Oracle® Transportation Management

Administration Guide

Release 6.2

Part No. E20093-05

August 2012

ORACLE®

Oracle Transportation Management Administration Guide, Release 6.2

Part No. E20093-05

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

Access to Oracle Support

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

Contents

Send Us Your Comments

Oracle Transportation Management Administration Guide, Release 6.2

Part No. E20093-05

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 would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

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 guide is for administrators who are responsible for installing and managing the Oracle Transportation Management system at your site. This manual provides step-by-step installation instructions for installing all Oracle Transportation Management software components. This manual does not cover the installation of any operating system that is required to run Oracle Transportation Management such as Linux or Windows Server. It is assumed that your IT staff will handle the installation and configuration of this software.

Change History

Date	Document Revision	Summary of Changes
10/10	-01	Initial release.
01/11	-02	Correct table list in the section "Manually Purge Transmissions" section. Added section on session bean monitoring and tuning.
06/11	-03	Added the following to the PCMIler section (2.4): **Note**: PCMiler *must be configured to return its distance in miles.*
11/11	-04	Updated the Analyzing Tables/Gathering Statistics section of the Database segment of the Performance Tuning chapter. The instructions in step 2 have been modified to reflect 11g.
8/12	-05	Added the Changing the hdowner Password for Fusion Transportation Intelligence section.

1. Architecture Overview

Oracle Transportation Management is built for interactive environments that leverage the internet as an information backbone to capture reference data such as updates to carriers, carrier rates, shipping schedules, locations, ports, and other logistics sources. The underlying architecture is a unique multi-enterprise data model with n-tier web and application servers, partitioned database server, native XML language, extensible style sheets, advanced data security, and optimal caching.

Oracle Transportation Management is highly distributed and collaborative. It is written in Java and uses Enterprise Java Beans, XML, HTML, and XSL technologies. The design is based on a thin HTML client/fat application architecture so that applications are easily deployable and accessible by multiple enterprises. It supports comprehensive integration via XML APIs to all of the application logic.

Server Overview

The following illustration describes the overall architecture of the Oracle Transportation Management environment.
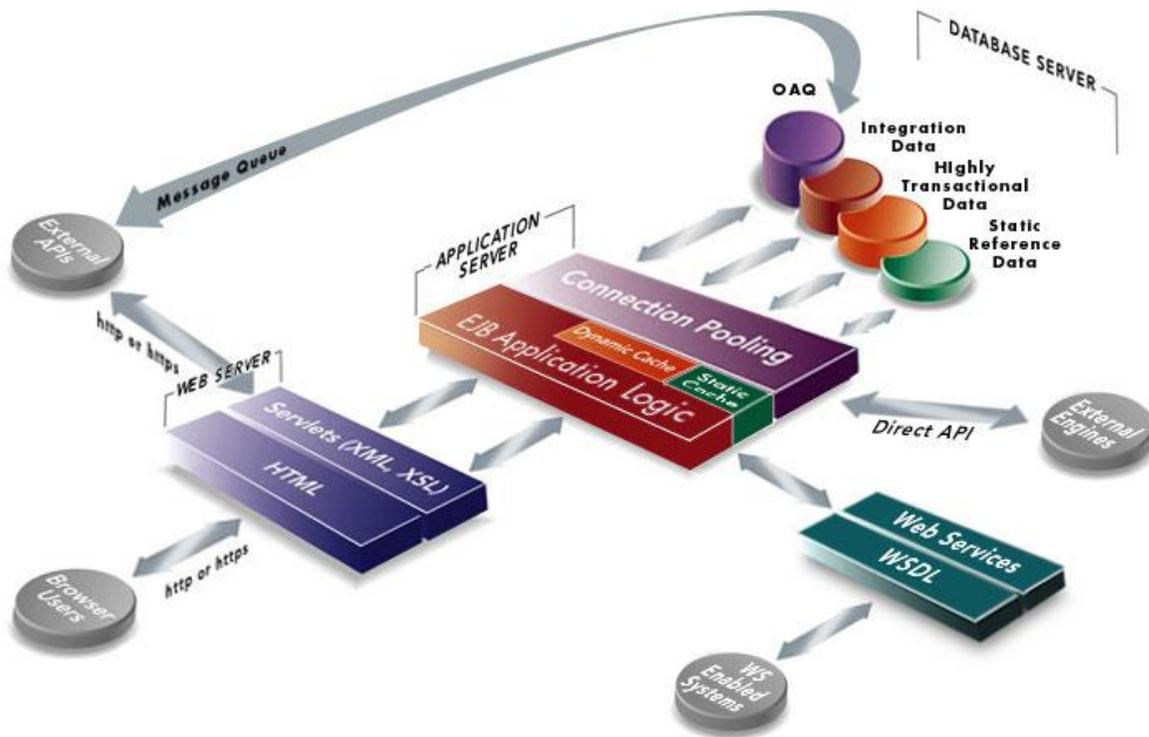
Each server is designed to perform specific functions that are briefly described in the following sections. From a user's perspective, all that you need to use to access Oracle Transportation Management is a standard Web browser such as Microsoft Internet Explorer or Mozilla Firefox.

Web Server

The web server controls the Oracle Transportation Management user interface and communicates with the application server to initiate the appropriate application logic. The Oracle Transportation

---

Management user interface is delivered to the user in HTML format and viewed using a browser such as Microsoft Internet Explorer or Mozilla Firefox.

The Oracle Transportation Management web server uses the Apache web server and the Tomcat Java servlet engine to manage the user interface and communications with the application server.

Application Server

The application server controls the application logic and communicates that logic between the web server and the Oracle Database. The application server runs Oracle WebLogic Server software or WebSphere Application Server from IBM (AIX only) to manage system-level details and operates at the center of the Oracle Transportation Management multi-tier architecture. In this architecture, business logic is executed in the application server, rather than in client applications. The resulting "thin" client, allows the client to manage the presentation layer, the application server to manage the business logic, and the back-end data services to manage the data.

> **Note:** WebSphere Application Server has not yet been certified with Oracle Transportation Management 6.2; there is no ETA on when this will be completed.

Database Server

The database server is the central repository for all the data entered in Oracle Transportation Management. The database server communicates with the application server to deliver the appropriate data to the user based on the Oracle Transportation Management application logic. The database server also communicates with the integration server to handle the import and export of Oracle Transportation Management data with other systems.

Oracle Transportation Management supports the following databases:

- Oracle 11gR2 (11.2.x on any supported platform)

Integration Server

The integration server is responsible for:

- Accepting Oracle Transportation Management XML from an Enterprise Application Integration (EAI) solution.
- Persisting XML transmission to a staging table.
- Accepting requests from the Oracle Transportation Management application server to build Oracle Transportation Management XML from staged data.
- Sending Oracle Transportation Management XML to an EAI solution.

The EAI solution is used as a secure data pipeline and data transformation engine and is responsible for:

- Sending and receiving EDI flat files to and from an EDI Van.
- Transforming EDI flat files to Oracle Transportation Management XML.
- Login security (communication is permitted only from registered IP addresses).
- Encryption security (using https).
- Guaranteed delivery (only if a partner server is on the remote side).

The integration server's job is to safely store XML transmissions and transform the XML transmissions into internal tables. The Oracle Transportation Management application server is responsible for initiating jobs that move data to and from staging tables and notifies the application server when new

data has arrived in the staging table. The application server must notify the Integration server when it has moved new data to the staging table.

The integration server does not perform data validation. Data validation occurs when data is transferred from the staging table to the application server tables. The staging table does not enforce referential integrity or constraints.

The Integration server and the application server communicate to the application server using Enterprise Java Beans (EJB) session bean calls. When an integration event occurs, the integration server notifies the application server using a session bean call. When the application service needs to communicate with the outside world, it does so directly, using an integration solution that provides a method to build transmissions and forward them to the EAI solution.

2. Configuring Optional Components

These components include:

- Locally-hosted Oracle MapViewer server
- Locally-hosted Oracle Spatial server
- Custom External Rating and Distance Engines
- GFI FAXmaker for Networks/SMTP 8.1
- RightFax
- PCMiler WorldWide & PCMiler Rail
- Rand McNally IntelliRoute Server
- SMC RateWare Server
- SMC RateWareXL Service
- Kewill FlagShip Server
- Python on a Client PC

Configuring Oracle Transportation Management for a Locally Hosted Oracle MapViewer

You can setup Oracle Transportation Management to integrate with a locally hosted MapViewer server during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use a locally hosted MapViewer server. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
# Locally-hosted MapViewer server
glog.map.baseurl=otmmapviewer.company.com
glog.map.service_name=otm_world
glog.map.basemap=otm_world.world_sample
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring Oracle Transportation Management for a Locally Hosted Oracle Spatial Instance

You can set up Oracle Transportation Management to integrate with a locally hosted Oracle Spatial server during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use a locally hosted Oracle Spatial server. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
# Locally-hosted Oracle Spatial server
OracleSpatial.host=otmspatial.company.com
OracleSpatial.port=7777
OracleSpatial.US_Canada.routeServlet=/routeserver/servlet/RouteServerServlet
OracleSpatial.WesternEurope.routeServlet=/routeserver_eu/servlet/RouteServerServlet
OracleSpatial.geocodeServlet=/geocoder/gcserver
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring Custom External Rating and Distance Engines

This section has been broken out into its own document. Please see the **External Programming Interface Guide** for more information.

Configuring FAXmaker for Fax Notifications

FAXmaker is an email-to-fax gateway that allows Oracle Transportation Management to send out fax notifications. Oracle Transportation Management sends an email via SMTP to a specific POP3 mailbox. FAXmaker checks this mailbox at regular intervals and converts the emails within to fax transmissions, if all security requirements are satisfied.

1. Install FAXmaker for Networks/SMTP 8.1 from GFI on a Microsoft Windows server and configure it to work with your modem(s). Refer to the FAXmaker documentation for details.

1. Test the modem to ensure that it gets a dial tone and can access an outside line.

2. Create a POP3 mailbox within your mail server that can be accessed by the FAXmaker server. The mailbox name must be identical to the email address for outgoing fax notifications that you entered during the Oracle Transportation Management application server install. Test this POP3 account using any email client that supports POP3. Please contact your System Administrators for more information as they will have greater insight into the makeup of your SMTP mail services.

3. Install the FAXmaker software according to the vendor's instructions.

4. Copy the Oracle Transportation Management fax template file (`<otm_install_path>\install\FAXmaker\G-Log.rtf` or `<otm_install_path>/install/FAXmaker/G-Log.rtf`) to FAX maker's coverpage directory (e.g. `C:\Program Files\FAXmaker\Coverpage\`).

5. Please note the following:
   a. Ensure that your modem is configured to send faxes ONLY; do not configure the modem to allow faxes to be received.
   b. Ensure that G-Log.rtf is the default coverpage and set the paper size to letter.
   c. Ensure that any special dialing requirements (such as dialing 7 to get an outside line) are set according to the vendor's documentation.
   d. The username and email address of the user you add to FAXmaker must be identical to the email address from which email notifications will come that you entered when you installed the Oracle Transportation Management application server. This allows emails that originate from this email address to be converted and faxed out. If an email is received from an email address that does not have an account, FAXmaker will bounce it.

6. Refer to the FAXmaker documentation for any troubleshooting assistance or contact GFI directly.

You can setup Oracle Transportation Management to integrate with FAXmaker during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use FAXmaker. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
# FaxMaker Settings – optional
glog.workflow.notify.faxmaker.email=fax@company.com
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring RightFax for Fax Notifications

RightFax is an email-to-fax gateway that allows Oracle Transportation Management to send out fax notifications. The Oracle Transportation Management application sends an email via SMTP to a specific POP3 mailbox. RightFax checks this mailbox at regular intervals and converts the emails within to fax transmissions, if all security requirements are satisfied.

1. Install RightFax 8.5 from Captaris on a Microsoft Windows 2000 server and configure it to work with your modem(s). Refer to the RightFax documentation for details.

2. Test the modem to ensure that it gets a dial tone and can access an outside line.

3. Create a POP3 mailbox within your mail server that can be accessed by the RightFax server. The mailbox name must be identical to the email address for outgoing fax notifications that you entered during the Oracle Transportation Management application server install. Test this POP3 account using any email client that supports POP3. Please contact your System Administrators for more information as they will have greater insight into the makeup of your SMTP mail services.

4. Install RightFax according to the vendor's installation guide and configure it to poll the POP3 mailbox created above.

You can setup Oracle Transportation Management to integrate with RightFax during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use RightFax. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
# RightFax Settings - optional
glog.fax.defaultHandler.routingMode=to
glog.fax.defaultHandler.routingPicture=/name={lastName},{firstName}/fax
={rawPhone}/
glog.fax.email=fax@company.com
```

Once this change has been made, restart your Oracle Transportation Management instance.

Fax Generation in Oracle Transportation Management

Fax generation is controlled via a plug-and-play interface, FaxHandler. Classes that implement this interface support the following two methods:

```
public MailAddress getAddress();
```
// returns the mail address for fax generation

```
public void handle(MailEvent event, FaxTopic faxTopic);
```
// modifies our standard text e-mail to conform to outgoing fax
// specifications.

The fax handler is specified via the property glog.fax.handler=<fax handler class name>

Oracle Transportation Management supplies a standard fax handler, glog.server.workflow.notify.DefaultFaxHandler, to support most fax generation software. This handler embeds routing information into one of the following 3 mail components:

- message body

- subject line
- to alias (i.e. the last name of the to party)

The embedding is accomplished with a picture string describing the modified format of the above component. You can specify original information from the text email to use in the modified component including:

- firstName
- lastName
- company
- phone
- rawPhone (punctuation or space)
- message
- subject

Replacements are represented by one of these identifiers surrounded by parenthesis; for example, {firstName}. Newline characters are preserved. To transform an email message body to one supporting FAXMaker, the picture string would be:

```
::{firstName},{company},{lastName},,{rawPhone}\n{message}
```

The following properties control the default fax handler:

```
glog.fax.defaultHandler.routingMode=<message|body|to>  (default:
message)
glog.fax.defaultHandler.routingPicture=<picture string>
     (default:
::{firstName},{company},{lastName},,{rawPhone}\n{message})
```

For RightFax, the properties default to:

```
glog.fax.defaultHandler.routingMode=to
glog.fax.defaultHandler.routingPicture=/name={lastName},
{firstName}/fax={rawPhone}/
glog.fax.email=fax@company.com
```

For FAXMaker, they default to:

```
glog.fax.defaultHandler.routingMode=message
glog.fax.defaultHandler.routingPicture=::{firstName},{company},{lastNam
e},, {rawPhone}\n{message}
glog.fax.email=fax@company.com
```

Configuring PCMiler WorldWide or PCMiler Rail

Install the PCMiler product as described in your PCMiler user's manual. Install the PCMiler Canada Postal Codes, if desired (PCMiler WorldWide only).

You can setup Oracle Transportation Management to integrate with PCMiler during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use PCMiler.

> **Note**: PCMiler must be configured to return its distance in miles.

To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines

and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
pcmiler.host=pcmiler.company.com
pcmiler.port=8145
```

Uncomment and modify the following lines for PCMiler Rail:

```
pcmiler.rail.host=pcmilerrail.company.com
pcmiler.rail.port=2001
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring Rand McNally IntelliRoute

> **Note:** Due to lack of support for 64-bit Java Development Kit (JDK), IntelliRoute is not currently supported as of the time of this writing. Please contact Rand McNally for an ETA on when it will be available.

Install the Rand McNally IntelliRoute server as described in your IntelliRoute user's manual. You will also need to create a user (e.g. otm), password, and location for the Oracle Transportation Management application to use. These activities are described in the IntelliRoute user's manual. Once the server is installed, you need to install the IntelliRoute Java API onto the Oracle Transportation Management application server as described in your IntelliRoute user's manual.

You can setup Oracle Transportation Management to integrate with IntelliRoute during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use IntelliRoute. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
intelliroute.host=otmir.company.com
intelliroute.port=1998
intelliroute.user=otm
intelliroute.password=changeme
intelliroute.location=company1
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring Rand McNally MileMaker

Install the MileMaker server as described in your user's manual.

You can setup Oracle Transportation Management to integrate with MileMaker during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use MileMaker. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
milemaker.host=otmmm.company.com
milemaker.port=1031
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring SMC RateWare

Install the SMC RateWare server and load the CZAR tariffs as described in your RateWare user's manual.

You can setup Oracle Transportation Management to integrate with RateWare during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use RateWare. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
glog.RatingEngine.Rateware.URL=otmsmc.company.com
glog.RatingEngine.Rateware.port=23700
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring SMC RateWareXL Service

You can set up Oracle Transportation Management to integrate with the RateWareXL Service during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use the RateWareXL Service. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
# external SMC RatewareXL - optional
glog.RatingEngine.RatewareXL.Username=SomeUserName
glog.RatingEngine.RatewareXL.Password=SomePassword
glog.RatingEngine.RatewareXL.License=qwerty12345
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring Kewill FlagShip

Install the Kewill FlagShip server as described in your FlagShip user's manual.

You can setup Oracle Transportation Management to integrate with Kewill FlagShip during the initial Oracle Transportation Management installation (as described earlier), or you can modify an existing installation to use Kewill FlagShip. To modify an existing Oracle Transportation Management installation, you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is usually under the `<otm_install_path>/glog/config` directory. Uncomment and modify the following lines and set the values on the right as appropriate (see the relevant section of the Oracle Transportation Management installation above for details on what each field's value should be):

```
glog.RatingEngine.Kewill.URL=otmkewill.company.com
glog.RatingEngine.Kewill.port=1200
glog.RatingEngine.Kewill.RefnumQual=KEWILL_CUST_NUM
```

Once this change has been made, restart your Oracle Transportation Management instance.

Configuring Python on a Client PC

This is a prerequisite to using the ClientUtil.py for exporting/importing from a remote Oracle Transportation Management instance, or for using sql2xml.py or xml2sql.py for exporting/importing from a database for which you have sql*net access.

1. Download Python 2.5 from the Python website (http://www.python.org/) and install it according to their install instructions.

2. Set the following environment variables according to the instructions for your OS (e.g. under Windows you would go to Start > Settings > Control Panel > System > Environment), where $PYTHON_INSTALL_DIR is the directory where Python was installed (e.g. D:\product\python):

```
PYTHON_HOME=$PYTHON_INSTALL_DIR
PYTHON_LIB=$PYTHON_HOME/lib/python2.5
PYTHONPATH=$PYTHON_LIB:$PYTHON_LIB/site-packages:$PYTHON_LIB/site-
packages/_xmlplus:$GLOG_HOME/utils/integration/python
```

You are now ready to use the Oracle Transportation Management client-side python scripts. Please refer to the Oracle Transportation Management Integration Guide and Oracle Transportation Management Data Management Guide for more information.

Python International Character Set Configuration

The site.py file under $PYTHON_LIB must be modified to change encoding = ascii to encoding = utf-8.

Note: Failure to make the above change will cause international characters to be garbled.

Configuring Outbound Connections to use Proxy Servers

Oracle Transportation Management can be configured to allow outgoing connections to be made using a non-authenticating HTTP proxy server. Once the proxy server is defined, you must define which connections will not go through the proxy server, as the default will be to use whatever is defined via the properties for all connections.

Note: For performance reasons it is highly recommended that there are no firewalls or proxy servers sitting between the main components of your Oracle Transportation Management architecture (i.e. WEB, APP & DB).

In order to set up to use a proxy server you need to edit the glog.properties file on the Oracle Transportation Management application server. This file is in the `<otm_install_path>/glog/config` directory. List the proxy server name and port number in the following two properties:

```
glog.integration.http.proxyHost=otmproxy.company.com
glog.integration.http.proxyPort=8080
```

The value for `proxyHost` can be a server name or an IP address. The value for `proxyPort` must be a number. Then, define which servers should not go through the proxy server. Multiple values can be specified, separated by a '|' character:

```
glog.integration.http.client.nonProxyHosts=internal-mail-
host.company.com|*.local.company.com|192.168.101.*
```

As shown in the example above, an '*' may be used as a wildcard in machine names and IP addresses.

This will cover all protocols other than FTP, which has its own set of properties:

```
glog.integration.ftp.proxyHost=otmproxy.company.com
```

```
glog.integration.ftp.proxyPort=8080
glog.integration.ftp.nonProxyHosts=integration.company.com|*.local.comp
any.com
```

Here again you may indicate which machines should not go through the proxy server when using FTP.

3. Starting and Stopping the Oracle Transportation Management Servers

Starting Oracle Transportation Management Servers

To start Oracle Transportation Management, please do the following:

1. Start the Oracle Transportation Management database, including any listeners
2. Start the Oracle Transportation Management web server
    a. On Microsoft Windows:
        i. Start the "Server Manager"
        ii. Expand "Configuration" and then click on "Services"
        iii. Locate the Oracle Transportation Management Tomcat service (e.g. otmwebtomcat62), right-click on it and select "Start"; wait 10 seconds before proceeding to the next step
        iv. Locate the Oracle Transportation Management Apache service (e.g. otmweb62), right-click on it and select "Start"
    b. On UNIX systems:
        i. Start the Oracle Transportation Management web daemon (e.g. /etc/init.d/otmweb62 start)
3. Start the Oracle Transportation Management application server
    a. On Microsoft Windows:
        i. Start the "Server Manager"
        ii. Expand "Configuration" and then click on "Services"
        iii. Locate the Oracle Transportation Management WebLogic service (e.g. otmapp62), right-click on it and select "Start"
    b. On UNIX systems:
        i. Start the Oracle Transportation Management app server daemon (e.g. /etc/init.d/otmapp62 start)
    c. It can take several minutes for the application server to fully start up, depending on your hardware
    d. **[WebLogic]** You can monitor the <otm_install_path>/logs/weblogic/console.log.0 file to determine when WebLogic has been fully initialized. Look for lines similar to the following:

```
INFO | 2009/03/09 23:26:12 | <Mar 9, 2009 11:26:12 PM EDT> <Notice>
<Server> <BEA-002613> <Channel "Default" is now listening on
10.143.205.28:7001 for protocols iiop, t3, ldap, http.>
INFO | 2009/03/09 23:26:12 | <Mar 9, 2009 11:26:12 PM EDT> <Notice>
<WebLogicServer> <BEA-000329> <Started WebLogic Admin Server "gc3-
otm-qa-62-wl" for domain "Otmv620" running in Production Mode>
INFO | 2009/03/09 23:26:12 | <Mar 9, 2009 11:26:12 PM EDT> <Notice>
<WebLogicServer> <BEA-000365> <Server state changed to RUNNING>
INFO | 2009/03/09 23:26:12 | <Mar 9, 2009 11:26:12 PM EDT> <Notice>
<WebLogicServer> <BEA-000360> <Server started in RUNNING mode>
INFO | 2009/03/09 23:26:26 | -- OTM Event: serverReadyc
```

3-1

e. **[WebSphere]** You can monitor the
<otm_install_path>/logs/websphere/console_out.log file to determine when
WebSphere has been fully initialized. Look for lines similar to the following:

```
[8/4/06 10:00:16:809 EDT] 0000000a WsServerImpl  A   WSVR0001I:
Server server1 open for e-business
[8/4/06 10:02:45:720 EDT] 00000041 SystemOut     O End startup, GC3
is ready
```

4. To check to see if everything has started up, point to http://<web server name> in your web
browser and log in.

    Note: While the application server is starting up, the web server will respond to all
    requests with a '503 (Service Unavailable)' error message. This will go away once the
    application server has fully started up.

Stopping Oracle Transportation Management Servers

To start Oracle Transportation Management, please do the following:

1. Stop the web server
    a. On Microsoft Windows:
        i. Start the "Server Manager"
        ii. Expand "Configuration" and then click on "Services"
        iii. Locate the Oracle Transportation Management Apache service (e.g.
            otmweb62), right-click on it and select "Stop"
        iv. Locate the Oracle Transportation Management Tomcat service (e.g.
            otmwebtomcat62), right-click on it and select "Stop"
    b. On UNIX systems:

    ```
    Stop the Oracle Transportation Management web daemon (e.g.
    /etc/init.d/otmweb62 stop)
    ```

2. Stop the application server
    a. On Microsoft Windows:
        i. Start the "Server Manager"
        ii. Expand "Configuration" and then click on "Services"
        iii. Locate the Oracle Transportation Management WebLogic service (e.g.
            otmapp62), right-click on it and select "Stop"
    b. On UNIX systems:
        i. Stop the Oracle Transportation Management application daemon (e.g.
            `/etc/init.d/otmapp62 stop`)

3. Stop the Oracle database, if necessary.

If you are going to restart Oracle Transportation Management, wait one minute between stopping and
restarting the servers. This gives the application server the time it needs to fully shut down.

4. Advanced Configuration: UI / Web Server

Note: Default paths are used below and may differ from your configuration.

Note: Whenever you edit the Oracle Transportation Management property files, only edit the glog.properties file. Any changes that you make to any other files will be overwritten every time an update is applied.

Browser Cookies Used in Oracle Transportation Management

The following browser cookies are used in Oracle Transportation Management.

Cookie Name	Personally Identifiable Information	Retention Policy	Effect of Refusal	Usage
JSESSIONID	There is no personally identifiable information collected or stored in this cookie.	The cookie only lasts as long as the browser is open; once the browser closes the cookie is discarded.	Oracle Transportation Management will not work without this cookie.	This cookie stores the Tomcat session ID, a random 32-character string.

Creating and Installing Secure Socket Layer (SSL) Certificates

Oracle Transportation Management is delivered with demonstration SSL certificates for Apache (using mod_ssl and OpenSSL). To run these servers in a production environment, you should replace these with your own real certificates, verified by a known Certificate Authority, such as Verisign, Thawte, etc. In a high-traffic SSL environment, you should use hardware SSL accelerators as they simplify certificate management, increase SSL performance, and greatly reduce the load on your servers.

Apache

1. Open a command prompt and change into `<otm_install_path>/apache/openssl/bin`, where `<otm_install_path>` is where you installed the Oracle Transportation Management software (Microsoft **Windows**: change into `<otm_install_path>\apache\bin` instead).

2. Create a RSA private key for your Apache server (it will be Triple-DES encrypted and PEM formatted).

   ```
   openssl genrsa -des3 -out server.key 1024
   ```

3. Enter your PEM pass phrase. This will be a password that you create. Make a backup copy of the server.key and remember your pass phrase (you will need it). You can see the details of the RSA private key with this command:

   ```
   openssl rsa -noout -text -in server.key
   ```

4. You can also create a decrypted PEM version of the RSA private key with this command:

   ```
   openssl rsa -in server.key -out server.key.unsecure
   ```

5. Create a Certificate Signing Request (CSR) with the server RSA private key (output will be in PEM formatted):

   ```
   openssl req -new -key server.key -out server.csr -config openssl.cnf
   ```

6. Please answer all of the questions to the best of your ability, especially the common name (your company's domain name).

7. You can view the details of your CSR via this command:

```
openssl req -noout -text -in server.csr
```

8. You now have to send this Certificate Signing Request (CSR) to a Certifying Authority (CA) for signing. The result is then a real certificate, which can be used for Apache. You now need to transfer the CSR to a commercial CA like Verisign or Thawte. Usually you will need to post the CSR into a web form, pay for the signing, and await the signed certificate that you then store into a server.crt file.

9. When you receive the file back from the Certifying Authority you will copy this file into <otm_install_path>/apache/conf/ssl.crt. Also copy server.key from <otm_install_path>/apache/openssl/bin to <otm_install_path>/apache/conf/ssl.key. The server.key file was created earlier.

10. Now you have two files: server.key and server.crt. These files can be used now. Edit your apache configuration file <otm_install_path>/apache/conf/httpd.conf. Scroll to almost to the end of the file and you will see two lines:

```
SSLCertificateFile  conf/ssl.crt/demo.cert
SSLCertificateKeyFile conf/ssl.key/demo.key
```

11. Replace demo.cert with the file name server.crt and replace demo.key with the file name server.key. Make sure that you put the file names at the end of the path. Notice server.crt is for SSLCertificateFile and server.key is for SSLCertificateKeyFile.

The Server.csr file is no longer needed.

12. Restart Apache for these new settings to take effect.

Enabling Automatic Web UI Login

Oracle Transportation Management can remember a user's login ID and password and auto-fill these fields. This feature stores the hashed login ID and password in a cookie within the end-user's web browser. This feature may present a security concern if enabled.

To enable, do the following:

13. Edit the `<otm_install_path>/glog/config/glog.properties` file on your web servers and add the following property:

```
glog.webserver.login.remember=true
```

14. Restart Oracle Transportation Management for this to take effect. The "Remember Me" check box will now appear on the login screen and browser-based users can selectively enable this feature for their login.

Enabling Reverse-Proxy Support

Oracle Transportation Management supports some reverse-proxy solutions. The reverse-proxy is expected to identify sites based on a URL-prefix (such as /otm) and then strip that prefix off before forwarding the URL to Oracle Transportation Management. Every link that Oracle Transportation Management sends to the browsers will contain the defined URL-prefix.

To enable reverse-proxy support, do the following:

1. Edit the `<otm_install_path>/glog/config/glog.properties` file on your web and application servers and modify the line to include the desired URL-prefix (ex. /otm):

```
glog.webserver.urlprefix=
```

2. Edit the index.htm file under `<otm_install_path>/apache/htdocs`. The following line needs to contain the URL-prefix:

```
<META http-equiv="refresh" content="0;
URL=/servlets/glog.webserver.servlet.umt.Login"/>
```

- For example, if your URL-prefix is /otm, it would look like:

```
<META http-equiv="refresh" content="0;
URL=/otm/servlets/glog.webserver.servlet.umt.Login"/>
```

3. Ensure that the `glog.webserver.URL` setting in your `glog.properties` files points to your reverse-proxy server, rather than the individual web server(s).

4. Restart Oracle Transportation Management to enable these changes. Oracle Transportation Management can now be accessed through the reverse-proxy, but may not be accessible directly. This is a limitation due to adjusting links to work with the reverse-proxy.

Enabling Single Sign On (SSO) Support

Oracle Transportation Management supports SSO, where a central application (the SSO provider) authenticates users and then passes the login information to Oracle Transportation Management, therefore bypassing the normal Oracle Transportation Management login process. *The burden of authentication then falls to the SSO provider -- Oracle Transportation Management will not provide any.*

For SSO access to Oracle Transportation Management, the invoking code needs to pass in the following parameter as part of the HTTP request:

```
appuid=/GUEST.ADMIN/
```

where `/GUEST.ADMIN/` is the user's GID. This can also be passed in as part of the HTTP header (see below to control this behavior). By default, the logic checks the header first and then the request.

There are several properties that control SSO. They must be set in the glog.properties file on the web server:

- `glog.security.sso=true`: is SSO allowed or not? It is false by default
- `glog.security.sso.appUidName=appuid`: to change the name of the UID field
- `glog.webserver.initial_page=url`: used if a redirect is not provided as part of the request. url can either be an Oracle Transportation Management servlet or a fully qualified URL:
    o `$glog.webserver.urlprefix$$glog.webserver.context$glog.webserver.util.FrameGC3Servlet`
    o `http://some.domain.com/some_page.html`
- `glog.security.sso.appUidLocation=X`: where X is one of the following:
    a. 1 (default): check HTTP header & then request parameter for the user name
    b. 2: check request parameter only
    c. 3: check HTTP header only

Hiding/Removing Oracle Transportation Management Online Help

The following directions need to be done for each Oracle Transportation Management web instance. Different instances can have different things done to them, e.g. help could be removed from the DMZ web instance, but left alone on the internal web instance.

Note: These changes should first be made on a test instance.

Note: All changed files should be backed up before proceeding.

Removing Online Help

Removing Oracle Transportation Management's online help can be accomplished by renaming or deleting the following directory:

```
<otm_install_directory>/apache/htdocs/html/help/webhelp
```

where "`<otm_install_directory>`" is the directory into which Oracle Transportation Management was installed. Users will now see a "Not Found" error message; this could be changed to a nicer message indicating online help is not available by re-creating the file:

`<otm_install_directory>apache/htdocs/html/help/webhelp/XX/gc3_help.htm` where XX is a 2-letter language code (e.g. "en" for English). This should be done for each localized language.

Hiding Online Help

Oracle Transportation Management's online help can be hidden or shown based upon the origin of the request to see the online help. For example, users coming from a particular machine (e.g. a firewall or reverse proxy server) could be denied access to online help, or users on a particular network subnet (e.g. internal users) could be granted access to online help.

There are two basic paradigms in this case: allow everyone to access the help by default, and restrict only some users; deny access to everyone by default, opening access back up for some users. Accomplishing either of these is done by editing the Apache configuration file and restarting the Oracle Transportation Management web instance. An example of each follows; in the examples, "`<otm_install_directory>`" refers to the directory into which Oracle Transportation Management was installed.

Restricted users will receive a "403 Forbidden" message. Please see the Apache documentation on the "ErrorDocument" directive for information regarding how to present a nicer error message; the `ErrorDocument` directive could be used inside of the `<Directory></Directory>` blocks shown below.

Allow Everyone to Access Help By Default, Restrict Specific Traffic

1. Edit the file `<otm_install_directory>/apache/conf/apache.conf`
2. Locate the line: `<Directory "<otm_install_directory>/apache/htdocs">`
   a. Add the following code immediately before the <Directory> line:

```
<Directory "<otm_install_directory>/apache/htdocs/html/help">
      Options None
      AllowOverride None
      Order allow, deny
      Deny from 192.168.1.2
      Allow from all
</Directory>
```

where 192.168.1.2 is the IP address of the machine from which you wish to restrict access. This could also be a partial IP address if you want to restrict access from a particular subnet (e.g. 192.168.1 to restrict access from any machine with an IP address in the range 192.168.1.1 - 192.168.1.254).

3. Restart the Oracle Transportation Management web instance.

Deny Everyone Access to Help By Default, Allowing Specific Traffic

1. Edit the file <otm_install_directory>/apache/conf/apache.conf
2. Locate the line: <Directory "<otm_install_directory>/apache/htdocs">
   a. Add the following code immediately before the <Directory> line:

```
<Directory "<otm_install_directory>/apache/htdocs/html/help">
      Options None
      AllowOverride None
      Order deny, allow
      Deny from all
      Allow from 192.168.1.2
</Directory>
```

where 192.168.1.2 is the IP address of the machine from which you wish to allow access. This could also be a partial IP address if you want to allow access from a particular subnet (e.g. 192.168.1 to allow access from any machine with an IP address in the range 192.168.1.1 - 192.168.1.254).

3.   Restart the Oracle Transportation Management web instance

Installing Multiple Web Servers

Multiple web servers can be utilized to increase the performance of Oracle Transportation Management. Generally, with more web servers you can maintain more simultaneous user connections into the Oracle Transportation Management server's web interface. You may also see increased performance in integration, since incoming integration files are posted to a servlet on the web server and are passed back to the application server. We recommend using a hardware load-balancer to spread the incoming requests among multiple web servers. Hardware solutions are available from many vendors including Alteon, Cisco, F5, and others.

Installing Translations

Under UNIX systems:

1.   Log in as the Oracle Transportation Management user.
2.   Change the directory to `<otm_install_path>/glog/oracle/script8`.
3.   Run the following command: `install_lang.sh <lang id> [<country id>]`.

Under Microsoft Windows:

1.   Log in as the Oracle Transportation Management user.
2.   Start > Run… > `cmd`.
3.   Change the directory to `<otm_install_path>/glog/oracle/script8`.
4.   Run the following command: `install_lang.sh <lang id> [<country id>]`

`where valid <lang id> and <country id> values are:`

Language	Language ID	Country ID
Chinese (Simplified)	zh	CN
Chinese (Traditional)	zh	TW
French (Canadian)	fr	CA
French (European)	fr	
German	de	
Italian	it	

Language	Language ID	Country ID
Japanese	ja	
Korean	ko	
Portuguese (Brazillian)	pt	BR
Russian	ru	
Spanish (European)	es	
Spanish (Latin American)	es	MX

To have notifications translated, the following property will need to be set in the `<otm_install_path>/glog/config/glog.properties` file:

```
glog.notify.localizeStylesheet=true
```

Modifying the Session Timeout

The default session timeout for the Oracle Transportation Management web server is 30 minutes. If you leave Oracle Transportation Management idle for 30 minutes, your session will timeout and you will need to log in again. For some Oracle Transportation Management installations, you may want to increase this timeout. Please note that increasing the session timeout will increase the load on your web server and may decrease the number of simultaneous users that can access the system. We don't recommend setting this above 60 minutes.

To change this setting, edit the web.xml files on your Oracle Transportation Management web server. These files are located on your web server(s): `<otm_install_path>/tomcat/conf/web.xml` and `<otm_install_path>/glog/gc3webapp/WEB-INF/web.xml`. Edit the following line in each file:

```
<session-timeout>30</session-timeout>
```

Change the setting from 30 to your desired timeout period, in minutes. Restart the Oracle Transportation Management instance as described in the chapter **Starting and Stopping the Oracle Transportation Management Servers**.

5. Advanced Configuration: Database

Note: Default paths are used below and may differ from your configuration.

Note: Whenever you edit the Oracle Transportation Management property files, only edit the glog.properties file. Any changes that you make to any other files will be overwritten every time an update is applied.

Configuring Oracle Transportation Management to Use a Different Database

At some point, you may need to point your Oracle Transportation Management instance to another database for testing, development, or fail-over purposes. Follow the directions below to configure Oracle Transportation Management to use this new database. It is assumed that the new Oracle Transportation Management database has already been created and populated with data using the database installation directions earlier in the Installation Guide. The database should also be at the same patch level as your Oracle Transportation Management instance to prevent any incompatibility problems.

1. Shutdown your Oracle Transportation Management web and application servers as described earlier in this guide.

2. Edit the file `<otm_install_path>/glog/config/glog.properties` and change the following properties. This should be done on each Oracle Transportation Management web and application server.

   ```
   dbserver=<db_server_fqdn>
   glog.database.sid=<oracle_sid>
   glog.database.connectstring=<oracle_connectstring>
   ```

3. **[WebSphere]** Edit the file
   `<otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01
   Cell/nodes/__APP_SERVER__Node01/resources.xml` (where __APP_SERVER__ is your application server's host name) and change the following line under resources.jdbc:JDBCProvider:

   ```
   <resourceProperties xmi:id="J2EEResourceProperty_1141837349928"
   name="URL" type="java.lang.String"
   value="jdbc:oracle:thin:@[DB_FQDN]:[DB_PORT]:[DB_SID]"
   description="This is a required property. The URL indicating the
   database from which the Data Source will obtain connections, such as
   'jdbc:oracle:thin:@localhost:1521:sample' for thin driver and
   'jdbc:oracle:oci8:@sample' for thick driver." required="true"/>
   ```

   Note: only change the values `[DB_FQDN]:[DB_PORT]:[DB_SID]`

4. Restart your Oracle Transportation Management web and application servers as described earlier in this guide.

5. You may also need to edit your tnsnames.ora files on each Oracle Transportation Management server.

International Characters in SQL*Plus Windows

Oracle Transportation Management supports multiple languages. However, you can only display characters for one language at a time. The following needs to be done to properly configure this application to work in another language:

- Specify NLS_LANG as a system or environment variable on your client. The NLS_LANG parameter has three components: language, territory and charset. Examples of NLS_LANG settings are:

  NLS_LANG = AMERICAN_AMERICA.WE8ISO8859P1
  NLS_LANG = JAPANESE_JAPAN.JA16EUC

  For more information on NLS_LANG see the Oracle National Language Support Guide.

  Important: The character set part of your NLS_LANG setting needs to correspond to your client application character set. For example, in an English Microsoft Windows environment, the usual character set of GUI applications is the operating system code page 1252, which corresponds to the Oracle character set WE8MSWIN1252. Therefore you would set your NLS_LANG to something like AMERICAN_AMERICA.WE8MSWIN1252. An incorrect NLS_LANG setting in this case would be AMERICAN_AMERICA.WE8ISO8859P15.

- If characters still do not display properly, then you should switch to a font that contains support for the characters you need. You can find more information on how to do this from Oracle. For example, http://otn.oracle.com/docs/tech/sql_plus/content.html.

  Note: To find out which font supports the characters/script you need to use the Character Map utility on Microsoft Windows Server.

Sorting

Proper sorting in languages other than English can currently not be done via the Oracle Transportation Management web interface. However, language-sensitive sorting of Oracle Transportation Management data is possible when you have direct access to the Oracle Transportation Management database with applications such as Toad and SQL*Plus by setting the desired sort order as an ALTER SESSION parameter. For example, by issuing the command ALTER SESSION SET NLS_SORT = 'SWEDISH'; the sort order for the current session will be changed to Swedish.

Replicated Operational Database Maintenance

After an upgrade or patch, the Replicated Operational Database (ROD) will not automatically be aware of new Oracle Transportation Management tables. You only need to run this when you complete all the upgrades on the Online Transaction Prcoessing Database (OLTP); in other words, if you are upgrading through two versions on the OLTP in one weekend, complete those upgrades first, and then upgrade the ROD.

Run the following command to create logs for new tables. It will skip tables that already have a log.

1. On the OLTP database in SQL*Plus, as user **GLOGOWNER** run:

    `@create_mview_logs`

2. On the OLTP database in SQL*Plus, as user **REPORTOWNER** run:

    `@create_mview_logs`

3. On the ROD database in SQL*Plus, as **GLOGOWNER** run:

    `@pre_dbupdate_rod.sql`

This script checks if any job is running and marks it as broken. If there are running jobs, you should see a statement like "Refresh job 504 is running. Please ask DBA to bounce the database". Once the database is restarted, no jobs will be running, which allows the next step to run much faster. If no refresh job is running, this script will say "No refresh job is running now. Please go ahead to migrate the rod database now".

Please note that all broken jobs will be enabled in step 3 during the execution of dbupdate_rod.sql.

4.   On the ROD database in SQL*Plus, as **GLOGOWNER** run:

```
@dbupdate_rod.sql
```

This script will build (or rebuild) the materialized views for any new/modified tables so that materialized views in ROD database are in sync with tables in OTLP database. Note that script dbupdate_rod will not refresh materialized views after they are build/rebuild. Materialized views will be refreshed later by refreshing jobs at their scheduled times.

Rebuilding an Existing Materialized View

If you want the new columns of a particular table to be regenerated, execute the following:

```
EXEC pkg_rod.build_mview('table_name');
```

It will drop and recreate the materialized view, empty. You can wait until the next refresh for it to populate, or you can do the following:

```
EXEC pkg_rod.refresh_one('table_name');
```

If you are recreating several materialized views, you may want to kick off the refresh for all views manually by executing the following:

```
EXEC pkg_rod.refresh_all;
```

Copy Database Data

When moving from initial implementation to production, or duplicate production to development, there are needs to copy the whole database or part of Oracle Transportation Management data between production and development/test environment. Depending on the purpose and requirement, copying an Oracle Transportation Management database/data can be achieved using different methods. It is recommended that Support be consulted to decide the best way of moving data.

Copy Database Files

An Oracle Database can be cloned by copying datafiles and other configuration files to the destination environment (usually another server). Procedures of this method can be found from Oracle Database support/document.

Features/Limitations:

- Simple and straightforward. There is no need to pre-create an Oracle Database in the destination machine.
- Fast. The total time needed is governed by physically transferring datafiles from source machine to destination machine.
- Whole database copy. If there are data in the source database that the client does not want to copy to the destination database, then data cleaning process should be carried out in the destination database after the copy.
- Source and destination servers must be the same platform, same operating system and same Oracle Database version between.
- Requires DBA skills.
- Requires down time in the source database if there is no hot-backup.

5-3

Oracle Database Export/Import Utility

Using the Oracle Database export/import method, the entire Oracle Transportation Management database structures can be copied from one database to another. To successfully copy an Oracle Transportation Management database, following guidelines are recommended:

1. Create all tablespaces used by Oracle Transportation Management first in the target database.
2. Check if all of Oracle Transportation Management required init.ora parameters are properly set in the target database.
3. Oracle Transportation Management database roles and schema users should be created in the target database before the import. This should be done by running Oracle Transportation Management supplied SQL scripts as described below.
4. Log in to the target database as user **SYS**.

    ```
    @create_glog_users.sql  -- Make sure no errors
    ```

5. Perform export on the source database. The easiest way is to do a FULL export. But if for any reason you do not want a FULL export, you can do a schema export. Oracle supplies a schema export parameter file (exp_GC3.par), where it lists all schemas the Oracle Transportation Management application needs.
6. Perform import on the target database. You can do a FULL import if the export was a FULL export. Alternatively you can do a schema import. Oracle supplies a schema import parameter file (imp_GC3.par), where it lists all schemas the Oracle Transportation Management application needs.
7. After the import finished successfully, there are post-import SQL scripts to be run in the target database. The steps to run these scripts are described in the Installation Guide in the Database Server section. It is important to follow the instructions to run the scripts. Ensure you log in to the database as the correct users when running the scripts.

Features/Limitations:

*   Can be used between different platforms and different operating systems.
*   Oracle Database version in the destination environment must be equal or higher than that of source environment.
*   Need to pre-create an Oracle Database in the destination environment.
*   Whole Oracle Transportation Management database copy. If there are data in the source database that the client does not want to be copied to the destination database, then data cleaning process should be carried out in the destination database after the import.
*   May take a long time to finish depending on the size of the database.

Domain Copy

Oracle Transportation Management provides utilities to copy domains between Oracle Transportation Management databases. There are two approaches to copy domains. The first one uses a PL/SQL procedure to generate INSERT statements with domain data to be copied from the source database. The insert statements can be run in the target database. The second approach uses Oracle Database TABLE mode export/import with WHERE clause to move domain data between source and destination databases. Please refer to the Oracle Transportation Management document titled "Data Management Guide" for details of this method.

Features/Limitations:

*   Can be used between different platforms and different operating systems.
*   Oracle Database version in the target environment must be equal or higher than that in the source environment if using export/import approach.

- Requires pre-creating an Oracle Database and installing an Oracle Transportation Management database in the target environment.
- Oracle Transportation Management table structures must be identical between the source and target database.
- Can just move one or more domain data between the databases.

6. Advanced Configuration: General

Note: Default paths are used below and may differ from your configuration.

Note: Whenever you edit the Oracle Transportation Management property files, only edit the glog.properties file. Any changes that you make to any other files will be overwritten every time an update is applied.

General Instructions on Upgrading Oracle Transportation Management from One Release to Another

Note: These instructions do not apply when simply applying a new roll-up.

Unless otherwise specified in release-specific instructions, please follow these steps when upgrading to a new Oracle Transportation Management version:

1. Upgrade and migrate your data to the latest RU of your current version of Oracle Transportation Management.
2. Install the new Oracle Transportation Management version.
3. Install the latest RU of the new version.
4. Migrate your data from the old version to the new version according to the new version's Installation Guide.
5. Run any RU-specific instructions that come out of step #3.

This will ensure that you have the most up-to-date copies of the migration scripts in place before running the migration itself. For example:

- Currently Version: 5.5.05.03
- Target Version: 6.0.3

Steps

1. Install and migrate to 5.5.05.07.
2. Install 6.0.
3. Install 6.0.3.
4. Run 6.0 migration scripts according to latest Migration Guide.
5. Run any 6.0.3-specific instructions.

Branding

Oracle Transportation Management upgrades typically include updated web files (XSL, html, jpg, gif, JS, etc.) to fix specific reported problems and to provide minor enhancements requested by customers.

Since each Oracle Transportation Management upgrade includes a list of updated files, you should apply the upgrade to your test site, re-brand the updated files, and then test everything. Once it has passed your testing, you should apply the upgrade and your new branded files to your production site. This procedure ensures that you have all of the fixes included in the latest upgrade and that your branding is not broken on your production site.

Oracle Transportation Management now supports branding by domain. You should brand into the following path, which eliminates the problem of overwriting branding when Oracle Transportation Management upgrades are installed.

`<otm_install_path>/apache/htdocs/xsl/branded` subdirectory

Contact the Support group to receive patch notifications or to download the latest Oracle Transportation Management patch.

Adding, Removing or Changing Properties

The `glog.properties` file contains settings which control Oracle Transportation Management's behavior. You may be instructed by the Administration Guide or Support to set particular property values. The property files are organized in a hierarchical structure using "`!include`" statements. Property settings made in a file after the "`!include`" statements override the values that are specified in the included files. Therefore it is mandatory that all changes to properties be made only in the `glog.properties` file. It is also important to make all changes below the "`!include`" statements or they will not be used.

A copy of the `glog.properties` file resides on the web server and the application server in the `<otm_install_path>/glog/config` directory. When instructed to change a property value, you will be instructed to make changes to only the web server, only the application server, or both. Some more advanced properties have a dynamic value. These properties have a comma-separated value, name-value pairing. An example of this type of property is the value for the topic group default queue sizes.

```
glog.workflow.topicGroup=batch,2
glog.workflow.topicGroup=report,2
```

In order to change this type of property, it is necessary to first remove the default value by adding a "`!remove`" command prior to specifying the new value for the property. For example, to change the default topic group size for the "batch" queue from the default value of "2" to the new value of "4", you would specify the following in the glog.properties file:

```
!remove glog.workflow.topicGroup=batch,2
glog.workflow.topicGroup=batch,4
```

For further information and complete documentation of all properties, please consult the online help topic "Properties".

> **Note**: All property changes made in any file other than "`glog.properties`" will be over-written during every upgrade.

Changing Logging Options

After any of the following settings are changed, you should restart the Oracle Transportation Management instance as described in the chapter **Starting and Stopping the Oracle Transportation Management Servers** for the changes to take effect.

Oracle Transportation Management

To modify the Oracle Transportation Management logging options, you need to log into your Oracle Transportation Management instance as the DBA.ADMIN user. Then navigate through the following menus: Data Management > Power Data > General > Log Files. Click Search to show all available log files. For complete details, please see the online help.

Apache

To modify your Apache log settings, you need to edit the configuration file, httpd.conf. This file is usually under `<otm_install_path>\apache\conf` on Microsoft Windows or

`<otm_install_path>/apache/conf` on UNIX. Refer to your Apache documentation or http://httpd.apache.org for more information.

Tomcat

To modify your Tomcat log settings, you need to edit the configuration file, server.xml. This file is usually under `<otm_install_path>\tomcat\conf` on Windows or `<otm_install_path>/tomcat/conf` on UNIX. Refer to your Apache documentation or http://jakarta.apache.org/tomcat for more information.

WebLogic

To modify your WebLogic log settings, you need to use the WebLogic console. Open a web browser and go to `http://<appserver_name>:7001/console` where `<appserver_name>` is the FQDN of your Oracle Transportation Management application server. You need to login as system, with the password you setup when installing WebLogic. Refer to your WebLogic documentation or http://egeneration.beasys.com/wls/docs92/index.html.

WebSphere

To modify your WebSphere log settings, you need to use the WebSphere console. Open a web browser and go to `https://<appserver_name>:9043/ibm/console` where `<appserver_name>` is the FQDN of your Oracle Transportation Management application server. You need to login as system, with the password you setup when installing WebSphere. Refer to your WebSphere documentation or http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp.

Changing Notification Settings

The Oracle Transportation Management server sends out a variety of notifications to users. Changing these settings involves modifying the glog.properties file on your Oracle Transportation Management application server as described below.

```
glog.mail.smtp.host=smtp.company.com
```

This setting defines the SMTP server that Oracle Transportation Management uses when sending email and fax notifications.

```
glog.workflow.notify.advisor.email=OTMAdvisor@company.com
```

This setting defines the email address that email and fax notifications will appear to come from. This email address should be valid and this email box should be monitored, so that bounced emails and delivery failures are caught. FAXmaker will also send fax success/failure responses to this email address.

```
glog.workflow.notify.advisor.fax=800-555-1212
```

This setting defines the fax number that fax notifications appear to come from.

```
glog.workflow.notify.FAXmaker.email=fax@company.com
```

This setting defines the e-mail address of the fax server. The fax server should monitor the corresponding mailbox and convert the incoming emails to faxes.

Changing Passwords

The following lists out all of the default users created as part of a standard Oracle Transportation Management installation, their default passwords, and how they can to be changed.

Encoding Values in glog.properties

Any value in the `<otm_install_path>`/glog/glog.properties file can be encoded. To do so, do the following:

1. Log into the server as the user that the Oracle Transportation Management application server runs as

2. Run the following commands (note the period at the start of the first line):

   - `. <otm_install_path>/install/gc3env.sh`
   - `java glog.util.appclass.Base64Encoding SOME_VALUE`

3. The encoded value will appear on the screen between angle brackets ("<" and ">"), for example the value "CHANGE" is encoded to "Q0hBTkdFTUU=":

   ```
   $ java glog.util.appclass.Base64Encoding CHANGEME

   <Q0hBTkdFTUU=>
   ```

4. Use the encoded value in the glog.properties file by placing a "{e" prefix on it, for example:

   ```
   glog.database.password={eQ0hBTkdFTUU=
   ```

   Note: The "{e" prefix syntax only works in the `glog.properties` file

Oracle Transportation Management Users

Default Users

A new Oracle Transportation Management installation has the following users created:

User ID	Default Password	Notes
DBA.ADMIN	CHANGEME	May not be deleted.
DBA.DEFAULT	CHANGEME	
SERVPROV.ADMIN	CHANGEME	May not be deleted.
SERVPROV.DEFAULT	CHANGEME	
GUEST.ADMIN	CHANGEME	May not be deleted.
GUEST.DEFAULT	CHANGEME	
GLOG.ADMIN	CHANGEME	
GLOG.DEFAULT	CHANGEME	
STAGE.ADMIN	CHANGEME	
STAGE.DEFAULT	CHANGEME	

User ID	Default Password	Notes
EBS.ADMIN	CHANGEME	
EBS.DEFAULT	CHANGEME	
E1.ADMIN	CHANGEME	
E1.DEFAULT	CHANGEME	
BLUEPRINT.ADMIN	CHANGEME	
BLUEPRINT.DEFAULT	CHANGEME	
system	CHANGEME	Change via the procedures listed Application Server Users section below. May not be deleted.
guest	CHANGEME	
ebs	ebs	
e1	e1	
blueprint	blueprint	
glog	glog	
glogdev	CHANGEME	

Unless otherwise noted above, passwords for Oracle Transportation Management users can be changed using the Oracle Transportation Management User Manager; refer to the online help for details.

Resetting Passwords

Passwords in the Oracle Transportation Management database are encrypted and are not presented as clear-text. If you lose a password for a user and need to reset it, you can log in as an Admin user for that domain and reset their password using the User Manager.

If the Admin user password is lost, or if you want to reset more than one password, you can do the following:

1. Log into the application server machine as the Oracle Transportation Management user
2. Setup your environment by running `<otm_install_path>/install/gc3env.sh` on UNIX or `<otm_install_path>/install/gc3env.cmd` on Windows.
3. cd to <otm_install_path>/oracle/script8
4. Run update_password.sh on UNIX or update_password.cmd on Windows with the following parameters:

    ```
    update_password.sh  <glog_properties_path> <user> <password>
    <ifPasswords>
    ```

    o **glog_properties_path**: directory where glog.properties is, e.g. <otm_install_path>/glog/config
    o **user**: one or more users separated by commas, or 'all', e.g. GUEST.ADMIN,SERVPROV.ADMIN

o `password`: the new password

o `ifPasswords`: one or more passwords to match against separated by commas, or 'all'

For example:

```
$ ./update_password.sh /opt/otm/glog/config GUEST.ADMIN foobar all
```

will change GUEST.ADMIN's password to 'foobar' regardless as to what it is currently set to.

```
$ ./update_password.sh /opt/otm/glog/config all foobar
CHANGEME,DEFAULT
```

will change ALL passwords to 'foobar', but only if they are currently set to 'CHANGEME' or 'DEFAULT'

Note: When running this procedure for all users, the application server's 'system' user password is changed as well. If you had previously changed this, you will need to change it back or change it as noted under the section ***Application Server Users*** below.

5. The changes will not take effect until Oracle Transportation Management is restarted.

Application Server Users

Default Users

Each type of application server has one or more users that are utilized in running Oracle Transportation Management, and each user has a separate set of instructions to be used if you want to change the password. None of these users may be deleted unless otherwise noted.

WebLogic

WebLogic creates no new users on its own, relying on the users that Oracle Transportation Management creates:

User ID	Default Password	Notes
system	CHANGEME	Used to start and stop the application as well as manage the WebLogic console. May not be deleted.

In order to change the 'system' user's password, do the following:

1. Ensure that Oracle Transportation Management application and web servers are stopped.
2. Setup your environment by running `<otm_install_path>/install/gc3env.sh` on UNIX or `<otm_install_path>\install\gc3env.cmd` on Windows.
3. cd to `<otm_install_path>/oracle/script8`.
4. Run `update_password.sh` on UNIX or `update_password.cmd` on Microsoft Windows with the following parameters:

   • `update_password.sh <glog_properties_path> <user> <password> <old-password>`

      o glog_properties_path: directory where glog.properties is, e.g. `<otm_install_path>/glog/config`

      o user: system or guest

      o password: the new password

      o old-password: the existing password

5. Run the following command; this will return the encoded value for your new password:
   - java glog.util.appclass.Base64Encoding <new_password>
     - password: the new password
6. On the application server(s) edit the file `<otm_install_path>/weblogic/weblogic.conf`
   a. **[system]** search for the string `var.WL_PW=CHANGEME` and change the value of this to the results of the previous step.
   b. **[system]** add or replace the string `jvm.arg=-DGC3EncodedPassword=<encoded-password>`, where `<encoded-password>` is the result of the previous step.
   c. **[guest]** add or replace the string `jvm.arg=-DGuestEncodedPassword=<encoded-password>`, where `<encoded-password>` is the result of the previous step.
7. On the Oracle Transportation Management web server(s) edit the file `<otm_install_path>/tomcat/bin/tomcat.conf`
   a. **[system]** search for the string `-DGC3EncodedPassword=` and change the value of this to the results of the previous step.
   b. **[guest]** search for the string `-DGuestEncodedPassword=` and change the value of this to the results of the previous step.
8. Restart Oracle Transportation Management

WebSphere

WebSphere creates no new users on its own, relying on the users Oracle Transportation Management creates:

User ID	Default Password	Notes
system	CHANGEME	Used to start and stop the application as well as manage the WebSphere console. May not be deleted.

In order to change the 'system' user's password, do the following:

1. Log into Oracle Transportation Management as DBA.ADMIN and use the Oracle Transportation Management User Manager to change the password for the 'system' user; refer to the online help for details.
2. On the application server, edit the following files, replacing the old password with the new password:
   a. `<otm_install_path>/websphere/profiles/default/bin/deployGC3.jacl`
   b. `<otm_install_path>/websphere/profiles/default/bin/deployGC3SS.jacl`
   c. `<otm_install_path>/websphere/profiles/default/bin/undeployGC3.jacl`
   d. `<otm_install_path>/websphere/profiles/default/bin/undeployGC3SS.jacl`
   e. `<otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01Cell/nodes/__APP_SERVER__Node01/servers/server1/server.xml` (where __APP_SERVER__ is your application server's host name)
   f. Look on the "`<jvmEntries`" line near the bottom of the file.
   g. `<otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01Cell/ security.xml` (where __APP_SERVER__ is your application server's host name)
      i. Log into your APP server machine as the Oracle Transportation Management user

ii. cd $WAS_INSTALL_DIR/lib (where $WAS_INSTALL_DIR is the directory where you installed the WebSphere application server, e.g. /opt/IBM/WebSphere/AppServer6)

iii. Type the following, all on one line, where "secret" is your new password; make note of the output, specifically the stuff between the quotes (e.g. encoded password == "{xor}LDo8LTor"):
../java/bin/java -cp securityimpl.jar:iwsorb.jar::ras.jar:wsexception.jar:bootstrap.jar:emf.jar:ffdc.jar com.ibm.ws.security.util.PasswordEncoder secret

iv. Look for the "<authDataEntries" line in the security.xml file that has "alias="system""

v. Replace the old "password="{xor}HBceERgaEho="" with the newly encoded secret "password="{xor}LDo8LTor ""

3. Log into the application server machine as the Oracle Transportation Management user

4. Setup your environment by running `<otm_install_path>/install/gc3env.sh` on UNIX or `<otm_install_path>/install/gc3env.cmd` on Windows.

5. Run the following command:
   `java glog.util.appclass.Base64Encoding <new_password>`

6. This will return the encoded value for your new password

7. On the Oracle Transportation Management web server(s) edit the file `<otm_install_path>/tomcat/bin/tomcat.conf`; search for the string '-DGC3EncodedPassword=' and change the value of this to the results of the previous step.

8. Restart Oracle Transportation Management

Oracle Database Users

Database User ID	Default Password	Notes
archive	archive	This user owns the _DMP tables used for archiving the data. May not be deleted.
glogdba	glogdba	This user has access to functions and packages owned by glogowner and reportowner, but does not itself own any tables, views, functions or packages. It must call the vpd.set_user stored procedure to set user context to view data. May not be deleted.
glogowner	glogowner	This user owns Oracle Transportation Management tables, views, functions and packages, can create or alter data structures within the database and can manipulate data. May not be deleted.
glogdev	glogdev	This user has complete access to the data. There is no need to call the vpd package to set user context when logged in as glogdev; does not own any tables, views, functions or packages. May not be deleted.
glogload	glogload	Used for loading data into glogowner and reportowner schemas. May not be deleted.

Database User ID	Default Password	Notes
reportowner	reportowner	This user owns the tables, views, functions and packages required for reporting, and can read the data. May not be deleted.
globalreportuser	globalreportuser	This user has read access to all the data in Oracle Transportation Management. It is mainly used for XSQL reporting. May not be deleted.
ftimaster	ftimaster	Used to create Oracle Data Integrator (ODI) Master repository.
ftiwork	ftiwork	Used to create ODI Work repository.
ftistage	ftistage	Used to create ODI staging area.

If you wish to change the passwords for these users, follow these steps:

1. Using SQL*Plus, log into the Oracle Transportation Management database as sys or system.
2. Run the following for each user that you wish to change:

   `alter user <user_name> identified by <new password>`

Some users have additional steps that need to be taken:

glogdba

1. Edit the `<otm_install_path>/glog/config/glog.properties` file on each Oracle Transportation Management web and application server and replace the value of "glog.database.password" with the new glogdba password.
2. **[WebSphere]** Edit the file
   `<otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01 Cell/ security.xml` (where __APP_SERVER__ is your application server's host name)
     a. Login to your APP server machine as the Oracle Transportation Management user.
     b. `cd $WAS_INSTALL_DIR/lib` (where `$WAS_INSTALL_DIR` is the directory where you installed the WebSphere Application Server, e.g. `/opt/IBM/WebSphere/AppServer6`).
     c. Type the following, all on one line, where "secret" is your new password; make note of the output, specifically the stuff between the quotes (e.g. encoded password == "{xor}LDo8LTor"):
        `../java/bin/java -cp securityimpl.jar:iwsorb.jar::ras.jar:wsexception.jar:bootstrap.jar:emf.jar:ffdc.jar com.ibm.ws.security.util.PasswordEncoder secret`).
     d. Look for the "`<authDataEntries`" line in the security.xml file that has "`alias="glogdba`"
     e. Replace the old "`password="{xor}HBceERgaEho=""` with the newly encoded secret "`password="{xor}LDo8LTor ""`.
3. Restart Oracle Transportation Management.

glogload

1. Edit the `<otm_install_path>/glog/config/glog.properties` file on each Oracle Transportation Management web and application server and replace the value of "`glog.database.load.password`" with the new glogload password.

globalreportuser

1. Edit the `<otm_install_path>/glog/config/XSQLConfig.xml` file on each Oracle Transportation Management web server and replace the value of "`<password>globalreportuser</password>`" with the new globalreportuser password.

Cleartext Passwords

The following files have cleartext passwords in them. Proper OS-specific measures should be taken to ensure that only privileged users have read-access to these files:

- `<otm_install_path>/glog/config/glog.properties`
- `<otm_install_path>/glog/config/XSQLConfig.xml`
- [WebLogic] `<otm_install_path>/weblogic/weblogic.conf`
- [WebLogic] `<otm_install_path>/weblogic/domains/otm/config/config.xml.fresh`
- [WebSphere] `<otm_install_path>/websphere/profiles/default/bin/deployGC3.jacl`
- [WebSphere] `<otm_install_path>/websphere/profiles/default/bin/deployGC3SS.jacl`
- [WebSphere] `<otm_install_path>/websphere/profiles/default/bin/undeployGC3.jacl`
- [WebSphere] `<otm_install_path>/websphere/profiles/default/bin/undeployGC3SS.jacl`

In addition, the following two files have passwords that can be easily decrypted:

- [WebSphere] `<otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01 Cell/nodes/__APP_SERVER__Node01/servers/server1/server.xml` (where __APP_SERVER__ is your application server's host name)x`
- [WebSphere] `<otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01 Cell/ security.xml` (where __APP_SERVER__ is your application server's host name)

Changing the hdowner Password for Oracle Fusion Transportation Intelligence

The following instructions describe in detail how to set up and change the hdowner password(s) in Oracle Fusion Transportation Intelligence. You will need to have System Administrator rights to be able to perform the following actions:

1. Obtain the Windows OBIEE Suite from eDelivery.Oracle.com and install it on your local windows machine.
2. Download the advanced_analytics.rpd repository file from your Oracle Fusion Transportation Intellitence box. You can find it in `/server/Repository`. For example: `/opt/oraclebi/server/Repository`.
3. Navigate to and select **advanced_analytics.rpd**.
4. When prompted for an ID/password, click **OK**, using **Administrator** as the ID and leave password blank.
5. When the RPD opens, you will be presented with three columns of information.

   Note: There are three places to change the password in the physical layer.

6. In the Physical column click the **+** sign for both **Advanced Analytics** and **Advanced Analytics UET**.
7. Double-click **Connection Pool**.
8. Select the Password field and type the new password for hdowner. Press **Enter**.
9. In the Confirm Password field, retype the new password for hdowner and click **OK**.
10. Repeat this process for NonVPD and UETNonVPD.

11. Click **File > Save** and select **No** when prompted about global consistency.

12. Upload the advanced_analytics.rpd repository file back to your Oracle Fusion Transportation Intelligence machine. Normally it is located in `/server/Repository`. For example: `/opt/oraclebi/server/Repository`.

13. Update the password for hdowner in the glog.properties files. This property is: `glog.database.fti.password`.

14. Retest the issue.

15. Migrate the solution as appropriate to other environments.

    Note: If you are unable to log in as described in step 4 by inputting Administrator and leaving the password blank, please follow the steps provided below:

Set security settings in the NQSConfig.INI in the `<path>\OracleBI\server\Config` folder and override `MINIMUM_PASSWORD_LENGTH` from 8 to 0:

```
MINIMUM_PASSWORD_LENGTH = 0;
```

> **Note**: If RPD is read-only, change it to allow changes. Then, you can change the RPD file.

Multi-byte Character Support in BI Publisher

Oracle BI Publisher PDF generation with multi-byte characters requires an extra font be copied into a specific directory. Please follow these instructions when Oracle Transportation Management is first installed, and every time the JDK is updated.

1. Log into each server where report transformation is being done.

2. Copy the files in the `<otm_install_path>/install/fonts` directory to the `<otm_install_path>/jdk/jre/lib/fonts` directory:

   ```
   cp <otm_install_path>/install/fonts
   <otm_install_path>/jdk/jre/lib/fonts
   ```

3. Restart the Oracle Transportation Management instance.

Default Units of Measure

By default, Oracle Transportation Management uses U.S. units of measure when saving data to the database. To change what units of measure Oracle Transportation Management uses, follow these steps.

1. Run the following SQL statement to find out what the current storage default is for each UOM:

   ```
   select TYPE,UOM_CODE,UOM_RANK,IS_STORAGE_DEFAULT from UOM where
   IS_STORAGE_DEFAULT = 'Y'
   ```

The current typical layout of the UOM table for an Oracle Transportation Management instance is as follows:

This layout shows that for this particular Oracle Transportation Management installation, the default
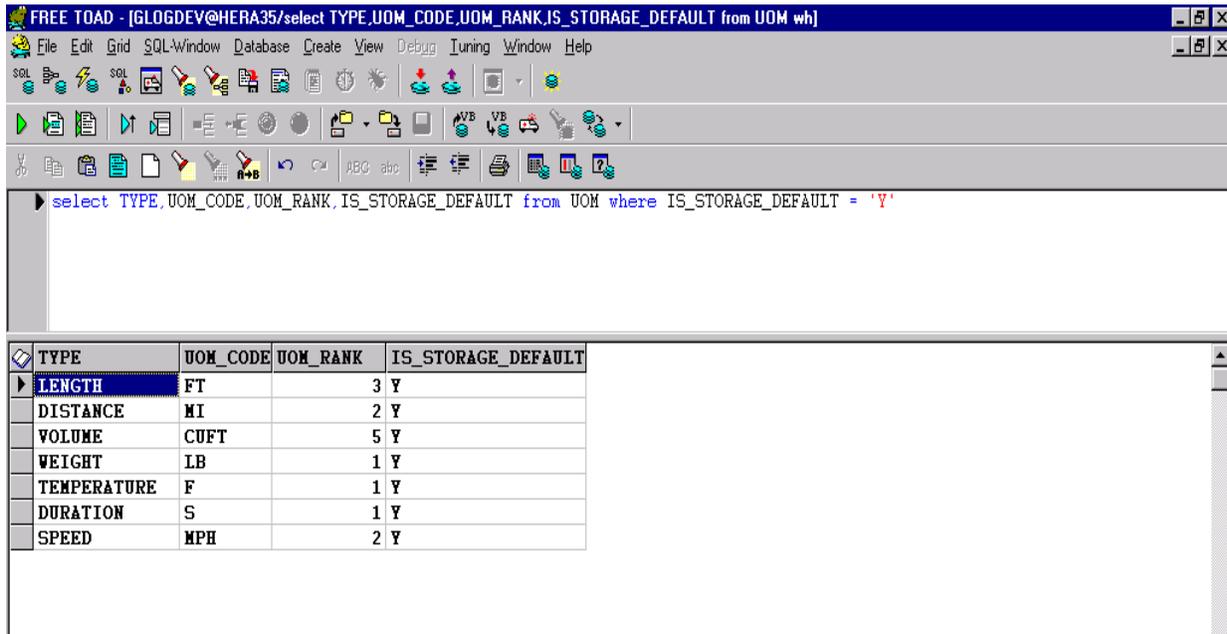
storage UOM for length is FT, for distance Miles…etc…

> **Note**: If you use a default currency other then USD, you will need to remove that currency record from UOM table after each consolidated update is installed.

2. Run the following SQL statement to change the default storage in your system for any UOM. (This example changes the default UOM for LENGTH from current FT to M = meters):

    ```
    UPDATE UOM SET IS_STORAGE_DEFAULT = 'Y' WHERE UOM_CODE = 'M'
    ```

3. You will also need to remove the 'Y' flag on the current length default of FT.

    ```
    UPDATE UOM SET IS_STORAGE_DEFAULT = 'N' WHERE UOM_CODE = 'FT'
    ```

The above is an example for length but the same SQL can be used to change the defaults for DISTANCE, SPEED, TEMPERATURE, VOLUME, and WEIGHT.

Changing Currency Settings

By default, Oracle Transportation Management uses US Dollars when saving costs to the database. Also by default, Oracle Transportation Management triangulates all currency conversions through US Dollars.

This example illustrates how Oracle Transportation Management stores a shipment cost record with the currency storage default set to two different currencies:

Total actual cost of shipment is 1000 JPY. If Oracle Transportation Management's currency storage default is USD (current default in all Oracle Transportation Management installations), Oracle Transportation Management stores this cost as follows:

Total_actual-cost	Total-actual-cost_currency_GID	Total_Actual_cost_base

Total_actual-cost	Total-actual-cost_currency_GID	Total_Actual_cost_base
1000	JPY	7.76

If instead Oracle Transportation Management's currency storage default is GBP:

Total_actual-cost	Total-actual-cost_currency_GID	Total_Actual_cost_base
1000	JPY	5.31

In the first instance, the rate of 7.76 represents the USD value of 1000 JPY converted at the current rate in Oracle Transportation Management (128.77) while in the second instance, the rate of 5.31 represents the GBP value of 1000 JPY converted at the current JPY/GBP rate in the system (188.08).

When to Change Currency Storage Default

There are two scenarios where you would like to change your currency storage default: either you only use one currency other than USD or you use multiple currencies but not USD.

Rates in One Single Currency

In this case, you only need to set your currency storage default to the currency you use. For example, Oracle Transportation Management stores a 100 GBP shipment cost as 100 in both the total cost and the total cost base fields so no currency conversion is needed.

1. Run the following insert statement (example using GBP as new storage default) as the DBA user:

   ```
   INSERT INTO UOM (TYPE, UOM_CODE, UOM_SHORT_DESCRIPTION,
   UOM_LONG_DESCRIPTION, UOM_RANK, IS_STORAGE_DEFAULT,
   IS_DISPLAY_DEFAULT, DOMAIN_NAME) VALUES ('CURRENCY', 'GBP', 'UK
   POUND', 'UK POUND', 1, 'Y', 'Y', 'DBA')
   ```

Rates in Several Currencies

In this case, you need to:

1. Decide what the currency storage default is for your Oracle Transportation Management installation.
2. Set your currency storage default by running the following insert statement (example using GBP as new storage default) as the DBA user:

   ```
   INSERT INTO UOM (TYPE, UOM_CODE, UOM_SHORT_DESCRIPTION,
   UOM_LONG_DESCRIPTION, UOM_RANK, IS_STORAGE_DEFAULT,
   IS_DISPLAY_DEFAULT, DOMAIN_NAME) VALUES ('CURRENCY', 'GBP', 'UK
   POUND', 'UK POUND', 1, 'Y', 'Y', 'DBA')
   ```

Oracle Transportation Management still needs currency rates to convert between the currencies you use. You can download rates from the IMF website. (This populates the DEFAULT rate in the CURRENCY_EXCHANGE_RATE table.) The problem is that all rates from the IMF are against USD. You have two alternatives to using USD centric IMF rates:

- Write an API that uses another source than the IMF to bring in exchange rates stated against your currency storage default.

  or

  Update the rates you need manually or use a CSV process while again entering your needed rates against your preferred currency. You can do this nightly, monthly or in any other frequency.
- In the property file glog.properties, set glog.currency.base to your currency storage default (e.g. EUR). This makes Oracle Transportation Management triangulate through the currency of your choice.

This means that Oracle Transportation Management will have all DEFAULT rates stated against your base currency and triangulates using your base currency.

Accounting for Missing Time Zones

The UTC pl/sql package depends on java running in Oracle to do time zone conversions. Some time zones are not accounted for in the java.util.TimeZone class. The solution to this problem involves a new (and optional) "missing_time_zone" table, which has the following structure:

Column	Value
TIME_ZONE_GID	NOT NULL VARCHAR2(128)
ALT_TIME_ZONE_GID	NOT NULL VARCHAR2(128)
ADJ_FACTOR	NOT NULL NUMBER

So for example, if you had a location that specified a time zone of "FOO", and you wanted "FOO" to be treated the same as "America/New_York", you would specify a missing_time_zone record of:

```
time_zone_gid = 'FOO'
alt_time_zone_gid = 'America/New_York'
adj_factor = 0
```

The adj_factor is added to the time_zone_offset. So if the time_zone_offset for America/New_York was -5, and you wanted 'FOO' to have an offset of -4, you could set the adj_factor to 1.

Some of the missing Australian time zones are accounted for in the java code itself. The defaults established in the code for the Australian time zones can be overridden via the missing_time_zone table.

BPEL Integration

BPEL Flows

Oracle Transportation Management provides BPEL integrations with Oracle E-Business Suite. These flows can be found in `<otm_install_path>/utils/integration/bpel/flows`. Each flow should have a README.TXT file in the zip file with further instructions. There are several other flows that ship with the other Oracle products. Refer to the respective product documentation for the details for those flows.

Oracle Transportation Management Properties Files for Oracle E-Business Suite Integration through BPEL

If integrating with Oracle E-Business Suite through Oracle BPEL Process Manager, you need to include/uncomment the ebs.properties file in the glog.properties file on the Oracle Transportation Management web and application servers. These files are usually under <otm_install_path>\glog\config on Windows or <otm_install_path>/glog/config on UNIX. In the glog.properties file, you would uncomment the following:

```
#!include ebs.properties
```

Your line would look like this:

```
!include ebs.properties
```

Once this change has been made, restart your Oracle Transportation Management instance.

Oracle Transportation Management Properties Files for BPEL Integration

If integrating with the Oracle BPEL Process Manager, but not necessarily integration with Oracle E-Business Suite, you will need to enable a property in the glog.properties file on the Oracle Transportation Management web and application servers. These files are usually under <otm_install_path>\glog\config on Windows or <otm_install_path>/glog/config on UNIX. In the glog.properties file, add the following:

```
glog.integration.enableParserInServlets=true
```

Once this change has been made, restart your Oracle Transportation Management instance.

Integration to BPEL

As of the Oracle Transportation Management 5.5 CU2 release, the recommended method for sending data from Oracle Transportation Management to a BPEL server is via web services that are configured in the External System Manager in the Oracle Transportation Management UI. Refer to the online help for details.

Recurring Processes

There are recurring processes in Oracle Transportation Management that are scheduled to run automatically at regular intervals. Several of these are configured and turned on out-of-the-box in a fresh installation or upgrade. Please consult the online help to determine what these processes are, to help understand what they do and to learn how to change their frequency interval or disable them altogether.

SMTP Authentication

If an SMTP server requires authentication this can be specified along with the name of the SMTP server itself. Wherever you specify an SMTP host you can also specify a username and password as follows:

- `username/password@smtp.company.com`: The supplied username and password are used to authenticate with the SMTP host.
- `username/{epassword@smtp.company.com`: Prepending "{e" to the password value indicates that the password is Base-64 encoded. The value will be decoded before being sent to the SMTP host. See the **Changing Passwords** section of the Administration Guide for more information on encoding passwords.

- `smtp.company.com`: either no authentication is needed or authentication is accomplished via IP address.

7. Advanced Configuration: LDAP

LDAP stands for the Lightweight Directory Access Protocol. It is important to remember that LDAP is in essence a protocol – a common language that various directory products can speak in order to communicate with users and applications -- and other directories. The TCP/IP-based LDAP protocol contains messages allowing an LDAP client (an application or user) to connect to, search, add to, delete from, and modify an LDAP server (the directory).

Overview

LDAP clients connect to an LDAP server as a user in the directory (sometimes called binding to the directory). The LDAP server may choose from a number of authentication protocols (see below) to validate the identity of the connecting user. Once connected, the LDAP user can search or modify the directory (if permission has been granted to perform these operations). In our case, Oracle Transportation Management is the LDAP client. A customer's LDAP directory is the server.

LDAP represents names in a standard format – the Distinguished Name, or DN (see below for more detail on DNs). This format contains name attributes like organization, country, organization unit, etc… Moreover, these attributes are arranged hierarchically. So, there can be multiple organizational units within an organization, and multiple organizations within a country. The directory is searched and organized hierarchically.

Each name is associated with one or more directory objects. These directory objects contain attributes that can be used for authentication, for populating databases, for applications, or any other number of uses.

Because of the focus on clients, the LDAP community also defined standards for the string representation of DNs (RFC 1779), search filters (RFC 1960), and attribute syntaxes (RFC 1778), for a C language-based API (RFC 1823), and for the format of URLs for accessing LDAP services (RFC 1959).

LDAP Schema

A directory schema specifies, among other things, the types of objects that a directory may have and the mandatory and optional attributes that each type of object may have. LDAP (version 3) defines a schema (RFC 2252 and RFC 2256) based on the X.500 standard. The schema includes common objects found in networks, such as countries, localities, organizations, users/persons, groups, and devices. It also defines a way for a client application to access the server's schema so that it can find out the types of objects and attributes that a particular server supports.

The LDAP schema has become one of the basic ways that different LDAP directories can interoperate. Corporations use the schema to store user, profile, organization, contact, and location information. Oracle Transportation Management relies on the username (and for local authentication, password) attributes. Currently, Oracle Transportation Management requires that the username be part of the distinguished name, and requires that the "GLUSER" attribute be added to each user object.

LDAP in Oracle Transportation Management

LDAP is used by Oracle Transportation Management to allow users to log into Oracle Transportation Management using standardized LDAP names, instead of, or in addition to Oracle Transportation Management usernames. Oracle Transportation Management allows authentication to be performed by a remote LDAP server -- a more secure, more centralized approach. Corporate users can securely log onto Oracle Transportation Management with the LDAP login names that they are used to and use everyday.

Oracle Transportation Management allows the LDAP users to be mapped to Oracle Transportation Management users in the LDAP directory itself. This way, Oracle Transportation Management permits a single Oracle Transportation Management user to be mapped to multiple LDAP users. This allows a generic Oracle Transportation Management user such as "GUEST" (GUEST may have primarily read-only access to limited information) to a large group of users, without giving the password to all of these users. LDAP allows Oracle Transportation Management user, security, and policy information to be centralized in one place – the LDAP directory.

In addition, Oracle Transportation Management contains multiple directory support (see NameSpaces below). This allows multiple LDAP directories to be consulted to find names. For instance, a logistics company may wish to authenticate shippers with a local LDAP directory, and service providers with an external LDAP directory. In addition, NameSpaces allow the same directory to be looked up using (for instance) different authentication protocols, or different branches of the same directory tree.

Oracle Transportation Management allows you to choose which directory is consulted upon login. Alternatively, a default search order can be configured, so that multiple directories can be looked up in turn to authenticate a login.

Limitations

Oracle Transportation Management does not support group authentication, group membership testing, or distribution lists. It simply supports username authentication upon login. Oracle Transportation Management does not support the use of user profile attributes from LDAP such as language, time zone, e-mail addresses, or any other user preferences. All user parameters are controlled within Oracle Transportation Management – the GLUSER attribute provides the linkage between an LDAP user and an Oracle Transportation Management user. The LDAP directory itself must be modified to contain the mapping (this in itself could be considered a limitation).

Oracle Transportation Management does not use the LDAP directory to store contact information, e-mail addresses, location information, or other centrally maintained pieces of information vital to large corporations using our product.

LDAP Server

Oracle Transportation Management does not contain an LDAP server. Many customers already have a corporate LDAP Server, and one of the major goals of this feature is to allow corporations to centralize user and security information -- not having it scattered in many different places. Oracle Transportation Management can be configured to talk to an LDAP server by defining a NameSpace.

Single Sign-on Support

With LDAP, Oracle Transportation Management supports the ability to have users login to Oracle Transportation Management using LDAP usernames that they are familiar with. Sometimes, third-party packages allow user to log into the package once. Thereafter, that person will not have to log in to each individual application that may be accessed subsequently. LDAP, as a technology, is often used in the implementation of single sign-on solutions.

Definitions

NameSpace

A NameSpace is where information about an LDAP directory is stored in Oracle Transportation Management. It is an Oracle Transportation Management term, and not an "LDAP term". It contains information such as the branch of the naming hierarchy to search for users, the URL of the LDAP server, the authentication methods to be employed, and the protocol version to use. A user logging in can choose which NameSpace to use for authentication, or use the default. In the default case, Oracle

Transportation Management allows multiple directories to be looked up one by one until successful authentication has taken place.

Distinguished Name

A Distinguished Name (or DN) is the standard format for naming within LDAP directories. Quite simply, a distinguished name is an ordered list of naming attributes. These attributes are often syntactically organized into a single string such as "cn=John Doe, ou=Marketing, o=Oracle, c=US" (see RFC 1779). This name consists of Common Name, Organizational Unit, Organization, and Country attributes. The directory uses these attributes to arrange objects in the directory hierarchically. So, there can be multiple organizational units within an organization, and multiple organizations within a country. This way, different branches of the LDAP "tree" can be searched independently. For instance, one might want to search only names within the organization "Oracle".

Oracle Transportation Management requires that the user ID field be part of the Distinguished Name (at least externally to an LDAP client). It also requires that each LDAP user object to be authenticated with Oracle Transportation Management be populated with the GLUSER attribute. The GLUSER attribute should not be part of the Distinguished Name.

A fully qualified DN identifies the name of an object within an LDAP directory. A relative DN identifies a branch of the naming tree, but does not necessarily address a schema object.

LDAP Authentication Protocol

An LDAP Authentication Protocol is used between an LDAP client and LDAP server to authenticate a user within the directory. Oracle Transportation Management supports simple authentication (clear text username and password), and some other authentication protocols (such as CRAM-MD5). Oracle Transportation Management also supports using no authentication at all (although this is somewhat pointless). These protocols can be used in both local and external authentication (see below).

LDAP Encryption Protocol (SSL)

Oracle Transportation Management allows encryption between the Oracle Transportation Management application server and the LDAP server. This insures that password information flowing between the LDAP server and Oracle Transportation Management is not intercepted. Oracle Transportation Management uses SSL to provide this encryption. SSL is a generic transport layer encryption/authentication solution. The LDAP directory server must support SSL in order to use this feature. Although SSL can in theory be used for authentication as well, SSL is used by Oracle Transportation Management to encrypt the communication between Oracle Transportation Management and the LDAP server.

LDAP Authentication Method

Oracle Transportation Management uses two major methods of authentication. Local authentication involves searching for a name object in the directory and extracting some attributes. These attributes are in turn used to authenticate the name. Most commonly, the password attribute is used to validate entries. It's called local authentication because the validation is performed "locally" (by the client). The client logs into the LDAP directory as a sort of super-user (called the Principal). The principal user is used to look up all other users in the directory.

Oracle Transportation Management can also authenticate users by attempting to directly connect to the directory as the user in question, instead of connecting as a principal user and then performing a lookup. This allows the LDAP directory to perform the authentication at the LDAP server. It's called external authentication, because the authentication is performed externally to the LDAP client. This method insures that sensitive authentication information (such as certificates or passwords) never leaves the LDAP server. In the LDAP world, this method of authentication is often called "binding" to the server.

LDAP Protocol Version

LDAP is a TCP/IP based protocol, and this protocol has two major revisions still in use. Version 2 contains most of the basic LDAP functionality. RFC 1777 defines what is now called version 2 of the LDAP (or LDAP v2). The LDAP v3 (RFC 2251) protocol is designed to address some of the limitations of LDAP v2 in the areas of internationalization, authentication, referral, and deployment. It also allows new features to be added to the protocol without also requiring changes to the protocol itself.

LDAP Directories

Organizing LDAP Directories For Oracle Transportation Management

Oracle Transportation Management requires that the user ID field be part of the Distinguished Name (at least externally to an LDAP client). It also requires that each LDAP user object to be authenticated with Oracle Transportation Management be populated with the GLUSER attribute. The GLUSER attribute should not be part of the Distinguished Name.

Each NameSpace contains a User DN field. This contains a relative DN that identifies the branch of the LDAP tree to search for users. Oracle Transportation Management searches this branch, and this branch only – meaning no sub-branches are searched. If you wish to search for sub-branches, you must define each sub-branch as a NameSpace, and use the default search order to search for them one-by-one (see NameSpaces above).

Many directories enforce trueness to the schema defined for a particular object. This means that the object's schema must be modified to contain the attribute GLUSER in order to have that attribute be successfully added. Sometimes, this schema checking can be turned off in the directory. Another alternative is to use an attribute that already exists in the schema (but is not populated). You can change the NameSpace configuration to define the attribute where the GLUSER information is contained, so that Oracle Transportation Management knows where to get the Oracle Transportation Management user mapping.

LDAP and the Oracle Transportation Management Login

If you have any doubt, choose Default. This most likely will be set up by the Oracle Transportation Management Administrator to serve most needs. The GC3 NameSpace allows logging in via the Oracle Transportation Management username and password (see below). The other choices represent LDAP directories that have been configured to work with Oracle Transportation Management.

The Oracle Transportation Management administrator has the ability to configure which directories are consulted when the Default option is chosen. In fact, when this option is chosen, the administrator has the ability to configure an ordered list of directories to search. Usually, this results in a successful authentication.

The GC3 NameSpace is a special NameSpace that identifies the Oracle Transportation Management realm itself (the usernames and passwords stored directly within Oracle Transportation Management). When you search the GC3 NameSpace, you are performing a search local to the Oracle Transportation Management product.

Configuring LDAP NameSpaces

The glog.ldap.properties file is read by the web server when the first user logs in. It is never read again. You can bounce the Tomcat instance (if you know how to do this) if you need to reload the properties. The application server will automatically adjust.

The following is a sample property file containing one NameSpace:

```
ldap.searchOrder=GC3, CorpDir
ldap.namespace.name=CorpDir
ldqp.namespace.CorpDir.authProtocol=simple
ldap.namespace.CorpDir.ldapUrl=ldap://localhost:389
ldap.namespace.CorpDir.principal=otmdev
ldap.namespace.CorpDir.credential=CHANGEME
ldap.namespace.CorpDir.userDN=o=Oracle, c=US
ldap.namespace.CorpDir.userNameAttribute=uid
ldap.namespace.CorpDir.glUserAttribute=gluser
ldap.namespace.CorpDir.userAuthentication=local
ldap.namespace.CorpDir.credentialAttribute=password
```

The search order parameter is global to all NameSpaces. The namespace.name parameter must precede the other NameSpace parameters. Other NameSpaces can be added below.

NameSpace Attributes

Here is a list of the attributes that comprise a NameSpace.

Attribute	Description	Valid Values
authProtocol	The authentication protocol to employ.	None simple (default) CRAM-MD5 (v3 external authentication only) DIGEST-MD5 (v3 external authentication only).
Name	Name Of the LDAP namespace – used in user interface display. Required	String (example: CompanyDir) Only alpha-numeric and underscores are allowed.
LdapUrl	URL of the LDAP server. Required	String (example: ldap://somehost:389)
Principal	User to log in as on LDAP server. Required if authentication=local	Distinguished Name (example: Uid=foo)
Credential	LDAP principal password. Required if authentication=local.	String

Attribute	Description	Valid Values
UserDN	Distinguished name of all users to be searched – the branch of the tree to search. Always specify the most "specific" attributes first – for example, supply Organizational Unit (ou) before Organization (o), which should be supplied before Country (c), etc… Required.	Distinguished Name (example: ou=people, o=acme.com,, c=US)
userNameAttribute	Name of the User ID attribute in the LDAP directory. Required.	String (default: uid)
glUserAttribute	Name of the GLUSER attribute in the LDAP directory. Required.	String (example: gluser)
userAuthentication	Type of authentication employed for this namespace.	Local = authentication based on downloaded attributes. external = an LDAP bind as the user in question (default)
credentialAttribute	Name of the password attribute in the LDAP directory. Required if authentication=local.	String (example: userpassword)
SSL	Connect to directory using SSL (true or false).	true false (default)
version	The version of the LDAP protocol (2 or 3).	2 (default) 3
ctxFactory	Java Naming and Directory Interface (JNDI) service provider to use.	String (default: com.sun.jndi.ldap.LdapCtxFactory)

The Distinguished Name is an ordered list of attributes, and the attributes must be listed by the most specific attribute first. For instance, OrganizationalUnit is more specific than Organization, but less specific than an individual user within that OrganizationalUnit. Similarly, a domain component attribute of Oracle is more specific than dc=com. When you specify a DN, you are really specifying a path from the node you are interested in up to the root of the directory tree.

Common Questions

Why can't I connect using local authentication, even though the password attribute is present?

There are a number of reasons why this could happen. The password attribute may not be visible to Oracle Transportation Management, even though it may be visible to other clients. The password could be encrypted – you might want to try changing the authenticationProtocol attribute. It is possible that the directory entry is not readable by the Principal being used.

Why does authentication fail for entries not directly below the UserDN in the directory tree?

This is a limitation of the LDAP client – the benefit is rapid lookups. However, many commercial directories allow entries to be indexed and placed in a single branch. In addition, the NameSpaces feature can be configured to allow multiple branches to be searched.

I am using local authentication and my principal user uses an encrypted password, but my users use simple authentication. How do I get authentication to work?

You cannot have the Principal using a different encryption algorithm than the individual users. If you really need this, you can use external authentication and bypass the special authentication for the Principal altogether.

If you have two user groups that use different authentication mechanisms, you can set up two NameSpaces that point to the same directory, but use different authentication methods. Then, set the default search order parameter (see above) to search the two namespaces.

8. Performance Monitoring

Performance monitoring is a necessary part of running a complex multi-tier application like Oracle Transportation Management. In most cases, a sudden change in performance is caused by a recent configuration change or an increase in transactional data. A formal change management process is critical for maintaining a stable production system and for quickly diagnosing performance issues should they happen. All system related configuration changes should be tested before applying to production. The date and time of the change should also be tracked. In a situation where this process is not being followed, a query of the insert/update date on important configuration tables may reveal a recent change that is causing an issue. It is also important to maintain benchmark statistics (i.e. orders and shipments processed in a typical day) to be able to quickly rule out a significant increase in data volume.

Operating System Monitoring Tools

Microsoft Windows

On Microsoft Windows, you can monitor system performance using the Task Manager or Performance Monitor. You should monitor general trends, such as CPU usage, memory usage, and memory swapping (out to virtual memory).

The Apache service starts up multiple instances of Apache to handle incoming requests. Each of these will show up as "Apache.exe", and can be monitored for performance statistics using general Microsoft Windows Server tools.

The Tomcat service starts up one Java Virtual Machine (JVM) and shows up as "java.exe". Unfortunately, this can be hard to find if you have multiple JVMs running on the same machine. Usually you can identify Tomcat by the amount of memory it is using, as normal Tomcat instances use large amounts of memory (1.6 GB or higher). You can monitor the performance statistics of Tomcat using general Microsoft Windows Server tools.

The WebLogic service starts up one JVM, but shows up as beasvc.exe, because of WebLogic's special service runner. You can monitor the performance statistics of WebLogic using general Microsoft Windows Server tools. You can also get detailed performance statistics from the WebLogic console. Follow these steps:

1.   Log into the WebLogic console (http://<appserver_name>:7001/console) as the user system.
2.   In the left frame, expand Otmv62 > Environment > Servers.
3.   Select the Oracle Transportation Management server (e.g. gc3-otm-machinename-62-wl).
4.   In the main frame, select the Monitoring tab.
5.   Then select the Performance tab.
6.   This screen shows you the following:
     - Request Throughput: The rate at which requests are processed. The higher the better.
     - Requests Waiting: The number of requests waiting to process. The lower the better.
     - Memory Usage: The amount of memory utilized by WebLogic. Keep in mind that this shows you how much memory WebLogic is using, within the memory allocated to the JVM. The amount of memory reported by your operating system will be higher, since the JVM reserves all of the memory that it has been configured to allocate, regardless of whether WebLogic is using it all or not. This was set during the Oracle Transportation Management installation. To get a true memory reading, you should force a garbage collection.

- Force Garbage Collection: This will force the JVM to free up all unused memory. This should be run in order to get a true memory reading from the main performance screen. Keep in mind that forcing a garbage collection will slow the server down and shouldn't be forced continually. The JVM usually manages garbage collections automatically as the server runs.

7. To monitor database connections, expand Services > JDBC > Connection Pools.

8. Select dbaPool.

9. Select the Monitoring tab and Monitor all instances of dbaPool.

10. This page shows you the initial, maximum, and high number of db connections. You should monitor whether the high number is at or near the maximum number of connections. If so, you should consider increasing the database connections in the dbaPool.

UNIX

On UNIX, you can monitor system performance using the ps, top, or sar. You should monitor general trends, such as CPU usage, memory usage, and memory swapping (out to virtual memory). All processes for Oracle Transportation Management should run under the user setup during the installation process. This can be helpful for identifying the processes.

The Apache service starts up multiple instances of Apache to handle incoming requests. Each of these will show up as "httpd", and can be monitored for performance statistics using general UNIX tools.

Tomcat

The Tomcat service starts up one JVM and shows up as "`<path>/ java –server –Xms1600m –`". Unfortunately this can be hard to find if you have multiple JVMs running on the same machine. Usually you can identify Tomcat by the amount of memory it is using, as normal Tomcat instances use large amounts of memory (1.6GB or more). You can monitor the performance statistics of Tomcat using general UNIX tools.

WebLogic

The WebLogic service starts up one JVM and shows up as "`<path>/ java –server –XX:MaxPer`" (differs slightly from the Tomcat instance). Unfortunately this can be hard to find if you have multiple JVMs running on the same machine. Usually you can identify WebLogic by the amount of memory it is using, as normal WebLogic instances use large amounts of memory (1.6 GB or more). You can monitor the performance statistics of WebLogic using general UNIX tools. You can also get detailed performance statistics from the WebLogic console. Follow the steps under the **Microsoft Windows** section in the **Operating System Monitoring Tools** section.

WebSphere

The WebSphere service starts up one JVM and shows up as "`<path>/ java –Xbootclasspath/p:`". Unfortunately this can be hard to find if you have multiple JVMs running on the same machine. Usually you can identify WebSphere by the amount of memory it is using, as normal WebSphere instances use large amounts of memory (1.6 GB or more). You can monitor the performance statistics of WebSphere using general UNIX tools. You can also get detailed performance statistics from the WebSphere console.

SAR

The "`sar`" command-line utility displays CPU activity. Running "`sar 10 10`" will run the `sar` command ten times in ten second intervals. The output will show basic CPU, %nice, %system, %iowait and % idle information.

```
Linux 2.4.21-37.0.1.ELsmp (spapp01)     01/30/2008
```

```
07:24:18 AM          CPU     %user    %nice    %system    %iowait    %idle
07:24:28 AM          all      2.88     0.00      0.28        0.00      96.85
07:24:38 AM          all      3.65     0.00      0.60        0.10      95.65
```

Oracle Transportation Management activity will show up under the %user column, high activity in the user column would be expected on an unresponsive system. If you see significant utilization in the system or iowait columns it could be an indication of other internal processes or swapping memory to and from disk that may need to be investigated further.

TOP

The "top" command-line utility produces a frequently-updated list of processes on most UNIX-based platforms. By default, the processes are ordered by percentage of CPU usage, with only the highest CPU consumers shown. The top command shows how much processing power and memory are being used. Running the "top" command will give you information on the system memory and swap space usage. The output on the screen is broken down into two sections, the top section tells you the current system time, up time, number of users and average load.

```
top – 08:57:58 up 97 days, 23:53, 3 users, load average: 0.06, 0.03,
0.00
Tasks: 79 total, 1 running, 78 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.0% us, 0.0% sy, 0.0% ni, 100.0% id, 0.0% wa, 0.0% hi, 0.0% si
Mem: 2049976k total, 2034984k used, 14992k free, 1848k buffers
Swap: 2048248k total, 1021960k used, 1026288k free, 14252k cached
```

It also shows you the total amount of memory available on the system, amount used and free. The same information is shown for the swap space allocated on the system, swap used, swap free and swap cached.

The bottom section of the output shows the top 20 processes listed by PID and user name, priority and other information. For many Oracle Transportation Management instances Java will likely show up as being the highest consumer of memory and CPU on the system.

PID	USER	PR	NI	VIRT	RES	SHRS	%CPU	%MEM	TIME+	COMMAND
1	root	16	0	4752	76	48 S	0.0	0.0	0:01.93	init
2	root	RT	0	0	0	0 S	0.0	0.0	0:01.84	migration/0
3	root	34	19	0	0	0 S	0.0	0.0	0:00.04	ksoftirqd/0
4	root	RT	0	0	0	0 S	0.0	0.0	0:01.86	migration/1
5	root	34	19	0	0	0 S	0.0	0.0	0:00.03	ksoftirqd/1
6	root	5	-10	0	0	0 S	0.0	0.0	0:00.01	events/0
7	root	5	-10	0	0	0 S	0.0	0.0	0:00.01	events/1
8	root	5	-10	0	0	0 S	0.0	0.0	0:00.00	khelper
9	root	13	-10	0	0	0 S	0.0	0.0	0:00.00	kacpid
33	root	5	-10	0	0	0 S	0.0	0.0	0:00.00	kblockd/0

Note: If you see other processes that are using significant system resources you should ask the administrator to follow up.

Oracle Transportation Management Tools

Configuration Data Collection

The Configuration Data Collection tool captures important configuration data that can be used in diagnosing a performance issue. Users have the option of displaying the information on the screen, or they can have the data collected to them via FTP or via email. Information sent via FTP and email is encrypted, but this can be disabled via the following properties:

```
glog.diag.output.ftp.encrypt=false
glog.diag.output.email.encrypt=false
```

Files that are not encrypted and sent via email are given the .ZOP extension.

In addition to this data, there is also an option to Export the Environment. This will send you the most recent access log from apache, config.xml.original from weblogic, console.log from Tomcat and WebLogic, Error.log from the Oracle Transportation Management logs directory, glog.exception.log, glog.properties, httpd.conf, mod_jk.log and mod_jk.conf from apache, tomcat.conf, weblogic.conf, weblogic.sh, workers.properties, several class files and the patch notes from the installed releases on the system. The list of files to be exported is stored in a new properties file called `glog.environment.export.properties`. You can also add files to the list on the fly by adding an additional `glog.environment.export` entry via the properties servlet.

Performance Data Collection

The Performance Data Collection utility can be used to capture performance statistics that can be used in diagnosing a performance issue. Users have the options of displaying the information on the screen or having it sent via FTP or email to a contact they have defined in the system.

Application Server Tools

Administration Console

The application servers used by Oracle Transportation Management have administration consoles that provide various monitoring utilities. This document will not cover the monitoring utilities, where they are, or how to use them. Please see your application server documentation for the specific details on accessing the console and how to use the utilities that are provided. However, there are certain tips that we will provide.

Monitoring Deployed Enterprise Java Beans

The administration console for the application server should have the ability to monitor Enterprise Java Beans (EJB). Oracle Transportation Management only uses stateless Enterprise Java Beans so keep this in mind when using the administration console to monitor these. Monitoring really only needs to be for heavily used EJBs like `workflow.WorkflowSession`.

JVM Tools

Console Logs

For the web and application servers, system monitoring utilities will only be partially effective since the memory of a Java application is typically static and limited by the Java heap size. They will be useful if multiple applications are being run on a single machine and the total physical memory may be limited. If the entire Java heap cannot be contained in physical memory performance issues will be very likely due to memory swapping during JVM garbage collection.

Memory monitoring within the Java heap can be performed by various Java monitoring tools or by reviewing the verbose garbage collection output in the console.log files. You should have garbage collection logging enable. This is done via one of the parameter settings in the "`.conf`" file for Tomcat, WebLogic and Oracle Application Server (OAS). Console logs for the web server and the application server should be reviewed to understand the memory allocation and garbage collection.

Thread Dumps

Thread dumps are critical when diagnosing almost any performance issue. They are particularly useful when diagnosing an issue where the application server is unresponsive. Thread dumps capture the current state (executing, blocked, or inactive) and a stacktrace indicating the exact code that is being executed for every thread. Thread dumps should be taken in a set of three thread dumps taken consecutively at 30 second intervals. If the performance issue continues for an extended period of time, it is extremely useful to capture multiple sets of thread dumps.

If there is very little activity in the thread dumps the cause of the performance issue may be fairly apparent. However, if there are many active threads it may be very difficult to deduce the exact cause of the performance issue. In this case, capturing the thread dumps during multiple occurrences of the issue is necessary.

Heap Dumps

A heap dump is a dump of all the objects that are currently in the Java heap. The heap dump is a critical piece of information when diagnosing a memory related issues. The heap dump can be captured and then be analyzed off-line. Unfortunately the support and technique for capturing a heap dump varies by JVM and version of the JVM. Oracle Support will help with capturing a heap dump when it is necessary.

Capturing JRockit Mission Control JRA Files

Mission Control is a run-time analysis tool that is included with the JRockit JVM. This utility captures and analyzes JRA (Java Run-time Analysis) files. This utility can be used to analyze performance, memory heap usage, garbage collection, and thread latency. Mission Control imposes minimal overhead on the running system and therefore can be used to analyze issues in a production environment. If the issue you are experiencing only occurs sporadically it can be helpful to capture JRAs both prior to the problem happening and while it is happening for comparison purposes. The following steps can be used to capture a JRA.

1. Find the Process Id (PID) of the application server and keep it for the next step. You can do this by running this command: `<OTM_INSTALL_DIR>/jdk/bin/jrcmd –p`

    *   The `<OTM_INSTALL_DIR>/jdk/bin/jrcmd –p` command will list all of the java processes running.

2. You can then make a jra recording from command line by running this command after replacing the information in between the :

