

Oracle® Business Intelligence Publisher

Converting Oracle Reports to Oracle Business Intelligence Publisher

11g Release 1 (11.1.1)

E28918-02

June 2013

Welcome to *Converting Oracle Reports to Oracle Business Intelligence Publisher*. This guide covers the following topics:

- [Overview](#)
- [Obtaining Oracle Reports to BI Publisher Conversion Assistant](#)
- [Prerequisites and Limitations of Oracle Reports to BI Publisher Conversion Assistant](#)
- [Running Oracle Reports to BI Publisher Conversion Assistant](#)
- [Compiling PL/SQL Packages on the Database](#)
- [Moving Converted Reports to the Oracle BI Publisher Repository](#)
- [Testing and Editing Converted Reports](#)
- [Troubleshooting Oracle Reports to BI Publisher Conversion Assistant](#)
- [Documentation Accessibility](#)

1 Overview

Oracle BI Publisher provides Oracle Reports to BI Publisher Conversion Assistant as a tool to convert reports from the Oracle Reports format to the Oracle BI Publisher format. The Conversion Assistant can be used for BI Publisher standalone or BI Publisher integrated with Oracle Business Intelligence Enterprise Edition.

In Oracle Reports, the data model (SQL query or extraction logic) and report layout specifications are contained in a single file. In BI Publisher, the data model and the layout are separate objects. Therefore, the Conversion Assistant generates several files from a single Oracle report file to compose the report in BI Publisher.

The Conversion Assistant generates a report definition file, data model file, and layout template file. These are the BI Publisher objects that are uploaded to the BI Publisher repository. When the Oracle report includes calls to package functions, the Conversion Assistant also creates the PL/SQL specification and body.

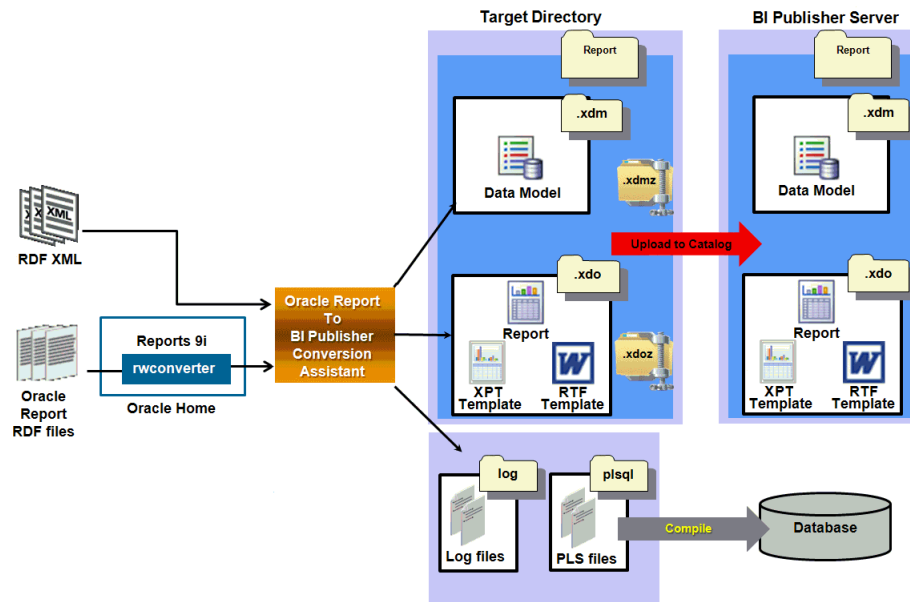
After the Conversion Assistant completes the report conversion, compile the .pls files on the database, and test the report to ensure that the output is as expected. If the report fails to display the expected results, review the log files (conversion log, upload log, and BI Publisher Server log).

In some cases, the data model and layout template may require manual adjustments to improve report output.

Some reports contain structures that the Conversion Assistant cannot convert. These structures must be manually implemented in the converted reports.

Figure 1 describes the conversion process.

Figure 1 Oracle Reports to BI Publisher Reports Conversion Process



The overall flow for the conversion process is as follows:

1. In BI Publisher, create a JDBC or JNDI data source connection that connects to the same database as the Oracle Report.
2. Run the Conversion Assistant.
3. (Conditional) Upload the report to the BI Publisher Catalog if you did not choose the automatic upload option during the conversion.
4. (Conditional) Compile the PL/SQL packages on the database.
5. Test the report and review the conversion log files to identify any manual modifications needed to complete the conversion.

