client. Note: All the supported configurations are listed in Oracle Machine Learning for R System Requirements.</td>
<td>A Sample Installation of Oracle Machine Learning for R</td>
</tr>
<tr>
<td>2. Verify supported platforms and system requirements.</td>
<td>Use the Platform Requirements table and Server Support Matrix to verify your environment meets the requirements for installation.</td>
<td>Oracle Machine Learning for R System Requirements</td>
</tr>
</tbody>
</table>
## Table 1-1  (Cont.) Tasks for Installing OML4R

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. Identify installation users for Oracle R Distribution and OML4R.</td>
<td>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).</td>
<td>User Requirements for OML4R Server</td>
</tr>
</tbody>
</table>
| 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 OML4R are available on Oracle’s Technology Network (OTN). For Oracle Database 18c and later, the OML4R installers are shipped with Oracle Database. | Oracle Database Software Downloads
Oracle R Distribution Downloads
Oracle Machine Learning for R Downloads |
| 5. Install and configure Oracle Database.                            | OML4R requires the 64-bit version of Oracle Database Enterprise Edition.                        | Install and Configure the Database for Oracle Machine Learning for R                                     |
| 6. Install and configure R.                                          | OML4R 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 OML4R Server components must be installed on the database server. | Install R for Oracle Machine Learning for R                                                                |
| 7. Install and configure OML4R Server.                              | OML4R includes several components on the server. Together these components enable an OML4R client to interact with an OML4R server. | Install Oracle Machine Learning for R Server
Install Oracle Machine Learning for R on Exadata |
| 8. Install Open Source R packages on the OML4R server.               | Embedded R execution with OML4R 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 Machine Learning for R |
| 9. Install and configure the OML4R client.                           | If a physical client is configured, then you must install the following OML4R components separately on each client computer:  
  - R 
  - Oracle Instant Client 
  - OML4R Client packages 
  - OML4R Client Supporting packages  
  If you wish to run the OML4R client through a web browser, then install RStudio Server on the database server (Linux only). | Install Oracle Machine Learning for R Client
Installing RStudio                                                  |
### Table 1-1  (Cont.) Tasks for Installing OML4R

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>10. Install Open Source R packages on the OML4R client.</td>
<td>R packages installed on the OML4R server must also be installed on the OML4R client.</td>
<td>R Package Installation Basics</td>
</tr>
<tr>
<td>11. Verify the OML4R Installation.</td>
<td>Test the OML4R installation by connecting to the OML4R client to the server and executing some OML4R functions.</td>
<td>Verify the OML4R Server Installation</td>
</tr>
</tbody>
</table>

---

### Oracle Machine Learning for R System Requirements

OML4R 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 Machine Learning for R Platform Requirements

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Hardware Platform</th>
<th>Description</th>
</tr>
</thead>
</table>
| Linux x86-64     | Intel and AMD     | • 64-bit Oracle Linux Releases 6, 7, and 8  
|                  |                   | • 64-bit Red Hat Enterprise Linux Releases 6 and 7 |
| Oracle Solaris on x86-64 (64-Bit) | Intel and SPARC | Oracle Solaris may be running on Oracle Exadata Database Machine.  
|                |                   | • 64-bit Oracle Solaris 10 update 11 through Oracle Solaris 11 for both SPARC and x86-64 (Intel) platforms  
| Oracle Solaris on SPARC-64 (64-Bit) |           | • Oracle SPARC SuperCluster  
|                |                   | • Oracle Solaris Studio (formerly Sun Studio) 12u3 or later  
| 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 Professional  

**Note:**

Starting with R-3.6.1, Linux 6 is no longer supported.

---

The following table shows the supported configurations of OML4R Server components. Oracle recommends that you use Oracle R Distribution, Oracle’s free distribution of R, with OML4R. You should install Oracle R Distribution before installing OML4R.
### Table 1-3  Oracle Machine Learning for R Configuration Requirements and Server Support Matrix

<table>
<thead>
<tr>
<th>OML4R Version</th>
<th>Open Source R or Oracle R Distribution</th>
<th>Oracle Database Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5.1</td>
<td>3.3.0, 3.6.1</td>
<td>11.2.0.4, 12.1.0.1, 12.1.0.2, 12.2.0.1, 18c, 19c, 21c</td>
</tr>
<tr>
<td>1.5</td>
<td>3.2.0</td>
<td>11.2.0.4, 12.1.0.1, 12.1.0.2</td>
</tr>
<tr>
<td>1.4.1</td>
<td>3.0.1, 3.1.1</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1, 12.1.0.2</td>
</tr>
<tr>
<td>1.4</td>
<td>2.15.2, 2.15.3, 3.0.1</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1</td>
</tr>
<tr>
<td>1.3.1</td>
<td>2.15.1, 2.15.2, 2.15.3</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1</td>
</tr>
<tr>
<td>1.3</td>
<td>2.15.1</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1</td>
</tr>
<tr>
<td>1.2</td>
<td>2.15.1</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1</td>
</tr>
<tr>
<td>1.1</td>
<td>2.13.2</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1</td>
</tr>
<tr>
<td>1.0</td>
<td>2.13.2</td>
<td>11.2.0.3, 11.2.0.4, 12.1.0.1</td>
</tr>
</tbody>
</table>

#### Note:
- The version of R must be the same on the server and on each client computer. Also, the version of OML4R must be the same on the server and on each client computer.

#### Note:
- R-3.6.1 has an OML4R 1.5.1 binary built under R-3.6.1 and is not compatible with the OML4R 1.5.1 built under R-3.3.0.

#### Note:
- Starting with Oracle Database 18c, to install OML4R you must use the installation scripts that come with the database. See [Install Oracle Machine Learning for R Server for Oracle Database 18c and Later](#).

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

**Verifying 64-Bit Architecture on Microsoft Windows**

OML4R only runs on 64-bit operating systems. You can determine if your Windows system is 64-bit by following these steps.
• On Windows 10:
  – Hold down the **Windows** key and press **I**.
  – In the Settings window, under **Device specifications**, find **System type**. This entry shows whether the system is 64 bit.

• On other versions of Windows, refer to the Windows documentation to find out how to display the bit version for your operating system.

**Related Topics**

• **Oracle R Distribution and OML4R**
  Oracle recommends that you use Oracle R Distribution, Oracle’s free distribution of R, with OML4R.
Install and Configure the Database for Oracle Machine Learning for R

This chapter explains how to install and configure Oracle Database to support OML4R Server.

This chapter contains these topics:

Install Oracle Database for Oracle Machine Learning for R

Installation instructions for Oracle Database.

OML4R requires the 64-bit version of Oracle Database Enterprise Edition or Standard Edition 2. For the supported platforms, see Oracle Machine Learning for R System Requirements.

To install Oracle Database, follow the installation instructions for your supported platform:

1. Go to the Oracle Database Documentation page in Oracle Help Center.
2. Select the version of Oracle Database to install.
3. In the Topics section, select Install and Upgrade.
4. In the section for your operating system, select the appropriate installation guide.

Note:

You can install OML4R 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.

Configure EXTPROC for Embedded R Execution

Oracle Database uses an external procedure agent named extproc to support external procedures.

An external procedure is a procedure invoked from a program that is written in a different language. OML4R uses extproc to support embedded R execution.

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.

About EXTPROC Configuration for OML4R

OML4R 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 OML4R, 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 execute only external procedures for OML4R:

```bash
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).

Enable extproc tracing by doing the following:

1. To your /extproc.ora file, add the following statement:

   ```bash
   SET TRACE_LEVEL=ON
   ```

2. Restart the database.
   Traces for all extproc operations are now recorded in the log files in the $ORACLE_HOME/log directory.

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:

```bash
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.

See Also:
Troubleshooting EXTPROC

Calling an OML4R embedded R function may result in an error if a database configuration problem exists.

If an attempt to call an OML4R 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 OML4R 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.
Install R for Oracle Machine Learning for R

This chapter explains how to install R for OML4R.

This chapter contains these topics:

**Related Topics**

- Install Oracle Machine Learning for R on Exadata

### About R and Oracle Machine Learning for R

OML4R 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.

**Note:**

The version of R must be the same on the server and on each client computer. Also, the version of OML4R must be the same on the server and on each client computer.

**See Also:**

- The table of configuration requirements and server support in *Oracle Machine Learning for R System Requirements* for the versions of R that are supported with OML4R
- *Oracle Machine Learning for R Licensing Information User Manual*
- R Project for Statistical Computing

### 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 Machine Learning for R. The supporting packages are introduced in *Client and Server Components of Oracle Machine Learning for R* and described in Table 6-2.
Oracle R Distribution and OML4R

Oracle recommends that you use Oracle R Distribution, Oracle's free distribution of R, with OML4R.

Oracle R Distribution offers significant advantages for OML4R.

Why Oracle R Distribution?

- Oracle R Distribution simplifies the installation of R for OML4R.
- Oracle R Distribution is supported by Oracle for customers of Oracle Machine Learning, 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 Configure Oracle R Distribution to Use MKL on the Client.)
- 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.

Open Source R and OML4R

Although Oracle recommends that you use Oracle R Distribution whenever possible, you can use open source R with OML4R.

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`

On UNIX systems, additional operating system packages may be required for R's configuration and build process to succeed.

See Also:

*The R Installation and Administration* manual at R Manuals for information about building R from source

Install Oracle R Distribution on Linux

Instructions for installing 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 Machine Learning for R, as described in the table of platform requirements in Oracle Machine Learning for R System Requirements. 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.

The following topics describe installing Oracle R Distribution:

Related Topics

- **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`.

**Install 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:
   ```bash
   # 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:
   ```bash
   ```

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-ol6.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)
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-3.6.1, use the command `yum install R-3.6.1`.

```
# 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 Machine Learning for R. Consult the table of configuration requirements and server support in Oracle Machine Learning for R System Requirements to determine which version of R you should use.

