Oracle® REST Data Services
Installation and Configuration Guide
Release 2.0
E25066-12

October 2014
Contents

Preface .................................................................................................................................................................. v
  Topic Overview .................................................................................................................................................... v
  Audience .......................................................................................................................................................... v
  Documentation Accessibility ................................................................................................................................ vi
  Related Documents ......................................................................................................................................... vi
  Conventions ..................................................................................................................................................... vi
  Third-Party License Information ................................................................................................................ vi

1 Installing Oracle REST Data Services
  1.1 About Oracle REST Data Services ........................................................................................................ 1-1
  1.2 Understanding the Installation Process ................................................................................................... 1-2
  1.2.1 Supported Java EE Application Servers .............................................................................................. 1-2
  1.2.2 System Requirements .......................................................................................................................... 1-2
  1.2.3 About Installing Oracle REST Data Services ...................................................................................... 1-2
  1.3 Installing Oracle REST Data Services .................................................................................................... 1-3
  1.4 Configuring Oracle REST Data Services .................................................................................................. 1-3
  1.4.1 About Using the Command-Line Interface ............................................................................................ 1-4
  1.4.2 About the Database Users Used by Oracle REST Data Services ....................................................... 1-4
  1.4.3 Downloading and Configuring Oracle REST Data Services ................................................................. 1-4
  1.4.4 Using SQL Developer Oracle REST Data Services Administration (Optional) ................................ 1-5
  1.4.4.1 About SQL Developer Oracle REST Data Services Administration ............................................... 1-5
  1.4.4.2 Configuring an Administrator User .................................................................................................... 1-6
  1.4.5 Using OAuth2 in Non HTTPS Environments ........................................................................................ 1-6
  1.5 Running in Standalone Mode .................................................................................................................... 1-7
  1.5.1 Starting in Standalone Mode .................................................................................................................. 1-7
  1.5.2 Stopping the Server in Standalone Mode ............................................................................................... 1-7
  1.5.3 About JServ Protocol (AJP) Support ...................................................................................................... 1-8
  1.6 Deploying to Oracle WebLogic Server ..................................................................................................... 1-8
  1.6.1 About Oracle WebLogic Server ............................................................................................................. 1-8
  1.6.2 Downloading and Unzipping Oracle REST Data Services ................................................................. 1-8
  1.6.3 Configuring Oracle REST Data Services .............................................................................................. 1-8
  1.6.4 Configuring Oracle Application Express Images .................................................................................... 1-9
  1.6.5 Launching the Administration Server Console ...................................................................................... 1-9
  1.6.6 Installing the Oracle WebLogic Server Deployment ............................................................................. 1-9
  1.6.7 Configuring WebLogic to Handle HTTP Basic Challenges Correctly ................................................. 1-11
Oracle REST Data Services Installation and Configuration Guide explains how to install and configure Oracle REST Data Services.

**Name Change:** Oracle REST Data Services was called Oracle Application Express Listener before Release 2.0.6.

### Topics:
- **Topic Overview**
- **Audience**
- **Documentation Accessibility**
- **Related Documents**
- **Conventions**
- **Third-Party License Information**

### Topic Overview

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installing Oracle REST Data Services</td>
<td>Describes how to download, install, configure, run, and deploy Oracle REST Data Services in various environments.</td>
</tr>
<tr>
<td>Configuring Oracle REST Data Services</td>
<td>Explains how to configure Oracle REST Data Services for connecting to multiple databases for routing requests.</td>
</tr>
<tr>
<td>About the Configuration File</td>
<td>Describes the Oracle REST Data Services configuration file.</td>
</tr>
<tr>
<td>Troubleshooting</td>
<td>Describes how to enable debug tracing and detailed request error messages using the Oracle REST Data Services configuration file.</td>
</tr>
</tbody>
</table>

### Audience

This document is intended for system administrators or application developers who are installing and configuring Oracle REST Data Services. This guide assumes you are familiar with web technologies, especially REST (Representational State Transfer), and have a general understanding of Windows and UNIX platforms.
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 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.

Related Documents

For more information and resources relating to Oracle REST Data Services, see the following the Oracle Technology Network (OTN) site:


Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</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>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that is displayed on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>

Third-Party License Information

Oracle REST Data Services contains third-party code. See the Oracle Database Licensing Information book for notices Oracle is required to provide.

Note, however, that the Oracle program license that accompanied this product determines your right to use the Oracle program, including the third-party software, and the terms contained in the following notices do not change those rights.
Installing Oracle REST Data Services

This section describes how to install and deploy Oracle REST Data Services. (REST stands for Representational State Transfer.)

---

**Name Change:** Oracle REST Data Services was called Oracle Application Express Listener before Release 2.0.6.

---

**Topics:**

- About Oracle REST Data Services
- Understanding the Installation Process
- Installing Oracle REST Data Services
- Configuring Oracle REST Data Services
- Running in Standalone Mode
- Deploying to Oracle WebLogic Server
- Deploying to GlassFish Server
- Deploying to Apache Tomcat
- Upgrading Oracle REST Data Services

### 1.1 About Oracle REST Data Services

Oracle REST Data Services is a Java EE-based alternative for Oracle HTTP Server and mod_plsql. The Java EE implementation offers increased functionality including a command line based configuration, enhanced security, file caching, and RESTful web services. Oracle REST Data Services also provides increased flexibility by supporting deployments using Oracle WebLogic Server, GlassFish Server, Apache Tomcat, and a standalone mode.

The Oracle Application Express architecture requires some form of web server to proxy requests between a web browser and the Oracle Application Express engine. Oracle REST Data Services satisfies this need but its use goes beyond that of Oracle Application Express configurations. Oracle REST Data Services simplifies the deployment process because there is no Oracle home required, as connectivity is provided using an embedded JDBC driver.
1.2 Understanding the Installation Process

This section offers an overview of Oracle REST Data Services and provides information about supported Java Platform, Enterprise Edition (Java EE) application servers and system requirements.

Topics:
- Supported Java EE Application Servers
- System Requirements

1.2.1 Supported Java EE Application Servers

Oracle REST Data Services supports the following Java EE application servers:

<table>
<thead>
<tr>
<th>Application Server</th>
<th>Supported Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle WebLogic Server</td>
<td>11g Release 1 (10.3.3) or later</td>
</tr>
<tr>
<td>GlassFish Server</td>
<td>Release 3 or later</td>
</tr>
<tr>
<td>Apache Tomcat</td>
<td>Release 6 or later</td>
</tr>
</tbody>
</table>

1.2.2 System Requirements

Oracle REST Data Services system requirements are as follows:

- Oracle Database (Enterprise Edition, Standard Edition or Standard Edition One) release 10.2.0.3 or later, or Oracle Database 10g Release 2 Express Edition or later.
- Oracle Application Express:
  - Oracle REST Data Services depends on Oracle Application Express for storing and creating RESTful Services.
  - If you want to use RESTful Services, then Oracle Application Express 4.2 or later is required.
  - Ensure that all steps of the Oracle Application Express installation process have been completed, including the step "Configure RESTful Services" in Oracle Application Express Installation Guide.
  - If you do not want to use RESTful Services, then the use of any Oracle Application Express version is supported.
- Java 6 Update 20 JDK or later.
- Java Servlet Specification 2.3 or later.
- Web browser requirements:
  - Microsoft Internet Explorer 8.0 or later.
  - Mozilla Firefox 3.0 or later.
  - Google Chrome 2.0 or later.