2 Obtaining Oracle Reports to BI Publisher Conversion Assistant

To obtain Oracle Reports to BI Publisher Conversion Assistant:

1. Go to the following website:
<http://www.oracle.com/technetwork/middleware/bi-publisher/downloads/index.html>
2. Download OR2BIPConvAssist.zip for your Windows or Linux environment and unzip the installation zip files to any directory on your machine. The following folders are created in the directory: bin, config, and lib.
3. Under the bin folder, use OR2BIPConvAssist.bat or OR2BIPConvAssist.sh to launch the Conversion Assistant.

3 Prerequisites and Limitations of Oracle Reports to BI Publisher Conversion Assistant

The following are prerequisites for running the Conversion Assistant:

- If the source Oracle reports are not in XML format, then you must have Oracle Reports *rwconverter 9i* in the Oracle Home.
- You must have a BI Publisher role with Write access to Shared Folders in the BI Publisher Catalog.

3.1 Converting Oracle Reports to XML Format

The Conversion Assistant uses Oracle Report XML (or RDF XML) to convert Oracle reports into BI Publisher reports. The conversion from RDF binary to RDF XML report formatting is supported for Oracle Reports *9i* and above.

The Conversion Assistant allows source reports to be in both formats - RDF binary and RDF XML.

The following requirements are necessary in order for the Conversion Assistant to convert reports:

- Oracle Reports Designer *9i* or later installed on the same machine as the Conversion Assistant.
- You must enter the Oracle Reports Home Path in the Conversion Assistant in order to call the *rwconverter* executable and convert the reports into Oracle Reports XML format.

Oracle Reports Designer is part of the Oracle Developer Suite 10g (10.1.2.0.2) and is available from:

<http://www.oracle.com/technology/software/products/ids/index.html>

3.2 Limitations

The Conversion Assistant has the following limitations:

- Format triggers cannot be converted, but the code is written to a log file for manual implementation. For more information on PL/SQL format triggers in converted layout templates, see [Section 7.2, "PL/SQL Format Trigger Logic Not Supported in Converted Layout Templates."](#)
- Charts can be converted in the XPT format, but not in the RTF format. If you create a RTF layout, use the Template Builder for Word to create charts. If you create a XPT layout, use the Layout Editor to further define the charts. For more information about the layout design tools, see *Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher*.
- Matrix reports cannot be converted. If you choose to create a RTF layout, use the Template Builder for Word's Pivot Table Wizard. If you choose to create a BI Publisher layout template, use the Layout Editor to create pivot tables. For more information about the layout design tools, see *Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher*.

4 Running Oracle Reports to BI Publisher Conversion Assistant

This section includes:

- [If the Source Report Is an Oracle Reports RDF File](#)
- [If the Source Report Is in Oracle Reports XML Format](#)
- [Output Files](#)

4.1 If the Source Report Is an Oracle Reports RDF File

If the source report is an Oracle Reports RDF File, perform the following steps:

1. Create the JDBC/JNDI Connection data source on the BI Publisher Administration page. This data source connects to the same database as the Oracle report. You must have the Administrator role to perform this step.
2. Launch the Conversion Assistant.
3. Select the source type, **RDF Binary (.rdf)**, and then click **Next**.
4. Enter the **Oracle Reports Home**, for example, C:\Oracle_1\Middleware\OracleFRHome1.

The Conversion Assistant requires the `rwconverter` from Oracle Reports in order to convert the report from RDF format to XML format. Specify the Oracle Reports home path where Oracle Reports Designer (9i or later version) is installed. The Conversion Assistant assumes that `rwconverter` is contained in the `bin` directory beneath the Oracle Reports home path specified.

5. Click **Next**.
6. Enter the following fields:
 - **Source Directory** - source directory for Oracle Reports files.
 - **Target Directory** - location to place the output files. For each converted report, the Conversion Assistant creates the following separate output objects in the **Target Directory**: Oracle BI Publisher Report folder (.xdo) containing the report definition (.xdo) and layout template (.rtf) files, the data model file (.xdm), zipped report definition file (.xdoz) and zipped data model file (.xdmz), the `plsql_pkgs` folder containing the PL/SQL packages, and `bipconvert.log` and `bipupload.log` files.

Note: You can use the zipped report definition file (.xdoz) and the zipped data model file (.xdmz) to manually upload reports to the BI Publisher Catalog.

- **Debug Mode** - specifies if the Conversion Assistant runs in debug mode, and writes debug statements to the log files.

Select **ON** to write conversion process details and any statement level error messages to a log file at the report level in the target directory.

