

Oracle® Transportation Management

Report Designer's Guide

Release 6.4.3

Part No. E92130-01

December 2017

Copyright Notice

Oracle® Transportation Management Report Designer's Guide, Release 6.4.3

Part No. E92130-01

Copyright © 2010, 2017, 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, then 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 fail-safe, 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 about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Contents

COPYRIGHT NOTICE	III
CONTENTS	IV
TABLE OF FIGURES	VI
TABLES	VI
SEND US YOUR COMMENTS	VII
PREFACE	VIII
CHANGE HISTORY	VIII
1. OVERVIEW	1-1
REPORT CONSUMER	1-1
REPORT DESIGNER	1-1
ACCESSING BI PUBLISHER	1-1
COMPLETE SUITE OF OBIEE.....	1-1
BI PUBLISHER COMPONENT	1-2
2. ARCHITECTURE	2-1
REPORT SYSTEM	2-1
REPORT CONTENT	2-1
REPORT GENERATION	2-2
IMPORTANT NOTES	2-2
3. CREATE A NEW REPORT IN BI PUBLISHER	3-1
REPORT COMPONENTS	3-1
WHAT YOU CAN CREATE OR EDIT	3-1
PROCESS OVERVIEW	3-2
DATA SOURCE	3-2
NOTES	3-3
REPORT FOLDERS	3-3
DATA MODELS	3-4
CREATING A DATA MODEL.....	3-4
EDITING A DATA MODEL.....	3-4
NOTES	3-4
UTILITY PACKAGES	3-5
PARAMETERS	3-9
DATE AND TIMESTAMP HANDLING	3-10
LAYOUTS	3-11
LAYOUT TEMPLATES	3-11
OVERALL PROCESS TO CREATE OR EDIT LAYOUTS	3-12
DELETING LAYOUT TEMPLATES.....	3-12
GENERATING SAMPLE REPORT DATA: PROCEDURE	3-12

LAYOUT EDITOR TEMPLATES	3-13
RTF AND EXCEL REPORT LAYOUT TEMPLATES.....	3-14
UPLOADING THE LAYOUT TEMPLATE FILE TO THE REPORT DEFINITION	3-15
CONFIGURING LAYOUT SETTINGS FOR REPORTS	3-15
4. USE BI PUBLISHER REPORT IN OTM.....	4-1
SETUP	4-1
REPORT SYSTEM.....	4-1
REPORT	4-1
DATA SECURITY	4-2
5. SAMPLE OTM REPORTS.....	5-1
6. EXPORT AND IMPORT REPORTS TO BI PUBLISHER REPOSITORY	6-1
CONFIGURE THE BI PUBLISHER CATALOG UTILITY	6-1
COMMAND LINE UTILITIES TO IMPORT AND EXPORT REPORT ARTIFACTS	6-2
EXPORT COMMAND	6-2
IMPORT COMMAND	6-2
OTHER OPTIONS TO MOVE THE REPORT ARTIFACTS	6-2
7. INTERMEDIATE FILE PERSISTENCE.....	7-1
8. PRINTING REPORTS	8-1
SET UP A PRINTER THAT MEETS ORACLE CLOUD REQUIREMENTS	8-1
CONFIGURE PRINTER IN BI PUBLISHER	8-1
PRINT FROM BI PUBLISHER	8-2
CONFIGURE PRINTER IN OTM	8-3
CONFIGURE THE PRINTERS	8-3
PRINT FROM OTM.....	8-3
OPTION 1	8-3
OPTION 2	8-3
ENABLE OTM PRINT LOGGING.....	8-4
VIEW LOGGING	8-4
DEBUGGING USING CUPS PRINT SERVER	8-5
IPP PRINTING FREQUENTLY ASKED QUESTIONS (FAQS)	8-5
9. BI PUBLISHER 10G MIGRATION	9-1
MIGRATING FROM STAND-ALONE BI PUBLISHER 10G	9-1
MIGRATING FROM EMBEDDED BI PUBLISHER 10G.....	9-1
MIGRATING FROM BI PUBLISHER 11G TO 12C	9-2
10. REPORT STORAGE AND CLEANUP	10-1
REPORT STORAGE	10-1
REPORT PERSISTENCE.....	10-1
GENERAL FILE CLEANUP	10-2
11. TROUBLESHOOTING	11-1

REPORT LOGGING	11-1
ORACLE TRANSPORTATION MANAGEMENT LOGGING	11-1
BI PUBLISHER LOGGING.....	11-1
FREQUENTLY ASKED QUESTIONS	11-1
HOW DO I CONFIGURE BI PUBLISHER REPORTS IN ORACLE TRANSPORTATION MANAGEMENT?.....	11-1
WHICH DATABASE USER IS USED FOR GENERATING REPORTS?	11-2
I AM UPGRADING FROM PRE 6.4 RELEASES TO 6.4.2. WHAT ARE THE STEPS I NEED TO FOLLOW?.....	11-2
CAN I USE THE DATA TEMPLATE AND FORMAT TEMPLATE I HAVE IN PRE 6.4 RELEASES FOR REPORTS IN 6.4.2?	11-3
CAN I USE PARAMETER OPERATORS IN 6.4.2?	11-3
CAN I USE CONFIGURE CONSOLIDATED REPORTS?	11-3
WHY DO I GET INVALID FORMAT REQUESTED ERROR WHEN TRYING TO VIEW SELECTED FORMAT OF THE REPORT?....	11-3
HOW DO I SEND MULTIPLE VALUES FOR A PARAMETER OF THE REPORT?	11-3
HOW DO I MOVE REPORT ARTIFACTS FROM ONE SERVER TO ANOTHER?.....	11-3
WHY DO I GET OUTOFMEMORYERROR: JAVA HEAP SPACE?	11-4
HOW DO I VIEW ALL OF THE SCHEDULED REPORTS?	11-4
SCHEDULED REPORTS ARE NOT BEING RECEIVED AS EMAILS, WHAT SHOULD I CHECK?	11-4
WHY ARE SCHEDULED REPORT RECEIVED AS A LINK AND NOT AS AN ATTACHMENT?	11-4
HOW DO I GENERATE REPORTS WITH A BAR CODE?.....	11-5
WHY DOES THE REPORT GIVE ME THE SAME OUTPUT FOR THE SAME SET OF PARAMETERS EVEN THOUGH I CHANGE THE USER?	11-5
HOW DO I DISABLE REPORT CACHING?	11-5
12. ADDITIONAL RESOURCES.....	12-1
13. APPENDIX: GENERAL PROPERTIES.....	13-1

Table of Figures

Figure 2-1: Reporting Architecture	2-1
--	-----

Tables

Table 3-1: Report Components	3-1
Table 3-2: Examples of Creating or Editing Reports	3-1
Table 3-3: PL/SQL Report Functions	3-5
Table 3-4: Standard Parameters	3-9
Table 3-5: Two Levels of Handling.....	3-15
Table 3-6: Layout Properties	3-16
Table 7-1: Intermediate File Persistence	7-1
Table 10-1: Report Lifespans.....	10-1
Table 13-1: General Properties	13-1

Send Us Your Comments

Oracle® Transportation Management Report Designer's Guide, Release 6.4.3

Part No. E92130-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: otm-doc_us@oracle.com

If you have problems with the software, contact Support at <https://support.oracle.com> or find the Support phone number for your region at <http://www.oracle.com/support/contact.html>.

Preface

This document is intended for Oracle Transportation Management clients, Oracle Transportation Management System administrators, or Oracle Transportation Management Consultants who have an interest in creating or configuring reports intended for use within the Oracle Transportation Management Application.

Change History

Date	Document Revision	Summary of Changes
12/2017	-01	Where appropriate, replaced the words "custom", "customizing", "customized" with one of the following "configuration", "configure", "edit", "update", "user-defined", or "customer-defined" per Fusion requirements. Added "Report Storage and Cleanup" from the online help.

1. Overview

Oracle Transportation Management's (OTM) report infrastructure uses Oracle Business Intelligence Publisher as a report generation component. It enables you to extract data from OTM/GTM, create a template to layout the data in a report, and generate the report to numerous formats. The OTM report infrastructure also enables you to schedule reports and deliver the reports to any delivery channel (email, printer, or store it in a content management system) required by your business. This guide is for report consumers and report designers.

Report Consumer

A report consumer may perform the following tasks

- Run and view reports from within OTM.
- Schedule reports to run at selected intervals and to be delivered via channels defined by your administrators.
- Link the report to a document definition in OTM.
- Configure the report to run on occurrence of an event using agent action.

Report Designer

A report consists of a data model, a layout, a set of properties, and a link to the report in OTM. A report designer may perform the following tasks:

- Create the report data model, including parameters and a list of values, using BI Publisher's report editor.
- Design the layout template of the report. The layout can be created using different tools depending on your output requirements, including Microsoft Word, Adobe Acrobat, Microsoft Excel, and Adobe Flex Builder.
- Set up runtime configuration properties for the report.
- Establish a link between OTM reports to BI Publisher's report.

It is strongly recommended that you refer to [BI Publisher Installation](#), [BI Publisher Administration Guide](#), [BI Publisher Report Designer's Guide](#), [BI Publisher User's Guide](#), the [OTM Administration Guide](#), and the [OTM Installation Guide](#) before you proceed with the below guide.

Accessing BI Publisher

BI Publisher can be made available in a few ways.