1.2.3 About Installing Oracle REST Data Services

To install Oracle REST Data Services:

1. Download and unzip Oracle REST Data Services. See Section 1.3, "Installing Oracle REST Data Services".
2. Configure Oracle REST Data Services. See Section 1.4, "Configuring Oracle REST Data Services".

3. Deploy Oracle REST Data Services. Deployment options include:
   - **Standalone Mode.** See Section 1.5, "Running in Standalone Mode".
   - **Oracle WebLogic Server.** See Section 1.6, "Deploying to Oracle WebLogic Server".
   - **GlassFish Server.** See Section 1.7, "Deploying to GlassFish Server".
   - **Apache Tomcat.** See Section 1.8, "Deploying to Apache Tomcat".

   **See Also:** Section 1.9, "Upgrading Oracle REST Data Services" and Appendix B, "Troubleshooting"

### 1.3 Installing Oracle REST Data Services

To install Oracle REST Data Services:

1. Download the file ords.version.number.zip from the Oracle REST Data Services download page. See:
   

   Note that the version.number in the file name reflects the current release number.

2. Unzip the downloaded zip file into a directory (or folder) of your choice:
   - UNIX and Linux: unzip ords.version.number.zip
   - Windows: Double-click the file ords.version.number.zip in Windows Explorer

### 1.4 Configuring Oracle REST Data Services

Before you deploy Oracle REST Data Services, you must configure it using a command-line interface. This section describes the various ways you can configure Oracle REST Data Services.

**Tip:** To quickly get started with Oracle REST Data Services execute the following command:

```
java -jar ords.war
```

You are prompted to enter some information, and it then starts in Standalone Mode.

**Topics:**
- About Using the Command-Line Interface
- About the Database Users Used by Oracle REST Data Services
- Downloading and Configuring Oracle REST Data Services
- Using SQL Developer Oracle REST Data Services Administration (Optional)
- Using OAuth2 in Non HTTPS Environments
1.4.1 About Using the Command-Line Interface

Oracle REST Data Services provides several command line commands. For example, you can configure the location where Oracle REST Data Services stores configuration files, configure the database Oracle REST Data Services uses, and start Oracle REST Data Services in standalone mode.

To display a full list of available commands, execute the following command:

```
java -jar ords.war help
```

A list of the available commands is displayed. To see instructions on how to use each of these commands, enter `help` followed by the command name, for example:

```
java -jar ords.war help configdir
```

1.4.2 About the Database Users Used by Oracle REST Data Services

Oracle REST Data Services uses the following database users:

<table>
<thead>
<tr>
<th>User Name</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APEX_PUBLIC_USER</td>
<td>Yes</td>
<td>If you use Oracle REST Data Services with Oracle Application Express, this is the database user used when invoking PL/SQL Gateway operations, for example, all Oracle Application Express operations. For information on unlocking the APEX_PUBLIC_USER, see “Configure APEX_PUBLIC_USER Account” in Oracle Application Express Installation Guide.</td>
</tr>
<tr>
<td>APEX_REST_PUBLIC_USER</td>
<td>Only if using RESTful Services</td>
<td>The database user used when invoking RESTful services</td>
</tr>
<tr>
<td>APEX_LISTENER</td>
<td>Only if using RESTful Services</td>
<td>The database user used to query RESTful Services definitions stored in Oracle Application Express</td>
</tr>
</tbody>
</table>

Each of these database users is created during the Oracle Application Express installation process.

1.4.3 Downloading and Configuring Oracle REST Data Services

**Note:** You must complete the configuration steps in this section before deploying to an application server.

To configure Oracle REST Data Services:

1. Download and unzip Oracle REST Data Services. See Section 1.3, "Installing Oracle REST Data Services".
2. Change your active directory to the directory where you unzipped Oracle REST Data Services.
3. Execute the following command:

```
java -jar ords.war
```
4. When prompted, enter the information required to complete the configuration process. Each prompt may have a default response enclosed in brackets [] . To choose the default response, press Enter without entering anything.

You are prompted to enter the following data:

- **Location to store configuration data** - Enter the location where you want to store Oracle REST Data Services configuration data. There is no default value.

- **Name of the database server** - Enter the name of the database server. The default value is localhost.

- **Database listener port** - Enter the port number that database is listening on. The default value is 1521.

- **Database service name or database SID** - Options include:
  - Database service name - Enter 1 to specify the database service name (the default value).
  - Database SID - Enter 2 to specify the database SID.

- **Database user name** - Enter the database user name to use for all PL/SQL Gateway operations. The default value is APEX_PUBLIC_USER.

- **Database user password** - Enter the database password for the user specified in the previous step. There is no default value.

- **Passwords for RESTful Services Database Accounts** - Options include:
  - Choose passwords - Enter 1 to specify passwords for APEX_LISTENER and APEX_REST_PUBLIC_USER accounts (the default value).
  - Re-use APEX_PUBLIC_USER password - Enter 2 to use the same password for APEX_LISTENER and APEX_REST_PUBLIC_USER as used for APEX_PUBLIC_USER.
  - Skip passwords - Enter 3 to skip configuring RESTful Services. Note that entering 3 disables RESTful Services.

**See Also:** Section 1.4.2, "About the Database Users Used by Oracle REST Data Services"

### 1.4.4 Using SQL Developer Oracle REST Data Services Administration (Optional)

This section describes how to use Oracle SQL Developer to administer Oracle REST Data Services.

**See Also:** "Oracle REST Data Services Administration" in Oracle SQL Developer User’s Guide

**Topics:**

- About SQL Developer Oracle REST Data Services Administration
- Configuring an Administrator User

#### 1.4.4.1 About SQL Developer Oracle REST Data Services Administration

Oracle SQL Developer (release 3.2 and later) enables you to administer Oracle REST Data Services using a graphical user interface. Using SQL Developer for Oracle REST Data Services administration is optional.
Using this graphical user interface, you can update the database connections, JDBC settings, URL mappings, RESTful connections, security (allowed procedures, blocked procedures, validation function and virus scanning), Caching, Pre/Post Processing Procedures, Environment, and Excel Settings. Oracle SQL Developer also provides statistical reporting, error reporting, and logging.

See Also: "Oracle REST Data Services Administration" in Oracle SQL Developer User’s Guide

1.4.4.2 Configuring an Administrator User

If you want to be able to administer Oracle REST Data Services using SQL Developer, then you must configure an administrator user as follows:

- Execute the following command:
  
  ```java -jar ords.war user adminlistener "Listener Administrator"
  ```

- Enter a password for the adminlistener user.

- Confirm the password for the adminlistener user.

- If you are using Oracle REST Data Services without HTTPS, follow the steps in Section 1.4.5, "Using OAuth2 in Non HTTPS Environments".

When using SQL Developer to retrieve and/or upload an Oracle REST Data Services configuration, when prompted, enter the credentials provided in the preceding list.

1.4.5 Using OAuth2 in Non HTTPS Environments

RESTful Services can be protected with the OAuth2 protocol to control access to nonpublic data. To prevent data snooping, OAuth2 requires all requests involved in the OAuth2 authentication process to be transported using HTTPS. The default behavior of Oracle REST Data Services is to verify that all OAuth2 related requests have been received using HTTPS. It will refuse to service any such requests received over HTTP, returning an HTTP status code of 403 Forbidden.