5. On Linux 6, install the R-core-extra RPM; for `rversion`, specify the version of R that you are installing:

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

For example, for R-3.3.0, the command is `yum install R-core-extra-3.3.0`. 
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:

```bash
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 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 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:**

- [https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-Rversion.el6.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-Rversion.el6.x86_64.rpm)
- [https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-devel-Rversion.el6.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-devel-Rversion.el6.x86_64.rpm)

**Oracle Linux 7:**

- [https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-Rversion.el7.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-Rversion.el7.x86_64.rpm)
- [https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-Rversion.el7.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-Rversion.el7.x86_64.rpm)
- [https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-extra-Rversion.el7.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-core-extra-Rversion.el7.x86_64.rpm)
- [https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-devel-Rversion.el7.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/R-devel-Rversion.el7.x86_64.rpm)
- [https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-Rversion.el7.x86_64.rpm](https://public-yum.oracle.com/repo/OracleLinux/OL7/addons/x86_64/getPackage/libRmath-Rversion.el7.x86_64.rpm)
Install Oracle R Distribution on Oracle Linux Using RPMs

If yum is not available due to lack of internet access, then you can install the RPMs directly and resolve the dependencies manually.

However, Oracle recommends that you use yum to install Oracle R Distribution, because yum automatically resolves RPM dependencies.

To download and install the RPMs, log in as root and execute the command `rpm -Uvh rpm_name` for each RPM listed in the following sections:

Oracle R Distribution 3.3.0 RPMs for Oracle Linux 7

Lists the Oracle R Distribution RPMs for Oracle Linux 7.

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

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

Oracle R Distribution 3.3.0 RPMs for Oracle Linux 6

Lists the Oracle R Distribution RPMs for Oracle Linux 6.

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

https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-3.3.0-2.el6.x86_64.rpm
https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-core-3.3.0-2.el6.x86_64.rpm
https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-core-extra-3.3.0-2.el6.x86_64.rpm
https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/R-devel-3.3.0-2.el6.x86_64.rpm
https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-3.3.0-2.el6.x86_64.rpm
https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-devel-3.3.0-2.el6.x86_64.rpm
https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64/getPackage/libRmath-static-3.3.0-2.el6.x86_64.rpm
Install Oracle R Distribution on Red Hat Enterprise Linux

Instructions on rebuilding the Oracle R Distribution RPMs on a Red Hat Linux system.

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:

\( R_{version} \) represents the version of Oracle R Distribution. Replace \( R_{version} \) with the \( R \) version you want to build. For example, replace \( R_{version} \) with \( \text{rpm -i /refresh/home/rpmbuild/RPMS/x86_64/R-core-3.3.0-2.el6.x86_64.rpm} \) for R-3.3.0.

To install Oracle R Distribution on Red Hat Enterprise Linux:

1. Create an RPM build directory structure:
   
   ```bash
   mkdir -p /rpmbuild/{BUILD,RPMS,SOURCES,SPECS,SRPMS}
   ```

2. Set up RPM tools to use your own build tree (to avoid root):
   
   ```bash
   echo '"$_topdir $(echo $HOME)/rpmbuild"' > /.rpmmacros
   ```

3. From Oracle public yum, download the source RPM (\( R_{version} \).elx.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 \( \text{rpmbuild/SRPMS} \) directory.

4. Rebuild Red Hat Enterprise Linux using `rpmbuild`.
   
   ```bash
   rpmbuild --rebuild /rpmbuild/SRPMS/R-\( R_{version} \).elx.src.rpm
   ```

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

   The binary RPMs are built and saved under \( /\text{rpmbuild/RPMS} \).

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

   ```bash
   # rpm -i path/rpmbuild/RPMS/R-\( R_{version} \).elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/R-core-\( R_{version} \).elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/libRmath-\( R_{version} \).elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/libRmath-devel-\( R_{version} \).elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/libRmath-static-\( R_{version} \).elx.x86_64.rpm
   # rpm -i path/rpmbuild/RPMS/R-devel-\( R_{version} \).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
```

## Install Oracle R Distribution on Oracle Solaris

Instructions for installing 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 Machine Learning for R, as described in the table of platform requirements in Oracle Machine Learning for R System Requirements. 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

Install Oracle R Distribution on IBM AIX

Instructions for installing Oracle R Distribution on IBM AIX.

Before installing Oracle R Distribution, verify that your version of IBM AIX is supported by Oracle Machine Learning for R, as described in the table of platform requirements in Oracle Machine Learning for R System Requirements. 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
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:
     
     ```bash
     $ export LIBPATH=/opt/freeware/lib64:/opt/freeware/lib:$LIBPATH
     ```
   • For csh:
     
     ```bash
     $ 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:

     ```bash
     $ 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.

   ```bash
   $ 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORD</td>
<td>Install all of the filesets in Oracle R Distribution.</td>
</tr>
<tr>
<td>ORD.core</td>
<td>Installs only the ORE.core fileset.</td>
</tr>
<tr>
<td>ORD.devel</td>
<td>Installs only the ORE.devel fileset, which requires the ORE.core fileset.</td>
</tr>
</tbody>
</table>

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

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

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

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

8. Add `installation_path/usr/bin` to the `PATH` environment variable:

   • For ksh:

     ```bash
     $ export PATH=installation_path/usr/bin:$PATH
     ```
Install Oracle R Distribution on Microsoft Windows

Instructions for installing Oracle R Distribution on Microsoft Windows.

Before installing Oracle R Distribution, verify that your version of Microsoft Windows is supported by Oracle Machine Learning for R, as described in the table of platform requirements in Oracle Machine Learning for R System Requirements.

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.

Configure Oracle R Distribution to Use MKL on the Client

Instructions for configuring 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:

Enable 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, $HOME/lib, and $ORACLE_HOME/lib to the LD_LIBRARY_PATH system environment variable. For example, in the Bash shell:
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`).

### 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:

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

### Enable 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.

   **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
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`).

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

Uninstall Oracle R Distribution

Instructions for uninstalling Oracle R Distribution.

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

Uninstall Oracle R Distribution on Windows

Instructions for 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.

Uninstall Oracle R Distribution on Linux

Instructions for uninstalling Oracle R Distribution on Linux.

To uninstall Oracle R Distribution on Linux, log in as root and execute the commands in the example in the order shown. 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 7:

```
rpm -e R-3.3.0-2.el7
rpm -e R-devel
rpm -e R-core
rpm -e R-core-extra
rpm -e libRmath-devel
rpm -e libRmath
rpm -e libRmath-static
```

Uninstalling Oracle R Distribution on Oracle Solaris

Instructions for uninstalling 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`

Uninstall Oracle R Distribution on IBM AIX

Instructions for uninstalling Oracle R Distribution on 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:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORD</td>
<td>Uninstall all of the filesets in Oracle R Distribution.</td>
</tr>
<tr>
<td>ORD.core</td>
<td>Uninstalls only the ORE.core fileset.</td>
</tr>
<tr>
<td>ORD.devel</td>
<td>Uninstall only the ORE.devel fileset.</td>
</tr>
</tbody>
</table>

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
```
Install Oracle Machine Learning for R Server

This chapter explains how to install and administer OML4R Server. This chapter includes these topics:

Related Topics
• Install Oracle Machine Learning for R on Exadata

About Oracle Machine Learning for R Server

OML4R includes components on the Oracle Database server that enable an OML4R client to interact with OML4R Server.

Note:
The version of OML4R must be the same on the server and on each client computer. Also, the version of R must be the same on the server and on each client computer.

The components are:
• Oracle R Distribution or open source R
• OML4R Server
  – The RQSYS schema
  – Metadata and executable code in sys
  – OML4R Server libraries in \$ORACLE_HOME/lib (Linux and UNIX) or %ORACLE_HOME%\bin (Windows)
  – OML4R R packages in \$ORACLE_HOME/R/library (%ORACLE_HOME%\R\library on Windows)

The OML4R 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 OML4R Packages).

See the following topics for additional information:

See Also:
• Oracle Machine Learning for R Configuration Requirements and Server Support Matrix for a list of supported R and OML4R versions.
• Figure 1-2 for an illustration of the server and client components of OML4R.
About the RQSYS Schema

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

It contains metadata, PL/SQL packages, and other executable code that is used internally by OML4R Server.

The OML4R Server installation process creates RQSYS as a locked account with an expired password. The rqsys user does not have the CREATE SESSION privilege.

Security Best Practices for OML4R

To minimize the risk of compromising the security of an OML4R Server in Oracle Database, Oracle recommends the following security best practices.

OML4R Server components in an Oracle Database instance include the locked and password-expired RQSYS schema, which contains and manages OML4R metadata. Users connect to OML4R Server through their database connection credentials. The RQADMIN role grants a user the privilege of creating R functions as scripts in the OML4R R script repository; those scripts can be executed using OML4R 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 OML4R user.
• Use parameters or the OML4R 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 OML4R 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
  The server script installation process creates a database role called RQADMIN.
• Control Memory Used by Embedded R
  How to control the memory used by embedded R execution.
Oracle Machine Learning for R Server Requirements

Before installing OML4R Server, verify your system environment, and ensure that your user ID has the proper permissions.

You should also have installed the OML4R Server prerequisites: Oracle Database and Oracle R Distribution or open source R.

Related Topics
- Install and Configure the Database for Oracle Machine Learning for R
- Install R for Oracle Machine Learning for R

System Requirements

Lists the system requirements for OML4R Server.

- The operating system must conform to the requirements specified in Oracle Machine Learning for R System Requirements.
- Oracle Database must be installed and configured as described in Install and Configure the Database for Oracle Machine Learning for R.

Note:
You can install OML4R Server in a pluggable database (PDB) in a multitenant environment. See Oracle Database Administrator's Guide.

- R must be installed as described in Install R for Oracle Machine Learning for R.

Environment Variables

Lists the environment variables required by OML4R Server.

Table 4-1    Environment Variable Requirements for Oracle Machine Learning for R Server

<table>
<thead>
<tr>
<th>Platform</th>
<th>Environment Variable Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>$ORACLE_SID must specify the service identifier (SID) of the database that will support OML4R.</td>
</tr>
<tr>
<td></td>
<td>$ORACLE_HOME must specify the home directory of the database identified by ORACLE_SID.</td>
</tr>
<tr>
<td></td>
<td>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 Create and Modify Environment Variables on Windows.</td>
</tr>
<tr>
<td>Linux</td>
<td>$LD_LIBRARY_PATH must include $ORACLE_HOME/lib.</td>
</tr>
<tr>
<td></td>
<td>$PATH must include $ORACLE_HOME/bin.</td>
</tr>
</tbody>
</table>
Table 4-1 (Cont.) Environment Variable Requirements for Oracle Machine Learning for R Server

<table>
<thead>
<tr>
<th>Platform</th>
<th>Environment Variable Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Solaris</td>
<td>$LD_LIBRARY_PATH must include $ORACLE_HOME/lib. $PATH must include $ORACLE_HOME/bin.</td>
</tr>
<tr>
<td>IBM AIX</td>
<td>$LIBPATH must include $ORACLE_HOME/lib. $PATH must include $ORACLE_HOME/bin.</td>
</tr>
<tr>
<td>Microsoft Windows</td>
<td>%PATH% must include %R_HOME%\bin\x64. The default value of %R_HOME% is C:\Program Files\R\R-3.2.0.</td>
</tr>
<tr>
<td></td>
<td>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 Create and Modify Environment Variables on Windows.</td>
</tr>
</tbody>
</table>

User Requirements

Lists the requirements for the operating system user who installs OML4R Server.

Table 4-2 User Requirements for OML4R Server Installer

<table>
<thead>
<tr>
<th>Platform</th>
<th>User Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux and UNIX</td>
<td>• Member of the dba group</td>
</tr>
<tr>
<td></td>
<td>• Has write access to $ORACLE_HOME/lib</td>
</tr>
<tr>
<td>Microsoft Windows</td>
<td>• Administrator access</td>
</tr>
<tr>
<td></td>
<td>• Member of the ora_dba group</td>
</tr>
<tr>
<td></td>
<td>• Has write access to %ORACLE_HOME%\bin</td>
</tr>
</tbody>
</table>

See the following topics for additional information:

About Operating System Authentication

Describes the operating system authentication used by OML4R Server.

The OML4R 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 OML4R 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 OML4R Server must belong to OSDBA.
Verify the Group Membership of Your User ID

Describes how to determine the group memberships required by OML4R Server.

As described in “About Operating System Authentication”, the Linux or UNIX user ID that runs the OML4R Server installation script must belong to the `dba` group. Membership in the `dba` group is also required for running other OML4R 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`.

Install Oracle Machine Learning for R Server for Oracle Database 18c and Later

Instructions for installing OML4R Server for Oracle Database 18c or later.

As of Oracle Database 18c, the `rqcfg.sql` installation script for OML4R Server is part of the database. The script is in the `$ORACLE_HOME/R/server/` directory.

The `rqcfg.sql` script enables the OML4R Server components that are part of the database, configures some aspects of the server, and installs some OML4R database objects.

Before installing OML4R Server, install R, as described in Install R for Oracle Machine Learning for R.

The OML4R Server packages are in the `$ORACLE_HOME/R/library/` directory. For Oracle Database 18c and 19c, the server packages in Oracle Database are built under R-3.3.0 and are compatible with R-3.3.0. If you are using R-3.3.0, you can install OML4R Server by running the `rqcfg.sql` installation script.

For use with R-3.6.1 or later, the OML4R Server packages are built under R-3.6.1. To use OML4R with R-3.6.1, you must download and install those OML4R Server packages before running the `rqcfg.sql` installation script.
Prepare to Use OML4R Server with R-3.6.1 or Later

Instructions for downloading and installing the OML4R Server packages built under R-3.6.1.

In Oracle Database 18c and 19c, the OML4R Server packages are in the `$ORACLE_HOME/R/library/` directory. For use with R-3.6.1, you can either rename the OML4R packages in that directory or you can delete them. You then download and install the OML4R Server packages that are compatible with R-3.6.1.

These instructions rename the original OML4R server packages and replace them with ones built under R-3.6.1.

1. Rename the OML4R packages in the `$ORACLE_HOME/R/library/` directory.

   ```
   $ cd $ORACLE_HOME/R/library
   $ mv ORE ORE.orig
   $ mv OREbase OREbase.orig
   $ mv OREcommon OREcommon.orig
   $ mv OREdm OREdm.orig
   $ mv OREdplyr OREdplyr.orig
   $ mv OREeda OREeda.orig
   $ mv OREembed OREembed.orig
   $ mv OREgeographics OREgeographics.orig
   $ mv OREmodels OREmodels.orig
   $ mv OREpredict OREpredict.orig
   $ mv OREserver OREserver.orig
   $ mv OREstats OREstats.orig
   $ mv ORExml ORExml.orig
   ```

2. Go to the Oracle Machine Learning for R Downloads page, accept the license agreement, and download the R-3.6.1 compatible OML4R Server packages to an installation directory, such as `/oml4rserver_install_dir/`.

   ```
   $ cd /oml4rserver_install_dir/
   $ unzip ore-server-platform-arch-version.zip
   ```

Note:
Use the same installation directory for all OML4R components.

3. Go to the installation directory and unzip the downloaded file.

   ```
   $ cd /oml4rserver_install_dir/
   $ unzip ore-server-platform-arch-version.zip
   ```
When you unzip the file, the `/server` directory is created and these files are extracted to it:

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

4. Go to the `/oml4rserver_install_dir/server` directory and execute the following commands:

```
$ cd /oml4rserver_install_dir/server
$ ORE CMD INSTALL ORE_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREbase_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREcommon_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREdm_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREdplyr_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREeda_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREembed_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREgraphics_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREmodels_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREpredict_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREserver_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL OREstats_version_R_arch-unknown-platform-gnu.tar.gz
$ ORE CMD INSTALL ORExml_version_R_arch-unknown-platform-gnu.tar.gz
```

If you are installing OML4R Server for the first time, then run the `rqcfg.sql` script. If you already have installed OML4R Server and are upgrading R from R-3.3.0 to R-3.6.1, then you do not need to, and should not, run the `rqcfg.sql` script.

### Install OML4R Server Using rqcfg.sql

Instructions for installing OML4R Server for Oracle Database 18c and later.

After installing R, install OML4R Server by running the `rqcfg.sql` script and providing values for the arguments in the following table.

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>define permtbl</td>
<td>Specify a permanent tablespace for the RQSYS schema.</td>
</tr>
<tr>
<td>define temptbl</td>
<td>Specify a temporary tablespace.</td>
</tr>
</tbody>
</table>
Table 4-3  (Cont.) Arguments to the rqcfg.sql Installation Script

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>define orahome</td>
<td>Specify the ORACLE_HOME directory.</td>
</tr>
<tr>
<td>define rhome</td>
<td>Specify the R_HOME directory.</td>
</tr>
</tbody>
</table>

1. At your operating system prompt, start SQL*Plus, log in to your PDB directly and run the rqcfg.sql script. To capture the log, spool the installation steps to an external file. The following example uses the PDB PDB1 and gives example values for the script arguments.

   $ sqlplus / as sysdba
   SQL> spool install.txt
   SQL> alter session set container=PDB1;
   SQL> ALTER PROFILE DEFAULT LIMIT PASSWORD_VERIFY_FUNCTION NULL;
   SQL> @$ORACLE_HOME/R/server/rqcfg.sql

   define permtbl = SYSAUX
   define temptbl = TEMP
   define orahome = /u01/app/oracle/product/18.0.0.0/dbhome_1
   define rhome = /usr/lib64/R

   Open the install.txt file to see if any errors occurred.

2. At your operating system prompt, go to the ORACLE_HOME/bin directory and grant read and execute permission to all users to the ORE directory.

   cd $ORACLE_HOME/bin
   chmod 755 ORE

3. Download and install the OML4R 1.5.1 supporting packages for your system as described in Install the OML4R Supporting Packages

Related Topics

- Uninstall OML4R Server from Oracle Database 18c or Later

Install Oracle Machine Learning for R Server for Oracle Database 12c and Earlier

Instructions for installing OML4R Server for Oracle Database 12c or earlier.

**Note:**

To install OML4R Server without needing to respond to visual prompts, use a batch mode installation such as that described in A Default Batch Installation.
To install OML4R Server:

1. Ensure that your system satisfies the requirements specified in Oracle Machine Learning for R Server Requirements.

2. Create an installation directory for the OML4R Server components. The directory can have any name. For example:

   /oml4r_server_install_dir

3. Download the OML4R Server installation files and supporting packages from the Oracle Machine Learning for R Downloads page on the Oracle Technology Network.

   a. Accept the license agreement and download the OML4R Server files for your platform to your installation directory.

   b. Accept the license agreement and download the OML4R 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:

   /oml4r_server_install_dir
   ore-server-platform-arch-version.zip
   ore-supporting-platform-arch-version.zip
   server.sh
   /server
   /supporting

   For Windows:

   \oml4r_server_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 OML4R Server, as described in A Default Interactive Installation. For details about the server script, see About the Server Script.

   For Linux or UNIX:

   ./server.sh

   For Windows:

   server.bat
**Note:**

Beginning in R-3.3.0, on an 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 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 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 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.

The following topic describes the OML4R Server script for Oracle Database 12c and earlier.

**About the Server Script**

A single script called server manages the installation and administration of OML4R Server for Oracle Database 12c and earlier releases.

You can rerun the server script whenever you need to install, uninstall, upgrade, or configure server-side components of OML4R.

The following topics provide details about the script:
Overview of Server Script Operations

Describes the operations you can perform with the server script.

The server script supports the following operations:

- Installs OML4R Server
- Uninstalls OML4R Server
- Upgrades OML4R 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:

- Install the OML4R Supporting Packages
- Create a Database User for Oracle Machine Learning for R

Server Script 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 OML4R 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 4-4  Server Script Command-Line Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-i, --install</td>
<td>Install or upgrade OML4R Server.</td>
</tr>
<tr>
<td></td>
<td>An installation or upgrade includes the following by default:</td>
</tr>
<tr>
<td></td>
<td>• Installation of the supporting packages if they are present.</td>
</tr>
<tr>
<td></td>
<td>• Creation or configuration of a database user if one does not exist.</td>
</tr>
<tr>
<td>-u, --uninstall</td>
<td>Uninstall OML4R Server:</td>
</tr>
<tr>
<td></td>
<td>• 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).</td>
</tr>
<tr>
<td></td>
<td>• 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).</td>
</tr>
<tr>
<td></td>
<td>See Uninstall Oracle Machine Learning for R.</td>
</tr>
<tr>
<td>-s, --setup-user</td>
<td>Create or configure a database user for OML4R (the default).</td>
</tr>
<tr>
<td>-y</td>
<td>Never prompt.</td>
</tr>
<tr>
<td>--ask</td>
<td>Interactive mode (the default).</td>
</tr>
<tr>
<td>--keep</td>
<td>When uninstalling OML4R Server, keep the R packages and libraries under Oracle home but remove the database objects. Allows OML4R support to be removed from a single database instance or pluggable database (PDB) without affecting other databases in Oracle home.</td>
</tr>
<tr>
<td></td>
<td>See Performing a Partial Uninstall.</td>
</tr>
<tr>
<td>--full</td>
<td>When uninstalling OML4R Server, remove the R packages and libraries under Oracle home in addition to the database objects.</td>
</tr>
<tr>
<td></td>
<td>See Performing a Full Uninstall.</td>
</tr>
<tr>
<td>--no-supp</td>
<td>When combined with --install, prevents installation of the supporting packages. By default the supporting packages are installed if they are available.</td>
</tr>
<tr>
<td>--supp</td>
<td>Install supporting packages (the default).</td>
</tr>
<tr>
<td>--pdb NAME</td>
<td>The name of a pluggable database (PDB) in a multitenant container database (CDB).</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>--perm PERM</td>
<td>Permanent tablespace for RQSYS.</td>
</tr>
<tr>
<td>--temp TEMP</td>
<td>Temporary tablespace for RQSYS.</td>
</tr>
<tr>
<td>--user-perm PERM</td>
<td>Permanent tablespace for OML4R user.</td>
</tr>
<tr>
<td>--user-temp TEMP</td>
<td>Temporary tablespace for OML4R user.</td>
</tr>
<tr>
<td>--user USER</td>
<td>OML4R database user name.</td>
</tr>
</tbody>
</table>

**Server Script Examples**

**Example of using the server script.**

See the following topics for examples of using the server script:
A Default Interactive Installation

Example of a default installation of OML4R Server.

If your Linux or UNIX or Windows system meets the requirements specified in System Requirements, then this command performs a default, first-time installation of OML4R Server:

For Linux or UNIX:

    ./server.sh

For Windows:

    server.bat

As shown in Example A-1, 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 OML4R. (Creates a user by default if one does not exist.) When creating a user, prompts for the permanent and temporary tablespaces.

A Default Batch Installation

Example of installing OML4R Server in batch mode.

This example shows an installation like the one in A Default Interactive Installation, but specified to run in batch mode.

For Linux or UNIX:

    ./server.sh  -y  --install  --setup-user  --perm SYSAUX  --temp TEMP
               --user-perm USERS  --user-temp TEMP  --user OML_USER

For Windows:

    server.bat  -y  --install  --setup-user  --perm SYSAUX  --temp TEMP
               --user-perm USERS  --user-temp TEMP  --user OML_USER

Verify the OML4R Server Installation

To verify the success of an OML4R Server installation for Oracle Database 12c and earlier using the server.sh script, you can view the log files. For an Oracle Database 18c and later installation, the SQL script rqcfg.sql returns any errors encountered.

For any installation, you can execute some functions to verify a successful installation.
For 12c and Earlier, View Log Files

The OML4R Server installation script server.sh 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:

```bash
cd ./oml4r_server_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 R Technologies in Data Warehousing.

Example 4-1    Run Examples to Verify the Server Installation

First execute these commands from an R instance directly on the database server and then execute them from the OML4R client.