Complete Suite of OBIEE

Install a complete suite of OBIEE, BI Publisher is available as one of the component. In this case, BI Publisher can be accessed by using the following URL.

```
http(s)://<host>:<port>/analytics
```

In this case the catalog used by OBIEE is used as the BI Publisher repository. All of the administration activities (i.e. changing roles and permission, creating data sources, configuring printers, etc.) are performed using 'Manage BI Publisher' link in the administration section of OBIEE.

BI Publisher Component

Install just the BI Publisher component. In this case BI Publisher can be accessed using the following URL. This same URL can be used even in case of complete OBIEE install too.

```
http(s)://<host>:<port>/xmlpserver
```

It is recommended to perform the administration tasks, data modeling and layout editing using the analytics URL rather than using the xmlpserver URL in case of complete OBIEE installation.

2. Architecture

Oracle Transportation Management supports generation of reports using an external reporting system such as BI Publisher. Report requests are sent via HTTP to an external report server. The report server may begin an interactive session or simply return the report content.

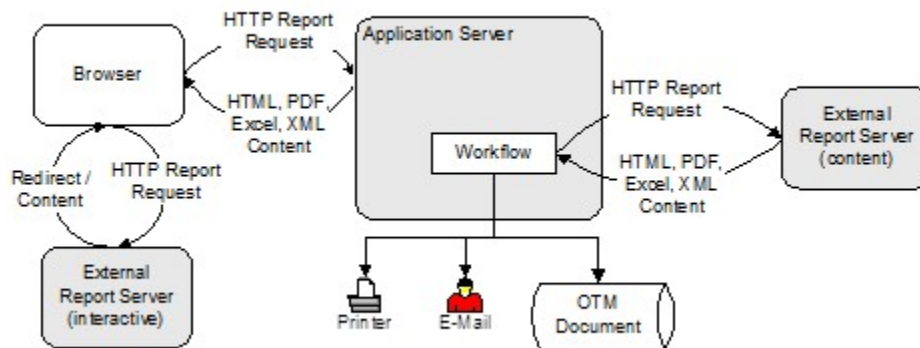


Figure 2-1: Reporting Architecture

Report System

Reports in OTM are always associated with a report system type that defines the communication type between OTM and the external report server. The following report systems are supported in Oracle Transportation Management:

- **BI Publisher:** A SOAP request is sent to the BI Publisher service and the SOAP response is parsed for content and errors. It includes details of the BI Publisher Server, such as report path prefix, host, port, and user credentials. These account for format, locale, and VPD user information.
- **Other via HTTP:** Use Other via HTTP for non-BI Publisher integration. It is assumed that the report path is a URL to the report with an optional {report} parameter that is replaced with the report path. This URL report parameter is appended as HTTP request parameters. Each request parameter name matches the name of the report parameter. This includes setting **&P_DESIRED_MIME_TYPE=<requested MIME type>**.

It is strongly recommended that you use BI Publisher as the report system type. This document describes the information pertaining to the BI Publisher report system type.

Report Content

The report content returned by the external report server can be used in different ways within OTM. You have to specify the report content type for each report in OTM. The following types of report content are supported in OTM.

- **Embedded:** Embedded content means that report content returned by the external report server can be distributed to a browser, email, printer, or stored in content management systems without needing links or nested frames. You may, however, get back an HTML page that references other URLs from the report server. If a third party generator returns embedded content, it can be used as a replacement for BI Publisher.

- **Browser Only:** Browser-only content means that report content returned by the external report server will be piped to a browser. Selecting this option lets you optionally include links or fields applicable to report content. OTM does not support distribution or storing of content for this option.
- **None:** This means that report content is not generated and returned by external report server, and an alternative URL link is used to generate the report on the report manager. These reports cannot be transactional as they do not accept business object information from OTM. The standard Run/Schedule links to the report are not supported, nor are the reports included on any action menu. Instead, there is a single Report link in the Third Party Reports section of the Report Manager that directs the browser directly to the third party URL. OTM does not add any implicit or explicit query parameters to the URL.

Report Generation

Report generation in Oracle Transportation Management can triggered as follows:

- **Ad Hoc Reports:** For ad hoc reports, an HTTP request is sent to the application server which sends an HTTP request to the report server and assumes the response, if successful, contains report content. This content is piped back through HTTP to the browser. The content may also be distributed via email, IPP printing, and/or stored with an associated business object.
- **Scheduled Reports:** Scheduled report requests, or requests triggered by a workflow agent, similarly send a HTTP request to the report server. The response content is distributed via email, printed via IPP, or stored with an associated business object.

The scheduling of the reports has nothing to do with the scheduling capabilities of the external report server. The scheduling is OTM's own scheduling which is used for forming the report requests and receiving the response.

Streamlined support is available when generating reports on a remote BI Publisher server or a server farm. Requests are sent directly from the application server to BI publisher via a SOAP web service call. Results or errors are returned by the service.

Important Notes

- All the report content generation only happens on the external reporting server. The support for embedded report content generation within OTM using query and format template has been removed.
- OTM no longer supports the use of the "Use Parameters Operator" on the Report manager. This means reports parameters do not have operators like "Begins With", "Same As", or "Ends With". Instead it is free-form text that you enter.
- Consolidated reports are not supported.
- It is strongly recommended to use the OTM reporting capability only for transactional reports¹. All analytical reports should be generated directly on the BI Publisher server and should not be triggered from OTM.
- By default, all reports are executed with "Use Report Parameter as Bind Variable" as Yes. This change in behavior was done to prevent performance and security issues. Additionally, the parameter operators give the query wild cards, make the query to hard parse, and may not go through indexes properly.

¹ A transactional report is one that is focused on a single business object. A shipment document, such as a Bill of Lading, is an example of a transactional report.

3. Create a New Report in BI Publisher

Use reports to generate and print documents for internal operations, external business transactions, or legal requirements. To meet specific requirements, you may need to create or edit reports to capture different data, or present data in another way. This section provides general guidelines to design new reports and use them in Oracle Transportation Management. Refer to [BI Publisher Report Designer's Guide](#) for more options and details.

Report Components

Each report has components that you can configure, as described in this table:

Table 3-1: Report Components

Report Component	Description	Tool for Configuring
Data model	Defines the data source, data structure, and parameters for the report. Multiple reports can use the same data model. Each report has one data model.	Data model editor in the application.
Layout	Defines the presentation, formatting, and visualizations of the data. A report can have multiple layouts. There are different types of layout templates, for example Excel and RTF.	Depending on the template file type: <ul style="list-style-type: none">• XPT: Layout editor in the application• RTF: Microsoft Word• PDF: Adobe Acrobat Professional• Excel: Microsoft Excel• eText: Microsoft Word
Properties	Specifies formatting and other settings for the report.	Report editor in the application

What You Can Create or Edit

This table gives just a few examples of creating or editing reports.

Table 3-2: Examples of Creating or Editing Reports

Task	Example
Edit the layout of a report.	Add your company logo to the report output.
Add a new layout to a report.	Design a new layout template that provides less detail than the existing template.
Edit a data model.	Add two fields to the data model used by a report so you can add those new fields to a customer-defined layout for the report.

Task	Example
Create a new report based on a new data model.	Create a new data model based on data from an external system, and create reports using the customer-defined data model.

Process Overview

Note: You must be assigned the BI Publisher Developer or BI Publisher Administrator role to create or edit the reports.

Creating a new report consists of the following steps:

1. Create the report data source. Your report data would come from the OTM operational database or archive database.
2. Create the report entry in the desired folder on the **Reports** page.
3. Open the **Report Editor**.
4. Specify the general properties for the report.
5. Define the **Data Model**. Your report data may come from a SQL query, an HTTP feed, a Web service, an Oracle BI Answers request, or a file.
6. Define the parameters that you want users to pass to the query and define lists of values for users to select parameter values.
7. Test your data model.
8. Design the layout template.
9. If you are designing an RTF template, load the data to the Template Builder for Word. Use the Template Builder in conjunction with the instructions in [Creating an RTF Templates](#) to build your report layout.
10. If you are designing a PDF template, follow the instructions in [Creating a PDF Templates](#) to build your report layout.
11. If you are using a predesigned PDF form (such as a government form), follow the instructions in [Creating a Layout from a Predefined PDF Form](#).
12. Upload your templates to the **Report Editor**.
13. (Optional) Add translations for your reports. See [Translating Reports](#).

Data Source

On-premise customers will need to configure Data Sources to connect to the corresponding Database. Follow the instruction at [Setting Up Data Sources](#). A typical setup uses the JDBC connection.

You can create a new data source to read the data from OTM. It is recommended that you create two data source as follows:

- OTMOLTP: To connect to OTM OLTP Schema
Data Source Name: **otmoltp**
Driver Type: Oracle **11g**
Database Driver Class: **oracle.jdbc.OracleDriver**
Connection String: **<otm oltp database connection string>**

Username: **globalreportuser**
Password: <**password for globalreportuser**>

- OTMARCHIVE: To connect to OTM Archive Schema
Data Source Name: **otmarchive**
Driver Type: Oracle **11g**
Database Driver Class: **oracle.jdbc.OracleDriver**
Connection String: <**otm archive database connection string**>
Username: **archive schema user**
Password: <**password for archive schema**>

Notes

It is highly recommend for security reasons to use the "globalreportuser" database user, provided by default, for the "otmoltp" data source. That user has the minimal rights needed to create and run reports.