This default behavior can be disabled in environments where HTTPS is not available as follows:

1. Locate the folder where the Oracle REST Data Services configuration is stored.

2. Edit the file named defaults.xml.

3. Add the following setting to the end of this file just before the </properties> tag.

   ```
   <entry key="security.verifySSL">false</entry>
   ```

4. Save the file.

5. Restart Oracle REST Data Services if it is running.

Note that it is only appropriate to use this setting in development or test environments. It is never appropriate to use this setting in production environments because it will result in user credentials being passed in clear text.

Note: Oracle REST Data Services must be restarted after making configuration changes. See your application server documentation for information on how to restart applications.
1.5 Running in Standalone Mode

Although Oracle REST Data Services supports the Java EE application servers described in Section 1.2.1, "Supported Java EE Application Servers", you also have the option of running in standalone mode. This section describes how to run Oracle REST Data Services in standalone mode.

Standalone mode is suitable for development use only, and is not supported for use in production deployments. Use a supported Java EE application server for production deployments.

Topics:
- Starting in Standalone Mode
- Stopping the Server in Standalone Mode
- About JServ Protocol (AJP) Support

1.5.1 Starting in Standalone Mode

To launch Oracle REST Data Services in standalone mode:

1. To start Standalone mode, execute the following command:
   ```
   java -jar ords.war
   ```
   If you have not yet completed configuration, you are prompted to do so.

   **Tip:** To see help on standalone mode options, execute the following command:
   ```
   java -jar ords.war help standalone
   ```

   **Note:** Standalone mode does not support HTTPS, so the steps in Section 1.4.5, “Using OAuth2 in Non HTTPS Environments” must be followed if you want to use RESTful Services that require secure access (including the SQL Developer Administration View).

2. When prompted, specify the location of the folder containing the Oracle Application Express static resources used by Oracle REST Data Services, or press Enter if you do not want to specify this location.

3. When prompted, enter the value of the HTTP port you want the standalone Oracle REST Data Services to listen on. The default port is 8080.

   You are only prompted for these values the first time you launch standalone mode.

   **Note:** Ensure that no other servers are listening on the port you choose. The default port 8080 is commonly used by HTTP or application servers including the embedded PL/SQL gateway.

1.5.2 Stopping the Server in Standalone Mode

To stop the Oracle REST Data Services server in standalone mode, at a command prompt press Ctrl+C.
1.5.3 About JServ Protocol (AJP) Support

Standalone mode supports the Apache JServ Protocol (AJP), enabling it to be integrated with web servers that support AJP (for example, Apache 2.2 and mod_jk). To enable the AJP, use the --ajp-port command line option with the standalone command to indicate the port to listen on (usually 8009). For example:

```
java -jar ords.war standalone --ajp-port 8009 --apex-images /usr/local/ords/images
```

**Note:** Enabling AJP implicitly disables HTTP. It is not possible to listen for both protocols simultaneously.

1.6 Deploying to Oracle WebLogic Server

This section describes how to deploy Oracle REST Data Services on Oracle WebLogic Server. It assumes that you have completed the installation process and are familiar with Oracle WebLogic Server. If you are unfamiliar with domains, managed servers, deployment, security, users and roles, refer to your Oracle WebLogic Server documentation.

**Topics:**
- About Oracle WebLogic Server
- Downloading and Unzipping Oracle REST Data Services
- Configuring Oracle REST Data Services
- Configuring Oracle Application Express Images
- Launching the Administration Server Console
- Installing the Oracle WebLogic Server Deployment
- Configuring WebLogic to Handle HTTP Basic Challenges Correctly
- Verifying the State and Health of apex and i

1.6.1 About Oracle WebLogic Server

You can download Oracle WebLogic Server from Oracle Technology Network. See:


To learn more about installing Oracle WebLogic Server, see Oracle Fusion Middleware Getting Started With Installation for Oracle WebLogic Server and Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server.

1.6.2 Downloading and Unzipping Oracle REST Data Services

For information, see Section 1.3, "Installing Oracle REST Data Services”.

1.6.3 Configuring Oracle REST Data Services

You must complete this step before deploying Oracle REST Data Services on WebLogic. For more information, see Section 1.4, "Configuring Oracle REST Data Services".
1.6.4 Configuring Oracle Application Express Images

Before you begin, you must create a web archive (WAR) file to reference the Oracle Application Express image files. Use the static command to create a web archive file named i.war:

```
java -jar ords.war static <apex directory>\images
```

Where:

- `<apex directory>` is the directory location of Oracle Application Express.

This command runs the `static` command contained in the `ords.war` file. It packages the Application Express static images into an archive file named `i.war`.

The created images WAR does not contain the static resources; instead, it references the location where the static resources are stored. Therefore the static resources must be available at the specified path on the server where the WAR is deployed.

**Tip:** Use `java -jar ords.war help static` to see the full range of options for the `static` command.

Use the `i.war` file to deploy to WebLogic in the following steps:

1. **Launching the Administration Server Console**
2. **Installing the Oracle WebLogic Server Deployment**
3. **Configuring WebLogic to Handle HTTP Basic Challenges Correctly**

1.6.5 Launching the Administration Server Console

To launch the Administration Server console:

1. Start an Administration Server.
2. Launch the WebLogic Administration Console by typing the following URL in your web browser:

```
http://<host>:<port>/console
```

Where:

- `<host>` is the DNS name or IP address of the Administration Server.
- `<port>` is the port on which the Administration Server is listening for requests (port 7001 by default).
3. Enter your WebLogic Administrator username and password.
4. If your domain is in *Production* mode, click the *Lock & Edit* button on the left-pane below the submenu Change Center. If your domain is in *Development* mode, this button does not appear.

1.6.6 Installing the Oracle WebLogic Server Deployment

**Tip:** The Oracle REST Data Services files, `ords.war` and `i.war`, must be available before you start this task. See Section 1.3, "Installing Oracle REST Data Services" and Section 1.6.4, "Configuring Oracle Application Express Images"

To install the deployment:
   The Summary of Deployments is displayed.
2. Click Install.
3. Specify the location of the ords.war file and click Next.
   The ords.war file is located in the folder where you unzipped the Oracle REST Data Services ZIP file. See Section 1.3, "Installing Oracle REST Data Services".
   Tip: WebLogic Server determines the context root from the file name of a WAR archive. If you need to keep backward compatibility, so that URLs are of the form http://server/apex/... rather than http://server/ords/..., then you must rename ords.war to apex.war before the deployment.

