Oracle® Transportation Management

Report Designer's Guide

Release 6.4.1

Part No. E68701-01

December 2015

ORACLE®

Copyright Notice

Contents

Send Us Your Comments

Oracle® Transportation Management Report Designer's Guide, Release 6.4.1

Part No. E68701-01

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 customizing reports intended for use within the Oracle Transportation Management Application.

Change History

Date	Document Revision	Summary of Changes
12/2015	-01	Added Section 6 - Printing Reports. This section covers issues relating to IPP and CUPS printing.

1. Reporting Overview

This document describes how to design and generate reports for Oracle Transportation Management. It is intended for Report designers and integrators who need to develop reports based off of Oracle Transportation Management data, and for system administrators who need to configure report generation. It is not intended to serve as an installation or users guide. Please consult the BIPublisher Installation Guide for details on installing BIPublisher; the online help for details on generating ad-hoc or scheduled reports from within Oracle Transportation Management.

Architecture

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

Each report may be associated with a specific report system. This system defines the communication between Oracle Transportation Management and the external report server. This may be an HTTP URL, in the case of third-party reporting tools, or connection information for a BI Publisher server.

Figure 1 summarizes the data flow.
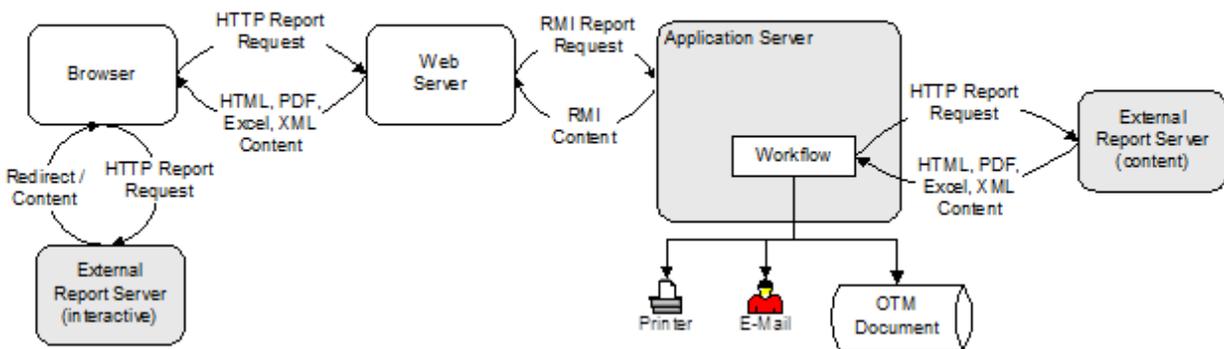
Figure 1: Reporting Architecture

If the reporting system does not handle embedded content, the link directly maps to the report system URL. After selecting the report, you are redirected to the external report system. This system may prompt for parameters and/or distribution, generate the report, and return report content to the browser. Oracle Transportation Management makes no assumptions regarding any returned content.

If the reporting system, though, handles embedded content, the report link connects to standard Oracle Transportation Management reporting screens. These screens allow you to run ad-hoc or scheduled reports. For ad-hoc reports, an HTTP request is sent to a web server which passes the request via RMI to an application server. The application server sends an HTTP request to the report server and assumes the response, if successful, contains report content. This content is piped back through RMI to the web server, and then through HTTP to the browser. The content may also be distributed via e-mail, IPP printing, and/or stored with an associated business object.

Scheduled report requests, or requests triggered by a workflow agent, similarly send an HTTP request to the report server. The response content is distributed via e-mail, printed via IPP, or stored with an associated business object.

Streamlined support is available when generating reports on a remote BI Publisher server or 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.

2. BIPublisher Configuration

The Oracle Transportation Management and Global Trade Management license includes a restricted use license for installing and running Oracle BIPublisher. The restriction limits the use of BIPublisher to reporting on data retrieved from data created by Oracle Transportation Management and Global Trade Management. The following steps assume that the correct version of BIPublisher has been installed per the Oracle Transportation Management Technical Architecture Guide.

Note: Oracle Business Intelligence Enterprise Edition installation includes BIPublisher. In order to generate BIPublisher Reports from within Oracle Transportation Management the following steps must be followed:

BIPublisher Data Source

Log into BIPublisher Administration Console

```
http://test.company.com:9704/xmlpserver/
```

Create a new Data Source pointing to the Oracle Transportation Management operational database. It is highly recommend for security reasons that the "globalreportuser" database user, provided by default, be used. This user has the minimal rights needed to create and run reports.