Start R using the ORE script and load the ORE library.

```r
% ORE
> library(ORE)
```

Connect to the server. This example connects as the user OML_USER.

```r
ore.connect("OML_USER", password="OML_USERpsw", conn_string="", all=TRUE)
```

Execute some functions.

```r
## Is the OML4R client connected to the OML4R server?
## The output of this function should be TRUE.
ore.is.connected()

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

## Push an R dataframe to a database table.
df <- data.frame(a="abc",
                 b=1.456,
                 c=TRUE,
                 d=as.integer(1))
of <- ore.push(df)

## Run the self-contained example code in the help files associated with
## the following functions.
example("ore.glm") # Builds an OML4R generalized linear model.
example("ore.stepwise") # Builds an OML4R stepwise least squares regression model.
example("ore.odmAI")  # Builds an OML4SQL attribute importance
model.
example("ore.doEval")  ## Runs an embedded R execution function.

Related Topics

• Connect OML4R Client to OML4R Server
  Instructions for connecting to an OML4R server.

Install OML4R Server in a Multitenant Environment

Information about installing OML4R Server in a Multitenant Environment.

You can install OML4R Server in one or more pluggable databases (PDBs) within a multitenant environment. OML4R Server must be installed in a pluggable database, not in the root database.

If you have more than one instance of OML4R 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.

See Also:

• Oracle Machine Learning for R User’s Guide for information about connecting to a pluggable database.
• Oracle Database Concepts for an introduction to multitenant architecture.
• Oracle Database Administrator’s Guide for information about managing a multitenant environment.
This chapter explains how to install Oracle R Distribution and OML4R Server on Oracle Exadata Database Machine. This chapter includes these topics:

About Oracle Machine Learning for R on Exadata

Exadata is an ideal platform for OML4R.

The parallel resources of R computations in OML4R take advantage of the massively parallel grid infrastructure of Exadata.

**Note:**

The version of OML4R must be the same on the server and on each client computer. Also, the version of R must be the same on the server and on each client computer. See the Oracle Machine Learning for R Configuration Requirements and Server Support Matrix for supported configurations.

**To install OML4R on Exadata:**

1. On each node:
   - Install Oracle R Distribution
   - Verify and configure the environment
   - Install OML4R Server and the supporting packages

2. On the first node only, create an OML4R user, if desired. Alternatively, configure an existing database user to use OML4R. See Create a Database User for Oracle Machine Learning for R.

You can simplify the process of installing OML4R on Exadata by using the Distributed Command Line Interface (DCLI).

**Related Topics**

- Install R for Oracle Machine Learning for R
- Oracle Machine Learning for R Server Requirements
- Install Oracle Machine Learning for R Server for Oracle Database 12c and Earlier
- Create a Database User for Oracle Machine Learning for R
- Install Oracle Machine Learning for R on Exadata Using DCLI
Install Oracle Machine Learning for R on Exadata Using DCLI

Using DCLI can simplify the installation of OML4R on Exadata.

With DCLI, you can use a single command to install Oracle R Distribution and OML4R 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
The following topics describe installing OML4R components using DCLI:

Install Oracle R Distribution Across Exadata Compute Nodes Using DCLI

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.

Important:
Before beginning the installation, review the instructions for installing Oracle R Distribution in Install R for Oracle Machine Learning for R.

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. Install Oracle R Distribution using yum if an internet connection is available. Otherwise, install the Oracle R Distribution and operating system dependencies manually. 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.
   
   a. Log in to My Oracle Support.
   
   b. Click **Contact Us**.
   
   c. If yum and internet access are unavailable, request access to this file through My Oracle Support.

   ```
   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 "Install Oracle R Distribution on Oracle Linux Using RPMs".

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
   ```

   For each node, the following command returns the output shown.

   ```
   $ 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.
   ```
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
```

Install OML4R Server Across Exadata Compute Nodes Using DCLI for 12c and Earlier

How to use DCLI to install OML4R Server across multiple Exadata Linux compute nodes for Oracle Database 12c and Earlier.

The DCLI commands are summarized in **DCLI Commands Summary for Oracle Machine Learning for R Server**.

---

**Note:**

Before beginning the installation, review the instructions for installing OML4R Server in **Install Oracle Machine Learning for R Server**.
To install OML4R Server on Exadata using DCLI for Oracle Database 12c and earlier, 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:$ORACLE_HOME/lib:$HOME_lib:$R_HOME/port/Linux-X64/lib
   ```

2. Go to the Oracle Machine Learning for R Downloads website.
   On the Downloads page, in the Linux 64-bit row, select Server, accept the license agreement, and download the file. To download the supporting packages, select Supporting, accept the license agreement, and download the file. The following files are downloaded for OML4R, where version is the OML4R, 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 OML4R Server and the supporting packages across nodes. For example:

   ```
   $ dcli -g nodes -l oracle mkdir -p /home/oracle/OML4R
   $ dcli -g nodes -l oracle -f ore-server-linux-x86-64-version.zip -d /home/oracle/OML4R/ore-server-linux-x86-64-version.zip
   $ dcli -g nodes -l oracle -f ore-supporting-linux-x86-64-version.zip -d /home/oracle/OML4R/ore-supporting-linux-x86-64-version.zip
   ```

4. Unzip the OML4R Server bundle on each node:

   ```
   $ dcli -t -g nodes -l oracle unzip /home/oracle/OML4R/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/OML4R/ore-supporting-linux-x86-64-version.zip -d /my_destination_directory/
   ```

6. Install OML4R Server components:

   ```
   $ dcli -t -g nodes -l oracle "cd /my_destination_directory; ./server.sh -y --perm permtablespace --temp temptablespace --user-perm usertablespace --user-temp usertemptablespace --user OML_USER"
   ```
Note:

The server script creates a user for OML4R. By default, the script does not grant the RQADMIN role to the user.

Any OML4R 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.

7. Verify OML4R loads.

```r
> library(ORE)
Loading required package: OREbase
Attaching package: OREbase
The following objects are masked from package:base:
  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
```

See Also:

Verify the OML4R Server Installation

Install OML4R Server Across Exadata Compute Nodes Using DCLI for 18c and Later

How to use DCLI to install OML4R Server across multiple Exadata Linux compute nodes for Oracle Database 18c and later.

To install OML4R Server on Exadata using DCLI for Oracle Database 18c and later, follow these steps:

1. Get a list of the compute nodes in the rack.

   In the following example, the `cat nodes` command lists the nodes for a two-node cluster.

   ```bash
   $ cat nodes
   exadb01
   exadb02
   ```
2. In a text editor, create a file that contains the names of all of the compute nodes in the rack. Specify each node name on a separate line. For example, the nodes file for a two-node cluster would contain entries such as the following:

```
exadb01
exadb02
```

3. 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 in which you will run the DCLI script. For example, you could specify values like the following in a bashrc file:

```
export ORACLE_HOME=/u01/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:$ORACLE_HOME/lib:$R_HOME/lib:$R_HOME/port/Linux-X64/lib
```

4. Option 1: On the first database node only, execute as sysdba the `rqcfg.sql` script from your PDB.

```
$ sqlplus / as sysdba;
SQL> alter session set container=PDBNAME;
SQL> @$ORACLE_HOME/R/server/rqcfg.sql
```

**Note:**

The `rqcfg.sql` script ships with Oracle Database 18c and later and resides in the `$ORACLE_HOME/R/server` directory. The script installs the OML4R Server components in the database and you only need to execute it once.

The `rqcfg.sql` script prompts you for the following input parameters:

- `define permtbl = permanent tablespace name for RQSYS schema`
- `define temptbl = temporary tablespace name for RQSYS schema`
- `define orahome = ORACLE_HOME path`
- `define rhome = R_HOME path`

Option 2: Execute the `rqcfg.sql` script from the Linux command line.

In the example, the user is system with the password `apassword`, the RQSYS schema is in SYSAUX and SYSAUX is assigned the temporary tablespace TEMP. The value for `ORACLE_HOME` is `/u01/app/oracle/product/18.0.0.0/dbhome_1` and the value for `R_HOME` is the Linux default path, `/usr/lib64/R`:

```
$ sqlplus -L -S system/apassword @$ORACLE_HOME/R/server/rqcfg.sql
SYSAUX TEMP /u01/app/oracle/product/18.0.0.0/dbhome_1 /usr/lib64/R
```
5. Download and install the OML4R supporting packages.

To download the supporting packages, go to the Oracle Machine Learning for R Downloads website. Select Supporting in the column for your version of the database, accept the license agreement, and download the ore-supporting-linux-x86-64-version.zip file.

Log in as root and copy the installers for the supporting packages across the nodes. For example:

$ dcli -g nodes -l oracle mkdir -p /home/oracle/OML4R

$ dcli -g nodes -l oracle -f ore-supporting-linux-x86-64-version.zip -d /home/oracle/OML4R/ore-supporting-linux-x86-64-version.zip

Unzip the supporting packages on each node:

$ dcli -t -g nodes -l oracle unzip /home/oracle/OML4R/ore-supporting-linux-x86-64-version.zip -d /my_destination_directory/

Install the OML4R supporting packages, as in the following example:

$ dcli -t -g nodes -l oracle R CMD INSTALL /my_destination_directory/supporting/* -l $ORACLE_HOME/R/library/  

Note:

The rqcfg.sql script creates an OML4R user. By default, the script does not grant the RQADMIN role to the user.

Any OML4R user can use an embedded R execution function, but only those with the RQADMIN role can create and drop the R scripts in the OML4R script repository in the database. Use caution when granting the RQADMIN role.

6. Verify the OML4R loads.

$ ORE

> library(ORE)
Loading required package: OREbase
Attaching package: OREbase
The following objects are masked from package:base:
   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
DCLI Commands Summary for Oracle Machine Learning for R Server

The DCLI commands used to install OML4R and the supporting packages on a Linux Exadata system are listed in the following example.

Example 5-3  DCLI Command Summary for OML4R 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 OML_USER;
exit;

dcli -t -g nodes -l oracle ORE -e "library(ORE)"

Related Topics

- Security Best Practices for OML4R
To minimize the risk of compromising the security of an OML4R Server in Oracle Database, Oracle recommends the following security best practices.

Install Oracle Machine Learning for R for Oracle RAC Without DCLI

How to install OML4R for an Oracle Real Application Clusters (Oracle RAC) database if DCLI is unavailable.

If the Distributed Command Line Interface (DCLI) is not available, you must install each of the following components individually on each database instance in the Oracle RAC cluster.

- R or Oracle R Distribution
- OML4R Server
- OML4R supporting packages

The first section contains installation instructions for Oracle Database 18c and later. The second section has instructions for Oracle Database 12c and earlier.

Install OML4R in an Oracle 18c and Later RAC Environment

Following these step to install Oracle R Distribution, OML4R, and the OML4R supporting packages.

1. Install Oracle R Distribution. See Install R for Oracle Machine Learning for R.
2. Start SQL*Plus, log in to your PDB directly and run the rqcfg.sql script. The following example uses the PDB PDB1 and gives example values for the script arguments.

```
SQL> sqlplus / as sysdba
SQL> alter session set container=PDB1;
SQL> ALTER PROFILE DEFAULT LIMIT PASSWORD_VERIFY_FUNCTION NULL;
SQL> @$ORACLE_HOME/R/server/rqcfg.sql
define permtbl = SYSAUX
define temptbl = TEMP
define orahome = /u01/app/oracle/product/18.0.0.0/dbhome_1
define rhome = /usr/lib64/R
```

3. At your operating system prompt, go to the ORACLE_HOME/bin directory and grant read and execute permission to all users to the ORE directory.

```
cd $ORACLE_HOME/bin
chmod 755 ORE
```

4. Create a directory to contain the OML4R 1.5.1 supporting packages for your system and change directories to it. To that directory, download the supporting package zip file as described in Install the OML4R Supporting Packages.

5. Extract the supporting packages.

6. For each package, at your operating system command prompt, run the following command.