   The Install Application assistant is displayed.
4. Select Install this deployment as an application and click Next.
5. Select the servers and/or clusters to which you want to deploy the application or module and click Next.
   Tip: If you have not created additional Managed Servers or clusters, you do not see this assistant page.
6. In the Optional Settings, specify the following:
   a. Name - Enter: ords
   b. Security - Select the following:
      Custom Roles: Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor
   c. Source accessibility - Select:
      Use the defaults defined by the deployment's targets
7. Click Next.
   A summary page is displayed.
8. Under Additional configuration, select one of the following:
   ■ Yes, take me to the deployment's configuration - Displays the Configuration page.
   ■ No I will review the configuration later - Returns you to the Summary of Deployments page.
9. Review the summary of configuration settings that you have specified.
10. Click Finish.
11. Repeat the previous steps to deploy the i.war file.
   In the optional settings, specify the following:
   a. Name - Enter: i
b. Security - Select:
   Custom Roles: Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor

c. Source Accessibility - Select:
   Use the defaults defined by the deployment's targets

12. If your domain is in Production Mode, then on the Change Center click **Activate Changes**.

### 1.6.7 Configuring WebLogic to Handle HTTP Basic Challenges Correctly

By default WebLogic attempts to intercept all HTTP Basic Authentication challenges. This default behavior needs to be disabled for Oracle REST Data Services to function correctly.

See your WebLogic documentation for the location of the WebLogic configuration file named: `config.xml`

Add the `<enforce-valid-basic-auth-credentials>` element to `config.xml` within the `<security-configuration>` element. The edited file should look like the following:

```xml
<enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-credentials>
```

Save the updated `config.xml` file, and restart WebLogic if it is running.

### 1.6.8 Verifying the State and Health of apex and i

In the Summary of Deployments, select the **Control** tab and verify that both the `apex` and `i` State are Active and the Health status is OK.

If `apex` or `i` are not Active, then enable them. In the Deployments table, select the check box adjacent to `apex` and/or `i`. Click **Start** and select **Servicing all requests** to make them active.

### 1.7 Deploying to GlassFish Server

This section describes how to deploy Oracle REST Data Services on GlassFish Server.

**Topics:**
- About GlassFish Server
- Downloading and Unzipping Oracle REST Data Services
- Configuring Oracle REST Data Services
- Configuring Oracle Application Express Images
- Launching the Administration Server Console
- Installing the GlassFish Server Deployment

**Tip:** This section assumes that you have completed the installation process and are familiar with GlassFish Server. If you are unfamiliar with domains, servers, applications, security, users and roles, see your GlassFish Server documentation.
1.7.1 About GlassFish Server
You can install Oracle REST Data Services with GlassFish Server. GlassFish Server is available for download from the Oracle Technology Network. See:

1.7.2 Downloading and Unzipping Oracle REST Data Services
For information, see Section 1.3, "Installing Oracle REST Data Services".

1.7.3 Configuring Oracle REST Data Services
You must complete this step before deploying Oracle REST Data Services on GlassFish. For more information, see Section 1.4, "Configuring Oracle REST Data Services".

1.7.4 Configuring Oracle Application Express Images
Before you begin, you must create a web archive (WAR) file to reference the Oracle Application Express image files. Use the static command to create a web archive file named i.war:

general: java -jar ords.war static <apex directory>

Where:

- <apex directory> is the directory location of Oracle Application Express.

The created images WAR does not contain the static resources; instead, it references the location where the static resources are stored. Therefore the static resources must be available at the specified path on the server where the WAR is deployed.

Tip: Use java -jar ords.war help static to see the full range of options for the static command.

Use the i.war file to deploy to GlassFish in the following steps:

1. Launching the Administration Server Console
2. Installing the GlassFish Server Deployment

1.7.5 Launching the Administration Server Console
At least one GlassFish server domain must be started before you start the Administration Console.

To launch the Administration Console:

1. Launch the Administration Console by typing the following URL in your web browser:

   http://localhost:4848

2. If prompted, log in to the Administration Console.

   Tip: You are prompted to log in if you chose to require an administration password at the time GlassFish server was installed.
1.7.6 Installing the GlassFish Server Deployment

**Tip:** The Oracle REST Data Services files, ords.war and i.war must be available before you start this task. See Section 1.3, "Installing Oracle REST Data Services" and Section 1.7.4, "Configuring Oracle Application Express Images".

To install the deployment:

1. On the navigation tree, click the Application node.
   The Applications page is displayed.
2. Click the Deploy button.
   The Deploy Applications or Modules page is displayed.
3. Select Packaged File to be Uploaded to the Server and click Browse.
4. Navigate to the location of the ords.war file, select the file, and click Open.
   The Deploy Applications or Modules page is displayed.
5. On the Deploy Applications or Modules page, specify the following:
   a. Type: Web Application
   b. Context Root: ords
      **Tip:** The Context Root value defaults to ords. However you can change it to apex if you need to keep backward compatibility, so that URLs are of the form http://server/apex/... rather than http://server/ords/....
   c. Application Name: ords
   d. Status: Enabled
   e. Description: Oracle REST Data Services
   f. Accept all other default settings and click OK.
6. Repeat the previous steps to deploy the i.war file. Clear the Context Root field so that the context root set in the sun-web.xml is used.

The Applications page is displayed. A check mark should appear in the Enabled field for ords

**Tip:** If a check mark does not appear in the Enabled column for ords, then select the check box next to ords and click Enable.

1.8 Deploying to Apache Tomcat

This section describes how to deploy Oracle REST Data Services on Apache Tomcat.

Topics:
- About Apache Tomcat
- Downloading and Unzipping Oracle REST Data Services
- Configuring Oracle REST Data Services
- Configuring Oracle Application Express Images
1.8.1 About Apache Tomcat

You can download Apache Tomcat from:
http://tomcat.apache.org/download-70.cgi

Tip: This section assumes that you have completed the installation process and are familiar with Apache Tomcat. If you are unfamiliar with domains, servers, applications, security, users and roles, see your Apache Tomcat documentation.

1.8.2 Downloading and Unzipping Oracle REST Data Services

For information, see Section 1.3, "Installing Oracle REST Data Services".

1.8.3 Configuring Oracle REST Data Services

You must complete this step before deploying Oracle REST Data Services on Apache Tomcat. For more information, see Section 1.4, "Configuring Oracle REST Data Services".

1.8.4 Configuring Oracle Application Express Images

To configure Oracle Application Express Images on Apache Tomcat:

- Copy the contents of the <apex directory>/images folder to <Tomcat directory>/webapps/i/.

Where:
- <apex directory> is the directory location of the Oracle Application Express distribution.
- <Tomcat directory> is the folder where Apache Tomcat is installed.

1.8.5 Installing the Apache Tomcat Deployment

Tip: The Oracle REST Data Services file ords.war must be available before you start this task. See Section 1.3, "Installing Oracle REST Data Services" and Section 1.8.4, "Configuring Oracle Application Express Images".

To install the Apache Tomcat deployment:

1. Move the ords.war file into the webapps folder where Apache Tomcat is installed.

Tip: Apache Tomcat determines the context root from the file name of a WAR archive. If you need to keep backward compatibility, so that URLs are of the form http://server/apex/... rather than http://server/ords/..., then you must rename ords.war to apex.war before moving it into to the webapps folder.

2. Access Oracle REST Data Services typing the following URL in your web browser:
http://<hostname>:<port>/ords

Where:
- `<hostname>` is the name of the server where Apache Tomcat is running.
- `<port>` is the port number configured for Apache Tomcat application server.

### 1.9 Upgrading Oracle REST Data Services

In Oracle REST Data Services 2.0, the format of configuration files has changed and RESTful Services (also known as Resource Templates) are now stored in the Oracle Application Express schema.

If the configuration folder specified during Section 1.4, "Configuring Oracle REST Data Services" contains an existing Oracle REST Data Services 1.1 configuration, the configuration and any RESTful Services are migrated to 2.0 format.
This section explains how to configure Oracle REST Data Services for connecting to multiple databases for routing requests, and it refers to other documentation sources for other configuration information.