- The above data source connections are defined for viewing standard reports and for designing any new reports using the globalreportuser and archive databases schema users.
- When the OTM connects with BI Publisher, it does not send the same DB credentials that are defined above.
- For Online reports the OTM application sends the database credentials defined by the PRIMARY_THIN data source of OTM application.
- For Offline reports the OTM application sends the database credentials defined by the ODS data source of OTM application.
- For Archive Reports the OTM application sends the database credentials defined by the ARCHIVE data source of OTM application.
- Since the users with which the report is designed and accessed may be different, any customer-defined tables need to be provided with necessary grants. For example, if you are designing an online report that involves any customer-defined table or view (other than the table or view provided by OTM), then you will have to provide grants for the table or view for both globalreportuser and the schema defined by the PRIMARY_THIN data source.
- You will need to provide grants for customer-defined packages if they are used in any of the queries.
- Reports run from BI Publisher server will not have any VPD context applied.
- Reports run from OTM run with the VPD Context of the user running the report.

Report Folders

All sample reports shipped with OTM are created in the Reports folder under Shared Folders. It is highly recommended to create a folder named "custom" under Shared Folders and create all new reports and their corresponding data models in a separate folder for each report (like Invoice Summary) under "custom". This will enable you to move the report artifacts across environments, such as Stage to Prod. For Cloud customers, the "custom" folder is mandatory in order to ensure their customer Reports are preserved during upgrades.

For more information on creating the folders please refer to the section [Creating a Folder or Subfolder](#) in the chapter [Managing Reports](#) in [Fusion Middleware User's Guide for Oracle Business Intelligence Publisher](#).

Data Models

A data model defines where data for a report comes from and how that data is retrieved and how the data is structured. If the existing data model cannot give you all the data that you need in your report, then you can either copy and edit an existing data model or create a new one.

Creating a Data Model

1. In the business intelligence (BI) catalog, click the New button and select Data Model under Published Reporting.
2. Optionally click the Data Model node in the Data Model pane to set properties for the data model.
3. Click the Data Set node in the Data Model pane to create or edit data sets, which determine where and how to retrieve data.
4. Click the New Data Set button and select a data set type. Select one of the data source created otmoltp or otmarchive. Most of the time you will be creating the data set using SQL Query which uses a Query Builder tool to define from where the data is to come from how to fetch the data.
5. Optionally, to limit the data included in the report output, click the Parameters node in the Data Model pane to define variables that users can set when they use the report.
6. Optionally, define other components of the data model.
7. Save your data model.

Editing a Data Model

1. To edit a predefined data model:
 - a. Find the data model in the BI catalog and click Copy.
 - b. Paste within Shared Folders > Custom in a subfolder that has a folder path similar to the folder that stores the original data model.
 - c. For the data model you pasted, click More, and select Edit.
2. Optionally click the Data Model node in the Data Model pane to set properties for the data model.
3. Click the Data Set node in the Data Model pane to create or edit data sets.
4. Perform steps 5 and 6 from the preceding Creating a Data Model task, as needed.

For more information on the data modeling, refer to the [Oracle® Fusion Middleware Data Modeling Guide for Oracle Business Intelligence Publisher](#).

Notes

Below are some of points to remember about the data model components while creating the data model.

- Event triggers will not be useful for the Logistics cloud as the customers will not have the access to create or edit PL/SQL packages, procedures.
- Lists of values will only be useful if the customers are trying to run the reports from the BI Publisher. The LOVs cannot be used from OTM application.
- Bursting definitions are not used by OTM for the distribution of reports to different destinations. OTM uses its own infrastructure to distribute the reports to email and printers.
- Custom Metadata is not supported for reports integrated to OTM.
- Though we have so many options for creating the Data Sets, OTM supports only data sets created using SQL Query.

Utility Packages

OTM provides a number of PL/SQL functions/procedures that can be used in BI Publisher Data Models. **Error! Reference source not found.** summarizes the available procedures and functions. For more information, please review the `create_rpt` scripts in `OTM_INSTALL_DIR>\glog\oracle\script8`.

Table 3-3: PL/SQL Report Functions

Package	Procedure/Function	Description	Parameters	Returns
vpd	set_user	Sets the user for VPD data security	Oracle Transportation Management user GID	--
	set_user_r	Sets the user and user role for VPD data security	Oracle Transportation Management user GID Oracle Transportation Management user role GID	--
	set_user_fct	Sets the user for VPD data security. Designed for use as the first query in a Data Template	Oracle Transportation Management user GID	true
	set_user_r_fct	Sets the user and user role for VPD data security. Designed for use as the first query in a Data Template	Oracle Transportation Management user GID Oracle Transportation Management user role GID	true
	get_gl_user	Returns the current user for VPD data security	--	Oracle Transportation Management user GID

Package	Procedure/Function	Description	Parameters	Returns
	fnc_set_user_r	Sets the user and user role for VPD data security	Oracle Transportation Management user GID Oracle Transportation Management user role GID	true
utc	get_local_date	Converts a UTC timestamp to a location time zone	UTC Timestamp Location GID	Timestamp in the location's time zone
	get_utc_date	Converts a local timestamp to a UTC timestamp based on a location time zone	Local Timestamp Location GID	UTC Timestamp
	get_time_zone	Returns the time zone for a location	Location GID	Time Zone
rpt_general	p_insert_log	Logs a record to REPORT_LOG	Unique Filename Report GID Report Job # User Domain up to 3 (name, value) pairs	--
	f_uom_base	Returns the default storage type for a unit of measure	Unit of measure	Default storage type
	f_date_diff	Returns a readable string representing the duration difference of two dates	Ending date Starting date	Duration string
	f_remove_domain	Strips the domain from a GID	GID	XID
	f_format_address	Returns a formatted address for a location	Location GID	Formatted, multi-line address for the location

Package	Procedure/Function	Description	Parameters	Returns
	f_corporation	Returns the corporation for a location	Location GID	Corporation GID
	f_location_refnum	Returns the value of a specific location reference number	Location GID Reference number qualifier	Reference number value
rpt_order	f_ob_refnum	Returns the value of a specific order base reference number	Order Base GID Reference number qualifier	Reference number value
rpt_ship	f_commodity	Returns the commodity name for a specific item	Item GID	Commodity name
	f_ob_party_location	Returns the location for a specific order base involved party	Order Base GID Involved party qualifier	Location GID
	f_or_party_location	Returns the location for a specific order release involved party	Order Release GID Involved party qualifier	Location GID
	f_order_base_gid	Returns the order base associated with an order release	Order Release GID	Order Base GID
	f_party_address	Returns a formatted address for an order release involved party	Order Release GID Involved party qualifier	Formatted, multi-line address for the location
	f_tender_accepted_by	Returns the carrier that accepted a shipment tender	Shipment GID	Service Provider GID
	f_packaging_form_code	Returns the packaging form code for a ship unit specification	Ship Unit Specification GID	Packaging Form Code GID
	f_capacity_rate_offering	Returns the rate offering for a capacity usage	Capacity Usage GID	Rate Offering GID

Package	Procedure/Function	Description	Parameters	Returns
	f_capacity_time_period	Returns the time period type for a capacity limit	Capacity Limit GID	Time Period Type
	f_lane_source	Returns the source for a lane	XLane GID	Source
	f_lane_destination	Returns the destination for a lane	XLane GID	Destination
	f_equipment_type_name	Returns the name for a equipment type	Equipment Type GID	Equipment Type Name
	f_sellside_cost	Returns the sell-side cost for a shipment	Shipment GID	Sell-side cost
	f_transport_mode_name	Returns the transport mode for a shipment	Shipment GID	Transport Mode
	f_get_ship_inv_party_address	Returns a formatted address for a shipment involved party	Shipment GID Involved Party Qualifier	Formatted, multi-line address for the involved party
	f_get_country_name	Returns the origin or destination country name for a shipment	Shipment GID 'O' for origin, 'D' for destination	Country name
	f_get_pol	Returns the port of lading for a shipment	Shipment GID	Port of lading
rpt_invoice	f_party_location	Returns an involved party location on the invoice	Invoice GID Involved Party Qualifier	Involved party location
	f_party_address	Returns a formatted address for an invoice involved party	Invoice GID Involved Party Qualifier	Formatted, multi-line address for the involved party
rpt_servprov	f_servprov_gid	Returns the Service Provider GID for a particular alias	Service Provider Alias Service Provider Alias Qualifier	Service Provider GID

Package	Procedure/Function	Description	Parameters	Returns
	f_alias	Returns the Service Provider alias	Service Provider GID Service Provider Alias Qualifier	Service Provider Alias

Parameters

Adding parameters to your data model enables users to interact with data when they submit or view reports. OTM passes standard and ad hoc parameters to the report generator². **Error! Reference source not found.** summarizes the standard parameters passed to every report.

Table 3-4: Standard Parameters

Name	Description	Comments
P_REPORT_GID	The requested report	External systems are responsible for mapping the Oracle Transportation Management report GID to a valid report. Alternatively, the report URL can embed the mapping within a request parameter.
P_DBCONN_TYPE	The data schema	OLTP for the Oracle Transportation Management transactional database; ODS for the offline analytical database; ARCHIVE for the archived data.
P_GL_USER	The user requesting the report	This may be used by external systems to enforce VPD data security on report queries.
P_ROLE_ID	The role of the requesting user	This may be used by external systems to enforce VPD data security on report queries.
P_DOMAIN	The domain of the requesting user	
P_LANGUAGE	The ISO language code requested for the report	For embedded reports, XLIFF translations are automatically applied.
P_COUNTRY	The ISO country code requested for the report	For embedded reports, XLIFF translations are automatically applied.
P_DISPLAY_NAME	A user-readable name for the report.	