Select **OFF** to write the details of the conversion process, including the complete error stack in case there are exceptions, to a log file at the component level in the target directory.
7. Click **Next**.
 8. Enter the data source connection name you created in step 1, and then click **Next**.

The data models included in the report conversion are automatically named with the data source name you specify in this step.

9. Select the Layout Type as **RTF Template**, **XPT Template**, or **Both**, and then click **Next**.
10. Select the **Upload to BI Publisher Server Catalog** check box to automatically upload the reports to the BI Publisher Server. If you select this check box, you must provide the BI Publisher Server URL (http://<server>:<port>/xmlpserver,) username and password, and the targeted output folder in the BI Publisher Catalog (under the Shared folder) where the converted reports are stored. If you specify a folder that does not yet exist in the BI Publisher Catalog, it is automatically created during the conversion process.

If you do not wish to automatically upload the converted reports to the BI Publisher Catalog, ensure that the **Upload to BI Publisher Server Catalog** check box is not selected. If this check box is not selected, the converted report files are only created in the **Target Directory** specified in step 6.

Note: If you choose to automatically upload the converted reports to the BI Publisher Server, the bipupload.log file is populated with the details in the **Target Directory**.

11. Preview the conversion settings, and then click **Convert**. The Conversion Assistant converts the Oracle Reports and creates a folder for each of the Oracle Reports in the **Target Directory**. If you selected **Upload to BI Publisher Server Catalog**, a folder is created for each report in the specified folder under the Shared folder. Each report folder contains the data model and report.
12. Compile the PL/SQL packages (.pls files) on the database. The PL/SQL packages are created in the **Target Directory** during the conversion process. For more information, see [Section 5, "Compiling PL/SQL Packages on the Database."](#)

Note: The Conversion Assistant populates the **Oracle DB Default Package** field in the data model with the package name defined in the report.

13. (Conditional) If you did not select **Upload to BI Publisher Server Catalog**, use the **Upload Resource** feature in the BI Publisher Catalog to upload each zipped data model and report file (.xdmz and .xdoz). Ensure that you use the BI Publisher UI (/xmlpserver) to upload both objects to the same folder in the BI Publisher Catalog.

For more information, see "Downloading and Uploading Catalog Objects" in *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

14. View the converted reports in the Report Viewer.
15. If the reports are not converted as expected, review the log files. For more information, see [Section 8, "Troubleshooting Oracle Reports to BI Publisher Conversion Assistant."](#)

4.2 If the Source Report Is in Oracle Reports XML Format

If the source report is in Oracle Reports XML format, follow these steps:

1. Create the JDBC/JNDI Connection data source on the BI Publisher Administration page. This data source connects to the same database as the Oracle report. You must have the Administrator role to perform this step.
2. Launch the Conversion Assistant.
3. Select the source type, **RDF XML (.xml)**, and then click **Next**.
4. Enter the following fields:
 - **Source Directory** - source directory for the Oracle Reports files.
 - **Target Directory** - location to place the output files. For each converted report, the Conversion Assistant creates the following separate output objects in the **Target Directory**: Oracle BI Publisher report folder (.xdo) containing the report definition (.xdo) and layout template (.rtf) files, the data model file (.xdm), zipped report definition file (.xdoz) and zipped data model file (.xdmz), the plsqli_pkgs folder containing the PL/SQL packages, and bipconvert.log and bipupload.log files.

Note: You can use the zipped report definition file (.xdoz) and the zipped data model file (.xdmz) to manually upload reports to the BI Publisher Catalog.

- **Debug Mode** - specifies if the Conversion Assistant runs in debug mode, and writes debug statements to the log files.

Select **ON** to write the details of the conversion process, including the complete error stack in case there are exceptions, to a log file at the component level in the target directory.

Select **OFF** to write conversion process details and any statement level error messages to a log file at the report level in the target directory.
5. Click **Next**.
 6. Enter the data source connection name you created in step 1, and then click **Next**.

The data models included in the report conversion are automatically named with the data source name you specify in this step.
 7. Select the Layout Type as **RTF Template**, **XPT Template**, or **Both**, and then click **Next**.
 8. Select the **Upload to BI Publisher Server Catalog** check box to automatically upload the reports to the BI Publisher Server. If you select this check box, you must provide the BI Publisher Server URL (http://<server>:<port>/xmlpserver,) username and password, and the targeted output folder in the BI Publisher Catalog (under the Shared folder) where the converted reports are stored. If you specify a folder that does not yet exist in the BI Publisher Catalog, it is automatically created during the conversion process.