**Note:** Oracle REST Data Services must be restarted after making configuration changes. See your application server documentation for information on how to restart applications.

**Topics:**
- Configuring Multiple Databases
- Configuring Security, Caching, Pre- and Post Processing, Environment, and Excel Settings
- Developing RESTful Services for Use with Oracle REST Data Services

## 2.1 Configuring Multiple Databases

Oracle REST Data Services supports the ability to connect to more than one database. This section describes different strategies for routing requests to the appropriate database.

**Topics:**
- About the Request URL
- Configuring Additional Databases
- Routing Based on the Request Path Prefix
- Routing Based on Request URL Prefix

### 2.1.1 About the Request URL

Oracle REST Data Services supports a number of different strategies for routing requests to the appropriate database. All of these strategies rely on examining the request URL and choosing the database based on some kind of match against the URL. It is useful to recap the pertinent portions of a request URL. Consider the following URL:

```plaintext
https://www.example.com/ords/sales/f?p=1:1
```

This URL consists of the following sections:
- Protocol: `https`
Host Name: www.example.com

Context Root: /ords

The context root is the location at which Oracle REST Data Services is deployed on the application server.

Request Path: /sales/f?p=1.1

This is the portion of the request URL relative to the context root.

For different applications, it may be important to route requests based on certain prefixes in the request path or certain prefixes in the full request URL.

There are two steps to configuring multiple databases:

1. Configuring the database connection information
2. Configuring which requests are routed to which database

2.1.2 Configuring Additional Databases

When you first configure Oracle REST Data Services, you configure a default database connection named: apex. You can create additional database connections using the setup command.

Tip: To see full help for the setup command type:
java -jar ords.war help setup

To create a database connection type the following:
java -jar ords.war setup --database <database name>

Where:

- <database name> is the name you want to give the database connection.

You are prompted to enter the information required to configure the database. See Section 1.4, "Configuring Oracle REST Data Services" for more information on the data that must be entered.

After you have configured the additional databases, define the rules for how requests are route to the appropriate database: see Section 2.1.3, "Routing Based on the Request Path Prefix" and Section 2.1.3, "Routing Based on the Request Path Prefix".

2.1.3 Routing Based on the Request Path Prefix

You create request routing rules using the map-url command.

Tip: To see full help for the map-url command type:
java -jar ords.war help map-url

If you want to route requests based just on matching a prefix in the request path portion of the URL, use the map-url command as follows:
java -jar ords.war map-url --type base-path --workspace-id <workspace name> <path prefix> <database name>

Where:
2.1.3.1 Example of Routing Based on Request Path Prefix

Assuming Oracle REST Data Services is deployed on a system named example.com at the context path /ords, then create the following rule:

```
java -jar ords.war map-url --type base-path --workspace-id sales_rest /sales sales_db
```

This rule means that any requests matching `https://example.com/ords/sales/...` are routed to the `sales_db` database connection. The `sales_rest` workspace defined within the `sales_db` database is searched for RESTful Services definitions.

The previous rule matches all of the following requests:

- `https://example.com/ords/sales/f?p=1:1`
- `https://example.com/ords/sales/leads/`
- `https://www.example.com/ords/sales/forecasting.report?month=jan` (If www.example.com resolves to the same system as example.com.)

The previous rule does not match any of the following requests:

- `http://example.com/ords/sales/f?p=1:1` (The protocol is wrong.)
- `https://example.com:8080/ords/sales/f?p=1:1  (The port is wrong: 443 is default for https, but don’t specify if using default.)`
- `https://example.com/ords/f?p=1:1` (Missing the /sales prefix.)
- `https://example.com/pls/sales/leads/` (The context path is wrong.)

2.1.4 Routing Based on Request URL Prefix

If you want to route requests based on a match of the request URL prefix, use the `map-url` command as follows:

```
java -jar ords.war map-url --type base-url --workspace-id <workspace name> <url prefix> <database name>
```

Where:

- `<workspace name>` is the name of the Oracle Application Express workspace where RESTful services for this connection are defined. This may be omitted if RESTful Services are not being used.
- `<url prefix>` is the prefix with which the request URL must start.
- `<database name>` is the name of the database connection.

2.1.4.1 Example of Routing Based on Request URL Prefix

Assuming Oracle REST Data Services is deployed on a system named example.com at the context path /ords, then create the following rule:

```
java -jar ords.war map-url --type base-url --workspace-id sales_rest https://example.com/ords/sales sales_db
```
This rule means that any requests matching `https://example.com/ords/sales/...` are routed to the `sales_db` database connection. The `sales_rest` workspace defined within the `sales_db` database is searched for RESTful Services definitions.

The previous rule matches all of the following requests:

- `https://example.com/ords/sales/f?p=1:1`
- `https://example.com/ords/sales/leads/`

The previous rule does not match any of the following requests:

- `http://example.com/ords/sales/f?p=1:1` (The protocol is wrong.)
- `https://example.com:8080/ords/sales/f?p=1:1` (The port is wrong: 443 is default for https, but don’t specify if using default.)
- `https://example.com/ords/f?p=1:1` (Missing the /sales segment of the base URL.)
- `https://example.com/pls/sales/leads/` (The context path is wrong.)
- `https://www.example.com/ords/sales/forecasting.report?month=jan` (The host name is wrong.)

### 2.2 Configuring Security, Caching, Pre- and Post Processing, Environment, and Excel Settings

To configure security, caching, pre- and post- processing, environment, and Excel settings, see Section 1.4.4, "Using SQL Developer Oracle REST Data Services Administration (Optional)".

### 2.3 Developing RESTful Services for Use with Oracle REST Data Services

For more information on how to develop RESTful Services for use with Oracle REST Data Services, see "Using RESTful Services" in Oracle Application Express SQL Workshop Guide.
The section describes the Oracle REST Data Services configuration file.

**Topics:**
- Locating Configuration Files
- Setting the Location of the Configuration Files
- Understanding the Configuration Folder Structure
- Understanding the Configuration File Format
- Understanding Configurable Parameters

### A.1 Locating Configuration Files

Use the `configdir` command to display the current location of the configuration files:

```
java -jar ords.war configdir
```

If the configuration folder has not yet been configured, the message: *The config.dir setting is not set*, is displayed. If it has been configured, the current value of the setting is displayed.

### A.2 Setting the Location of the Configuration Files

To change the location of the configuration folder use the `configdir` command:

```
java -jar ords.war configdir </path/to/config>
```

Where:
- `</path/to/config>` is the location where the configuration files are stored.

### A.3 Understanding the Configuration Folder Structure

The configuration folder has the following structure:

```
./
|--defaults.xml
|--apex.properties
|--url-mapping.xml
|--conf/
```
Global settings that apply to all database connections are stored in `defaults.xml`. Settings specific to a particular database connection (for example, the default `apex` connection) are stored in `conf/<db-name>.xml`, where `<db-name>` is the name of the database connection.

If the database connection uses RESTful services, then two additional files: `<db-name>_al.xml` and `<db-name>_rt.xml` store the configuration for the APEX_LISTENER and APEX_REST_PUBLIC_USER database users respectively.