Make sure the data source name is *otmoltp* for being able to edit and view the sample reports provided by Oracle Transport Management in the report editor provided by bi publisher.

A data source needs to be set up inside of BIPublisher using the following steps:

1. Log into BIPublisher
2. Click **Administration**.
3. Click **Data Sources > JDBC Connection**.
4. Click **Add Data Source**.
5. Enter Connection Details

    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>*

Oracle Transportation Management Report System

Log into Oracle Transportation Management and create a Report System. Configuration of Report Systems is available on the Oracle Transportation Management menu at Business Process Automation > Power Data > Document Generation > 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. The following properties should be configured based on the previously installed BIPublisher instance.

*   glog.bip.externalFarm.host=

- glog.bip.externalFarm.port=9704[1]
- glog.bip.externalFarm.user=bipreportuser
- glog.bip.externalFarm.password=
- glog.bip.externalFarm.reportPath=/

Though the use of property macros is not required for the DEFAULT report system, it facilitates the movement of an integrated Oracle Transportation Management/BI Publisher system from a test platform to a production platform.

Sample Reports

Several sample reports are included as part of the OTM software installation. Since Oracle Transportation Management does not automatically install BIPublisher, loading the sample reports into the BIPublisher 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 BIPublisher 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 BIPublisher report server are on two different machines, make sure to copy the report artifacts from the app server to a folder on the BIPublisher report server, e.g. temp_otm_reports

Import/Export reports to BI Publisher Repository

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

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.

---

[1] Note that the BIP_PORT column on the REPORT_SYSTEM is numeric and does not support property macros. Instead, the **glog.bip.externalFarm.port** property always overrides the contents of the DEFAULT report system if specified. In a future release, the BIP_PORT will be reset to be VARCHAR.

---

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 bach 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.

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> /otm/ subfolders=true
overwrite=true
```

`<temp_otm_reports>` is the directory where the report artifavts are placed.

2-3

3. BIPublisher Report Design

This section provides guidelines for designing reports to be integrated with Oracle Transportation Management. It is not meant to be an exhaustive description of BI Publisher, but to present a streamlined approach to quickly build reports. It requires installation and a basic understanding of both products.

Oracle Transportation Management includes a set of analytic and transactional[2] reports available to all report users. The guidelines and sample reports provided by the Oracle Transportation Management installation can be used as a starting point for new reports.

To create an external report, the report designer:

1. Creates the report in BIPublisher. Refer to BIPublisher documentation.
2. Registers the report for execution in Oracle Transportation Management. See the **Report** section.
3. Registers custom parameters passed from Oracle Transportation Management to the report. See the **Parameterization** section.

Report Registration

After a report has been created, it is necessary to register the report in Oracle Transportation Management. Before registering the report, it is necessary to determine the Report Path for the report. Log into BIPublisher Administration Console and obtain the Report Path by viewing the Report and running the action Share Report Link > Current Page.

```
Ex. http://test.company.com:9704/xmlpserver/OTMUsers.xdo
```

The Report Path in this example is `"/OTMUsers.xdo"`

The following steps should be followed in order to register a report for execution from within Oracle Transportation Management:

1. Log into Oracle Transportation Management. They must have rights to create report data.
2. Navigate to **Business Process Automation > Power Data > Document Generation > Reports**.
3. Create a new report. By default, the report will be created in your domain.
4. Select the **Report System.** See **BIPublisher Configuration** section.
5. Specify the **Report Path** identified above.
6. Select the **Select Via UI** check box. This allows the report to be explicitly run by Oracle Transportation Management users.
7. Select **Default Display Format**, i.e. PDF. Note: there is a limitation that embedded images do not work with the HTML display format.
8. Select the **Use Report Parameters as Bind Values** check box.
9. Register custom parameters to be passed from Oracle Transportation Management to the report. See the **Parameterization** section.

---

[2] 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.

Parameterization

Oracle Transportation Management passes standard and ad hoc parameters to the report generator[3].
Table 1 summarizes the standard parameters passed to every report.

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.
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.	
P_DATE_FORMAT	The Oracle date format for both input parameters and output fields.	For input parameters, use the `TO_DATE(:P_MY_DATE, :P_DATE_FORMAT)` function. For output parameters, use the `TO_CHAR(field, :P_DATE_FORMAT)` function. This should be reserved for fields that are date-only.

---

[3] If external report generators do not support these parameters, their content type should be set to `None`.

Name	Description	Comments
P_DATE_TIME_FORMAT	The Oracle timestamp format for both input parameters and output fields.	For input parameters, use the `TO_DATE(:P_MY_TIMESTAMP, :P_DATE_TIME_FORMAT)` function. For output fields, use the `TO_CHAR(field, :P_DATE_TIME_FORMAT)` function. This should be reserved for fields that have both date and time.

Table 1: Standard Parameters

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**.[4]

For BI Publisher, all parameters are available for use in the **SQL Template**. Each parameter is mapped into a bind variable matching the Report Parameter Name with a string value. If any bind variable in the SQL template is not specified in the report parameters, BI Publisher fails to generate data.

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

Bar Codes

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".

---

[4] 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.

4. Advanced Content

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 preferences[5]. 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
```