If you do not wish to automatically upload the converted reports to the BI Publisher Catalog, ensure that the **Upload to BI Publisher Server Catalog** check box is not selected. If this check box is not selected, the converted report files are only created in the **Target Directory** specified in step 4.

Note: If you choose to automatically upload the converted reports to the BI Publisher Server, the bipupload.log file is populated with the details in the **Target Directory**.

9. Click **Convert**. The Conversion Assistant converts the Oracle reports and creates a folder for each report in the **Target Directory**. If you selected **Upload to BI Publisher Server Catalog**, a folder is created for each report in the specified folder under the Shared folder. Each report folder contains the data model and report.
10. Compile the PL/SQL packages (.pls files) on the database. The PL/SQL packages are created in the **Target Directory** during the conversion process. For more information, see [Section 5, "Compiling PL/SQL Packages on the Database."](#)

Note: The Conversion Assistant populates the **Oracle DB Default Package** field in the data model with the package name defined in the report.

11. (Conditional) If you did not select **Upload to BI Publisher Server Catalog**, use the **Upload Resource** feature in the BI Publisher Catalog to upload each zipped data model and report file (.xdmz and .xdoz). Ensure that you use the BI Publisher UI (/xmlpserver) to upload both objects to the same folder in the BI Publisher Catalog.

For more information, see "Downloading and Uploading Catalog Objects" in *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

12. View the converted reports in the Report Viewer.
13. If the reports are not converted as expected, review the log files. For more information, see [Section 8, "Troubleshooting Oracle Reports to BI Publisher Conversion Assistant."](#)

4.3 Output Files

The Conversion Assistant generates the following output files in your target directory for each converted report:

- Report definition (format: REPORT.xdo)
- Zipped report definition (format: REPORT.xdoz)
- Data model (format: REPORT.xdm)
- Zipped data model (format: REPORT.xdmz)
- RTF layout template if you select the layout type as either RTF Template or Both during the conversion (format: REPORT.rtf)
- XPT layout template if you select the layout type as either XPT Template or Both during the conversion (format: REPORT.xpt)
- Default PL/SQL package specification (format: REPORTS.pls)
- Default PL/SQL package body (format: REPORTB.pls)
- Log files (format: REPORT.log)

For example, assume that you want to convert a report called invoice.rdf located in the source directory D:\reports\pay, and you use the Conversion Assistant to enter the target directory D:\BIPublisher_reports\invoice.

The following output files are generated:

- Report definition: C:\BIPublisher_reports\invoice\invoice.xdo_report.xdo
 - Zipped report definition: C:\BIPublisher_reports\invoice\invoice.xdoz
 - RTF layout template: C:\BIPublisher_reports\invoice\invoice.xdo\invoice.rtf
- If you select the layout type as either XPT Template or Both during the conversion, the following XPT layout template is also generated: C:\BIPublisher_reports\invoice\invoice.xdo\invoice.xpt
- Data model: C:\BIPublisher_reports\invoice\invoice.xdm_datamodel.xdm
 - Zipped data model: C:\BIPublisher_reports\invoice\invoice.xdmz
 - PL/SQL package specification: C:\BIPublisher_reports\plsql_pkgs\invoice\invoiceS.pls
 - PL/SQL package body: C:\BIPublisher_reports\plsql_pkgs\invoice\invoiceB.pls
 - Report conversion log file: C:\BIPublisher_reports\bipconvert.log
 - Report upload log file: C:\BIPublisher_reports\bipupload.log

5 Compiling PL/SQL Packages on the Database

Many converted Oracle reports generate a PL/SQL package specification file and a PL/SQL package body file as follows:

- <report_name>S.pls
- <report_name>B.pls

Run the PL/SQL package files against your Oracle Database using the following commands to create the PL/SQL package specification and body.

```
SQL> @C:\BIPublisher_reports\invoice\invoiceS.pls
SQL> @C:\BIPublisher_reports\invoice\invoiceB.pls
```

6 Moving Converted Reports to the Oracle BI Publisher Repository

This step is only required if you did not direct the Conversion Assistant to automatically upload the converted reports to the Oracle BI Publisher Catalog. You can manually upload the converted report components using one of the following methods:

- [Method 1: Move the .xdo and .xdm folder files to the BI Publisher Repository Path in the File System](#)
- [Method 2: Upload the .xdoz and .xdmz zip files using the BI Publisher UI](#)

6.1 Method 1: Move the .xdo and .xdm folder files to the BI Publisher Repository Path in the File System

To use this method, you must have access to the server file system. Oracle recommends that you use this method only when you have a large number of converted reports.