```
<OTM_INSTALL_DIR>/jdk/bin/jrcmd <PID> jrarecording time=300
filename=<OTM_INSTALL_DIR>/logs/perfIssue.xml.zip
threaddumpinterval=30s
```

Please note the time is in seconds and this can be modified to any number of seconds. The directory and filename can also be changed.

The JRA utility will write to the file for the entire time interval specified. Every time you run it, you would probably want a new filename to help differentiate between the different records. You can always just save the file off somewhere else and rename it. JRA will record until the time interval is over, and then stop. There is no manual intervention needed to turn off the profiler (JRA recording).

Database Monitoring Tools

Oracle Enterprise Manager includes several tools which are critical to monitoring the performance of the Oracle Database. Two of these tools are Automatic Workload Repository (AWR) and Automatic Database Diagnostic Monitor (ADDM), both of which are included in the Diagnostics Pack.

Automatic Workload Repository

By default snapshots of the relevant data are taken every hour and retained for seven days. The data is accessed via a report by running `awrrpt.sql` or `awrrpti.sql`. The automated workload repository administration tasks have been included in Enterprise Manager. The Automatic Workload Repository page is accessed from the main page via Administration > Workload Repository (under the Workload section).

Automatic Database Diagnostic Monitor

ADDM analyzes data in the Automatic Workload Repository to identify potential performance bottlenecks. For each of the identified issues it locates the root cause and provides recommendations for correcting the problem. An ADDM analysis task is performed and its findings and recommendations stored in the database every time an AWR snapshot is taken.

The best place to start viewing ADDM reports is Enterprise Manager. The Performance Analysis section on the Home page is a list of the top five findings from the last ADDM analysis task. Specific reports can be produced by clicking on the Advisor Central link, then the ADDM link. The resulting page allows you to select a start and end snapshot, create an ADDM task and display the resulting report by clicking on a few links. Alternatively the `addmrpt.sql` script can be used to create an ADDM report from SQL*Plus.

Network Monitoring

Due to the heavy traffic between the client, web, application, and database, a small change in network latency can have a significant impact on system performance. System monitoring tools will be helpful in diagnosing network performance issues. Generally speaking though, network performance issues are one of the most difficult to identify. Maintaining benchmarks of good network performance that can be referenced during performance issues is a good practice.

Diagnosing a Performance Issue

Steps for Diagnosing a Performance Issue

1. Determine the scope of the problem.
   - Is this issue affecting one user or many users?
   - Can new users login to Oracle Transportation Management?

- Are agents and integrations not processing?
- Record the exact timeframe of when the issue occurred.

2. Monitor CPU, I/O, and network activity.
   Use a system monitoring tool to capture and review whether there is abnormally high CPU, network, or I/O activity on the web/app/database.

3. Check the exception logs.
   Capture and review the exception.log file for unusual exceptions.

4. Review logging that is enabled.
   Review what logging is enabled via Logging Overview page and potentially suppress all logging.

5. Review workflow activity.
   Capture and review Event Diagnostics servlet or Event Queue Application Performance Diagnostic for a backlog of unprocessed events.

6. Review console.log files.
   Capture and review the console.log file looking for excessive garbage collection activity.

7. Capture thread dumps.
   Perform multiple sets of three thread dumps (30 seconds apart) on the web/applications servers.

8. Restart the server.
   If the problem persists, a restart may be necessary to resolve the issue.

9. Review change log for recent configuration changes.
   Most performance problems are caused by a recent configuration change. Review the log of recent changes and provide this information to Support.

Additional steps for Database Related Issue

10. Capture AWR/ADDM reports.

11. Review reports for SQL which may need tuning.

12. Check for stale and partial database statistics.

9. Performance Tuning

JVM

The JVMs supported by Oracle Transportation Management vary by platform. Each JVM has a set of parameters which can be used to tune the performance of the JVM. Most of these parameters affect the memory management process referred to as garbage collection. Oracle Transportation Management ships with a default set of these parameters to be used as a starting point for JVM tuning. Changing these parameters can have a significant impact on the overall performance of the system. Therefore, changes should be well documented and made incrementally with thorough performance testing.

Changing Memory Settings

As your server load varies, you may need to change the amount of memory that is allocated to the Oracle Transportation Management instance. You can change the amount of memory allocated to both Tomcat and WebLogic/WebSphere.

Tomcat is the Java servlet server that Oracle Transportation Management uses. Increasing the amount of memory that it can utilize allows more Oracle Transportation Management concurrent users and improves the speed of some integration tasks.

WebLogic/WebSphere is the application server that Oracle Transportation Management runs within. Increasing the amount of memory that is has improves performance of business logic functions.

After making any memory changes, you need to stop and restart Oracle Transportation Management, as described starting on page in the chapter **Starting and Stopping the Oracle Transportation Management Servers**. The instructions below describe how to make these changes.

UNIX and Microsoft Windows

To change the amount of memory that Tomcat uses, edit <otm_install_path>/tomcat/bin/tomcat.conf and modify the following lines:

```
jvm.arg=-Xms[TOMCAT_MEMORY]m
jvm.arg=-Xmx[TOMCAT_MEMORY]m
```

[TOMCAT_MEMORY] will be some integer value that represents the amount of memory (in megabytes) to use. For instance, if you changed this to 2.2GB, the line would read:

```
jvm.arg=-Xms2200m
jvm.arg=-Xmx2200m
```

 [WebLogic] To change the amount of memory that WebLogic uses edit
`<otm_install_path>/weblogic/weblogic.conf`:

```
jvm.arg=-Xms[WEBLOGIC_MEMORY]m
jvm.arg=-Xmx[WEBLOGIC_MEMORY]m
```

If you change this value to 2.2GB, it would look like:

```
jvm.arg=-Xms2200m
jvm.arg=-Xmx2200m
```

[WebSphere] Under WebSphere, this change would be made to the file <otm_install_path>/websphere/profiles/default/config/cells/__APP_SERVER__Node01Cell/nodes/__APP_SERVER__Node01/perftuners.xml (where __APP_SERVER__ is your application server's host name):

```
<param name="MaxMX">[APP_JVM_MEMORY]</param>
```

If you change this value to 2.20GB, it would look like:

```
<param name="MaxMX">2200</param>
```

In all cases, restart your Oracle Transportation Management instance to make these changes take effect.

Apache/Tomcat Webserver

If your installation is going to have a large number of simultaneous users it may be necessary to tune some Apache and Tomcat settings. It is necessary to do performance testing and monitoring in order to tune these settings for an Oracle Transportation Management installation. Depending on the volume of users and the type of activity performed by these users it may also be necessary to setup additional webserver instances.

1. maxThreads in OTM/tomcat/conf/server.xml:

The maximum number of request processing threads to be created by this connector, which therefore determines the maximum number of simultaneous requests that can be handled. If not specified, this attribute is set to 200. If an executor is associated with this connector, this attribute is ignored as the connector will execute tasks using the executor rather than an internal thread pool.

For example:

```
<Connector excecutor="tomcatThreadPool" address="10.60.15.158"
maxThreads="1000" port="8009" protocol="AJP/1.3" URIEncoding="UTF-8"/>
```

1. maxClients in OTM/apache/conf/http.conf

The following properties in the http.conf file control the number processes/threads that will be created within Tomcat. Connection attempts beyond these limits will normally be queued, up to a number based on the "ListenBacklog" directive.

For example:

- # StartServers: number of server processes to start
- # MaxClients: maximum number of server processes allowed to start
- # MinSpareThreads: minimum number of threads which are kept spare
- # MaxSpareThreads: maximum number of threads which are kept spare
- # ThreadsPerChild: number of threads per server process
- # ServerLimit: maximum number of server processes
- # MaxRequestsPerChild: maximum number of requests a server process serves

  <IfModule worker.c>

```
StartServers       10
ServerLimit        40
MaxClients         1000
MinSpareThreads    25
MaxSpareThreads    75
ThreadsPerChild    25
MaxRequestsPerChild 0
```

Application Server

Logging

Although logging is a vital function in Oracle Transportation Management, excessive logging is very common cause of poor performance. This is particularly true of bulk planning processes. The Log Files and Logging Overview pages can be used to review what logs are currently enabled. You can also temporarily disable all logging by setting the following property:

```
glog.log.suppressAll=true
```

Setting this property can be a quick method of determining whether logging is the cause of a performance issue.

> **Note**: Log File and Logging Overview pages on the Configuration and Administration menu can only be used to display and configure details about log files enabled on the application server. Logging on the web server can only be controlled by glog.properties.

Another way to determine if logging is impacting performance is to look at the rate at which log files are turning over. On the server running Oracle Transportation Management, you can perform a quick scan of the directory containing the Oracle Transportation Management logs files. If this reveals a log file that is turning over a 10MB log file several times per second, it is very likely to be impacting performance.

LogIds with a suffix of "Debug" or "Details" have the potential to log significant amounts of data and should be avoided unless directed to be by Support. Ad-hoc logs are the most dangerous because they generate logging regardless of the user logged in. On the other hand, User logs only write to the log file when that particular user is logged in and using Oracle Transportation Management. In some scenarios user logs can still have a significant impact on performance, even if that particular user is not logged in. This logging happens because there is a certain amount of overhead in generating a log message. The overhead occurs before Oracle Transportation Management determines, based on the logged in user, that it does not need to write the message to the log file. For this reason, having many user logs with detailed logging enabled can have a significant impact on performance.

Enterprise Java Beans

The Enterprise Java Bean caches are maintained by the J2EE container (WebLogic, Oracle Application Server, etc.) in which Oracle Transportation Management is running.

Entity Bean Caches

Under WebLogic, there is a separate cache for each entity bean and it has a configurable size and timeout. Under Oracle Application Server, all entity beans are stored in one global cache. In general, the dependence of Oracle Transportation Management on entity beans has been greatly reduced. Loading data through entity beans has been replaced by direct loading via SQL queries. In addition, most static data is loaded into a business object cache so the size and contents of the entity bean cache are rarely a cause of performance issues.

Tuning Bean Caches

These are the steps to change the cache size for an EJB when using Weblogic.

1. Log on to the physical server where the application server is installed.
2. Navigate to `<otm_home>/weblogic/domains/otm/applications/otm/GC3App.ear`
3. Back up the exiting GC3App.ear file before making any changes.

---

4. The Bean count that needs to be increased can usually be identified from the error message. Be careful when selecting the bean; make sure that the right bean is selected. The example below is for the Contact bean.

5. Use these commands to extract the Contact.jar file and the weblogic-ejb-jar.xml from it.

```
jar -xvf GC3App.ear Contact.jar
jar -xvf Contact.jar META-INF/weblogic-ejb-jar.xml
```

6. Edit the `/weblogic-ejb-jar.xml` file and update the max beans in cache setting.

   a. vi META-INF/weblogic-ejb-jar.xml

   b. Look for this line:

   ```
   <max-beans-in-cache>350</max-beans-in-cache>
   ```

   c. Change the value.

   d. Save changes to the /weblogic-ejb-jar.xml

7. Add the modified `/weblogic-ejb-jar.xml` back into the Contact.jar and GC3App.ear file using the following commands.

```
jar -uvf Contact.jar META-INF/weblogic-ejb-jar.xml
jar -uvf GC3App.ear Contact.jar
```

8. Re-start the web and application servers for the changes to take effect.

9. Retest the issue.

It is recommended that you place the GC3App.ear file on a local system and open it with Winzip, then open the .jar file with Winrar. Edit the weblogic-ejb-jar.xml with notepad; editing with MS Word is not recommended because it adds extra characters when the document is saved. Update the file in the .jar and then move the .jar into the ear file. You may only be able to open the file in WinZip to get the name of the .jar file and then use the instructions listed above to make the actual change.

Oracle Transportation Management Business Object Caches

The Oracle Transportation Management Business Object caches are maintained by Oracle Transportation Management. The majority of Oracle Transportation Management Business Objects caches use a Least Recently Used (LRU) strategy to maintain the cache. When an LRU cache reaches its maximum, a one-for-one exchange is made for the new object and the least recently used object in the cache. Most static data used by Oracle Transportation Management business logic is maintained in one of these caches. The application-tier utility page is available under the DBA Diagnostics menu to review statistics on these caches.

The size of a Business Object Cache can have a significant impact on performance. The efficiency of a cache is measured by its hit ratio. A low hit ratio is a possible indication of an undersized cache. If a cache has reached its capacity and the hit ratio is low (less than 0.80), performance may be impacted. Increasing the maximum size of this cache may increase system performance. Temporary changes can be made to the cache using the diagnostic screen, but the changes will revert to the default upon restart. To permanently change the size of a cache the appropriate glog.property must be set. For example, the size of the Rate Offering cache is set by the following property:

```
glog.cache.TRateOfferingCache.capacity=2000
```

It is important to note that increasing the size of the cache has the adverse affect of increasing memory usage, so changes should be done incrementally and with thorough testing.

Business Number Generator (BNG) Contention

The Business Number Generator is an Oracle Transportation Management mechanism for creating IDs based on a complex set of business rules. Since the IDs are based on a sequence, it is necessary to generate the IDs one at a time in order to prevent duplicate IDs. For this reason, a process may have to wait for another process to finish generating an ID. This waiting can manifest itself as a

performance issue. The impact is even more significant in a scalability environment since the synchronization must be coordinated across application servers.

It is a good practice to eliminate unnecessary use of BNG. For example, the Bill of Lading 'BM' Shipment reference number can be disabled with the following property:

```
glog.shipment.createBMRefnum=false
```

The Oracle Database sequence number generator has less overhead than the BNG and should be used instead of the BNG when possible. For example, the `s_ship_unit_gid` can be changed from the BNG to an Oracle Database sequence number by setting the following property:

```
glog.server.bngenerator.oracleSequence.xid.S_SHIP_UNIT_XID.DEFAULT=S_SH
IP_UNIT_SEQUENCE
```

DirectSQLUpdate

DirectSQLUpdate is an agent action which allows you to update an attribute of an Oracle Transportation Management business object using a SQL statement. It is equivalent to a database trigger, but the agent action is capable of refreshing Oracle Transportation Management business caches to reflect the modified data. Performance tuning of SQL statements in DirectSQLUpdate is important. It is also important to optimally configure the use of the cache refreshing options in the action. For any given DirectSQLUpdate, you have the choice to refresh the cache or not. When choosing to refresh the cache you have the further option to Parse SQL. If you have multiple DirectSQLUpdate actions which update the same Oracle Transportation Management business object in a single agent, you should configure the agent to only refresh the cache on the last update. An exception to this rule is if actions in the agent itself are dependent upon the modified data.

The option to Parse SQL should be used whenever possible. By default, the refresh logic needs to refresh all child data for the business object that is being updated. For instance, a DirectSQLUpdate action in a shipment agent needs to refresh the related shipment, shipment stops, ship units, etc. When SQL parsing is enabled, the agent will attempt to determine the specific business object being updated, and only refresh this object.

Workflow Thread Tuning

Oracle Transportation Management workflow is based on a set of business topics and workflow thread groups. A Topic is the Oracle Transportation Management terminology for a particular workflow process. For example, AutoMatchInvoice is a topic. Each Topic is associated with a thread group by a glog.property. For example, the following property associates the AutoMatchInvoice Topic with the agentPlanning Thread Group.

```
glog.workflow.topic.group.glog.server.agent.business.invoice.AutoMatchI
nvoice=agentPlanning
```

A Thread Group is a set of workflow threads dedicated to processing a set of workflow topics. Each thread group is given a number of threads that can be used to simultaneously process topics. When all threads in a group are busy processing a topic, all additional topics are placed in a queue until a thread becomes available. Oracle Transportation Management computes statistics for the amount of time a topic spends waiting to be processed and the amount of time spent processing the topic. These statistics can be reviewed with the Event Diagnostics page available on the DBA Diagnostics menu.

A long average wait time and/or a backlog of queued events may indicate a need to increase the number of threads in a thread group. It is possible to temporarily add a new thread in the Event Diagnostic page. Permanent changes to thread groups must be made with a change to the corresponding glog.property. It is important to note that properties like the number of threads for a thread group need to be explicitly removed in order to override the value. The whole key and value must match the whole key and value that is being overridden. For example:

```
!remove glog.workflow.topicGroup= agentPlanning,3
glog.workflow.topicGroup= agentPlanning,4
```

This will change the number of agentPlanning threads from 3 to 4, assuming the default value was 3.

Adding more threads will not always increase performance. Another possibility is that some slower topics are in the same thread group as a higher volume, but short duration topic. The faster topics may be prevented from running by the slower topics. Adding more threads to this thread group may not have a significant impact on performance. However, moving the slow topics to a new thread group may improve overall system throughput by allowing the shorter topics to complete more quickly. The following properties are an example of moving a topic to a new thread group:

```
glog.workflow.topicGroup=slowEventGroup,1
glog.workflow.topic.group.glog.server.agent.business.shipment.SlowTopic
= slowEventGroup
```

Automatic Timeout for Stuck Threads

For any event queue, we can set an automatic timeout that effectively kills the thread and redirects work to a new thread of the same name. To use this feature, you need to set:

```
glog.eventQueueMonitor.polling=<polling interval in seconds>
glog.eventQueueMonitor.timeout.<queue>=<timeout in seconds>
```

The polling interval should generally be 1/2 the smallest timeout.

For example:

```
glog.eventQueueMonitor.polling=60
"glog.eventQueueMonitor.polling.transport - SMTP"=120
"glog.eventQueueMonitor.polling.transport - HTTP"=240
```

In this example, the system will check every minute on the health of the SMTP and HTTP threads. If an SMTP thread has been active longer than 2 minutes, it is interrupted and another SMTP thread is created in its place. If an HTTP thread has been active longer than 4 minutes, it is interrupted and another HTTP thread is created in its place.

> **Note**: The double quotes around the property key since the queue names have spaces.

Multi-threading Bulk Plan Processing

Parts of the bulk plan process can be configured to process multiple threads simultaneously. This behavior is enabled by the following property:

```
glog.task.on=true
```

Adding more threads to these thread groups may increase the performance of a very large bulk plan. The following are the two key properties to control this functionality.