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.

---

[5] Note that the Reporting has never applied user time preference. Time preference is assumed to be HH24:MI:SS.

When running BIPublisher reports, VPD security is automatically enforced by the system. The database connection passed to the BI Publisher web services maintains the user and role in context, applying VPD policies on any queries.

External report generators other than BIPublisher, 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.

Using Utility Packages

Oracle Transportation Management Package Reference

Oracle Transportation Management provides a number of PL/SQL functions/procedures that can be used in BIPublisher Data Models. Table 2 summarizes the available procedures and functions. For more information, please review the `create_rpt` scripts in `OTM_INSTALL_DIR>\glog\oracle\script8`.

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
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

4-1

Package	Procedure/Function	Description	Parameters	Returns
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
	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

Package	Procedure/Function	Description	Parameters	Returns
	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
	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

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

Table 2: PL/SQL Report Functions

Adding Custom Packages

To add a custom package for use with SQL templates, the report designer:

- Creates the PL/SQL package specification and the package body definition in the Oracle Transportation Management `reportowner` DB schema
- Compiles both the PL/SQL package specification and body to make sure the code is error free
- Provides grants and permissions so Oracle Transportation Management users can access the custom package. The following scripts are provided under the installation and should be run as `reportowner`.

  ```
  <OTM_INSTALL_DIR>/glog/oracle/script8/reportowner_grants.sql
  <OTM_INSTALL_DIR>/glog/oracle/script8/create_public_synonyms.sql
  ```

5. 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 `REPORT_DETAILS` 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 BIPublisher documentation for more details on this topic

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:

Property	Description	Default
glog.bipreports.content.persist	If true, content files are not removed	false
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 e-mails attachments are not removed	false
glog.bipreports.report.persist.printer	If true, report files used for IPP printing are not removed	false

6. 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.

To add a printer for reports, use the Printer manager[6]. Table 1 lists all printer properties, their values, defaults and description.

Field	Use	Options	Default	Comments
Host	IPP host			
Port	Port number		631	
Path	IPP printer path			For CUPS servers, this is typically /printers/<printer name>. For Windows servers, this is typically /printers/<printer name>/.printer
Authentication	HTTP Authentication	None Basic Digest	None	
User	HTTP Authentication User			
Password	HTTP Authentication Password			
Encryption	Message encryption type	None SSL	None	
Character Set	Request character set		UTF-8	
Language	Request language		en	This does not define the language of the report, only the language for print logs.

---

[6] Accessed from the main Oracle Transportation Menu via **Business Process Automation > Power Data > Printers**.

6-1

Field	Use	Options	Default	Comments
Orientation	Image orientation on the printed page	Portrait Landscape	Portrait	Orientation can be counter-intuitive when using OS utilities to convert content to Postscript. E.g. CUPS use of gs to convert PDF to Postscript automatically reorients the PDF to landscape if it's margins exceed portrait. The IPP orientation property is applied after this conversion. Thus, specifying a Landscape orientation for a PDF report can result in reorienting it to Portrait. On some printers this will print cropped; on others, the user may be prompted to feed special size paper into a manual feed. Generally, explicit orientation should be reserved for Postscript content.
Sides	Instructions on two-sided printing	One sided Two sided – Long edge Two sided – Short edge Duplex Tumble	One sided	Duplex = Two sided – Long edge Tumble = Two sided – Short edge
# Copies	Number of document copies to print		1	
Media	Paper tray/size specification	Tray 1 Tray 2 Tray 3 A3 A4 A5 B4 B5	Tray 1	Many printers will force the paper tray based on orientation and content size.
Printer Format	Document format expected by the printer	Postscript PDF Text Octet Stream	Postscript	Generally, this should be left at Postscript. For printers with native PDF support it can be set to PDF, especially for those reports or documents with PDF content. See section 1.2 for a discussion of content mapping to report format.