### A.4 Understanding the Configuration File Format

Configuration files use the standard Java XML properties file format, where each configuration setting contains a key and a corresponding value. The following is an example of a `defaults.xml` file:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
  <entry key="db.connectionType">basic</entry>
  <entry key="db.hostname">localhost</entry>
  <entry key="db.port">1521</entry>
  <entry key="db.sid">orcl</entry>
  <entry key="jdbc.DriverType">thin</entry>
  <entry key="jdbc.InitialLimit">3</entry>
  <entry key="jdbc.MinLimit">1</entry>
  <entry key="jdbc.MaxLimit">10</entry>
  <entry key="jdbc.MaxStatementsLimit">10</entry>
  <entry key="jdbc.InactivityTimeout">1800</entry>
  <entry key="jdbc.statementTimeout">900</entry>
  <entry key="jdbc.MaxConnectionReuseCount">1000</entry>
</properties>
```

### A.4.1 Understanding the url-mapping.xml File Format

The `url-mapping.xml` file stores the rules that route requests to the appropriate database when more than one database is configured. The following is an example of a `url-mapping.xml` file:

```xml
<pool-config xmlns="http://xmlns.oracle.com/apex/pool-config">
  <pool name="sales_db" uri-pattern="/sales" type="base-path" workspace-id="sales_rest/>
</pool-config>
```
A.5 Understanding Configurable Parameters

Table A–1 lists editable parameters for the defaults.xml and (db-name).xml configuration files.

**Tip:** Oracle recommends users to use the Oracle REST Data Services command-line interface and Oracle SQL Developer Oracle REST Data Services Administration to edit the configuration files. For more information, see Section 1.4, "Configuring Oracle REST Data Services" and "Oracle REST Data Services Administration" in Oracle SQL Developer User’s Guide.

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>apex.docTable</td>
<td>string</td>
<td>Name of the document table used by Application Express.</td>
<td>MYDOCTABLE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Defaults to FLOWS_FILES.WWV_FLOW_FILE_OBJECTS$.</td>
<td></td>
</tr>
</tbody>
</table>
| apex.excel2collection       | boolean | Indicate whether to place your Excel files into an Oracle Application Express collection. Supported values:
|                             |         | ■ true                                                                      | false         |
|                             |         | ■ false (default)                                                          |               |
|                             |         | If value is true, then either apex.excel2collection.onecollection or apex.excel2collection.useSheetName should be set to true. |               |
| apex.excel2collection.name  | string  | The name of the apex collection. The name is required if apex.excel2collection.onecollection is true. | mycollection  |
| apex.excel2collection.onecollection | boolean | Indicate whether to put all Excel worksheets into a single collection. Supported values:
|                             |         | ■ true                                                                      | false         |
|                             |         | ■ false (default)                                                          |               |
| apex.excel2collection.useSheetName | boolean | Indicate whether to create a collection for each Excel worksheet, and uses each worksheet name for the corresponding collection name. Supported values:
|                             |         | ■ true                                                                      | false         |
|                             |         | ■ false (default)                                                          |               |
| cache.caching               | boolean | Supported values:                                                          | true          |
|                             |         | ■ true                                                                      |               |
|                             |         | ■ false (default)                                                          |               |
|                             |         | For caching to be enabled, this must be set to true and the procedureNameList must have a procedure. |               |
### Understanding Configurable Parameters

#### Table A–1  (Cont.) Oracle REST Data Services Configuration Files Parameters

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>cache.directory</td>
<td>string</td>
<td>The directory location for the cache files.</td>
<td>C:\data\cachefiles</td>
</tr>
<tr>
<td>cache.duration</td>
<td>string</td>
<td>Supported values:</td>
<td>days</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ days (default)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ minutes</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ hours</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Required for expire cache type.</td>
<td></td>
</tr>
<tr>
<td>cache.expiration</td>
<td>numeric</td>
<td>Required for expire cache type.</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Defaults to 7.</td>
<td></td>
</tr>
<tr>
<td>cache.maxEntries</td>
<td>numeric</td>
<td>Required for lru cache type.</td>
<td>500</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Defaults to 500.</td>
<td></td>
</tr>
<tr>
<td>cache.monitorInterval</td>
<td>numeric</td>
<td>Interval time is specified in minutes.</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the cache type is expire, Oracle REST Data Services, checks the cache every <code>NN</code> minutes for files that have expired. For example, if the monitorInterval is 60, then it checks the cache every 60 minutes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Defaults to 60.</td>
<td></td>
</tr>
<tr>
<td>cache.procedureNameList</td>
<td>string</td>
<td>Specify the procedure names to allow for caching of their files.</td>
<td>p, download_file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Procedure names can contain the wildcard characters asterisk (<em>) or question mark (?). Use an asterisk (</em>) to substitute zero or more characters and a question mark (?) to substitute for any one character.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Each procedure name must be separated by a comma.</td>
<td></td>
</tr>
<tr>
<td>cache.type</td>
<td>string</td>
<td>Supported values:</td>
<td>lru</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ expire</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ lru (default)</td>
<td></td>
</tr>
<tr>
<td>db.connectionType</td>
<td>string</td>
<td>The type of connection. Supported values:</td>
<td>basic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ basic</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ tns</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ customurl</td>
<td></td>
</tr>
<tr>
<td>db.customURL</td>
<td>string</td>
<td>The JDBC URL connection to connect to the database.</td>
<td>jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP))(HOST=myhost)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=ora111.us.example.com)))</td>
</tr>
<tr>
<td>db.hostname</td>
<td>string</td>
<td>The host system for the Oracle database.</td>
<td>myhostname</td>
</tr>
</tbody>
</table>
### Table A–1 (Cont.) Oracle REST Data Services Configuration Files Parameters

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>db.password</td>
<td>string</td>
<td>The password of the specified database user. Include an exclamation at the</td>
<td>password4user</td>
</tr>
<tr>
<td></td>
<td></td>
<td>beginning of the password so that it can be stored encrypted.</td>
<td></td>
</tr>
<tr>
<td>db.port</td>
<td>numeric</td>
<td>The database listener port.</td>
<td>1521</td>
</tr>
<tr>
<td>db.servicename</td>
<td>string</td>
<td>The network service name of the database.</td>
<td>orall1.example.co m</td>
</tr>
<tr>
<td>db.sid</td>
<td>string</td>
<td>The name of the database.</td>
<td>orall1</td>
</tr>
<tr>
<td>db.tnsAliasName</td>
<td>string</td>
<td>The TNS alias name that matches the name in the tnsnames.ora file.</td>
<td>MY_TNSALIAS</td>
</tr>
<tr>
<td>db.tnsDirectory</td>
<td>string</td>