```
glog.workflow.topicGroup=planningCommit,2
glog.workflow.topicGroup=planningBuild,2
```

As described previously these values can only be overridden by using a `!remove` command in the property file before specifying the new value. Please include your changes in the corresponding section of the glog.properties file of the Oracle Transportation Management application server as in the example below:

```
For example:
#-------------------------------------------------------------------
# Custom Thread Properties - Beginning
```

```
#------------------------------------------------------------------
# Place all thread changes and properties here.
# This section will be used during GC3 Upgrades
#------------------------------------------------------------------

# Added by OTMUser on 07/15/2010
!remove glog.workflow.topicGroup=integration
glog.workflow.topicGroup=integration,20

#------------------------------------------------------------------
# Custom Thread Properties - End
#------------------------------------------------------------------
```

There are additional properties to control the behavior of the multi-threading. The default values and a description of each of these properties are available in the glog.task.properties file.

Database

Analyzing Tables/Gathering Statistics

The Oracle Transportation Management database performs best when optimizer statistics are gathered using scripts provided by Oracle Transportation Management. Oracle Transportation Management scripts gathers statistics by performing full compute for all the objects including indexes, columns with histograms, partitions etc.

Oracle Database also has automatic statistics gathering job GATHER_STATS_JOB, which is enabled by default when a database is created, or when a database is upgraded from an earlier database release. This job gathers statistics on all objects in the database that have missing statistics or stale statistics (stale - the underlying object has been modified significantly, i.e. more than 10% of the rows). The Scheduler runs this job during a maintenance window (by default, 10pm to 6am). This automatic job gathers statistics by estimate and not in as much detail as the Oracle Transportation Management job. Over time, it tends to override detailed statistics gathered by the Oracle Transportation Management job for fast growing tables, hence automatic job should be replaced by Oracle Transportation Management statistics job.

Below are the steps to verify that statistics are gathered by the Oracle Transportation Management statistics job.

1. In SQL*Plus, as user **GLOGOWNER**, run:

    ```
    select min(last_analyzed) from user_tables;
    ```

If it returns a date older than two weeks then Oracle Transportation Management statistics job is not running since that day or it is not scheduled. In that case, schedule a weekly job using any job scheduler, and have it run <otm_install_dir>/oracle/script8/gather_table_stats.sql during a low-activity period or off-peak hours.

> **Note**: gather_table_stats.sql also runs gather_column_histograms.sql, so both should be in same directory.

2. Disable Oracle's automatic job. In SQL*Plus, as user **SYS** (or any DBA account), run:

    ```
    BEGIN
    DBMS_AUTO_TASK_ADMIN.DISABLE(
    client_name => 'auto optimizer stats collection',
    operation => NULL,
    window_name => NULL);
    END;
    /
    ```

```
    commit;
```

This will ensure Oracle Transportation Management database has up to date statistics with full compute on all objects.

Database File Storage Performance

The two most commonly used forms of database file storage are: Storage Area Network (SAN) and Network-Attached Storage (NAS). NAS provides both storage and filesystem access whereas SAN provides only block-based storage and leaves file system concerns on the client side. Typical SAN protocols are SCSI, Fibre Channel, iSCSI, ATA over Ethernet, or HyperSCSI. A NAS on the other hand communicates over Ethernet and TCP/IP. Because NAS communicates over standard network protocols there is a stronger possibility of a poorly performing network affecting the data access rate. The latency from a database server to the NAS device should be less than 150 microseconds (0.15 ms) and preferably less than 100 microseconds (0.10 ms).

Full Export and Re-import

As data is added to tables the database may need to add additional database extents to store the data and indexes. The general consensus among database administrators is that an increasing number of extents does not affect performance of the database. However, in practice performance gains have been seen by completing a Full export and import.

Checking for Stale and Partial Oracle Database Statistics

The following query can be used to check for stale schema statistics:

```
select table_name,  last_analyzed,num_rows, sample_size from dba_tables
where last_analyzed is not null and owner = 'GLOGOWNER'  and
last_analyzed < (SYSDATE - 7) order by last_analyzed asc
```

Statistics having a 'last_analyzed' date older than one week, should be updated.

The following query can used to check if a partial analyze was run recently.

```
select table_name,  last_analyzed,num_rows, sample_size from dba_tables
where last_analyzed is not null and owner = 'GLOGOWNER'  and
sample_size < num_rows order by last_analyzed desc
```

A sample_size less than the num_rows is an indication that 'estimated' statistics were used. These tables should be re-analyzed with full statistics.

Maintaining Oracle Database Statistics

The complex queries run within Oracle Transportation Management depend heavily on the Oracle Database schema statistics. It is important that all tables and indexes are analyzed at least weekly by analyzing the whole table/index, not a sample. The Oracle 10g Database has an automatic statistics gathering job GATHER_STATS_JOB, which is enabled by default when a database is created, or when a database is upgraded from an earlier database release. This job gathers statistics on all objects in the database which have missing statistics or Stale statistics. Stale - the underlying object has been modified significantly (more than 10% of the rows). The Scheduler runs this job during a maintenance window.

This default scheduled job gathers statistics by estimate. This method does not perform the same detailed analysis as the GATHER_TABLE_STATS.sql script that is provided for Oracle Transportation Management. Over time, the Oracle 10g Database default GATHER_STATS_JOB tends to override detailed statistics gathered by Oracle Transportation Management script for fast growing tables. For

this reason, the GATHER_STATS_JOB should be disabled. Please consult MetaLink note 467668.1 for more details on this subject.

Customers should schedule the Oracle Transportation Management scripts to run at least once a week. A common performance problem occurs just after a customer goes live. During this period a significant amount of data is loaded and/or created by Oracle Transportation Management. It may be necessary to monitor the growth of tables by comparing the actual number of rows in the table to the number of rows captured in the schema statistics. If the table has had significantly growth (> 10%), it is recommended that the table and any related indexes be analyzed.

To run the Oracle Transportation Management provided gather table stats script manually go to the Application servers `<otm home>/glog/oracle/script8` directory. From the script8 directory log into the database as "glogower" using SQLPlus and run the following SQL command.

```
@gather_table_stats.sql
```

> **Note**: Clients should be warned to schedule this process to run when activity on the system is low due to potential performance impacts on the system.

SQL Tuning

Another common cause of performance issues is slow performing queries and high number of executions of queries. There are a number of mechanisms that can be used to identify these issues. The most effective is the AWR/ADDM reports provided by the Oracle Database. Another technique is to temporarily enable SQL logging in Oracle Transportation Management and review the logs for repetitive SQL statements.

Once a poorly performing SQL statement is identified, an Explain Plan should be performed. Within Oracle Transportation Management this can be performed in SQL Servlet by pre-pending the SQL statement with "explain plan for". Performing an explain plan outside of Oracle Transportation Management will supply more information about the query. When running it outside Oracle Transportation Management, it is important to execute the stored procedure call (`vpd.set_user()`) before running the query. Otherwise the explain plan will not be evaluated with VPD applied as it would be when run within the context of Oracle Transportation Management.

Determining the Origin of a SQL Statement

SQL statements that are part of Oracle Transportation Management originate from Oracle Transportation Management base code, custom saved queries, and reports. The performance tuning of custom reports and saved queries are the responsibility of the customer. Saved queries provide a mechanism to create a query which can be used in various parts of Oracle Transportation Management including Finders (Search pages), Business Monitors, and Automation Agents. The following query can be used to search for a SQL statement which may be generated from a saved query:

```
select saved_query_gid from saved_query where sql_find_all like
'%querytext%'or sql_check one like '%querytext %'
```

The 'querytext' should be just part of the SQL statement, since the SQL statement is slightly modified depending on where in Oracle Transportation Management the SQL statement is being used. For instance, the Business Monitor prefixes all SQL statements with "`select count(*)…`".

INIT.ORA Parameters

The values of INIT.ORA tuning parameters can have a significant impact on performance. The Oracle Transportation Management Administration Guide provides the latest set of recommended settings for Oracle Transportation Management. These settings are guidelines that have been established based on feedback from various customer scenarios. There is not one set of parameters that will be optimal for

all installations. These parameters need to be tuned for each Oracle Transportation Management instance by an experienced Database Administrator.

Tuning a RAC Database Installation Using Services

RAC is the multi-node clustering capability for the Oracle Database. Oracle Transportation Management is certified to work in a RAC configuration. A RAC configuration will allow the database to scale, particularly in a situation where the database experiences high CPU. The default configuration of RAC is to load-balance the work across all nodes in a cluster. In some scenarios this configuration may result in significant contention between the RAC nodes due to the global buffer cache. Evidence of this issue can be seen in the AWR/ADDM reports with high wait times due to the "gc cr request" and/or "gc buffer busy" events.

An Oracle Transportation Management large, multi-threaded bulkplan process is a scenario that may experience this problem when attempting to commit a large number of shipments across multiple nodes in a cluster. It is possible to achieve a performance improvement by dedicating a particular Oracle Transportation Management process to a sub-set of the nodes in the cluster in order to reduce contention. This can be accomplished by defining services on the RAC nodes and then associating the service in the database connect string of particular nodes in the Oracle Transportation Management Application Server. This assumes that Oracle Transportation Management has been configured to use Oracle Transportation Management scalability and that particular processes are being handled by particular applications servers in the Oracle Transportation Management cluster.

The following is an example of how these services could be defined and used:

Service Name	Primary Instance(s)	Available Instance(s)
otm_oltp	rotm021, r0tm022	rotm023
otm_batch	rotm023	rotm021, rotm022

To use these services, users need to configure new connect strings in the corresponding glog.properties file for each application server like the following:

otm_oltp

```
glog.database.dbaOnly.t2client.connectionURL=jdbc:oracle:thin:@(DESCRIP
TION=(ADDRESS=(PROTOCOL=TCP)(HOST=otm-rac01-
vip)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=otm-rac02-
vip)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=otm-rac03-
vip)(PORT=1521))(LOAD_BALANCE=yes)(CONNECT_DATA=(SERVER=DEDICATED)(SERV
ICE_NAME=otm_oltp)))\
```

otm_batch

```
glog.database.dbaOnly.t2client.connectionURL=jdbc:oracle:thin:@(DESCRIP
TION=(ADDRESS=(PROTOCOL=TCP)(HOST=otm-rac01-
vip)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=otm-rac02-
vip)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=otm-rac03-
vip)(PORT=1521))(LOAD_BALANCE=yes)(CONNECT_DATA=(SERVER=DEDICATED)(SERV
ICE_NAME=otm_batch)))
```

Changing Database Pool Size

As your server load varies, you may need to change the number of connections in the database pool. This may increase the performance of your Oracle Transportation Management instance. To determine if you have a database connection bottleneck, follow the instructions provided under the **Microsoft Windows** section in the **Operating System Monitoring Tools** section.

After making any database pool changes, you may need to stop and restart Oracle Transportation Management, as described in the chapter **Starting and Stopping the Oracle Transportation Management Servers**. Also, before you change your pool size, consult a qualified Oracle DBA and ensure that your database can handle the new number of connections. If you set your pool connections higher than your database can support, Oracle Transportation Management will not be able to startup and function correctly. The instructions below describe how to make these changes.

1. Log into your Oracle Transportation Management system (`http://<otm_webserver_name>`) as the user DBA.ADMIN.
2. In the left frame expand Configuration and Administration > System Administration > Data Source Manager.
3. Select search.
4. Select the desired database pool.
5. You can now edit the initial capacity, maximum capacity, and capacity increment. If resources permit, you should set the initial and maximum capacity to the same value and save some overhead involved with expanding and shrinking the database pool.

Purging Excess Reports from the REPORT_LOG Table

Purge Process for UNIX

1. Log into the UNIX machine on which the report server is installed.
2. Run the shell script

   `purgeRepLog.sh`

The script prompts for the nine parameters that are listed later in this section. Based on the parameters passed, the script deletes the records from the database and deletes the PDF files.

> **Note**: The USERID used to logon to the UNIX machine should have delete permissions on the PDF files created by the report server.

Purge Process for Windows

1. Log into the Windows machine on which the Report server is installed.
2. Run the batch file purgeRepLog.bat with the parameters that are list later in this section (in the order specified).

Based on the parameters passed, the script deletes the records from the database and deletes the PDF files.

Parameters

These parameters should be passed to the procedure in the order listed below:

- Start Date – this is the starting date from which the records are to be purged. % can be used to imply all dates. Default value is NULL. The format for the field is DD-MON-YYYY.
- End Date – this date is to date till which the records are to be purged. % can be used to imply all dates. Default value is NULL. The format for the field is DD-MON-YYYY.

- Domain Name – Domain name for which the records and files are to be purged. % indicates all domains. Default value is NULL
- GL User – GL User ID for which the records and files are to be purged. '%' can be used to indicate all users. Default value is NULL
- Report GID – the Report GID for which the records and files are to be purged. % indicates all the Report GIDs. Default value is NULL
- Job Number – the Job Number for which the records and files are to be purged. % indicates all the job numbers. Default value is NULL
- DB User Name – the database user name
- DB Password – the database password
- Connect string – the database SID or the connect string. Default value is the value of the environment variable $ORACLE_SID.

Manually Purging Transmissions from the Database

Manually purging is an alternative to truncating the partitioned tables. Learn more about the scheduled jobs that truncate partitioned tables later in the chapter. Truncating data is much faster than purging, but purging allows you to more exactly specify what data to purge.

Transmission_purge deletes data associated with xml blobs used for integrating Oracle Transportation Management with external systems. In general, the xml blobs are temporary, so there is no downside the getting rid of them.

Your DBA or System Administrator should set up a nightly or weekly background job to run purges as described below.

Manually Purge Transmissions

To purge the transmission tables:

1. Log into sql*plus as **glogowner**. (`glogowner/{password}@{tnsname}`)
2. `sql>exec domainman.transmission_purge(<age_in days>, <total_minutes>, <batch size>);`

   Note: The old procedure transmission purge "`domainman.transmission_purge(<age_in days>, <where clause>,<totalminutes>,<keep status clob flag>, <batch size);`" is deprecated (still workable).

   Tender collaboration records (TENDER_COLLABORATION), which have their own partition, are not able to be purged or truncated with this procedure. They can be purged with associated shipments if the parameter PURGE_TENDER_TRANSACTION=TRUE or manually purged with sql command "`exec domainman. orphan_tender_purge( <total_minutes>,<batch size>);`"

   Transmissions with shipment events (IE_SHIPMENT_STATUS) are purged just like other elements.

The transmission purge process accepts the following arguments:

Age in days (required): the age of the transmissions that you want to purge.

If you have no idea of how many transmissions you have that are older than 30 days (for example), you can do the following query:

```
sql> select count(*) from i_transmission where sysdate-create_date >
30;
```

Total Minutes (defaults to 60): The maximum number of minutes you would like the total purge process to take. After processing a set of rows, the procedure will check the total time. If it exceeds the inputted time, the process will wrap up the existing task, and stop. This is helpful if you have a tight time schedule in which you can perform this task.

Batch size (default 5000): Limits the number of transmission records that should be processed at a time. The smaller your rollback segment, the lower this number should be set. If you get a rollback segment error, rerun the process with a lower batch size specified.

The transmission purge process deletes records from the following tables:

```
I_TRANSMISSION
I_TRANSMISSION_ACK
I_TRANSMISSION_PGROUP
I_TRANSMISSION_REFNUM
I_TRANSMISSION_REPORT
I_TRANSACTION
I_TRANSACTION_ACK
I_TRANSACTION_DETAIL
I_TRANSACTION_REFNUM
I_LOG
I_LOG_DETAIL
```

Examples:

```
exec domainman.transmission_purge(90,30,1000);
```

deletes all transmission records (for above tables) that are older than 90 days except tender. The process would run for 30 minutes and would process 1000 records at a time.

> **Note**: Transmission_purge is an expensive operation. It should be run during off-peak hours.

Manually Purge Problem Records

If you also want to purge the PROBLEM table:

1. Log into SQL*Plus as **glogowner**. (glogowner/{password}@{tnsname})
2. sql> exec partit.purge_partit_tables(90); to purge all database records in this table older than 90 days.

Partitioned Tables

Because Oracle Transportation Management contains several integration and logging tables that can become quite large very quickly, these tables have been partitioned to allow for quick purges of older data. By partitioning the tables, a particular partition (segment) can be truncated, instead of records being individually deleted, which is inefficient for large amounts of data.

Range Partitioning

Here is an example of partitioning, or segmenting, a table where a certain year and month drive the partition the data should be placed in. This is an example of partitioning by range:

Range	Partition
Jan-Mar 2001	sales_q1

Range	Partition
Apr-Jun 2001	sales_q2
Jul-Sep 2001	sales_q3
Oct-Dec 2001	sales_q4

```
CREATE TABLE sales
        ( invoice_no NUMBER,
             sale_year  INT NOT NULL,
             sale_month INT NOT NULL)
    PARTITION BY RANGE (sale_year, sale_month)
        ( PARTITION sales_q1 VALUES LESS THAN (2001, 04)
                TABLESPACE data,
            PARTITION sales_q2 VALUES LESS THAN (2001, 07)
                TABLESPACE data,
            PARTITION sales_q3 VALUES LESS THAN (2001, 10)
                TABLESPACE data,
            PARTITION sales_q4 VALUES LESS THAN (2002, 01)
                TABLESPACE data );
```

Oracle Transportation Management Partitioned Tables

The Oracle Transportation Management database has groups of partitioned tables. Most of them are range partitioned. However, the ranges are not hard-coded. Instead, a partition_key column has been added to each partitioned table, which is populated with a trigger. The value of the calculated key determines into which partition the data should be placed, based upon the INSERT_DATE. This adds to flexibility in how the tables are segmented.

For integration related partitioned tables, each table has been defined as a quarterly, a monthly, or a weekly table. The frequency to which the table has been associated drives how the partitions are populated, and also the frequency of the purge. Each of these tables has four partitions (except for I_transaction and I_transmission, which have eight to differentiate between inbound and outbound transmissions).

Quarterly	Monthly	Weekly
	EXPLANATION	I_TRANSACTION (outbound)
	GL_LOGIN_HISTORY	I_TRANSMISSION (outbound)
	IE_INROUTE_SERVPROV	
	IE_MARKS	
	IE_SHIPMENTSTATUS	
	IE_SHIPMENT_REFNUM	
	IE_SHIP_GROUP_REFNUM	
	IE_SS_EQUIPMENT	
	IE_SS_REMARK	

Quarterly	Monthly	Weekly
	IE_SS_STOP	
	IE_SS_S_EQUIPMENT	
	IE_S_SHIP_UNIT	
	IE_S_SHIP_UNIT_LINE	
	I_LOG	
	I_LOG_DETAIL	
	I_TRANSACTION_DETAIL	
	PROBLEM	
	I_TRANSACTION (inbound)	
	I_TRANSMISSION (inbound)	

Here is how it works:

If the table is defined as quarterly, then one quarter will go into each partition. In other words,

Quarter	Partition
Jan-Mar	partition 1
Apr–Jun	partition 2
Jul–Sep	partition 3
Oct–Dec	partition 4

If the table is monthly, then the data is segmented as follows:

Month	Partition
Jan	partition 1
Feb	partition 2
Mar	partition 3
Apr	partition 4
May	partition 1

Month	Partition
June	partition 2
…	

If the table is weekly, then the data is divided as follows:

Week	Partition
Jan 1 –7	partition 1
Jan 8-14	partition 2
Jan 15- 21	partition 3
Jan 22-28	partition 4
Jan 29-Feb 4	partition 1
Feb 5–11	partition 2
…	

Every table reuses its partitions, because the intention is that before the end of the cycle, the oldest partition is purged in preparation for the new cycle. In other words, for a monthly table, on April 30th, partition 1 should be purged to remove January's data, which will then be used for May.

Truncate Jobs for Partitioned Tables

In order to make truncating the partitioned tables as seamless as possible, a script is provided that submits an Oracle Database job to automatically purge the oldest partition, on the last day of the cycle. These jobs should be submitted as a final step in any migration, logged in as **GLOGOWNER**.

1. Enter SET SERVEROUTPUT ON to be able to see feedback.
2. Run Exec partit.submit_job('WW');
3. Run Exec partit.submit_job('MM');
4. Run Exec partit.submit_job('Q');

These jobs are set to run at 1 AM on the last day of the cycle by default.

Schedule Jobs at a Different Time

You can specify a different time if you would like, as an optional second parameter. For example, the following command will have the job run at 2:30am. You can resubmit these jobs whenever you need to change the runtime, since it will auto-delete the previous job, and create it again with the new time.

```
SQL> exec partit.submit_job('WW','02:30');
Job 1164 has been submitted.
It will run next on 13-MAY-2002 02:30
```

Remove Jobs

If you need to remove the job, you can execute the partit.remove_job procedure:

```
SQL> EXEC PARTIT.REMOVE_JOB('WW');
PL/SQL procedure successfully completed.
```

Since the job is submitted as a standard Oracle job, you may want to read Oracle Database documentation about Oracle Database jobs, and the DBMS_JOB package that we utilize in our procedures.

If you need to manually run the purge, you can do so by running purge_partitioned_tables.sql logged in as GLOGOWNER. You will be prompted for the frequency. You may want to run this script if you are not automating the purge, or if it fails and you would like to run it manually. For example, it can look like this:

```
SQL>  @purge_partitioned_tables
```

This routine will truncate the oldest partition for each table that has been identified to be purged at the inputted frequency. The valid values are 'Q' for quarterly, 'MM' for monthly, or 'WW' for weekly, or 'D' for daily. By purging the old data, it can prepare room for the next week, month, or quarter.