```
ORE CMD INSTALL package
```

## Install OML4R in an Oracle 12c and Earlier RAC Environment

Following these steps to install Oracle R Distribution, OML4R, and the OML4R supporting packages.

> **Note:**
>
> You can perform steps 2 and 3 simultaneously by first extracting the OML4R supporting packages bundle in the same directory from which you execute the server.sh script. (For Microsoft Windows, the script is server.bat.)

1. Install Oracle R Distribution. See Install R for Oracle Machine Learning for R.

2. Execute the server.sh script from the OML4R Server installer bundle. See Install Oracle Machine Learning for R Server

3. Install the OML4R supporting packages. See Install Oracle Machine Learning for R Server for Oracle Database 12c and Earlier

When you execute the server.sh script on node 1, it installs the OML4R packages on the operation system in the $ORACLE_HOME/R/library directory. It also installs and configures the database components of OML4R. While running the script, you can create a new database user when prompted to do so. You can create a user while running the script only during the execution of the server.sh script on the first node.
When you execute the `server.sh` script on each subsequent node, the script only installs the OML4R packages on the operation system.
Install Oracle Machine Learning for R Client

This chapter explains how to install OML4R Client. This chapter includes these topics:

About OML4R Client

Lists the components of OML4R Client.

OML4R includes several components that must be installed separately on each client computer.

Note:

The version of OML4R must be the same on the server and on each client computer. Also, the version of R must be the same on the server and on each client computer.

Components of OML4R Client

- R (See Install R for Oracle Machine Learning for R)
- Oracle Database Client Software
- OML4R packages
- OML4R supporting packages

The OML4R Client components can be installed in any order.

The following sections have information about the components.

See Also:

- Oracle Machine Learning for R Configuration Requirements and Server Support Matrix for a list of supported R and OML4R versions.
- Figure 1-2 for an illustration of the client and server components of OML4R

About Oracle Database Client Software

ROracle requires an installation of Oracle Database client.

ROracle is one of the supporting packages used by OML4R. It requires an installation of Oracle Database client software to enable communication between an R client and an Oracle Database instance. 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.

### About the OML4R Packages

The OML4R packages are a set of Oracle proprietary packages that support OML4R. These packages are required on each client computer and on the server computer that hosts OML4R Server. On the server, the OML4R packages are installed automatically by the OML4R Server installation script.

**Note:**

The version of the OML4R packages on the client must match the version of the OML4R packages on the server.

#### Table 6-1  OML4R Packages

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORE</td>
<td>The top-level package for OML4R.</td>
</tr>
<tr>
<td>OREbase</td>
<td>Corresponds to the open source R base package.</td>
</tr>
<tr>
<td>OREcommon</td>
<td>Contains common low-level functionality for OML4R.</td>
</tr>
<tr>
<td>OREedm</td>
<td>Exposes Oracle Data Mining algorithms through R.</td>
</tr>
<tr>
<td>OREdplyr</td>
<td>Transparently implements dplyr data manipulation functions for ore.frame and ore.numeric objects.</td>
</tr>
<tr>
<td>OREeda</td>
<td>Contains functions for exploratory data analysis.</td>
</tr>
<tr>
<td>OREembed</td>
<td>Supports embedded R.</td>
</tr>
<tr>
<td>OREgraphics</td>
<td>Corresponds to the open source R graphics package.</td>
</tr>
<tr>
<td>OREmodels</td>
<td>Contains functions for advanced analytical modeling.</td>
</tr>
<tr>
<td>OREPredict</td>
<td>Enables scoring data in Oracle Database using R models.</td>
</tr>
<tr>
<td>ORESTats</td>
<td>Corresponds to the open source R stats package.</td>
</tr>
<tr>
<td>ORExml</td>
<td>Supports XML translation between R and Oracle Database.</td>
</tr>
</tbody>
</table>

### About the OML4R Supporting Packages

The supporting packages are a set of open source packages that support the OML4R packages.

#### Table 6-2  OML4R Supporting Packages

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arules</td>
<td>Provides the infrastructure for representing, manipulating, and analyzing transactional data and patterns (frequent itemsets and association rules).</td>
</tr>
</tbody>
</table>
Table 6-2  (Cont.) OML4R Supporting Packages

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cairo</td>
<td>Supports graphic rendering on an OML4R server.</td>
</tr>
<tr>
<td>DBI</td>
<td>A database interface definition for communication between R and Oracle Database.</td>
</tr>
<tr>
<td>png</td>
<td>Supports the reading and writing of PNG images for OML4R objects.</td>
</tr>
<tr>
<td>randomForest</td>
<td>Provides classification and regression based on a forest of trees using random inputs.</td>
</tr>
<tr>
<td>ROracle</td>
<td>Oracle Database interface for R-based OCI.</td>
</tr>
<tr>
<td>statmod</td>
<td>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.</td>
</tr>
</tbody>
</table>

Install Oracle Database Instant Client

OML4R requires Oracle Database client software.

Oracle Instant Client is suitable for most configurations of OML4R.

This topic includes these sections:

Related Topics

- About Oracle Database Client Software
  ROracle requires an installation of Oracle Database client.

Installing Oracle Database Instant Client on Windows

Instructions for installing Oracle Database Instant Client on Windows.

To Install Oracle Instant Client on Windows:

1. Create an installation directory for the OML4R client components. For example:
   ```
c:\oml4rclient_install_dir
   ```
2. Go to the Oracle Database Instant Client website.
3. In the Get Oracle Instant Client section, select Downloads.
4. On the Oracle Instant Client Downloads page, select Instant Client for Microsoft Windows (x64).
5. Choose Instant Client Package - Basic for your version of Oracle Database.
6. Save the file in the installation directory that you created in Step 1. For example:
   ```
c:\oml4rclient_install_dir\instantclient-basic-windows.x64-12.1.0.2.0.zip
   ```
7. Unzip the file. The files are extracted into a subdirectory called instantclient_version, where version is your version of Oracle Database. For example:
   ```
c:\oml4rclient_install_dir\instantclient_12_1
   ```
8. Return to the Instant Client Downloads for Microsoft Windows (x64) page.
9. 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.
    
    ```
    c:\oml4rclient_install_dir\instantclient-sdk-windows.x64-12.1.0.2.0.zip
    ```

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

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:\myoml4rclient\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:\myoml4rclient\instantclient_12_1`.
    
    d. Under **System variables**, edit `PATH` to include `c:\myoml4rclient\instantclient_12_1`.
    
    ![See Also:](Create and Modify Environment Variables on Windows)

---

**Install 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:

### Install Oracle Instant Client from a Zip File

Instructions for installing Oracle Instant Client from a zip file.

1. Create an installation directory for the OML4R client components. For example:

    ```bash
    mkdir oml4rclient_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:

    ```
    \oml4rclient_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:

```
\oml4rclient_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 instantclient_12_1
ls
  /help
  /sdk
  /vc10
  /vc11
```

### Install Oracle Instant Client on Linux from RPMs

Instructions for installing Oracle Instant Client from RPMs.

1. Create an installation directory for the OML4R client components. For example:

```
mkdir oml4rclient_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`. 

---

**Chapter 6**

**Install Oracle Database Instant Client**

---

**ORACLE**
10. After installing Oracle Instant Client, add the path of the Oracle Instant Client libraries to `LD_LIBRARY_PATH`. For example:

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

### Install the Oracle Machine Learning for R Packages

Install the OML4R packages on each client computer.

The OML4R packages are automatically included in the installation on the server.

This topic includes these sections:

#### Install the OML4R Packages on Windows

Instructions for installing the OML4R packages on Windows.

1. Download the OML4R packages from the Oracle Machine Learning for R Downloads website.
2. Accept the license agreement and select the OML4R packages for your platform. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:

   ```shell
   c:\oml4rclient_install_dir\ore-client-win-x86_64-1.5.1.zip
   ```

   **NOTE:** Choose the same installation directory for all OML4R 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 OML4R packages.

4. Choose one of the following methods to install the OML4R 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:

       ```r
       install.packages("oml4rclient_install_dir\client/"), repos=NULL)
       ```

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

       ```text
       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:
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
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 OREgraphics_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:\oml4rc\client_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_64-1.5.1.1.zip

Install the OML4R Packages on Linux or UNIX

Instructions for installing the OML4R packages on Linux or UNIX.

1. Download the OML4R packages from the Oracle Machine Learning for R Downloads page on the Oracle Technology Network.

2. Accept the license agreement and select the OML4R packages for your platform. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:
NOTE: Choose the same installation directory for all OML4R 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/OREEdm_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


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 OREEdm_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

Install the OML4R Supporting Packages

Install the OML4R supporting packages on each client computer and on the server that hosts OML4R Server.

This topic includes these sections:

Install the Supporting Packages on Windows

Instructions for installing the supporting packages on Windows.

1. Download the supporting packages from the Oracle Machine Learning for R Downloads website.
2. Select the **Supporting** packages for your platform and accept the license agreement. Download the zip file to the installation directory that you created for Oracle Instant Client. For example:

```
c:\oml4rclient_install_dir\ore-supporting-win-x86_64-1.5.1.zip
```

**NOTE:** Choose the same installation directory for all OML4R 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, OML4R packages, and OML4R 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("oml4rclient_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 arules_1.5-0.zip
        R CMD INSTALL Cairo_1.5-9.zip
        R CMD INSTALL DBI_0.6-1.zip
        R CMD INSTALL png_0.1-7.zip
        R CMD INSTALL randomForest_4.6-12.zip
        R CMD INSTALL statmod_1.4.29.zip
        ```
        Each package installation generates this message:
        ```
        package 'package_name' successfully unpacked and MD5 sums checked
        ```
Install the Supporting Packages on Linux or UNIX

Instructions for installing the supporting packages on Linux or UNIX.

1. Download the OML4R supporting packages from the Oracle Machine Learning for R Downloads website.

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:

   /oml4rclient_install_dir/ore-supporting-platform-arch-1.5.1.zip

3. Unzip the file:

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

   If you are using Linux 8, then unzip the ore-supporting-linux8-x86-64-1.5.1.zip file.

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

   /supporting/arules_1.5-0_R_arch-unknown-platform.tar.gz
   /supporting/Cairo_1.5-9_R_arch-unknown-platform.tar.gz
   /supporting/DBI_0.6-1_R_arch-unknown-platform.tar.gz
   /supporting/png_0.1-7_R_arch-unknown-platform.tar.gz
   /supporting/randomForest_4.6-12_R_arch-unknown-platform.tar.gz
4. Change to `/oml4rclient_install_dir/supporting`

5. Execute the following commands to install the supporting packages on the client:

   R CMD INSTALL arules_1.5-0_R_arch-unknown-platform.tar.gz
   R CMD INSTALL Cairo_1.5-9_R_arch-unknown-platform.tar.gz
   R CMD INSTALL DBI_0.6-1_R_arch-unknown-platform.tar.gz
   R CMD INSTALL png_0.1-7_R_arch-unknown-platform.tar.gz
   R CMD INSTALL randomForest_4.6-12_R_arch-unknown-platform.tar.gz
   R CMD INSTALL ROracle_1.3-1_R_arch-unknown-platform.tar.gz
   R CMD INSTALL statmod_1.4.29_R_arch-unknown-platform.tar.gz

6. Execute the following commands to install the supporting packages on the database server. When you install on the server, use the `ORE` command (instead of the `R` command used for the client, as shown in the previous step). This installs the packages to `$ORACLE_HOME/R/library` instead of the default location, which is `/usr/lib64/R/library` on Linux.

   ORE CMD INSTALL arules_1.5-0_R_arch-unknown-platform.tar.gz
   ORE CMD INSTALL Cairo_1.5-9_R_arch-unknown-platform.tar.gz
   ORE CMD INSTALL DBI_0.6-1_R_arch-unknown-platform.tar.gz
   ORE CMD INSTALL png_0.1-7_R_arch-unknown-platform.tar.gz
   ORE CMD INSTALL randomForest_4.6-12_R_arch-unknown-platform.tar.gz
   ORE CMD INSTALL ROracle_1.3-1_R_arch-unknown-platform.tar.gz
   ORE CMD INSTALL statmod_1.4.29_R_arch-unknown-platform.tar.gz

**For Linux, Verify Cairo and png Dependencies**

The `cairo` and `png` packages require the presence of these operating system dependencies:

- Cairo requires the `cairo-devel` package.
- `png` requires the `libpng-devel` package.

To verify the presence of these dependencies, do the following.

1. Execute the following commands:

   $ rpm -qa libpng-devel
   $ rpm -qa cairo-devel

   If the RPMs are installed, then the name of the RPM is returned.

   $ rpm -qa cairo-devel
   cairo-devel-1.15.12-3.el7.i686
   cairo-devel-1.15.12-3.el7.x86_64
   $ rpm -qa libpng-devel
   libpng-devel-1.5.13-7.el7_2.x86_64
   libpng-devel-1.5.13-7.el7_2.i686
2. If that output is not returned, the RPMs are not installed. To install them, execute the following commands as root:

$ yum install cairo-devel
$ yum install libpng-devel

Connect OML4R Client to OML4R Server

Instructions for connecting to an OML4R server.