After you move the .xdo and .xdm folder files to the BI Publisher repository path in the file system, the next step depends on your implementation:

If BI Publisher is deployed standalone (not part of Oracle BI Enterprise Edition), move the .xdo and .xdm folder files to the BI Publisher repository path in the file system. Ensure that you place each report definition (.xdo) and data model (.xdm) pair in the same folder. You do not need to restart the BI Publisher Server.

If BI Publisher is integrated with the BI Presentation Services Catalog, perform the following steps:

1. Move the .xdo and .xdm folder files to the BI Publisher repository path in the file system. Ensure that you place each report definition (.xdo) and data model (.xdm) pair in the same folder.
2. Navigate to the BI Publisher Server Administration page, and then select **Server Configuration**.
3. On the Server Configuration page, in the Catalog region, click **Upload to BI Presentation Catalog**. You are not required to restart the BI Publisher Server.

6.2 Method 2: Upload the .xdoz and .xdmz zip files using the BI Publisher UI

To use this method, you must have Write permissions on the BI Publisher Catalog folder to which you are uploading and you must assigned one of the following roles: BI Publisher Administrator, BI Publisher Template Designer, BI Publisher Developer, or BI Author.

1. Log in to the BI Publisher Server (for example, `http://www.<host>:<port>/xmlpserver`)
2. Select the folder in the BI Publisher Catalog to which you want to upload the report.
3. In the Tasks region, click **Upload**.
4. Click **Browse** to locate the report component (.xdoz or .xdmz) to upload. Repeat this step for each object, ensuring that you upload each report definition (.xdoz) and data model (.xdmz) pair to the same folder.

7 Testing and Editing Converted Reports

After you have successfully converted the Oracle reports, compiled the required PL/SQL packages, and moved the report definition and data model files into the BI Publisher Catalog, test the reports. Most converted reports run as expected without further modification of the data model; however, it is likely that the layout requires additional modifications to achieve the desired appearance. The following sections describe common issues with converted reports.

7.1 Summary Columns Moved to the Select Clause

Occasionally, when you convert a more complex Oracle report, the data model or PL/SQL contains minor errors and requires manual corrections. The Conversion Assistant moves all formula columns to the select clause of the SQL query in the data model. In most cases, this does not cause a problem. However, if any argument to the formula is a summary column, errors occur because the summary column is not calculated at the same time as the query is executed.

To correct this problem, remove the formula from the select clause and implement the formula in the data model. For more information on formula columns, see "*Structuring Data*" in *Oracle Fusion Middleware Data Modeling Guide for Oracle Business Intelligence Publisher*.

7.2 PL/SQL Format Trigger Logic Not Supported in Converted Layout Templates

Many Oracle reports use simple "if" formatting logic. The Conversion Assistant automatically converts this logic into equivalent XSL-FO and inserts the code into form fields in converted layout templates. However, there is no support for PL/SQL in converted layout templates. The Conversion Assistant does not convert any PL/SQL format trigger logic present in the report. Instead, the Conversion Assistant writes all the format trigger code to a log file. You must implement any corresponding PL/SQL logic as XSL code.

To aid in this process, the resulting template contains form fields that hold the format trigger names that are called; these fields are highlighted in red. You can then refer to the log file to find the actual PL/SQL code used in the original Oracle report.

You must rewrite the PL/SQL triggers using BI Publisher code syntax or XSL in the converted template form fields.

8 Troubleshooting Oracle Reports to BI Publisher Conversion Assistant

The Conversion Assistant produces two log files (bipconvert.log and bipupload.log) to assist you with troubleshooting the report conversion process. These log files capture exceptions at the statement level or in debug mode depending on which options you selected during report conversion.

After the you upload the reports to the BI Publisher Catalog, you can view the reports in the Report Viewer. If the report is not displayed as expected in the Report Viewer, review the errors captured on the screen or review the following BI Publisher log file: (MWHOME\user_projects\domains\bifoundation_domain\servers\AdminServer\logs\bipublisher\bipublisher.log).

Also, if you encounter problems with a converted report, test the data model as follows:

- If the data model is successful, verify the data in the XML output. If the data contains special characters, the report may fail while parsing the data.
- If the data model fails, check the data source connection. Next, verify that the .pls files were compiled successfully on the database to create the package. Finally, verify that the result data is accurate.

9 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.

Converting Oracle Reports to Oracle Business Intelligence Publisher, 11g Release 1 (11.1.1) Oracle Business Intelligence Publisher E28918-02

Copyright © June 2013 Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