```
Please enter a frequency (Q, MM, WW, D): MM

Purging partition 2...
        Purged partition 2 for table GL_LOGIN_HISTORY
        Purged partition 2 for table IE_INROUTE_SERVPROV
        Purged partition 2 for table IE_MARKS
        Purged partition 2 for table IE_SHIPMENTSTATUS
        Purged partition 2 for table IE_SHIPMENT_REFNUM
        Purged partition 2 for table IE_SHIP_GROUP_REFNUM
        Purged partition 2 for table IE_SS_EQUIPMENT
        Purged partition 2 for table IE_SS_REMARK
        Purged partition 2 for table IE_SS_STOP
        Purged partition 2 for table IE_SS_S_EQUIPMENT
        Purged partition 2 for table IE_S_SHIP_UNIT
        Purged partition 2 for table IE_S_SHIP_UNIT_LINE
        Purged partition 2 for table I_LOG_DETAIL
        Purged partition 2 for table I_LOG
        Purged partition 2 for table I_LOG_DETAIL
        Purged partition 2 for table I_TRANSACTION_DETAIL
        Purged partition 2 for table PROBLEM

        Purged inbound i_transaction records
        Purged inbound i_transmission records

PL/SQL procedure successfully completed.
```

Pinning Large Object in Memory

This section is for a production database and an experienced DBA.

Oracle Database requires space in the System Global Area (SGA) for stored packages and functions. If SGA space is fragmented, there may not be enough space to load a package or function. Oracle Database recommends pre-allocating space in the SGA shared pool for frequently used packages, procedures and functions. Pinning objects in the shared pool can improve database performance, if it is done correctly. For large packages, when pinned they do not need to be loaded and parsed from the database again and again. Pinning large package also helps to avoid ORA-04031 error (unable to allocate xxx bytes of shared memory).

It is recommended that, in an Oracle Transportation Management production database, following objects be pinned in the SGA:

```
Package:
    VPD
    PARTIT
    PKG_PURGE
    RRL
    USER_EXIT_HELPER
    RPT_GENERAL
```

Database Space Monitoring

A DBA should periodically monitor Oracle Transportation Management database(s) space usage. There are several tables containing a column with data type of CLOB. These columns can have great amount of data. Some of the CLOB tables can be loaded quite often. One example is I_TRANSMISSION and I_TRANSACTION table. These tables contain transmission/transaction's XMLs. If there are inbound /outbound transmissions in and out of system frequently, the disk space usage of these two tables can grow very quickly.

Scheduled Jobs

Oracle Transportation Management database uses Oracle Database scheduled job to perform certain activities inside the database. Some of the jobs are set up automatically after the database is installed. The others are optional and can be implemented by a DBA. An Oracle Transportation Management database DBA should monitor these jobs regularly.

Length Semantics

Oracle Transportation Management uses character semantics for new string columns and all of the PL/SQL objects to support multi-byte character encoding scheme. It is highly recommended to use character semantics during customization of Oracle Transportation Management database schemas. Change NLS_LENGTH_SEMANTICS parameter at session level before adding new column or compiling PL/SQL object during customization as shown below.

```
alter session set NLS_LENGTH_SEMANTICS=CHAR;
```

For further information about NLS_LENGTH_SEMANTICS, please refer to *Oracle Database Globalization Support Guide*.

Real Application Clusters (RAC)

Oracle Transportation Management may be used in conjunction with RAC by making the following changes to the <otm_install_path>/glog/config/glog.properties file.

Comment out the lines starting with the following by putting a # at the front of the line:

```
dbserver=
glog.database.sid=
glog.database.connectstring=
glog.database.port=
```

Add the following lines right after the above section. Any text in **bold** needs to be supplied with a valid value (each block of code is actually one single line -- there should be no line breaks):

```
glog.database.dbaOnly.t2client.connectionURL=jdbc:oracle:thin:@(DESCRIP
TION=(ADDRESS_LIST=(LOAD_BALANCE=ON)(ADDRESS=(PROTOCOL=TCP)(HOST=DB_FQD
N_HOST_1_NAME)(PORT=DB_PORT_HOST_1))(ADDRESS=(PROTOCOL=TCP)(HOST=DB_FQD
N_HOST_2_NAME)(PORT=DB_PORT_HOST_2)))(CONNECT_DATA=(service_name=RAC_SE
RVICE_NAME)))
```

```
glog.database.dbathin.t2client.connectionURL=$glog.database.dbaOnly.t2c
lient.connectionURL$

glog.database.DBA.t2client.connectionURL=$glog.database.dbaOnly.t2clien
t.connectionURL$

glog.database.DBADirect.t2client.connectionURL=$glog.database.dbaOnly.t
2client.connectionURL$

glog.database.migration.t2client.connectionURL=$glog.database.dbaOnly.t
2client.connectionURL$

glog.database.dbathin.t2client.connectionURL=$glog.database.dbaOnly.t2c
lient.connectionURL$

glog.database.dbaglogowner.t2client.connectionURL=$glog.database.dbaOnl
y.t2client.connectionURL$

glog.database.dbareportowner.t2client.connectionURL=$glog.database.dbaO
nly.t2client.connectionURL$

dbserver=DB_FQDN_HOST_1_NAME,DB_FQDN_HOST_2_NAME

glog.database.sid=RAC_SERVICE_NAME
```

Integration

Suppressing Persistence XML

There is a capability to suppress storing the full transmission XML in the `i_transmission` table, and the full transaction XML in the `i_transaction` table for the RemoteQuery interface. Each of these should reduce the space used by the raw XML in the database and help performance by reducing the time to persist a transmission.

The full transmission XML is suppressed by setting the following property:

```
glog.integration.transmission.suppressPersistFullXml = true
```

> **Note**: The TransmissionHeader will be stored in the `i_transmission` table even when this property is enabled.

The Remote Query Transaction XML is suppressed by setting the following property:

```
glog.integration.transaction.suppressPersist.RemoteQuery = true
```

As of Oracle Transportation Management version 6.0, both of these properties are enabled by default.

Synchronization of Master Data

Repetitive updates of embedded elements like locations and items should be avoided since they produce unnecessary processing. Location and item masters should be synchronized independently of orders instead of embedding within the order. This practice will reduce inserts into I_TRANSACTION and I_LOG tables, thus increasing performance by reducing storage needs.

OutXMLProfile for Outbound Integration and Notification

Out XML Profiles allow you to generate outbound XML and exclude portions of outbound XML with a high degree of control. This is done by specifying what XML builder class files should be excluded when

generating XML documents or by selecting an XML template on which you can base the outbound XML. Outbound XML includes integration and notification. Unnecessarily large outbound integration is a common cause of performance issues. Reducing the size of the XML will greatly reduce the load on the database and application server.

Inbound Integration Options

Large Transmission Servlet

The LargeTransmissionServlet can be used for sending exceptionally large transmission(s) into Oracle Transportation Management. The difference with WMServlet is that parsing of the XML is handled in the servlet, and there is suppression of storing the complete transmission in the database. The individual transactions are stored in the database. It persists only the individual transactions, not the transmission. Refer to the Integration Guide for more information on this subject.

OAQ

Oracle Advanced Queuing (OAQ) provides an alternate way of sending and receiving XML transmissions to/from Oracle Transportation Management. The main benefit to using OAQ is the added level of guaranteed message delivery provided by a persistent message queue. An additional benefit is the elimination of the web and application server from the processing of persisting a transmission. This has the benefit of reducing the load on the web and application servers. Refer to the Data Management Guide for details on the setup for use of Oracle Advanced Queuing.

Web Services

You can send data to Oracle Transportation Management via a web service call to the Oracle Transportation Management application server. The performance advantage of this approach is the elimination of the web tier from the process of persisting the XML. This should increase UI performance by reducing the overall load on the web server. Note that support for receiving inbound integration via a web service call is currently only supported on the WebLogic and Oracle Application Server platforms. Refer to the Integration Guide for more information on this subject.

User Interface

Finder Page Size

A Finder Page is the Oracle Transportation Management terminology for the standard result page which is returned when running an Oracle Transportation Management finder query. The number of records returned per page is configurable via the "Finder Page Size" user preference. Increasing this from the default value of 25 has a direct impact on the performance of loading this page due to the increased amount of data that needs to be processed and displayed. Values greater than 100 are likely to make the performance unacceptable.

User Favorites

Favorites allow you to limit initial `Find ()` results so that your favorite results are displayed first. For example, you can create a favorite for locations. In that favorite you specify the top 10 locations which you use on a regular basis. Then whenever you use the `Find ()` button the system first displays those 10 locations. If you want to find a different location you are given the option to search for more locations.

Adding Search Fields to Finders (Grid-Flattening)

Grid flattening enables administrators to present "pseudo fields" for search, result, and other manager pages that are customized to the needs of their users. Pseudo fields "flatten" a grid by presenting field

labels that are more specific than the default fields. For example, a PO Number field could be added to the order base search page, so users can search for purchase orders directly by their numbers. Grid flattening is used to make it appear as if data from a child database table is on a parent database table. Using grid flattening on search and result pages should be done with caution since the resulting query is more complex and therefore susceptible to slower performance.

Active Table

Oracle Transportation Management implements data security via the Oracle Database feature Virtual Private Database (VPD). VPD provides an ability to dynamically append where conditions to SQL statements at runtime. Oracle Transportation Management uses this capability for its external predicate functionality and for cross-domain visibility. By default, a "`where`" condition is appended to a SQL statement for each table in the SQL. These additional where conditions can have an adverse affect on performance.

Depending upon a customer's particular setup in Oracle Transportation Management, these where conditions may be unnecessary. A capability has been implemented to control which tables have where conditions appended when running an Oracle Transportation Management finder query. The behavior for finders is controlled by the following property:

```
glog.query.vpdActiveTables.default=value
```

The possible values are:

- none = Do not apply VPD active table.
- primary = Apply VPD active table to the primary table of the query.
- comma-delimited tables = Apply VPD active table(s) for the supplied list.

Optionally, this behavior can be overridden for a specific Oracle Transportation Management query class with a property like:

```
glog.query.vpdActiveTables.QueryClassName=value
```

For example:

```
glog.query.vpdActiveTables.glog.server.query.invoice.InvoiceQuery=prima
ry
```

The default behavior is to use primary table as active table for all finder set queries except following queries:

```
glog.server.query.preference.QueryTableQuery
glog.server.query.notify.StylesheetProfileQuery
```

Prior to version 6.0 the active table behavior was controlled by a property like:

```
glog.query.vpdOptimization.glog.server.query.invoice.InvoiceQuery=activ
e
```

Status Queries

Queries based on status can be complex and often suffer from poor performance. Some of this complexity is due to the implementation of domains such that status types are created within each domain. Based on a customer's particular domain setup it may be possible to optimize these status queries.

The following glog.property can be used to control the domains from which status types will be queried:

```
glog.query.statusOptimization=value
```

The valid values are:

- none – No optimization so all domains are used.
- currentDomain – Status types from the current domain only are used.
- childDomains – Status types from the current and child domains are used.
- grantedDomains – Status types from all granted domains are used.

The following property controls how many domains will be included before the logic reverts to the same behavior as "none":

```
glog.query.statusOptimizationDomainMax=value
```

Manager Layout Producer Configuration

The XML document for an object is created by a series of "producers". These producers are associated with objects within Oracle Transportation Management. Each manager in Oracle Transportation Management has an XML document associated with it. When you create a custom manager layout you can create or remove fields from that manager. The system creates an XML document containing relevant data that is displayed in that manager. However the XML document may contain data that is no longer needed in a custom manager that has had a lot of fields removed. By deleting the associated producers from the custom manager the XML document will be smaller, allowing the page to load more quickly. Refer to the "Manager Layout: Producers" topic in online help for more information.

Parameterized Saved Queries

The UI finder mechanism in Oracle Transportation Management is a generic engine which enables flexible querying of any business object in Oracle Transportation Management. A SQL query is generated dynamically based on the user's input in the finder query page. Due to the potential complexity of the queries and the generic nature of the engine, it is not possible to always generate SQL which will perform optimally when executed. Saved queries provided a mechanism for an end-user to create a query using custom SQL. Users can create a saved query using SQL that is formatted in a manner which performs more efficiently. This also provides a mechanism to specify a hint in the SQL which will get passed to the Oracle SQL optimizer.

In addition to a standard saved query, there is an ability to create a saved query with parameters that can be fulfilled at runtime. For more information on this topic refer to "How To Enable the Use of Parameterized Saved Queries" in the online help.

Limiting Child Records in a Manager

The Oracle Transportation Management UI architecture is based on an XML document that is created by querying a record and all of its related child records. In some scenarios the number of child records can cause the XML document to be extremely large and therefore cause high memory usage and slow performance. The number of child records returned is configurable via the glog.property:

```
glog.query.limit.glog.server.query.orderbase.QueryName=limit
```

The QueryName is the name of the corresponding query and the limit indicates the maximum number of child records to return. If the number of child records exceeds this value, the manager will not display any child records of this type.

For example:

```
glog.query.limit.glog.server.query.order.ShipUnitLineQuery=100
```

As of version 6.0 many, managers have these properties set by default.

Functional Security CheckAccess Cache Sizes

Oracle Transportation Management functional security is cached by user. If your installation is going to have a large number of simultaneous users it may be necessary to increase the size of the functional security CheckAccess cache to be equal to the peak number of users expected in order to obtain optimal performance. In the glog.properties file on the Oracle Transportation Management web server you should configure the following property. The default value is 100.

For example:

```
glog.cache.CheckAccess.capacity=nnn
```

The following property controls the number of items caches per user. It can also be adjusted as necessary. For example:

```
glog.cache.CheckAccessItem.capacity=200
```

Scalability

Oracle Transportation Management scalability can be used to solve some performance problems. However, scalability should only be used after the single application server has been tuned properly. Adding more application servers to an installation that is performing poorly may actually make the performance worse. For instance, if the network or database is the performance bottle-neck, adding more application servers will only exaggerate the issue.

One reason to add another scalability node would be that a single application server is running out of memory or performing excessive garbage collection. Since the memory that can be allocated to a single JVM is limited, adding a scalability node will expand the total available memory. In this scenario, the additional scalability node may even be on the same physical hardware (assuming it has sufficient memory and processing power). If the processing power (CPU) is being maxed out on a single application server, adding another scalability node on another physical server will increase the total processing power.

Miscellaneous

Optimizing Report Generation

Distribution of Report Components

The generation of reports in Oracle Transportation Management is delegated to the Oracle BI Publisher components. These components include:

- **Template Retrieval**. BI Publisher XML templates defining data generation, report transformation and report translation are stored in the database. The first step in report generation is their retrieval.
- **Data Generation**. Given a data generation template, BI Publisher applies input parameters to the template and constructs a data XML stream from database queries.
- **Report Transformation**. BI Publisher transforms the data XML stream and the format template into a desired output document. This document may be HTML, XML, PDF, RTF, or an Excel spreadsheet.

- **Distribution**. For ad-hoc reports, the user may request the report be served directly back to their browser. Alternatively, they may want the report to an e-mail recipient via SMTP. Scheduled reports are either sent directly via e-mail or transferred to an Oracle Transportation Management Contact. This leverages the Notification subsystem to notify interested parties of the report via e-mail, fax or message center[1], attaching or linking to the report contents.

These components typically use significant CPU and memory resources. By default, they are run on an application server to:

- perform all data queries behind a firewall;
- avoid degradation of web server response;
- manage SMTP backup through the Oracle Transportation Management event queues;
- avoid disk storage of reports on an open web server; and
- scale report generation via application-tier scalability[2].

There may be cases, though, where one or more of these components should run in the web-tier. For example, very large reports are more efficiently generated on the requesting web server. RMI, the transport protocol between the web servers and application servers, defaults to a maximum 10 MB message size. Trying to generate such reports on an application server may require larger RMI buffers with a subsequent impact on application-tier and web-tier memory to support the data transfer.

The following table lists properties which control report architecture:

Component Property	Description	Requirements	Default
glog.bipreports.appTier.distribute	if true, ad-hoc report e-mails are distributed via the application server; otherwise they are distributed via the web server	if false, requires SMTP server access on the web server	true
glog.bipreports.appTier.query	if true, ad-hoc report queries for templates and data are performed on the application server; otherwise, they are performed on the web server	if false, requires database access from the web server	true
glog.bipreports.appTier.transform	if true, report transformation is performed on the application server; otherwise, it is performed on the web server		true

Note: These properties have no impact on scheduled reports: they always run on the application server.

When reports are generated on the web server, web scalability can be used to scale generation limit the impact of report generation on a core cluster of web servers. See the Application Scalability Guide for more information.

---

[1] an internal Oracle Transportation Management message system

[2] via the new REPORTS application function

Management of Report Files

During report generation, a number of files may be stored in the server's file system. These include data content XML, embedded images for HTML and the final reports. Oracle Transportation Management must have full create/read/write/delete privileges to these files and their directories to properly stage reports.

The following properties control the location of reporting files on each tier:

Property	Holds	Default
glog.bipreports.report.path	final reports	temp
glog.bipreports.app.tempDir	data content files for the application server	temp/bipublisher
glog.bipreports.app.ImageDir	HTML embedded images for the application server	temp/bipublisher
glog.bipreports.web.tempDir	data content files for the web server	bipublisher
glog.bipreports.web.ImageDir	HTML embedded images for the web server	bipublisher3

Intermediate report files are typically removed once the final report is generated. Final report files are managed with a Least Recently Used (LRU) algorithm. The default number of reports maintained by a running server is 200. Use the following property to modify this value:

```
glog.cache.Reports.capacity=<report capacity>
```

If a server generating reports fails or is cycled, existing report files will no longer be managed by the LRU algorithm. The following set of properties provides a backup to clean out old report and image files.

Property	Holds	Default
glog.bipreports.cleanup.<type>.threshold	# of days to hold files of this type	Reports: 30 Images: 30
glog.bipreports.cleanup.<type>.frequency	# of minutes between file scans	Reports: 360 Images: 360
glog.bipreports.cleanup.<type>.files	comma-delimited list of file wildcards to remove	Reports: report-*, *.html, *.rtf, *.pdf, *.xls Images: xdo*

---

[3] This must be underneath the document root to properly server images to the browser.

Valid types include: `content`, `report`, `webImages` and `appImages`.

Logging BI Publisher Activity

To diagnose errors during data content generation or report transformation, use the following properties to enable and control BI Publisher logging:

Property	Description	Default
glog.bipreports.log.filename	base name of BI Publisher log	bipublisher.log
glog.bipreports.log.level	detail level for BI Publisher logging. Valid values are: 6 = UNEXPECTED 5 = ERROR 4 = EXCEPTION 3 = EVENT 2 = PROCEDURE 1 = STATEMENT 0 = OFF	4
glog.bipreports.log.maxSize	maximum number of bytes in any one log file	1000000
glog.bipreports.log.numBackups	number of log backups before cycling	5

10. Troubleshooting Known Issues

Installation

There are known problems with Windows X server Terminal Emulators and InstallAnywhere on Solaris. InstallAnywhere is the installation program used by both WebLogic and Oracle Transportation Management.

Symptoms: Installer started from Windows via remote X session appears to fail immediately - or in debug mode, displays an error containing the sting "An unexpected exception has been detected in native code outside the VM" and will leave a core dump in your current directory.

Resolution: Install from the server console (recommended) or log out of your X session, reset your X server and log back in.

Starting Oracle Transportation Management

After installing Oracle Transportation Management on Microsoft Windows Server, make sure that you reboot your server before you start Oracle Transportation Management. There are environment settings that will not take effect until the server is rebooted.

Symptoms: Apache won't startup correctly. You get an error stating "Cannot load apache/modules/mod_ssl.so into server: (126) The specified module could not be found." This error occurs when openssl.exe is not in the path.

Resolution: Restart your server before starting Oracle Transportation Management.

After installing Oracle Transportation Management on Solaris, make sure that you log out of your session before you start Oracle Transportation Management. There are environment settings that will not take effect until you log out and log back in.

Symptoms: Apache, Tomcat, or WebLogic/WebSphere will not startup correctly.

Resolution: Log out of your server and then log back in before starting Oracle Transportation Management. When Oracle Transportation Management starts, Tomcat opens a connection with WebLogic/WebSphere to cache certain information (UOM fields, dropdown boxes, etc). If WebLogic/WebSphere has not fully started before Tomcat is started, this information is not cached and certain fields will remain blank.

Symptoms: Oracle Transportation Management has been started but some fields and drop-down boxes are not populated as they should be.

Resolution: Restart Oracle Transportation Management.

503 (Service Unavailable) Errors

While the application server is starting up, the web server will respond to all requests with a"503 (Service Unavailable)" error message. This will go away once the application server has fully started up. See **Starting Oracle Transportation Management Servers** for more information.

Uninstalling Oracle Transportation Management

Oracle Transportation Management must be manually uninstalled; there is no automated uninstall option. Follow the directions below to completely remove Oracle Transportation Management from your system.

1. If running on Microsoft Windows, remove the Oracle Transportation Management services. The default names for these services are otmapache, otmtomcat, and otmweblogic. The srvinstw.exe utility from the Microsoft Windows Resource Kit can help you remove these services.
2. Remove the Oracle Transportation Management installation directory and all of its contents (e.g. d:\gc3 on Windows and /opt/gc3 on UNIX).
3. If running on UNIX, remove the Oracle Transportation Management init scripts. These scripts are installed under /etc/init.d and their default names are otmweb and otmapp.
4. Modify your PATH and CLASSPATH environmental variables and change them back to their original state.

Browsers

Symptoms: Your browser displays stale or incorrect data while using Oracle Transportation Management.

Resolution: Configure your browser to "Check for newer versions of stored pages: Every visit to the page". This will ensure that your browser displays the correct data and not cached pages. Depending on your browser, this option is usually under the "Cache" or "Temporary Internet files" section of the "Options" or "Preferences" window.