² If external report generators do not support these parameters, their content type should be set to **None**.

Name	Description	Comments
P_DATE_FORMAT	The Oracle date format for both input parameters and output fields.	<p>For input parameters, use the TO_DATE(:P_MY_DATE, :P_DATE_FORMAT) function.</p> <p>For output parameters, use the TO_CHAR(field, :P_DATE_FORMAT) function.</p> <p>This should be reserved for fields that are date-only.</p>
P_DATE_TIME_FORMAT	The Oracle timestamp format for both input parameters and output fields.	<p>For input parameters, use the TO_DATE(:P_MY_TIMESTAMP, :P_DATE_TIME_FORMAT) function.</p> <p>For output fields, use the TO_CHAR(field, :P_DATE_TIME_FORMAT) function.</p> <p>This should be reserved for fields that have both date and time.</p>

The above defined list of standard parameters can be used in any of the report you are designing. If you define any of these standard parameters as a parameter for the report, you do not have to define it as a parameter when defining the report in OTM.

You will have to define any other parameter required for the report apart from the standard parameters as a report parameter when defining the report definition in Oracle Transportation Management.

For sending multiple values for a parameter of the data model, please refer to FAQ section [How do I send multiple values for a parameter of the reports?](#)

Date and Timestamp Handling

Report designers should take special care when writing queries involving DATE columns. This includes:

- Converting date parameters for where clause comparison. All date parameters are sent as strings, formatted according to your date preferences. To compare one to a data field, the designer should use Oracle's TO_DATE function, applying the standard P_DATE_FORMAT parameter:

```
where accessorial_cost.effective_date > TO_DATE(:P_EARLIEST_DATE,
:P_DATE_FORMAT)
    and accessorial_cost.effective_date < TO_DATE(:P_LATEST_DATE,
:P_DATE_FORMAT)
```

- Converting timestamp parameters for where clause comparison. Like dates, timestamp parameters are sent as strings, formatted according to your date/time preferences³. To compare one to a timestamp field, the designer should use Oracle's TO_DATE function, applying the standard P_DATE_TIME_FORMAT parameter:

```
where shipment.start_time > TO_DATE(:P_EARLIEST_START, :P_DATE_TIME_FORMAT)
      and shipment.start_time < TO_DATE(:P_EARLIEST_START,
      :P_DATE_TIME_FORMAT)
```

- Applying date and time preferences. Depending on the use case, designers may want to apply user preferences to dates displayed on the final report. If so, the SQL query should convert selected date and timestamp values using Oracle's TO_CHAR function:

```
select TO_CHAR(accessorial_cost.effective_date, :P_DATE_FORMAT) ...
select TO_CHAR(shipment.start_time, :P_DATE_TIME_FORMAT) ...
```

- Accounting for UTC storage. Nearly all timestamp fields in Oracle Transportation Management are converted to UTC before persisting to the database. A report designer who simply queries shipment.start_time, for example, receives the time in UTC. To convert the stored time to the application or report server's time zone, use the vpd.gmt_offset function:

```
select TO_CHAR(shipment.start_time-(vpd.gmt_offset/24), :P_DATE_TIME_FORMAT)
...
```

- Accounting for UTC storage. To compare timestamp fields to some offset of current time, designers can either convert the field or use vpd.gmt_sysdate. This function returns the current time in UTC. E.g. to query shipments starting in the next three days:

```
select shipment.gid
where shipment.start_time > vpd.gmt_sysdate
and shipment.start_time < vpd.gmt_sysdate+3
```

Layouts

The layout determines what and how data is displayed on report output. Each report has at least one layout template. This topic describes the following aspects of report templates:

- Layout templates
- Layout template types
- Overall process of managing layouts
- Deleting layout templates

Layout Templates

To configure a layout, you edit the layout template, which:

- Defines the presentation components, such as tables and labeled fields.
- Maps columns from the data model to these components so that the data is displayed in the right place.
- Defines font sizes, styles, borders, shading, and other formatting, including images such as a company logo.

Layout Template Types

There are a few types of template files to support different report layout requirements.

³ Note that the reporting has never applied user time preference. Time preference is assumed to be HH24:MI:SS.

- **RTF:** Rich text format (RTF) templates created using Microsoft Word.
- **XPT:** Created using the application's layout editor, these templates are for interactive and more visually appealing layouts.
- **eText:** These templates are specifically for Electronic Data Interchange (EDI) and electronic funds transfer (EFT) information.

You can also create and edit other types of templates using Adobe PDF, Microsoft Excel, Adobe Flash, and XSL-FO.

Overall Process to Create or Edit Layouts

Editing or creating report layout, for example using Microsoft Word or the layout editor, involves making the actual changes to the template file. But that task is just one part of the entire process for configuring layouts.

1. Copy the original report and save the customer-defined version in Shared Folders - custom in the business intelligence (BI) catalog. You create or edit templates for the customer-defined copy of the report.
Tip: You can use the Customize option if the original is a predefined report.
2. Review report settings for online viewing.
3. Generate sample data for the report.
4. Edit or create the layout template file.
5. Upload the template file to the report definition. Skip this step if you're using the layout editor.
6. Configure the layout settings.

Deleting Layout Templates

To remove a layout template for a report:

1. Select your report in the BI catalog and click Edit.
2. In the report editor, click View a list.
3. Select the layout template and click Delete.

Generating Sample Report Data: Procedure

Depending on the type of report layout changes you're making, sample data can be required or optional. You generate sample data, and then load it for use with your layout so that you can map data fields to layout components. For example, for the Start Date table column in your layout, you can set it so that the data displayed in that column comes from the Start Date field in the sample data.

You can generate sample data from the:

- Report data model
- Report viewer
- Scheduler

Generating Sample Data from the Data Model

Follow these steps:

1. Select your data model in the business intelligence (BI) catalog and click Edit. Alternatively:
 - a. In the catalog, find the report to generate sample data for and click Edit.
 - b. Click the data model name in the report editor.
2. In the data model editor, click View Data.

3. Enter values for any required parameters, select the number of rows to return, and click View.
4. To save the sample data to the data model, click Save As Sample Data.
5. If you're designing a .rtf template, click Export to save the file locally.
6. Save the data model.

Saving Sample Data from the Report Viewer

For reports that are enabled for online viewing, you can save sample data from the report viewer:

1. Select the report in the BI catalog.
2. Click Open to run the report in the report viewer with the default parameters.
3. On the Actions menu, click Export, then click Data.
4. Save the data file.

Saving Sample Data from the Scheduler

For reports that are enabled for scheduling (not necessarily as a scheduled process), you can save sample data from the scheduler:

1. Select the report in the BI catalog.
2. Click Schedule.
3. On the General tab, enter values for any report parameters.
4. On the Output tab, ensure that Save Data for Republishing is selected.
5. Click Submit.
6. Open the Report Job History page.
7. On the global header, click Open, then click Report Job History.
8. Select your report job name in the Job Histories table.
9. On the details page, under Output and Delivery, click the XML Data Download icon button.

Layout Editor Templates

The layout editor in the application provides an intuitive, drag-and-drop interface for creating pixel-perfect reports with PDF, RTF, Excel, PowerPoint, and HTML output. The layout template files you create with this tool have an .xpt extension. The layout editor tool is the only editing tool that provides dynamic HTML output. Users can interact with this output in a browser, for example by sorting, applying filters, and so on.

Prerequisite

Make sure that sample data is generated from the data model that your report is using.

Using the Layout Editor

To configure XPT templates:

1. Select the report in the business intelligence (BI) catalog and click Edit.
2. In the report editor, click Edit to update a template.
Or, click Add New Layout and select a template type under the Create Layout section.
3. Create or edit the layout.
4. Click Save to save the layout to the report definition.

RTF and Excel Report Layout Templates

You can use Microsoft Word or Microsoft Excel to create or edit RTF and Excel layout templates, in addition to the layout editor in the application. If you use Word or Excel directly, you must download and install the appropriate add-in so that the Microsoft application has the features you need to design report layouts.

Note: If you're designing a new layout for your report, consider using the layout editor instead unless you are an experienced layout designer.

Installing the Add-In

1. Open the Reports and Analytics work area, or the Reports and Analytics pane if available in other work areas.
2. Click the Browse Catalog button.
3. Click Home.
4. Under the Get Started pane, click Download BI Desktop Tools.
5. Select the add-in for the type of template you're working with.
 - Template Builder for Word: RTF templates
 - Analyzer for Excel: Excel templates
6. Save and then run the installer.

Creating and Editing RTF Report Layout Templates: Procedure

An RTF template is a rich text format file that contains the layout instructions to use when generating the report output. Use Microsoft Word with the Template Builder for Word add-in to design RTF templates.

Prerequisites

Install the Template Builder for Word add-in, and generate sample data.

Using Template Builder for Word

To configure an RTF template:

1. If you are editing an existing layout:
 - a. Select your report in the business intelligence catalog and click Edit.
 - b. In the report editor, click the Edit link of the layout to download the RTF file. If you are creating a new layout, skip this step.
2. Open the downloaded RTF template file in Microsoft Word. Or, if you're creating a new template, just open Microsoft Word.
3. Load the sample data that you generated.
4. Edit or create the layout template.
5. Save the file as Rich Text Format (RTF).

eText Report Layout Templates: Explained

An eText template is an RTF-based report template that is used for Electronic Funds Transfer (EFT) and Electronic Data Interchange (EDI). The template is applied to an input XML data file to create a flat text file that you transmit to a bank or other organizations. Use Microsoft Word to create or edit eText templates.