To connect an OML4R client to an OML4R server, start R using the `ORE` script:

% ORE
R> library(ORE)

The following examples connect as user `OML_USER` with password `OML_USERpsw`:

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

```r
ore.connect(user="OML_USER", sid="orcl", host="servername",
password="OML_USERpsw",
port=1521, all=TRUE)
```

**Note:**

To avoid specifying the password and other connection details in embedded R scripts, you can use Oracle Wallet. See Create an Oracle Wallet for an Oracle Machine Learning for R Connection.

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

```r
ore.connect("OML_USER", password="OML_USERpsw", conn_string="", all=TRUE)
```

**See Also:**

This chapter describes administrative tasks for maintaining and optimizing OML4R.

This chapter contains these topics:

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
Upgrade Oracle Machine Learning for R

You can upgrade OML4R to the current release from any previous release by reinstalling the product.

Note:

Upgrading from OML4R 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 OML4R and migrate your data:

1. Ensure that you have the version of R that is required for the release of OML4R that you are installing.
   
   See the table of configuration requirements and server support in Oracle Machine Learning for R System Requirements for the R requirement.
   
   If you are don’t need to upgrade R, proceed to Step 2.
   
   If you do need to upgrade R, do the following:
   
   a. Back up your OML4R user schema, data store objects, R scripts, and the RQSYS schema.
   
   b. Remove the Oracle R Distribution RPMs or open source R components.
   
   c. Install the required R version, then proceed to Step 2.
   
2. To upgrade OML4R Server for Oracle Database Release 12c or earlier, run the server.sh or server.bat script to perform an installation.
   
   When the script detects an earlier version of OML4R Server, it asks if you want to upgrade. Type Yes to start the upgrade. (Type No to abort the process.)
   
   See About the Server Script for details.
3. To upgrade OML4R Client, re-install the OML4R packages and supporting packages. You do not need to uninstall the current packages before installing the new packages. See Install Oracle Machine Learning for R Client for instructions.

Migrate Oracle Machine Learning for R Data

OML4R Server includes migration scripts that you can run to migrate the RQSYS schema and OML4R 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 OML4R 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_screxport.pl for exporting the RQSYS schema and all OML4R user data to a dump file.
- imp — contains the script ore_destimport.pl for importing the RQSYS schema and all OML4R user data from the dump file created by ore_screxport.pl.
- oreuser — contains scripts for exporting and importing data for a specific OML4R user. Instructions for running the migration scripts are provided in the README.

Migrate Oracle Machine Learning for R After a Database Upgrade

After upgrading your Oracle Database, you must migrate your OML4R Server components to the new ORACLE_HOME.

If you do not migrate the OML4R Server components to the new ORACLE_HOME, then running an R function using OML4R embedded R execution results in errors such as:

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

The components of OML4R Server are:

- The Oracle Database schema RQSYS and schema-related objects
- Oracle Database shared libraries for supporting OML4R clients
- OML4R 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 OML4R packages must also be installed to the new database location.

You can easily do this by running the OML4R Server installation script against the new ORACLE_HOME. Doing so creates a new path to the ORACLE_HOME in the OML4R metadata.

The following steps illustrate migrating OML4R 1.5.1 from an initial database installation on a Linux system 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 OML4R are not upgraded, only migrated to the new ORACLE_HOME.

⚠️ WARNING:

When migrating to an Oracle Database Release 18c or later database instance, use the `rqcfg.sql` script. Do not use the `server.sh` or `server.bat` script.

1. Before migrating the OML4R components, back up the RQSYS schema and OML4R user schema.

2. Run the OML4R Server installation script against the new ORACLE_HOME.

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

3. As the sysdba user, verify that the OML4R configuration script is pointing to the new ORACLE_HOME:

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

<table>
<thead>
<tr>
<th>NAME</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_HOME</td>
<td>/usr/lib64/R</td>
</tr>
<tr>
<td>R_LIBS_USER</td>
<td>/u01/app/oracle/product/12.2.0.1/dbhome_1/R/library</td>
</tr>
<tr>
<td>VERSION</td>
<td>1.5.1</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

4. As the sysdba user, verify that the OML4R dependent libraries `ore.so` and `librqe.so` are in the new ORACLE_HOME:

   ```sql
   SQL> SELECT library_name, file_spec FROM all_libraries WHERE owner = 'RQSYS';
   ```

<table>
<thead>
<tr>
<th>LIBRARY_NAME</th>
<th>FILE_SPEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>RQ$LIB</td>
<td>/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/ore.so</td>
</tr>
<tr>
<td>RQELIB</td>
<td>/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/</td>
</tr>
<tr>
<td>librqe.so</td>
<td></td>
</tr>
</tbody>
</table>

5. Finally, test the OML4R installation against the upgraded ORACLE_HOME by connecting your OML4R client to the OML4R server and running OML4R examples, such as those in the Correlating Data and Using the ore.tableApply Function topics in Oracle Machine Learning for R User's Guide. These examples use the `iris` data set in the `datasets` package that is included in an R distribution.

---

Uninstall Oracle Machine Learning for R

Instructions for uninstalling OML4R

This topic contains these sections:
Related Topics

- Uninstall Oracle R Distribution
  Instructions for uninstalling Oracle R Distribution.

Uninstall OML4R Server from Oracle Database 18c or Later

How to uninstall OML4R from Oracle Database Release 18c or later.

The rquncfg.sql script uninstalls the database functions and procedures that are associated with OML4R. It does not remove the OML4R libraries in $ORACLE_HOME/lib because these are shipped with Oracle Database. Also, it does not remove the OML4R packages in $ORACLE_HOME/R/library.

To uninstall OML4R Server components, run the rquncfg.sql script.

1. Change directories to $ORACLE_HOME/R/server.

   $ cd $ORACLE_HOME/R/server

2. If you are using a PDB, connect to it.

   $ ALTER SESSION SET CONTAINER = pdbname;

3. In SQL, run the uninstall script. The script takes a single input, which is the $ORACLE_HOME location. In the following example, the value of the ORACLE_HOME environment variable is /u01/app/oracle/product/18.0.0/dbhome_1.

   SQL> @rquncfg.sql
   Session altered.
   Enter value for 1: /u01/app/oracle/product/18.0.0/dbhome_1

Uninstall OML4R Server from Oracle Database 12c and Earlier

To uninstall OML4R 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.

WARNING:

Do not use the server.sh script to uninstall Oracle Machine Learning for R components from Oracle Database Release 18c or Release 19c. Doing so results in important files being deleted from the database. Instead, for an 18c or later database, use the rquncfg.sql script.

This topic has the following sections:

Related Topics

- Uninstall OML4R Server from Oracle Database 18c or Later
  How to uninstall OML4R from Oracle Database Release 18c or later.
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 OML4R Server in Oracle home.

If OML4R 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 OML4R 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 OML4R Server, then the other shared instances or PDBs will no longer support OML4R Server. You can easily restore OML4R 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 OML4R Server:

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

Related Topics

• **Uninstall OML4R Server from Oracle Database 18c or Later**
  How to uninstall OML4R from Oracle Database Release 18c or later.

Performing a Full Uninstall

A full uninstall removes the RQSYS schema metadata and PL/SQL code from the database and removes all OML4R Server libraries and R packages from Oracle home.

The following commands each perform a full uninstall of OML4R 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 OML4R Server support. See **Performing a Partial Uninstall** for details.
Related Topics

- [Uninstall OML4R Server from Oracle Database 18c or Later](#)
  How to uninstall OML4R from Oracle Database Release 18c or later.

Uninstall OML4R Client

Instructions for uninstalling OML4R Client.

To uninstall the OML4R packages and supporting packages, start R and type the commands listed in the following example.

**Example 7-1  R Commands for Uninstalling OML4R Packages**

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

Install Additional R Packages on Linux or UNIX

On Linux and UNIX platforms, the OML4R 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:

```bash
ORE CMD INSTALL R_package_name
```

Create a Database User for Oracle Machine Learning for R

The `server` script installation process automatically creates or configures a user for OML4R if one does not already exist.

**Example 7-2  Creating an OML4R User**

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

---

[1] Uninstall OML4R Server from Oracle Database 18c or Later

How to uninstall OML4R from Oracle Database Release 18c or later.
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 OML4R. If you specify a user that does not already exist, the script creates the user.

Example 7-3 Creating an OML4R User in SQL*Plus

You can create an OML4R 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 oml_username IDENTIFIED BY password
   DEFAULT TABLESPACE default_tablespace_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 oml_username;

   See Also:

   Oracle Database SQL Language Reference for details about granting privileges to a user

About the RQADMIN Role

The server script installation process creates a database role called RQADMIN.
When the RQADMIN role is granted to an OML4R 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 OML4R user.

**Note:**

Any OML4R user can execute embedded R, but only OML4R users with the RQADMIN role can create and drop the R scripts.

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 oml_username;
```

**Caution:**

Use caution when granting the RQADMIN role. Only users that require OML4R administrative privileges should have this role.

Related Topics

- Security Best Practices for OML4R
  To minimize the risk of compromising the security of an OML4R Server in Oracle Database, Oracle recommends the following security best practices.

Create and Modify 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 7-2. On Windows systems, you must be an administrator to create or modify environment variables.

**To create or modify environment variables on Windows 10:**

1. On the Windows taskbar, right-click the Windows icon and select System.
2. In the Settings window, under Related Settings, click Advanced system settings.
Figure 7-1 Advanced System Settings in Windows

3. On the Advanced tab, click Environment Variables.
Figure 7-2  Environment Variables Dialog in Windows

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

5. After creating or modifying the environment variable, click **Apply** and then **OK** to have the change take effect.

**Note:**

The graphical user interface for creating environment variables may vary slightly, depending on your version of Windows.
Create an Oracle Wallet for an Oracle Machine Learning for R 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 OML4R connection:

1. Start Oracle Wallet Manager:
   - (Linux and UNIX) At the command line, enter `owm`.
   - (Windows) Select **Start**, **Programs**, **Oracle-HOME_NAME**, **Integrated Management Tools**, **Wallet Manager**.

2. To create the wallet, follow the instructions in the Oracle Database documentation for your supported platform:
   a. For Oracle Database 12c and later, go to the Oracle Database Documentation page in Oracle Help Center.
   b. Select your version of Oracle Database.
   c. In the Topics section, select **Security**.
   e. See the chapter Using Oracle Wallet Manager.
   For Oracle Database 11c, Release 11.2.0.4, see Using Oracle Wallet Manager in *Oracle Database Advanced Security Guide*.

3. Locate the connection string for the OML4R database in `tnsnames.ora`. For example:
   ```
   mydb_test =
   (DESCRIPTION =
    (ADDRESS =
     (PROTOCOL = TCP)
     (HOST = myserver)
     (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 OML4R 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)
```
To Configure an Oracle Wallet for Use with External Procedures

1. Create a wallet store.

   ```bash
   $ mkstore -create -wrl /home/oracle/wallet
   ```

   When prompted to do so, assign a username and password. This example uses the database user OML_USER with the password `apassword` and the PDB ORCL.

2. Assign wallet credentials.

   ```bash
   $ mkstore -wrl /home/oracle/wallet -createCredential ORCL oml_user apassword
   ```

3. In SQL*Plus, log in as OML_USER using the wallet.

   ```bash
   $ sqlplus /@ORCL
   ```

4. Show the user.

   ```sql
   SQL> show user;
   USER is "OML_USER"
   ```

Example 7-4    Testing the Wallet Connection

This example tests using embedded R execution in the wallet connection in an OML4R session. The example uses the `iris` data set that is in the `datasets` package that is included in an R distribution.