Field	Use	Options	Default	Comments
CUPS	Set for CUPS servers		unset	This should be set for CUPS, unset for Windows IPP. CUPS servers provide automatic format support for text, PDF and image (gif, jpeg, png, tiff) content.
Use Chunked Body	Set if the IPP HTTP message supports chunking		unset	
Use Full URL	Set if the IPP HTTP message should use a full URL in the request header		unset	
Page Ranges	Page ranges for printing		all pages	Supports specification of multiple comma-delimited ranges: • "3" : only print page 3. • "2-5" : print page 2-5. • "1,3-5" : print page 1 and 3-5.

Table 3 - Printer Fields

Handling Document Formats

To spool general content to an IPP printer, IPP may need to convert the content from a document format into a format understood by the printer. e.g., a Postscript printer requires a PDF document be converted before it is spooled to the printer. OTM and BI Publisher support the following type of printers:

- Postscript: understands application/postscript documents without conversion
- PDF: understands application/pdf documents without conversion
- Text: understands text/* documents without conversion
- Octet Stream: understands a proprietary document type (e.g. application/ms-word) without conversion

Most IPP and CUPS printers are Postscript printers and all documents must be converted to Postscript before spooling to the printer.

IPP supports two types of conversions:

- implicit. CUPS implementations of IPP automatically convert PDF to Postscript via the pdftops utility. They also support direct rendering of text and image files to the CUPS printer.[7]

---

[7] OTM uses BI Publisher to interact with IPP. As such, an OTM user cannot directly configure the CUPS printers. Text and image rendering may require tuning via the CUPS printer configuration file (printers.conf). This file allows the user to specify text and image parameters like characters per inch, lines per inch and percent scaling.

- explicit. By passing a command-line to IPP, the server can apply a caller-specified transformation utility to the document, resulting in printer-compliant content.

To utilize the implicit CUPS format conversions, a CUPS printer should be specified with the CUPS Server box checked. This leverages conversions configured directly in CUPS to map document content to the native printer format.

If your IPP implementation does not support the conversion of a particular document format to native printer format, you must add an explicit conversion via the Printer Conversions power data. These are global conversion utilities, invoked by BI Publisher, to convert your content to something the printer can understand. Each printer conversion represents a command-line utility taking a document input file `{infile}` to create a printer spool file `{outfile}`. e.g.:

```
gs –sOutputFile={outfile} {infile}
```

could be used on a Windows platform to convert PDF documents to Postscript via the `Ghostscript` open source utility.

If an explicit converter cannot be found to map a document format to printer format, the print request fails.

7. BIPublisher 10g Migration

Oracle Transportation Management updated the following reports to BIPublisher 11g.

- BILL_OF_LADING
- DOMESTIC_PACKING_LIST
- INVOICE
- DEMURRAGE_FINANCIAL
- AES_FILING
- CUSTOMER_COMMERCIAL_INVOICE
- PICKUP_SUMMARY

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 BIPublisher instance installed by the customer. Refer to **BIPublisher Configuration** section for more details on this subject.

> **Note:** 10g reports were not removed from existing installations during migration. These reports were intentionally left in order to allow implementers to reference the reports while migrating to 11g.

Migrating from Stand-alone BIPublisher 10g

Customers that have an existing external BIPublisher 10g instance, may be able to migrate existing reports to BIPublisher 11g. Please refer to the "Upgrading Oracle Business Intelligence 10g to 11g" tutorial" for more details. Particular attention should be paid to the "Post Upgrade Considerations for BI Publisher 11g" section.

http://www.oracle.com/webfolder/technetwork/tutorials/obe/fmw/bi/bi11115/upgrade/upgrade_to_11g.htm

Migrating from Embedded BIPublisher 10g

Customers migrating from OTM embedded BIPublisher will need to do more work to migrate existing reports. The SQL queries defined in BIPublisher 10g Data Templates will need to be rebuilt as BIPublisher 11g Data Models. It may be possible to upload the BIPublisher 10g Format Template into the BIPublisher 11g catalog, however it is highly recommended that customers recreate 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 BIPublisher instance.

1. Log into Oracle Transportation Management.
2. Navigate to **Business Process Automation > Power Data > Document Generation > Reports**.
3. Select an existing report.
4. Remove the **Query Template** and **Format Template**.
5. Select the **Report System** created previously**. See **BIPublisher Configuration** section for more detail.
6. Specify the **Report Path** corresponding to the Report in BIPublisher.
7. Click **Finish.**

8. 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