File Format

Because the output is for electronic communication, not printing, you must follow specific format instructions for exact placement of data on the template. You design eText templates using tables.

- Each record is represented by a table.
- Each row in a table corresponds to a field in a record.
- The columns of the table specify the position, length, and value of the field.

Special Commands

You must set up special handling of the data from the input XML file. This table describes the two levels of handling and where you declare the corresponding commands.

Table 3-5: Two Levels of Handling

Level	Example	Setup
Global	Character replacement Sequencing	Declare global commands in separate setup tables.
Record	Sorting	Declare functions in command rows, in the same table as the data.

Uploading the Layout Template File to the Report Definition

If you're creating or editing a report layout using the layout editor, the layout is automatically saved to the report definition, so you can skip this step. For all other layout types, for example RTF, upload the template file to the report definition after you're done making layout changes.

Uploading the Template File

1. Select your report in the business intelligence catalog and click Edit.
2. In the report editor, click View a list.
3. In the table that lists the layouts, click Create.
4. Under Upload or Generate Layout, click Upload.
5. In the Upload Template File dialog box:
 - a. Enter a layout name.
 - b. Browse for and select the layout template file that you created or edited.
 - c. Select the template file type.
 - d. Select the locale, which you can't change once the template file is saved to the report definition.
 - e. Click Upload.
6. Save the report definition.

Configuring Layout Settings for Reports

As part of creating or editing layout, you can set report properties related to layout. These settings determine, for example, which layouts users can choose from when viewing or scheduling the report. The settings apply only to your report.

Setting Layout Properties

1. Select your report in the business intelligence catalog and click Edit.
2. In the report editor, click View a list.
3. Set layout properties, some of which are described in this table.

Table 3-6: Layout Properties

Setting	Usage
Output Formats	Depending on the requirements for a report, you may want to limit the output file formats (for example, PDF or HTML) that users can choose. The available output formats vary depending on the template file type.
Default Format	When multiple output formats are available for the report, the default output format is generated when users open the report in the report viewer.
Default Layout	When multiple layouts are available for the report, you must select a default layout to present it first in the report viewer.
Active	Active layouts are available for users to choose from when they view or schedule the report.
View Online	Select this check box so that layouts are available to users when they view the report. Otherwise, the layout is available only for scheduling the report.

4. Click Save Report.

For more information on designing the Layouts refer to [Oracle® Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher](#).

4. Use BI Publisher Report in OTM

This section provides general guidelines to use the newly created report in Oracle Transportation Management. You would need to refer to Oracle Transportation Management online help topics about Reports for more options and details.

Note: The Report System is automatically configured for Cloud customers, so this section is not relevant to Cloud customers.

Setup

Note: You must be assigned the OTM Administrator role to use the newly created report in Oracle Transportation Management.

Using a new report consists of the following steps:

1. Create the report systems that represent the connection to BI Publisher Server.
2. Create the report, by specifying report details like content type, report system, report URL, default format type, report parameters, and other details.
3. Test your Oracle Transportation Management Report.

Report System

The report system is used to define the BI Publisher host, port, user, password, and report path prefix common to all reports sent to that host. Oracle Transportation Management provides a report system named "DEFAULT". This report system uses property macros to substitute property values at run time. These properties facilitate easy move from Stage to Prod BI Publisher instances. The following properties should be configured:

- glog.bip.externalFarm.host=
- glog.bip.externalFarm.port=9704
- glog.bip.externalFarm.user=bipreportuser
- glog.bip.externalFarm.password=
- glog.bip.externalFarm.reportPath=

Report

The report is used to define the link to BI Publisher report from Oracle Transportation Management by specifying the report path. Additional details that are specified include report system, content type, display format type, report parameters, and other details. You can use "Select via UI" to allow the report to explicitly run as an OTM user.

Report Parameters

Oracle Transportation Management passes standard and ad hoc parameters to the report generator⁴. For more details on the standard parameters please refer to the **Parameters** section under the Data Models section.

The report designer may add additional, ad-hoc parameters to a report using the Report Parameters grid of the Report Manager. Before submitting a report request, you are presented with a list of these additional parameters and must enter information for any marked as **mandatory**.⁵

⁴ If external report generators do not support these parameters, their content type should be set to **None**.

For external reports other than BI Publisher, report parameters are added as HTTP request parameters to the URL.

Data Security

Oracle Transportation Management implements data security via Oracle's Virtual Private Database (VPD). Given a user and their role, the system limits row access to tables. The default VPD policies implement a domain model, where a user's domain determines their read/write privileges. Specific implementations, however, can enhance this model to restrict access based on other columns and grant access across domains.

When running BI Publisher reports from within OTM, VPD security is automatically enforced by the system. The database credentials passed to the BI Publisher web services has the user and role in context, applying VPD policies on any queries that are executed for the report.

External report generators other than BI Publisher, however, have two options regarding data security:

- Suppress it. The standard `glogdba` database user requires a VPD context. If a report logs in as `glogowner`, however, VPD is suppressed. The report queries have access to all data. This may be appropriate for transactional reports where the data is restricted to a particular business object. Alternatively, each report can implement its own data security model independent of Oracle Transportation Management.
- Set the user context before issuing any queries. The `GLOGOWNER.VPD` package provides the following procedures and functions to set the context for VPD:

```
procedure set_user (user VARCHAR2);  
procedure set_user_r (user VARCHAR2, user_role varchar2);  
  
function set_user_fct (user VARCHAR2);  
function set_user_r_fct (user VARCHAR2, user_role varchar2);
```

By passing the standard `P_GL_USER` (and optionally `P_ROLE_ID`) parameter to one of these functions, VPD returns the proper rows.

⁵ Note that transactional reports have at least one mandatory parameter matching the business object type. If requested from a transactional manager, the system automatically populates this parameter. E.g., a shipment document with one mandatory `P_SHIPMENT_ID` parameter with a **Query Name** of `BUY_SHIPMENT` can be selected from the shipment manager. The `P_SHIPMENT_ID` parameter is automatically populated with the relevant shipment GID.

5. Sample OTM Reports

Some sample reports are included as part of the OTM software installation. Since Oracle Transportation Management does not automatically install BI Publisher, loading the sample reports into the BI Publisher repository is a post-installation process. The sample reports are located in the OTM installation in the following directory, where <otm_install> is the directory where OTM is being installed:

```
<otm_install>/otm/
```

The BI Publisher catalog utility enables administrators and report developers to import Reports into the BI Publisher catalog. For more details on this subject, please refer to the “Moving Catalog Objects between Environments” section of the Oracle Fusion Middleware Administrator’s Guide for Oracle Business Intelligence Publisher. The following section gives sample instructions used to do the import the reports.

If the Oracle Transportation Management app server and BI Publisher report server are on two different machines, make sure to copy the report artifacts from the app server to a folder on the BI Publisher report server, e.g. temp_otm_reports.

6. Export and Import Reports to BI Publisher Repository

This chapter provides general guidelines on how to move report between environments. One can use the same guidelines to import the sample reports. There are two major artifacts for any report once completely designed.

- Report.xdo files with corresponding configuration files
- Report.xdm files with corresponding configuration files

More detailed explanation of the report artifacts is given in section "[What Files Are Moved](#)" in the chapter [Moving Catalog Objects between Environments](#) in [Oracle® Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher](#).

Configure the BI Publisher Catalog Utility

On-Premise customers have the option to use a command line utility for importing/exporting report artifacts. Cloud customers should refer to the "[Other Options to Move Report Artifacts](#)" section. The below steps are provided only for the reference purpose. Please refer to section [Preparing to Use the BI Publisher Catalog Utility](#) in the chapter [Moving Catalog Objects between Environments](#) in [Oracle® Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher](#).

Assume [BI Publisher Home] to be the directory where BI Publisher is installed. The following steps and commands are given for Linux machine with bash shell.

1. Make a directory "BIPCatalogUtil" under [BI Publisher Home]/.

```
mkdir [BI Publisher Home]/BIPCatalogUtil
```

2. Go to [BI Publisher Home]/Oracle_BI1/clients/bipublisher.
3. Extract contents of BIPCatalogUtil.zip in the above location to go to [BI Publisher Home]/BIPCatalogUtil.
4. Unzip BIPCatalogUtil.zip -d [BI Publisher Home]/BIPCatalogUtil
5. Edit xmlp-client-config.xml. This configuration file is located in the BIPCatalogUtil/config directory.
6. Specify the BI Publisher instance URL ("bipurl") and the user name and password of the BI Publisher instance from which you must export or to which you must import.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
    <comment>BIP Server Information</comment>
    <entry key="bipurl">http://sta00XXX.example.com:14001/xmlpserver/</entry>
    <entry key="username">OPERATIONS</entry>
    <entry key="password">welcome</entry>
</properties>
```

If you do not want to store this information in the configuration file, then at the time of import/export you can also set the bipurl, username, and password as parameters in the command line to overwrite values defined in xmlp-client-config.xml.

7. Add below variables to your environment

```
export BI_EXPORT_HOME=[BI Publisher Home]/BIPCatalogUtil
export PATH=$BI_EXPORT_HOME/bin:$PATH
export BIP_LIB_DIR=$BI_EXPORT_HOME/lib
export BIP_CLIENT_CONFIG=$BI_EXPORT_HOME/config
```

Add the above commands into your `.bashrc` file if you are using a batch shell, so that every time a user logs into a shell, these commands are automatically executed.

8. Close the terminal and launch a new Terminal.

Command Line Utilities to Import and Export Report Artifacts

The following commands are used to export and import reports:

Export Command