```r
ore.doEval(function(){print("TEST")})

TEST_WALLET_DF
function() {
    return(as.data.frame(length(iris)))
}

ore.scriptLoad("TEST_WALLET_DF")

ore.doEval(FUN.NAME="TEST_WALLET_DF")
length(iris)
```

Listing for This Example

```r
> ore.doEval(function(){print("TEST")})
[1] "TEST"
>
> TEST_WALLET_DF
function() {
    return(as.data.frame(length(iris)))
}
>
> ore.scriptLoad("TEST_WALLET_DF")
>
> ore.doEval(FUN.NAME="TEST_WALLET_DF")
>    length(iris)
1 5

**Note:**

In embedded R execution, an R function that creates a database connection will fail because Oracle Database does not support recursive external procedures. To connect an embedded R execution function to a database, use the ore.connect special control argument.

### Control Memory Used by Embedded R

How to control the memory used by embedded R execution.

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>
<thead>
<tr>
<th>Keyword</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIN_VSIZE</td>
<td>32M</td>
<td>Minimum R vector heap memory</td>
</tr>
<tr>
<td>MAX_VSIZE</td>
<td>4G</td>
<td>Maximum R vector heap memory</td>
</tr>
<tr>
<td>MIN_NSIZE</td>
<td>1M</td>
<td>Minimum number of R cons cells</td>
</tr>
<tr>
<td>MAX_NSIZE</td>
<td>20M</td>
<td>Maximum number of R cons cells</td>
</tr>
</tbody>
</table>

**Example 7-5** 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);
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 OML4R. 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
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 Sample Installation of Oracle Machine Learning for R

Steps in a typical installation of OML4R Server on a Linux server running Oracle Database 12c, Release 12.1.0.2, and OML4R Client on a Windows system.

Note:
This appendix describes an initial installation of OML4R. If OML4R components already exist on your client or server, refer to Upgrade Oracle Machine Learning for R.

This appendix contains these topics:

About the Oracle Machine Learning for R 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 OML4R.
  – 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.
Install Oracle Machine Learning for R on the Server

Instructions for installing OML4R on the server.

To install OML4R on the server computer, first verify that Oracle Database is installed and that the environment is configured as specified in About the Oracle Machine Learning for R Sample Installation Environment. Next, complete these steps in the specified order:

1. Verify the environment.
2. Install Oracle R Distribution
3. Install Oracle Machine Learning for R Server

These steps are described in the following topics:

Verify the Environment

A checklist for the OML4R Server requirements.

Table A-1  Checklist for Oracle Machine Learning for R Server Requirements

<table>
<thead>
<tr>
<th>Question</th>
<th>Sample Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the Linux version?</td>
<td>% cat /etc/redhat-release</td>
</tr>
<tr>
<td></td>
<td>Enterprise Linux Server release 6.4</td>
</tr>
<tr>
<td>Do you have access to the internet?</td>
<td>Start a browser</td>
</tr>
<tr>
<td>Can you log in as root?</td>
<td>% sudo -su</td>
</tr>
<tr>
<td></td>
<td>Password: __________ #</td>
</tr>
<tr>
<td></td>
<td>% exit</td>
</tr>
<tr>
<td></td>
<td>%</td>
</tr>
<tr>
<td>Is Oracle Database installed?</td>
<td>% SQLPLUS / as sysdba</td>
</tr>
<tr>
<td></td>
<td>Copyright (c) 1982, 2017, Oracle. All rights reserved.</td>
</tr>
<tr>
<td></td>
<td>Connected to: Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bitProduction</td>
</tr>
<tr>
<td></td>
<td>With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options</td>
</tr>
<tr>
<td></td>
<td>&gt; exit</td>
</tr>
<tr>
<td></td>
<td>%</td>
</tr>
<tr>
<td>What is the value of $ORACLE_HOME?</td>
<td>% echo $ORACLE_HOME</td>
</tr>
<tr>
<td></td>
<td>/myhome/product/12.1.0.2/dbhome_1</td>
</tr>
<tr>
<td>What is the value of $ORACLE_SID?</td>
<td>% echo $ORACLE_SID</td>
</tr>
<tr>
<td></td>
<td>orcl</td>
</tr>
<tr>
<td>Does $LD_LIBRARY_PATH include $ORACLE_HOME/</td>
<td>% echo $LD_LIBRARY_PATH</td>
</tr>
<tr>
<td>lib?</td>
<td>/myhome/product/12.1.0.2/dbhome_1/lib:....</td>
</tr>
</tbody>
</table>
Table A-1  (Cont.) Checklist for Oracle Machine Learning for R Server Requirements

<table>
<thead>
<tr>
<th>Question</th>
<th>Sample Answer</th>
</tr>
</thead>
</table>
| Does \$PATH include \$ORACLE_HOME/bin? | % echo \$PATH
/myhome/product/12.1.0.2/dbhome_1/bin:....... |
| 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/ ... |

Install Oracle R Distribution

Example of installing 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 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 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:


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
Install Oracle Machine Learning for R Server

OML4R Server includes the RQSYS schema in Oracle Database and OML4R packages and shared libraries.

To install OML4R Server:

1. Verify the environment according to Table A-1.

2. Create an installation directory for the OML4R Server components. The directory can have any name. For example:

   /myhome/myomlserver/

3. Download the OML4R Server installation files and supporting packages from the Oracle Machine Learning for R Downloads website.
   a. Accept the license agreement and download the OML4R Server files for your platform to your installation directory.
   b. Accept the license agreement and download the OML4R 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/myomlserv
   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 OML4R 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.

6. On Oracle Linux 6, set LD_LIBRARY_PATH to the location of the files installed by the R-core-extra RPM:

   export LD_LIBRARY_PATH=/usr/lib64/R/port/Linux-X64/lib
Example A-1  A Default, First-Time Installation of OML4R Server

hcearwigger@myserver> ./server.sh

Oracle Machine Learning for R 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
  TEMP
    TEMPORARY tablespace to use for RQSYS [list]: TEMP
Choosing RQSYS password
  Password to use for RQSYS: XXXXXX

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
Install Oracle Machine Learning for R on the Client

To install OML4R on the client computer, first verify that the Microsoft Windows environment meets the requirements.

The requirements are specified in About the Oracle Machine Learning for R Sample Installation Environment.

Next, complete these steps:

1. Install Oracle R Distribution on the Windows client
2. Install Oracle Instant Client
3. Install the OML4R packages
4. Install the OML4R supporting packages

These steps are described in the following topics:

Install Oracle R Distribution on the Windows Client

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

See Also:

- Oracle Machine Learning for R System Requirements
- Verifying 64-Bit Architecture on Microsoft Windows
To install Oracle R Distribution on Windows:

1. Go to the Oracle R Distribution downloads page.
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.

Install Oracle Instant Client

Oracle Machine Learning for R 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 OML4R client components. The directory can have any name. For example:
   
   c:\myoml4rclient

2. Navigate to the Oracle Database Instant Client website.

3. Click the **Download Now** button.

4. On the Oracle Instant Client Downloads page, select **Instant Client for Microsoft Windows (x64)**.

5. Under **Version 12.1.0.2.0**, select **Instant Client Package - Basic** for Oracle Database 12.1.

6. 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:\myoml4rclient\instantclient-basic-windows.x64-12.1.0.2.0.zip

7. 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:

   myoml4rclient
   instantclient_12_1
   vc10
   vc11
   vc12

8. Return to the Instant Client Downloads for Microsoft Windows (x64) page.

9. Accept the license agreement and select **Instant Client Package - SDK**. Save the file in the directory that you created in Step 1.
   
   c:\myoml4rclient\instantclient-sdk-windows.x64-12.1.0.2.0.zip

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

```
myoml4rclient
    instantclient_12_1
      help
      sdk
      vc10
      vc11
      vc12
```

11. 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:\myoml4rclient\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:\myoml4rclient\instantclient_12_1`.
- e. Under **System variables**, edit `PATH` to include `c:\myoml4rclient\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.

### Install the Oracle Machine Learning for R Packages

Example of installing the Oracle Machine Learning for R packages.

Follow these steps to download and install the OML4R packages:

**To download the OML4R packages:**

1. Go to the Oracle Machine Learning for R Downloads website.
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**.

   `c:\myoml4rclient\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:
To install the OML4R 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:

   ```r
   install.packages("c:/myoml4rclient/client/ORE_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREbase_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREcommon_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREdm_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREdplyr_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREeda_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREembed_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREgraphics_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREmodels_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREpredict_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/client/OREstats_1.5.1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/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
   ```

Install the Oracle Machine Learning for R Supporting Packages

Example of installing the OML4R supporting packages.

Follow these steps to download and install the OML4R supporting packages:

To download the OML4R supporting packages:

1. Go to the Oracle Machine Learning for R Downloads website.

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.

   ```
   c:\myoml4rclient\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
   ```
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:/myoml4rclient/supporting/ROracle_1.3-1.zip", repos=NULL)
   install.packages("c:/myoml4rclient/supporting/DBI_0.5.zip", repos=NULL)
   install.packages("c:/myoml4rclient/supporting/png_0.1-7.zip", repos=NULL)
   install.packages("c:/myoml4rclient/supporting/Cairo_1.5-8.zip", repos=NULL)
   install.packages("c:/myoml4rclient/supporting/arules_1.1-9.zip", repos=NULL)
   install.packages("c:/myoml4rclient/supporting/randomForest_4.6-10.zip", repos=NULL)
   install.packages("c:/myoml4rclient/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
   ```

Verifying the Oracle Machine Learning for R Installation

To verify that the basic functionality of OML4R is working, establish a connection to an OML4R server and execute several basic functions.

**Note:**

To start and use OML4R, your database user must have the privileges required for OML4R installation. See User Requirements for details.

**Example A-2 Connecting to an OML4R Server**

To connect the an OML4R client to an OML4R server:

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

   The R Console is displayed.

2. Type this command to start OML4R:

   ```
   $ ORE
   R> library(ORE)
   ```

3. Type this command to connect to the OML4R server. The following example connects user OML_USER to the database orcl on the server host serv1 using port 1521:

   ```
   > ore.connect(user="OML_USER", sid="orcl", host="serv1",
   > password="OML_USERpsw",
   > 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 function returns `TRUE`:

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

**Example A-3  Listing the Database Tables Accessible in a Schema**

The `ore.ls` function lists the `ore.frame` proxy objects that correspond to database tables in the environment for a schema. In the following example, `TABLE1` and `TABLE2` exist in the current schema:

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

**Example A-4  Pushing an R Data Frame to the Database**

The `ore.push` function pushes a local R object into an OML4R object of the appropriate data type in the database. The following example creates an R `data.frame` and pushes it an `ore.frame` object in the database.

```r
df <- data.frame(a="abc",
               b=1.456,
               c=TRUE,
               d=as.integer(1))
of <- ore.push(df)
```

**Example A-5  Executing an Embedded R Function**

The `ore.doEval` function executes the specified function in an R engine on the database server and returns the results. This example declares a function in the `ore.doEval` invocation.

```r
> ore.doEval(function() { 123 })
[1] 123
```
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 Machine Learning for R 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:

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
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 https://cran.r-project.org/src/contrib/arules_1.1-9.tar.gz
```

Then, install the package using the command R CMD INSTALL:

```r
$ 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"
```

---

**Set the R Repository**

Instructions for 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 specify the CRAN mirror to use. The following code sets the R package repository to the Seattle CRAN mirror at the start of each R session.

```
cat("Setting Seattle repository")
r = getOption("repos")
r["CRAN"] = "http://cran.fhcrc.org/"
options(repos = r)
rm(r)
```

---

**About R Package Installation for Oracle Machine Learning for R**

Embedded R execution with OML4R 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 OML4R are the same as for open source R. The database-side R engine just needs to know where to find the R packages.

The OML4R installation is performed by the 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 OML4R 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 the oracle user and accessible by all OML4R users: `$ORACLE_HOME/R/library`. 

---

---

---
To install a package on the database server so that any R user can use it 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 OML4R packages.