<td>The directory location of your tnsnames.ora file.</td>
<td>C:\ORACLE\NETWORK\ADMIN</td>
</tr>
<tr>
<td>db.username</td>
<td>string</td>
<td>The name of the database user for the connection.</td>
<td>APEX_PUBLIC_USER</td>
</tr>
<tr>
<td>debug.debugger</td>
<td>boolean</td>
<td>Indicate whether to display debugging messages on the application server</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>console.</td>
<td></td>
</tr>
<tr>
<td>debug.printDebugToScreen</td>
<td>boolean</td>
<td>Indicate whether to display error messages on the browser.</td>
<td>false</td>
</tr>
<tr>
<td>error.keepErrorMessages</td>
<td>boolean</td>
<td>Indicate whether to retain the error messages.</td>
<td>true</td>
</tr>
<tr>
<td>error.maxEntries</td>
<td>numeric</td>
<td>Specify the total number of error messages to retain.</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Defaults to 50.</td>
<td></td>
</tr>
<tr>
<td>icap.port</td>
<td>numeric</td>
<td>Specify the Internet Content Adaptation Protocol (ICAP) Port to virus scan</td>
<td>5555</td>
</tr>
<tr>
<td></td>
<td></td>
<td>files. The icap.port is required to have a value.</td>
<td></td>
</tr>
<tr>
<td>icap.server</td>
<td>string</td>
<td>Specify the Internet Content Adaptation Protocol (ICAP) Server name to virus</td>
<td>servername</td>
</tr>
<tr>
<td></td>
<td></td>
<td>scan files. The icap.server is required to have a value.</td>
<td></td>
</tr>
<tr>
<td>jdbc.DriverType</td>
<td>string</td>
<td>The JDBC driver type. Supported values:</td>
<td>thin</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ thin</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ oci8</td>
<td></td>
</tr>
</tbody>
</table>
### Table A–1 (Cont.) Oracle REST Data Services Configuration Files Parameters

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>jdbc.InactivityTimeout</td>
<td>numeric</td>
<td>Specify how long an available connection can remain idle before it is closed. The inactivity connection timeout is in seconds. Defaults to 1800.</td>
<td>1800</td>
</tr>
<tr>
<td>jdbc.InitialLimit</td>
<td>numeric</td>
<td>Specify the initial size for the number of connections that will be created. Defaults to 3.</td>
<td>3</td>
</tr>
<tr>
<td>jdbc.MaxConnectionReuseCount</td>
<td>numeric</td>
<td>Specify the maximum number of times to reuse a connection before it is discarded and replaced with a new connection. Defaults to 1000.</td>
<td>1000</td>
</tr>
<tr>
<td>jdbc.MaxLimit</td>
<td>numeric</td>
<td>Specify the maximum number of connections. Defaults to 10.</td>
<td>10</td>
</tr>
<tr>
<td>jdbc.MaxRows</td>
<td>numeric</td>
<td>Specify the maximum number of rows that will be returned from a query when processing a RESTful service and that will be returned from a nested cursor in a result set. Affects all RESTful services generated through a SQL query, regardless of whether the resource is paginated. Defaults to 500.</td>
<td>300</td>
</tr>
<tr>
<td>jdbc.MaxStatementsLimit</td>
<td>numeric</td>
<td>Specify the maximum number of statements to cache for each connection. Defaults to 10.</td>
<td>10</td>
</tr>
<tr>
<td>jdbc.MinLimit</td>
<td>numeric</td>
<td>Specify the minimum number of connections. Defaults to 1.</td>
<td>1</td>
</tr>
<tr>
<td>jdbc.statementTimeout</td>
<td>numeric</td>
<td>Specify how long a borrowed (in use) connection can remain unused before it is considered as abandoned and reclaimed. The abandoned connection timeout is in seconds. Defaults to 900.</td>
<td>900</td>
</tr>
<tr>
<td>log.logging</td>
<td>boolean</td>
<td>Indicate whether to retain the log messages. Supported values: true, false (default)</td>
<td>true</td>
</tr>
<tr>
<td>log.maxEntries</td>
<td>numeric</td>
<td>Specify the total number of log messages to retain. Defaults to 50.</td>
<td>50</td>
</tr>
</tbody>
</table>
### Understanding Configurable Parameters

#### About the Configuration File

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>log.procedure</td>
<td>boolean</td>
<td>Indicate whether procedures are to be logged.</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Supported values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ true</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ false (default)</td>
<td></td>
</tr>
<tr>
<td>misc.defaultPage</td>
<td>string</td>
<td>The default page to display. The Oracle REST Data Services home page, apex, is commonly used.</td>
<td>apex</td>
</tr>
<tr>
<td>procedure.postProcess</td>
<td>string</td>
<td>Specify the procedure name(s) to execute after executing the procedure specified on the URL. Multiple procedure names must be separated by commas.</td>
<td>SCHEMA1.SUBMIT.REQUEST, FINISHTASK</td>
</tr>
<tr>
<td>procedure.preProcess</td>
<td>string</td>
<td>Specify the procedure name(s) to execute prior to executing the procedure specified on the URL. Multiple procedure names must be separated by commas.</td>
<td>SCOTT.PREPROC1, INITIALIZE, PKG1.PROC</td>
</tr>
<tr>
<td>security.disableDefaultExclusionList</td>
<td>boolean</td>
<td>Supported values:</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ true</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ false (default)</td>
<td></td>
</tr>
<tr>
<td>security.exclusionList</td>
<td>string</td>
<td>Specify a pattern for procedures, packages, or schema names which are forbidden to be directly executed from a browser. Procedure names can contain the wildcard characters asterisk (<em>) or question mark (?). Use an asterisk (</em>) to substitute zero or more characters and a question mark (?) to substitute for any one character.</td>
<td>customer_account, bank*, employe?</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Separate multiple patterns using commas.</td>
<td></td>
</tr>
<tr>
<td>security.inclusionList</td>
<td>string</td>
<td>Specify a pattern for procedures, packages, or schema names which are allowed to be directly executed from a browser. Procedure names can contain the wildcard characters asterisk (<em>) or question mark (?). Use an asterisk (</em>) to substitute zero or more characters and a question mark (?) to substitute for any one character.</td>
<td>apex, p, v, f, wwv.<em>, y</em>, c*</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Separate multiple patterns using commas.</td>
<td></td>
</tr>
</tbody>
</table>
### Oracle REST Data Services Configuration Files Parameters (Cont.)

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>security.maxEntries</td>
<td>numeric</td>
<td>Specify the maximum cache size. Defaults to 2000.</td>
<td>2000</td>
</tr>
<tr>
<td>security.requestValidationFunction</td>
<td>string</td>
<td>Specify a validation function to determine if the requested procedure in the URL should be allowed or disallowed for processing. The function should return true if the procedure is allowed; otherwise, return false.</td>
<td>CHECK_VALID_PROCEDURE</td>
</tr>
<tr>
<td>security.verifySSL</td>
<td>boolean</td>
<td>Indicate whether HTTPS is available in your environment. Supported values:</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ true</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ false (default)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you change the value to false, see Section 1.4.5, &quot;Using OAuth2 in Non HTTPS Environments&quot;.</td>
<td></td>
</tr>
</tbody>
</table>
This appendix contains information on troubleshooting.

Topics:
- Enabling Debug Tracing
- Enabling Detailed Request Error Messages
- Configuring Application Express Static Resources with Oracle REST Data Services

B.1 Enabling Debug Tracing

To enable debug tracing, add the following setting to the Oracle REST Data Services configuration file named: defaults.xml:

```xml
<entry key="debug.debugger">true</entry>
```

When this setting is present in defaults.xml, detailed logging information that may help with problem diagnosis is appended to the Oracle REST Data Services log output. This setting should not be enabled on production systems due to the performance impact of outputting large amounts of data to the log.

B.2 Enabling Detailed Request Error Messages

To enable detailed request error messages, add the following setting to the Oracle REST Data Services configuration file named: defaults.xml:

```xml
<entry key="debug.printDebugToScreen">true</entry>
```

When this setting is present in defaults.xml, any request that produces an error response includes a detailed message, including a stack trace. This setting must not be enabled on production systems due to the risk of sensitive information being revealed to an attacker.

B.3 Configuring Application Express Static Resources with Oracle REST Data Services

When using Oracle REST Data Services, a blank page might be displayed when attempting to access an Oracle Application Express page, for example, when attempting to display https://example/ords/. This problem is caused by an improper configuration of Application Express static resources, which causes the JavaScript and CSS resources required by Application Express not to be found and the Application Express page not to render correctly.
The specific cause can be any of the following:

- Forgetting to ensure that the Application Express static images are located on the same server as the Oracle REST Data Services instance
- Forgetting to deploy i.war on WebLogic Server or GlassFish
- Specifying an incorrect path when using the `java -jar ords.war static` command to generate i.war
- Configuring Application Express to use a nondefault context path for static resources (/i) and not specifying the same context path (using the `--context-path` option) when using `java -jar ords.war static`
- Moving, renaming, or deleting the folder pointed to by i.war after deploying i.war
- When running in Standalone mode, entering an incorrect path (or not specifying a path) when prompted on the first run of Standalone mode
- When running in Standalone mode, entering an incorrect path with the `--static-images` option
- Upgrading to a new version of Application Express and forgetting to reconfigure and redeploy i.war to point to the static resources for the new Application Express version, or in Standalone mode forgetting to update the location by using the `--apex-images` option

To help in diagnosing the problem, you can try to access the `apex_version.txt` file. For example, if your Application Express deployment is located at `https://example.com/ords/` and your static resources have been deployed at `https://example.com/i/`, use a browser to access the following URL:

`https://example.com/i/apex_version.txt`

If you get a 404 Not Found error, then check the preceding list of possible specific causes, including i.war not being deployed or not pointing to a folder containing Application Express static resources.

If a plain text file is displayed, it should contain text like the following:

```
Application Express Version: 4.2.1
```

Check that the version number matches the version of Application Express that is deployed on the database. If the numbers do not match, check if you have made an error mentioned in the last item in the preceding list of possible specific causes, because Oracle REST Data Services is not configured to use the correct version of the Application Express static resources to match the Application Express version in the database.

If you need help in solving the problem, check the information in this book about creating and deploying i.war for your environment, such as WebLogic Server or Glassfish.

You can also get detailed help on the static listener command by entering the following at a command prompt:

```
java -jar ords.war help static
```
Index

A
Apache Tomcat, 1-2
  about, 1-14
  configuring Oracle REST Data Services images, 1-14
  deploying to, 1-13
  downloading, 1-14

C
command-line interface, 1-4
configdir command, A-1
  locating configuration files, A-1
  locating configuration folder, A-1
configuration file editable parameters
  apex.docTable, A-3
  apex.excel2collection, A-3
  apex.excel2collection.name, A-3
  apex.excel2collection.onecollection, A-3
  apex.excel2collection.useSheetName, A-3
  cache.caching, A-3
  cache.directory, A-4
  cache.duration, A-4
  cache.expiration, A-4
  cache.maxEntries, A-4
  cache.monitorInterval, A-4
  cache.procedureNameList, A-4
  cache.type, A-4
  db.connectionType, A-4
  db.customURL, A-4
  db.hostname, A-4
  db.password, A-5
  db.port, A-5
  db.servicename, A-5
  db.sid, A-5
  db.tnsAliasName, A-5
  db.tnsDirectory, A-5
  db.username, A-5
  debug.debugger, A-5
  debug.printDebugToScreen, A-5
  error.keepErrorMessages, A-5
  error.maxEntries, A-5
  icap.port, A-5
  icap.server, A-5
  jdbc.DriverType, A-5
  jdbc.InactivityTimeout, A-6
  jdbc.InitialLimit, A-6
  jdbc.MaxConnectionReuseCount, A-6
  jdbc.MaxLimit, A-6
  jdbc.MaxRows, A-6
  jdbc.MaxStatementsLimit, A-6
  jdbc.MinLimit, A-6
  log.logging, A-6
  log.maxEntries, A-6
  log.procedure, A-7
  misc.defaultPage, A-7
  procedure.postProcess, A-7
  procedure.preProcess, A-7
  security.disableDefaultExclusionList, A-7
  security.exclusionList, A-7
  security.inclusionList, A-7
  security.maxEntries, A-8
  security.requestValidationFunction, A-8
  security.verifySSL, A-8
  statementTimeout, A-6
configuration files, A-1
  format of, A-2
  locating using configdir command, A-1
configuration folder
  setting location, A-1
  structure of, A-1

D
database users, 1-4
defaults.xml
  configuration file
    Oracle REST Data Services, B-1
    enabling debug tracing, B-1
    enabling detailed request error messages, B-1
defaults.xml, file format, A-2
deploy options
  Apache Tomcat, 1-13
deployment options
  GlassFish Server, 1-11
  Oracle WebLogic Server, 1-8
downloading
  Apache Tomcat, 1-14
  GlassFish Server, 1-12
  Oracle WebLogic Server, 1-8
GlassFish Server, 1-2  
  about, 1-12  
  creating a WAR file for images, 1-12  
  deploy to, 1-11  
  downloading, 1-12  
  installing the deployment, 1-13, 1-14  
  graphical user interface administration, 1-5

installation options  
  standalone mode, 1-7  
  installation overview, 1-2

Java EE application servers  
  about supported, 1-2

multiple database configuration, 2-1  
  about the request URL, 2-1  
  configuring additional databases, 2-2  
  routing request rules, 2-2  
  routing requests based on URL prefix, 2-3

OAuth2, default behavior, 1-6  
  Oracle GlassFish Server  
  launching Administration Console, 1-12  
  Oracle REST Data Services  
  about, 1-1  
  about configuring, 1-3  
  about upgrading, 1-15  
  administering with graphical user interface, 1-5  
  caching, 2-4  
  configuration file  
    defaults.xml, B-1  
    configuration files, A-1  
    configuring, 1-4, 2-1  
    configuring multiple databases, 2-1  
    configuring with command-line interface, 1-4  
    database users, 1-4  
    developing RESTful services, 2-4  
    downloading, 1-3  
    environment, 2-4  
    Excel settings, 2-4  
    installation overview, 1-2  
    installing, 1-3  
    pre- and post-processing, 2-4  
    running in standalone mode, 1-7  
    security, 2-4  
    system requirements, 1-2  
  Oracle REST Data Services configuration file  
    enabling debug tracing, B-1  
    enabling detailed request error messages, B-1  
  Oracle WebLogic Server, 1-2

RESTful services  
  developing, 2-4

SQL Developer Oracle REST Data Services  
  Administration, 1-5  
  standalone mode  
    about JServ Protocol (AJP) support, 1-8  
    starting, 1-7  
    stopping the server, 1-7  
    standalone mode, running in, 1-7  
    structure of configuration folder, A-1  
    supported Java EE application servers, 1-2  
    system requirements, 1-2

troubleshooting, B-1  
  enabling debug tracing, B-1  
  enabling detailed request error messages, B-1

url-mapping.xml  
  file format, A-2  
  request rules routing, A-2