```
BIPCatalogUtil.sh -exportfolder catalogpath=[folder-name-to-export]  
basedir=[dest-folder-to-place-exported] subfolders=true extract=true  
overwrite=true
```

Import Command

```
BIPCatalogUtil.sh -import basedir=<temp_otm_reports> subfolders=true  
overwrite=true
```

<temp_otm_reports> is the directory where the report artifacts are placed.

Other Options to Move the Report Artifacts

For many users and report designers, the command line utilities are difficult to use. They may not have access to the environments where they will have to execute these commands. In such case you can use the Download and Upload tools in the report UI.

For more information on this please refer to the section [Performing Tasks on Catalog Objects](#) and [Downloading and Uploading Catalog Objects](#) in the chapter Managing Objects in the BI Publisher Catalog in [Oracle® Fusion Middleware User's Guide for Oracle Business Intelligence Publisher](#).

7. Intermediate File Persistence

During report generation, content and report files are written to the disk before being distributed. By default, these files have a short lifespan. Once the report is generated, the content file is deleted. Once the report is distributed, the report file is deleted. For troubleshooting, it may be useful to view these files. They can be persisted via the following properties:

Table 7-1: Intermediate File Persistence

Property	Description	Default
glog.bipreports.report.persist.browser	If true, report files used for browser views are not removed.	false
glog.bipreports.report.persist.attachment	If true, report files used for email attachments are not removed.	false
glog.bipreports.report.persist.printer	If true, report files used for IPP printing are not removed.	false

8. Printing Reports

Oracle Transportation Management supports the printing of reports to an Internet Printing Protocol (IPP) compliant printer with specific support for a Common UNIX Printing System (CUPS) printer.

Set Up a Printer that Meets Oracle Cloud Requirements

When setting up an on-site printer, please note that the printer must meet the following prerequisites.

- The printer must support printing over the internet for which there are two options:
 - The printer is IPP (Internet Printing Protocol) enabled i.e. it natively supports printing over the internet.
 - If the printer doesn't support printing over the internet which can be the case for older printers, you need to set up an IPP print server at your location. Examples of IPP printer servers are CUPS (Common UNIX Printing System) and Windows IPP Print Server. For information on setting up CUPS or Windows IPP print servers and connecting network printers to them, refer to the CUPS or Windows IPP software vendor documentation.

Note: It is strongly recommended that a print server (CUPS or Windows) be used instead of connecting directly to the printer. This type of setup ensures that the printer is IPP enabled and that it can accept and process cloud-based print requests. A print server also allows for easier debugging through the availability of server level logs.

Note: CUPS refers to setting up a CUPS print server on site at your facility as a gateway to a printer. We use the CUPS print server URL to configure that printer in Oracle Business Intelligence Publisher (BI Publisher). It does not refer to setting up a CUPS server within BI Publisher which is no longer supported as of Release 9 (see [Doc ID 2089912.1](#)).

- The print server must present a valid SSL certificate signed by a trusted CA (Certificate Authority) such as GoDaddy, Verisign etc. Self-signed SSL certificates are NOT supported.
- The on-site printer must be accessible from the Oracle Public Cloud via the internet over a secure connection. If there is a firewall that is protecting customer internal servers, then a firewall policy/rule needs to be configured to allow incoming traffic to the onsite print server. For list of trusted Oracle IP addresses to add to white list for your firewall, please open a Support Request.
- The print server host must be registered with the McAfee Site Address Filter. Please review the Trusted URL Registration section of the Oracle Logistics Cloud Getting Started Guide for more details.
- The printer should be set up with the authentication option enabled.
- The printer should be using either Basic or Digest authentication schemes. NTLM is currently not supported.

Configure Printer in BI Publisher

This document lists the steps required to enable IPP printing either through a printer running from a CUPS server or directly via IPP.

If you are going to use a printer running through CUPS, your system administrator needs to add the printer to the CUPS server. Once your CUPS printer is set up, complete the following:

Log into the BI Publisher console.

1. Click **Sign In**.
2. Click **Administration**.
3. Click the **Printer** tab.
4. Click **Add Server**.
5. For a CUPS or direct IPP printer:
6. Enter the **Server Name**. You will need to enter this into OTM later, so make note of it.
7. Enter the URI for your server. For CUPS the form is "ipp://<CUPS host>:631/printers/<printer name>

IPP uses the TCP port 631 for printing, so any firewalls between the client and the server must be configured to allow bi-directional traffic on that port. Please consult your network administrator if you think any configuration changes are necessary.

8. Check with your system administrator for any Filter requirements.
9. It is required to use SSL Encryption. Select **SSL** for the **Encryption Type** in the Security section.
10. It is also required to protect the Printer/Print Server using User Authentication. Please enter the corresponding details in the **Username**, **Password**, and **Authentication Type** fields. The **Authentication Type** must be **Basic** or **Digest**.
11. The Proxy Server needs to be configured for the Printer. Open an SR to request Cloud Operations to configure the Proxy Server settings for BI Publisher. In a future release, these will automatically be configured.
12. Click "**Apply**".

A Direct IPP printer is set up the same way except for one difference. Instead of going through the CUPS server, you set up the URI to point to the server and IPP port directly. The URI would be entered in the following format, "ipp://<PRINTER_FQDN>:631"

Print from BI Publisher

To ensure BI Publisher functionality, you can print a test page with your printers.

1. Click **Home > Report Job > Search**.
2. Click the "+" sign next to "Reports".
3. Double click "domestic_packing_list".
4. Highlight the "domestic_packing_list" report.
5. Click **Open**.
6. Select the **Output** tab.
7. Click the **Add Destination** button.
8. Select a printer. You can select all your printers by clicking the **Add Destination** button an appropriate number of times.
9. Click the **Schedule** button. Ensure the Frequency is correct, in this case "Once".
10. Select **Run Now**.
11. Click **Submit**.
12. In the **Submit** pop-up, enter a **Report Job Name**.
13. Click **OK**.
14. Go to your physical printer and ensure that your jobs were printed.
15. Sign out of the BI Publisher console.

Configure Printer in OTM

After confirming that printing is working from within BI Publisher, you need to configure OTM to use the printer defined in BI Publisher.

Configure the Printers

1. Navigate to **Business Process Automation > Power Data > Document Generation > Printers**.
2. Click **New**.
3. Enter the following:
 - **Printer ID:** A printer ID
 - **Printer Name:** The server name you gave your printer when you set up the printer in BI Publisher.
4. No other fields are necessary. The system uses the following fields if specified: Orientation, Sides, Number of Copies, and Media.
5. Click **Finished**. Repeat for as many printers as you have set up in BI Publisher.

Print from OTM

Option 1

1. Navigate to **Business Process Automation > Document Manager**.
2. Click **Search**.
3. Select the check box next to a document.
4. Click **Actions > Send Content**.
5. In the Printers grid, enter your printer(s).
6. Click **Save**.
7. Click **Submit**.

You should get a Confirmation dialog box.

Option 2

1. Navigate to **Shipment Management > Shipment Management > Buy Shipments**.
2. Click **Search**.
3. Select the check box next to a shipment.
4. Click **Actions > Business Process Administration > Reports > Domestic Packing List**.
5. Enter the following:
 - a. **Report Format:** PDF
 - b. **Delivery Method:** Print
 - c. **Printer ID**
6. Click **Submit**.

You should get an Information screen saying, "The report print request has been submitted to <YOUR_PRINTER>".

Enable OTM Print Logging

Additionally you can turn on some printing logging as follows:

1. Navigate to **Configuration and Administration > Power Data > General > Log Files**.
2. Select a **Type** of System.
3. Click **Search**.
4. Edit the SYSTEM log.
Note: To edit the SYSTEM log you must be logged into OTM as DBA.ADMIN.
5. Add the following log IDs to the SYSTEM log:
 - a. Print
 - b. PrintDebug
 - c. PrintDetails
6. Click **Finished**.

View Logging

1. Open any Process Management screen. For example, Shipment Management > Process Management.
2. Under Logs, click System.
3. In the **From** field, change the time to when you submitted the first print job.
4. In the **Log** field, select SYSTEM.
5. In the **Severities** field, leave default setting.
6. Select the following IDs:
 - a. Print
 - b. Print Debug
 - c. PrintDetails
7. Leave the **Top Level Process** field blank.
8. Click **View Results**.

You should see something like the following:

```
Debug Print Sending print: Domestic Packing List to <YOUR_PRINTER> (<#>)
[transport - IPP - 1]
Debug PrintDebug File to print
/middleware/app/otm641/otm/temp/prt714623267893139703.pdf [transport - IPP - 1]
Debug PrintDebug BIP Print Service URL: http://pod-primary-1.otmgtm-
cloud.oracle.com:8704/xmlpserver/services/v2/ScheduleService [transport - IPP -
1]
Debug PrintDebug BIP Print Soap Request:
Debug PrintDebug BIP Print Soap Response:
Debug Print Print successful: Domestic Packing List to <YOUR_PRINTER> (<#>)
[transport - IPP - 1]
```


Debugging Using CUPS Print Server

It is highly recommended to use a CUPS printer server. Using a print server will make it easier to diagnose issues since you can view the status of the printer and printer jobs.

1. Log into the printer server host.
2. Click the **Printers** tab.

You will see a list of CUPS printers and ancillary information. Look for the "Status".

3. Click on your printer in the queue **Name** column.
4. Click **Show Active Jobs**.
5. Scroll to the bottom of the page, and you should see a job "spooling".

If your job spools it's in the printer queue which indicates that your job has been recognized and is being processed.

IPP Printing Frequently Asked Questions (FAQs)

Is there an alternative means of printing reports, documents, etc.?

Yes. Reports can be viewed and printed on a local printing using a web browser.

What is the standard Protocol for printing directly from the Oracle Logistics Cloud?

IPP over HTTPS.

Does direct printing from the Oracle Logistics Cloud require an SSL certificate from a Certification Authority?

Yes, the print server must present a valid SSL certificate signed by a trusted certificate authority such as GoDaddy.

Can a self signed SSL certificate be used, rather than a certificate from a Certification Authority?

No. Self signed certificates are not supported.

Can the print server be the printer itself, or, must a print server be, for example, a CUPS (Common UNIX Printing System) server?

The IPP print server can be the printer itself. Newer printers already have built in IPP over HTTPS support. Consult the printer's user guide for specific setup requirements and availability.

If the printer does not natively support IPP over HTTPS, then a CUPS server (or print server that supports IPP over HTTPS) can be used. For CUPS (or similar) based setup, ERP Cloud BI Publisher will connect to the print server, and then the print server would connect to the printer. All required setups will need to be completed and tested by Cloud customers.

Is it mandatory to use SSL and client authentication?

Yes. For cloud customers, those are enforced by default and should not be modified by the customer.

How to test that my printer is working?

You can run any report and redirect the print output to the newly setup printer. If the job completes successfully and the printer receives the print request then the setup is good.

Can we use client side certificate instead of username/password for cloud printer authentication?

No. Supported authentication methods are basic or digest both of which use username/password.

Are there any troubleshooting tips in case the print request doesn't go through?

Review and make sure the prerequisite steps are followed.

Verify the print server logs to check if it received the print request. Refer the IPP print server, CUPS or Windows IPP software vendor documentation for more details on where to find the logs.

In case you are still unable to receive the print requests on the print server from Oracle cloud, please log an SR to engage Oracle Support in debugging any setup issues on Oracle cloud. When logging the SR, please provide the following details:

- Screen shot of print server setup in BI Publisher
- Complete printer URL

9. BI Publisher 10g Migration

Oracle Transportation Management provides the following reports as the sample reports.

- BILL_OF_LADING
- DOMESTIC_PACKING_LIST
- INVOICE
- DEMURRAGE_FINANCIAL
- AES_FILING
- CUSTOMER_COMMERCIAL_INVOICE
- PICKUP_SUMMARY
- AWARD_BY_CARRIER
- AWARD_BY_LANE
- BID_ROUND_STATUS
- LTL_LANE_SUMMARY
- LTL_SHIPMENT_HISTORY
- MANUAL_AWARDS_REPORT
- OUTLIER_BIDS
- SAVINGS_BY_LANE
- WINNING_VS_OTHER_BIDS

These reports are available for new installations and systems that were migrated from a version prior to 6.4. However, these reports need to be loaded into the BI Publisher instance installed by the customer.

Migrating from Stand-alone BI Publisher 10g

If you have an existing external BI Publisher 10g instance, you can migrate the existing reports to BI Publisher 11g. Please refer to the [Upgrading Oracle BI Publisher](#) in the [Oracle® Fusion Middleware Upgrade Guide for Oracle Business Intelligence](#) for more details. Particular attention should be paid to the “Post Upgrade Considerations for BI Publisher 11g” section.

Once you have upgraded to 11g, you can upgrade to 12c.

Migrating from Embedded BI Publisher 10g

If you are migrating from the embedded BI Publisher, you will need to do more work to migrate existing reports. The SQL queries defined in BI Publisher 10g Data Templates will need to be rebuilt as BI Publisher 11g Data Models. It may be possible to upload the BI Publisher 10g Format Template into the BI Publisher 11g catalog, however it is highly recommended that you re-create reports using the new 11g Layout Template whenever possible. The 11g Layout Template is a WYSIWYG report designer which can greatly simplify report design and maintenance.

In addition to migrating the report itself, it will be necessary to modify the existing Report Registration in Oracle Transportation Management. The follow changes should be made to have a report execute using the external BI Publisher instance.

1. Log into Oracle Transportation Management.
2. Navigate to **Business Process Automation > Power Data > Document Generation > Reports**.
3. Select an existing report.

4. Select the **Report System** created previously. See BI Publisher Configuration section for more detail.
5. Specify the **Report Path** corresponding to the Report in BI Publisher.
6. Click **Finish**.

Migrating from BI Publisher 11g to 12c

If you are migrating from BI Publisher 11g to 12c (12.2.1.2.x), you can migrate the existing reports to BI Publisher 12c. Please refer to [Oracle® Fusion Middleware Upgrade Guide for Oracle Business Intelligence](#) for more details. Particular attention should be paid to the “Post Migration Steps for BI Publisher” section.

10. Report Storage and Cleanup

Report Storage

Depending on your use of reports, you may want to adjust how temporary report files are stored. These files are used to provide report content to emails, printers and the browser and are managed via common Store functionality. A store represents a stream of data:

- **In memory:** If the data is generally small, a customer can configure the data to be stored in memory. This increases performance at the cost of memory usage.
- **On disk where the caller determines the data lifespan:** This is referred to as an 'owned' cache as the items are owned by the caller. An owned cache is typically used when the data has a short, deterministic lifespan; or to allow debugging of persistent content. Owned caches leverage the standard `glog.util.cache.TemporaryFileCache`, which in turn extends our common `glog.util.cache.OwnedCache`.
- **On disk in an LRU file cache:** This store leverages the standard `glog.util.cache.TemporaryFileLRUCache`, which in turn extends our common `glog.util.cache.LRUCache`.

The following list contains properties that control the report store. Note that the store can be monitored via App-Tier Caches accessible as `DBA.ADMIN` under Technical Support. The cache name is Reports.

- `glog.bipreports.report.store`
- `glog.bipreports.report.path`
- `glog.bipreports.report.prefix`
- `glog.cache.Reports.capacity`

Report Persistence

Regardless of the store type, most report use cases are synchronous. A user requests a report from the browser and either receives a synchronous HTTP response or immediately generates an email with a report attachment. In both these cases, caller code can remove the content or report as soon as it's used. Only when reports are retrieved asynchronously by a user via an email link does a report store truly need to manage generated files.

The following table details the lifespan of reports. For each use case, you can suppress the automatic removal of data and rely on the store or periodic cleanup (see below) to limit file growth. This is recommended for asynchronous reports but may also be useful for debugging of synchronous reports.

Table 10-1: Report Lifespans

Use Case	Report Removal	Suppress Removal With	Default
Run report; Display on Screen	After HTTP response is streamed	<code>glog.bipreports.report.persist.browser</code>	false
Run report; Email	After creation of e-mail attachment	<code>glog.bipreports.report.persist.attachment</code>	false
Schedule report Print Document (Agent action)	After the report is first served to a remote user	<code>glog.bipreports.report.persist.log</code>	true

Use Case	Report Removal	Suppress Removal With	Default
Printing of the Report	After creation of print attachment to send to printer	<code>glog.bipreports.report.persist.printer</code>	false

General File Cleanup

The default implementations of the report store automatically control file growth on the application servers. There are a number of reasons, though, why a general cleanup mechanism is needed:

- If Oracle Transportation Management application servers fail or are cycled, the owned and LRU stores lose all state. Any files currently on disk will remain.
- If you choose to suppress report removal for debugging, files can build up on disk.

To address these cases, the report engine includes a cleanup poller that wakes up periodically and scans the specified directories for files older than a given threshold. These files are then removed.

Properties include:

- `glog.bipreports.cleanup.type=report`
- `glog.bipreports.cleanup.report.path=<directory to scan>`. This defaults to `$glog.bipreports.report.path$`
- `glog.bipreports.cleanup.report.files=<comma-delimited list of file wildcards to clean. See WildcardFileSearch for syntax.>`. This defaults to `report-*, *.html, *.rtf, *.pdf, *.xls`.
- `glog.bipreports.cleanup.report.threshold=<# of days old a file must be for cleaning>`. This defaults to 30 days.
- `glog.bipreports.cleanup.report.frequency=<# of minutes between cleanup checks>`. This defaults to 360 minutes.

11. Troubleshooting

When generating a report, errors can occur in BI Publisher. If fatal, an exception is thrown to Oracle Transportation Management. For an ad-hoc report request, the exception is shown to the user; for scheduled or automated reports, it is written to the exception log.

Report Logging

Two types of logging are available to diagnose report issues:

- Oracle Transportation Management logging. The `Report` log ID outputs the major steps of report creation. The `ReportDetails` log ID provides more detailed information.
- BI Publisher logging. The BI Publisher engine maintains a separate logging subsystem to track data generation and transformation progress. Please refer to BI Publisher documentation for more details on this topic

Oracle Transportation Management Logging

As mentioned above, you can enable logging in Oracle Transportation Management by enabling the `Report` and `ReportDetails` log IDs. This logging will can give the details of which report is being accessed and the SOAP request being posted to the BI Publisher external server web service.

BI Publisher Logging

You can enable detailed logging on the BI Publisher server to track what is being done by BI Publisher while generating the report. For more information on the type of log files, the categorization of the log levels, and configuration of the log files refer to chapter 12 [Diagnostics and Performance Monitoring](#) in the administrators guide at [Oracle® Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher](#).

Frequently Asked Questions

How do I configure BI Publisher reports in Oracle Transportation Management?

This guide talks about this in more detail. Below are the high level steps.

1. Configure the BI Publisher Data Source.
 - a. Give data source name as `otmoltp`
 - b. Give the database username as `globalreportuser`.
2. Design Report in BI Publisher.
 - a. Define a data model.
 - b. Define a report with layout for the data model.
 - c. Test the report using BI Publisher server directly.
3. Define a report system for connecting to the BI Publisher server or use the report system `DEFAULT` with property macros pointing to `glog.bip.externalFarm.*` properties.
 - a. The bi publisher user with which you created a report and user with which you want to access the report need not be the same. Please refer to [section 2.3 Permission Grants](#) and section [3.8 Configuring Users and Roles](#) in the [Oracle® Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher](#).
4. Define the report definition with the report system.
 - a. Provide the report absolute path. This will be the path in the URL for accessing the report from the BI Publisher server.

- b. Take the Report path prefix defined in the report system into consideration since the absolute path provided for the web service will be a concatenation of the report path prefix and the absolute path given in the report definition.
- c. Define report parameters. The parameter name should match the parameter name defined for the data model in BI Publisher.

Which database user is used for generating reports?

The application uses the database user pointed by the PRIMARY_THIN data source defined in OTM for generating online reports. Typically it is glogdba.

For generating offline reports which connect to a replicated offline database, the application uses the database user pointed by the ODS data source defined in OTM.

For generating archive reports which connect to an archive schema, the application uses the database user pointed by the ARCHIVE data source defined in OTM.

Another important point to note here is the reports run from the BI Publisher server will not have any VPD context applied. Reports run from OTM run with the VPD Context of the user running the report.

I am upgrading from pre 6.4 releases to 6.4.2. What are the steps I need to follow?

In pre 6.4 releases, reports are generated in two ways:

Embedded Report Generation

A report definition is linked to a data template (query template) and the format template. Using the Java API provided by the BI Publisher we used to generate the reports with in OTM. This was available from release 6.1 to 6.3.7.

In this case we used to have a data template or query template which used to generate the structured data i.e. xml data and a format template which defines the layout of the report.

When upgrading from 10g to 11g reports one will have to understand the changes made to the report architecture. The following link provides more details on the upgrade from 10g to 11g.

https://docs.oracle.com/middleware/11119/core/FUGBI/upgrade_bip.htm#FUGBI308

Follow the steps given under section 6.5.2 for migrating the data templates to data models given in the below link

https://docs.oracle.com/middleware/11119/core/FUGBI/upgrade_bip.htm#FUGBI589

In step 4, please use the Upgrade E-Business Suite Data Template Directory option to specify the path of the data templates to be upgraded.

You can use the same format template as before provided the path to access the elements are same as before.

Using the data model and format template you will have to create a BI Publisher 11g report.

Integration with External BI Publisher Stand Alone Server

Integration is provided with a standalone BI Publisher report server which has the report artifacts (data template and format template). OTM will call into the stand alone server using web services and get the report.

The following link provides details on BI Publisher upgrade.

https://docs.oracle.com/middleware/11119/core/FUGBI/upgrade_bip.htm#FUGBI308

Can I use the data template and format template I have in pre 6.4 releases for reports in 6.4.2?

No. Please follow the steps outlined in the section “**I am upgrading from pre 6.4 releases to 6.4.2. What are the steps I need to follow?**”

Can I use Parameter Operators in 6.4.2?

No. Every report defined in 6.4.2 or migrated to 6.4.2 is assumed to be a report with parameters being used as bind values. If you have reports with parameter operators, you should migrate these reports to use the parameters as bind values.

Can I use configure consolidated reports?

In pre 6.4 releases the consolidated reports are used to run the report for multiple business objects using the One Of operator. These reports are used only when using the Run Report web action. Since the parameter operators are deprecated, this functionality will not work in 6.4.2. It is assumed that all of the reports configured with OTM are transactional reports. So any use of consolidated reports is currently not supported.

Why do I get Invalid format requested error when trying to view selected format of the report?

After designing the report and registering it with OTM, when trying to generate the report for a particular format, if there is an error like:

```
oracle.xdo.webservice.exception.OperationFailedException:  
PublicReportService::generateReport for reportAbsolutePath  
[/Reports/pickup_summary/pickup_summary.xdo] failed: due to  
oracle.xdo.servlet.data.DataException: Invalid format requested: excel2000  
[000717] Invalid format requested: excel2000
```

This is because the report is not enabled for the format being requested.

Follow the steps as per the below link.

- [Selecting Output Formats](#)

How do I send multiple values for a parameter of the report?

Follow the steps as per the below link

<https://docs.oracle.com/middleware/12212/bip/BIPDM/GUID-FD8752C2-D398-4F8F-BB80-7C1347021FA4.htm#BIPDM310>

When defining a new parameter enable the check box for Text field contains comma-separated values.

How do I move report artifacts from one server to another?

Once a report is designed in the test environment, we will want to move the report from test environment to production environment.

Please follow the steps as outlined in chapter 6 of this guide.

Why do I get OutOfMemoryError: Java heap space?

Oracle Transportation Management is integrated with BI Publisher to send the report requests and receive the generated output as response. OTM is not equipped to manage very large responses that can be received from the BI Publisher server. Typically OTM assumes that transactional reports which tie to one business object are configured.

If analytical reports are configured, these reports can return a large amount of data which in turn can occupy the space in the java heap and can lead to OutOfMemoryError.

While OTM can receive a small amount of data for the transactional reports, it may not have enough space to handle the large amount data that can be generated in case of analytical reports.

It is recommended that you use transactional reports within OTM for this reason. The analytical report can still be used by leveraging the BI Publisher report server capabilities.

How do I view all of the scheduled reports?

Use the menu link **View Scheduled Jobs** at **Business Process Automation > Reporting > View Scheduled Jobs**.

Use **Business Process Automation > Reporting > View Completed Jobs** to view the report jobs which were generated during the scheduled period.

Note that the scheduling that happens in the application has no relation with the scheduling that BI Publisher provides. OTM uses the BI Publisher server only to send the report generation SOAP requests.

Scheduled reports are not being received as emails, what should I check?

A scheduled report, even though it got generated, can back up on the mail queue in the application if the mail queue does not have enough threads to process the queued items. The report once generated will be queued up for email processing. The queue used for this is "transport – SMTP". Check the number of threads given for this queue and check the backlog on this queue. If there are backlog items on the queue, please increase the number of threads for this queue using the following property.

```
glog.process.threadGroup.mail
```

By default it has a value of 1. Please increase this value but do not increase it to a high value which will result in burdening of the mailing system.

Why are scheduled report received as a link and not as an attachment?

Read the online help for Security Level that can be configured for each of the report.

Even if the security level allows you to attach, there is a limit of size of the reports which can be attached. The property that governs this is

```
glog.notify.report.maxPdfSize
```

You should also check the property

```
glog.mail.maxContentSize
```

This property defines the maximum content of the email size. So if you decide to increase the first property you should check that the limit does not exceed the second property.

How do I generate reports with a bar code?

For information on how to create reports which need to print barcodes, please refer to the "[Using the Barcode Fonts Shipped with BI Publisher](#)" section in the "[Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher](#)". You will be able to use only three bar code fonts shipped with bi publisher. Please refer to the above documentation for more information.

Why does the report give me the same output for the same set of parameters even though I change the user?

When a report is designed in BI Publisher, you need to pay attention to the caching properties of the reports. If the caching is enabled when you have the same set of parameter the report can be cached for a certain period of time. Please refer to the following link for more details.

<https://docs.oracle.com/middleware/12212/bip/BIPRD/GUID-921F37EA-3956-4244-BFBF-2B942DF10E22.htm#BIPRD2125>

Though it is a different application user, the report has no way of telling that it is different user unless, the user is also defined as one of the report parameters.

How do I disable report caching?

Please refer to the section "Why does the report give me the same output for the same set of parameters even though I change the user?"

12. Additional Resources

Oracle Business Intelligence Suite Enterprise Edition Documentation Library:

- http://download.oracle.com/docs/cd/E10415_01/doc/nav/portal_booklist.htm

A BI Publisher developer's diary

- <http://blogs.oracle.com/BIDeveloper/>

BI Publisher Forum

- <http://forums.oracle.com/forums/forum.jspa?forumID=245>

13. Appendix: General Properties

See the “Advanced Configuration: Customer-defined Properties” chapter in the Administration Guide for instructions on how to manage reserved properties.

Table 13-1: General Properties

Property	Reserved	Description
glog.bipreports.app.tempDir	Yes	This sets the base directory for the app-tier bi publisher report generation. Default: \$temp.dir\$/bipublisher/
glog.bipreports.cleanup.type=report glog.bipreports.cleanup.report.path=<directory to scan>. glog.bipreports.cleanup.report.files=<comma-delimited list of file wildcards to clean>. See WildcardFileSearch for syntax.>. glog.bipreports.cleanup.report.threshold=<# of days old a file must be for cleaning>. glog.bipreports.cleanup.report.frequency=<# of minutes between cleanup checks>.	Yes	A cleanup type specifies some group of files in a common directory to scan. See the Report Designer’s Guide.
glog.bipreports.report.path=<path>	Yes	Directory to store final reports. Only valid for owned or LRU stores. Default: \$temp.dir\$ (e.g. /opt/otm-61-wl/temp)
glog.bipreports.web.tempDir	Yes	This sets the base directory for the web-tier bi publisher report generation. Default: /bipublisher/temp/