```
$ wget https://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.

If you are using an OML4R client, install the package in the same way 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.

## About CRAN Task Views

CRAN 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, then you can install every package in that view using the `ctv` package, which automates package installation.

### Install the `ctv` Package and Task Views

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")
```

### Use and Manage Packages

To use a package, start R and load packages one at a time with the `library` command.
Load the \texttt{arules} package in your R session.

\begin{verbatim}
R> library(arules)
\end{verbatim}

Verify the version of \texttt{arules} installed.

\begin{verbatim}
R> packageVersion("arules")
[1] '1.1.9'
\end{verbatim}

Verify the version of \texttt{arules} installed on the database server using embedded R execution.

\begin{verbatim}
R> ore.doEval(function() packageVersion("arules"))
\end{verbatim}

View the help file for the \texttt{apropos} function in the \texttt{arules} package.

\begin{verbatim}
R> ?apropos
\end{verbatim}

Over time, your package repository will contain more and more packages, especially if you are using the system-wide repository in which others are also adding 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 \texttt{installed.packages} command:

\begin{verbatim}
R> myLocalPackages <- row.names(installed.packages())
R> myLocalPackages
\end{verbatim}
Installing RStudio

This appendix provides tips for installing RStudio Server for use with Oracle Machine Learning for R on Linux. This appendix includes these topics:

About RStudio

Describes 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 Machine Learning for R, however RStudio is not included with OML4R. If you want to use RStudio, you must install and license it separately.

See Also:

- GNU Affero General Public License for details about AGPL
- RStudio for details about RStudio

Install 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 Machine Learning for R:

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. Create the file `/etc/rstudio/rserver.conf`. Add the values of `R_HOME` and `ORACLE_HOME`.

   ```
   sudo vi /etc/rstudio/rserver.conf
   rsession-ld-library-path=R_HOME/lib:ORACLE_HOME/lib
   ```

   Note: The default value of `R_HOME` on Linux is `/usr/lib64/R`.

3. Create the configuration file `/usr/lib64/R/etc/Renviron.site`. Supply the values of `ORACLE_HOME`, `ORACLE_HOSTNAME`, and `ORACLE_SID`. For example, using the BASH shell:

   ```
   cd /usr/lib64/R/etc
   sudo vi Renviron.site
   ORACLE_HOME=ORACLE_HOME
   ORACLE_HOSTNAME=ORACLE_HOSTNAME
   ORACLE_SID=ORACLE_SID
   ```
Install 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.

4. Restart the RStudio Server service as sudo or root:

   `sudo rstudio-server restart`

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.
Oracle R Distribution Packages

The table in this section lists the packages in Oracle R Distribution that are used by Oracle Machine Learning for R.

See Also:
- Table 6-1 for a list of the packages in Oracle Machine Learning for R
- Table 6-2 for a list of the open source packages that ship with Oracle Machine Learning for R

Table D-1  Oracle R Distribution Packages Used by Oracle Machine Learning for R

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Package Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>base</td>
<td>The R Base Package</td>
</tr>
<tr>
<td>boot</td>
<td>Bootstrap Functions (originally by Angelo Canty for S)</td>
</tr>
<tr>
<td>class</td>
<td>Functions for Classification</td>
</tr>
<tr>
<td>cluster</td>
<td>Cluster Analysis Extended Rousseeuw et al</td>
</tr>
<tr>
<td>codetools</td>
<td>Code Analysis Tools for R</td>
</tr>
<tr>
<td>compiler</td>
<td>The R Compiler Package</td>
</tr>
<tr>
<td>datasets</td>
<td>The R Datasets Package</td>
</tr>
<tr>
<td>foreign</td>
<td>Read Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, dBase</td>
</tr>
<tr>
<td>graphics</td>
<td>The R Graphics Package</td>
</tr>
<tr>
<td>grDevices</td>
<td>The R Graphics Devices and Support for Colours and Fonts</td>
</tr>
<tr>
<td>grid</td>
<td>The Grid Graphics Package</td>
</tr>
<tr>
<td>KernSmooth</td>
<td>Functions for kernel smoothing for Wand &amp; Jones (1995)</td>
</tr>
<tr>
<td>lattice</td>
<td>Lattice Graphics</td>
</tr>
<tr>
<td>MASS</td>
<td>Support Functions and Datasets for Venables and Ripley's MASS</td>
</tr>
<tr>
<td>Matrix</td>
<td>Sparse and Dense Matrix Classes and Methods</td>
</tr>
<tr>
<td>methods</td>
<td>Formal Methods and Classes</td>
</tr>
<tr>
<td>mgcv</td>
<td>GAMs with GCV/AIC/REML smoothness estimation and GAMMs by PQL</td>
</tr>
<tr>
<td>nlme</td>
<td>Linear and Nonlinear Mixed Effects Models</td>
</tr>
<tr>
<td>nnet</td>
<td>Feed-forward Neural Networks and Multinomial Log-Linear Models</td>
</tr>
<tr>
<td>parallel</td>
<td>Support for parallel computation, including random-number generation</td>
</tr>
<tr>
<td>RFO</td>
<td>Classification based on a forest of trees using random inputs</td>
</tr>
<tr>
<td>rpart</td>
<td>Recursive Partitioning</td>
</tr>
<tr>
<td>spatial</td>
<td>Functions for Kriging and Point Pattern Analysis</td>
</tr>
<tr>
<td>Package Name</td>
<td>Package Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>splines</td>
<td>Regression Spline Functions and Classes</td>
</tr>
<tr>
<td>stats</td>
<td>The R Stats Package</td>
</tr>
<tr>
<td>stats4</td>
<td>Statistical Functions using S4 Classes</td>
</tr>
<tr>
<td>survival</td>
<td>Survival analysis, including penalised likelihood.</td>
</tr>
<tr>
<td>tcltk</td>
<td>Tcl/Tk Interface</td>
</tr>
<tr>
<td>tools</td>
<td>Tools for Package Development</td>
</tr>
<tr>
<td>translation</td>
<td>Bindings for the Google Translate API v2</td>
</tr>
<tr>
<td>utils</td>
<td>The R Utils Package</td>
</tr>
</tbody>
</table>
Index

C
CDB, 4-11
client
  components, 6-1
  installing, 6-1, A-6
  requirements, A-1
client/server architecture, 1-1
  connecting to the server, 4-15, 6-12

D
database
  configuring extproc, 2-1
  connecting to, 6-12
  installing, 2-1
  PDB, 2-1, 4-15, 7-6
  requirements, 1-5, 2-1
database user
  creating, 4-11, 7-7
  requirements, 4-4
dba group, 4-5, A-1
DCLI, 5-1
Distributed Command Line Interface, 5-1

E
  embedded R execution, 1-1, A-10
  scripts requiring RQADMIN role, 7-8
  environment variables
    creating on Windows, 7-9
    requirements, 4-3
Exadata
  installing server on, 5-1
  extproc, 2-1
  troubleshooting, 2-3

I
IBM AIX, 4-4
  requirements, 1-5
  upgrade restriction, 7-2
installation
  verifying server, 4-13

installation scripts
  for 12c and earlier, 4-8
  for 18c and later, 4-5
installing
  client, 6-1
  example for Oracle Database 12c and earlier, A-1
  Oracle Database Instant Client, 6-3
  overview, 1-2
  server for 12c and earlier, 4-8
  server for 18c and later, 4-5
  server on Exadata, 5-1
  user requirements for, 4-4, A-1
installation
  verifying, A-10
Instant Client, 6-1, A-7
  installing on Linux or UNIX, 6-4
  installing on Windows, 6-3

L
LD_LIBRARY_PATH, A-1
Linux
  requirements, 1-5

M
Math Kernel Library, 3-2, 3-12
Microsoft Windows
  requirements, 1-5
  verifying 64-bit architecture, 1-5
migrating
  data, 7-3
  server components, 7-3
multitenant architecture, 2-1, 4-15, 7-6

O
Oracle Call Interface, 6-1
Oracle Data Mining rebranded, vii
Oracle Database
  configuring extproc, 2-1
  installing, 2-1
  requirements, 1-5, 2-1
Oracle Database Client, 1-2, 6-1, A-7
Oracle Database Instant Client, 6-1, A-7
Oracle Linux
requirements, 1-5
Oracle Machine Learning for R
client components, 1-2
server components, 1-2
Oracle Machine Learning for R packages
described, 6-2
Oracle public yum, A-3
Oracle R Advanced Analytics for Hadoop
rebranded, vii
Oracle R Distribution
advantages, 3-2
example of installing, A-3
turing of installing in a non-default
R_HOME, 7-1
installing on Exadata with DCLI, 5-3
installing on IBM AIX, 3-9
installing on Linux, 3-2
installing on Linux using RPMs, 3-6
installing on Microsoft Windows, 3-12
installing on Oracle Solaris, 3-8
installing on Red Hat Enterprise Linux, 3-7
overview, 3-2
requirements, 1-5
Oracle R Enterprise rebranded, vii
Oracle Solaris, 4-4
requirements, 1-5
Oracle Solaris Studio, 3-9
Oracle Wallet, 7-12
ORE script, 7-7, B-2
ore.connect, 6-12, A-10

P

packages
installing on Windows, A-8
Oracle Machine Learning for R, 6-2
supporting, 6-2, 6-8
PDB, 2-1, 4-11, 4-15, 7-6

R

and Oracle Machine Learning for R, 3-1
installing on Windows, 3-12, A-6
memory usage, 7-14
open source, 3-1, 3-2
rebranding
Oracle Data Mining, vii
Oracle R Advanced Analytics for Hadoop, vii
Oracle R Enterprise, vii
Red Hat Enterprise Linux
requirements, 1-5
ROracle package, 3-1, 6-1

RQADMIN role
about, 7-8
dexample of granting, 5-10
security, 4-2
rqcfg.sql installation script, 4-5
RQSYS schema
security, 4-2
rquncfg.sql script, 7-5

S

scripts
embedded R execution, 7-8
ORE, 4-13, 7-7, B-2
rqcfg.sql, 4-5
rquncfg.sql, 7-5
server, 4-10
server syntax, 4-11
server.bat, 4-8
server.sh, 4-8
security
best practices, 4-2
server
components, 4-1
environment variables, 4-3
installing, A-2
installing for 18c and later, 4-5
installing on Exadata, 5-1
installing on Exadata with DCLI for 12c and
earlier, 5-5
installing on Exadata with DCLI for 18c and
later, 5-7
installing with rqcfg.sql script for 18c and
later, 4-5
installing with server script for 12c and
earlier, 4-8
migrating components after database
upgrade, 7-3
uninstalling with rquncfg.sql for 18c and later,
7-5
uninstalling with server.sh from 12c or earlier,
7-5
verifying installation, 4-13
server script
eamples, 4-12
for Oracle Database 18c and later, 4-5
full uninstall, 4-11, 7-6
overview, 4-10
partial uninstall, 4-11, 7-6
requirements, 4-3
syntax, 4-11
upgrading server, 7-2
SQL transparency, 1-1
Sun Performance Library, 3-9
supporting packages described, 6-2
installing on Linux or Unix, 6-10
installing on Windows, 6-8, A-9
system requirements, 1-5

U
uninstalling, 7-4
client, 7-7
Oracle R Distribution, 3-14
server from a 12c or earlier database, 7-5
server from an 18c or later database, 7-5
upgrading
database and migrating Oracle Machine Learning for R components, 7-3

upgrading (continued)
server, 7-2

V
versions, 1-5

W
wallets
creating Oracle, 7-12

Y
yum, 3-3, A-3