

Oracle® Transportation Management

Migration Guide

Release 6.2

Part No. E20095-04

December 2012

Copyright © 2001, 2012, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

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

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

CONTENTS	III
SEND US YOUR COMMENTS	V
PREFACE	VII
CHANGE HISTORY	VII
1. DATABASE MIGRATION	1-1
GENERAL NOTE ON MIGRATIONS	1-1
UPGRADING FROM VERSION 3.7 TO 4.0	1-1
UPGRADING FROM VERSION 4.0 TO 4.5	1-3
UPGRADING FROM VERSION 4.5 TO 5.0	1-7
APPLYING GC3 4.5 SERVICE RELEASES	1-7
MIGRATING GC3 4.5 DATABASE TO 5.0	1-8
UPGRADING FROM VERSION 5.0 TO 5.5	1-12
CUSTOM HELP	1-12
APPLYING ORACLE TRANSPORTATION MANAGEMENT 5.0 CONSOLIDATED UPDATES	1-12
MIGRATING GC3 5.0 DATABASE TO 5.5	1-12
UPGRADING FROM VERSION 5.5 TO 6.2	1-15
CUSTOM HELP	1-15
APPLYING ORACLE TRANSPORTATION MANAGEMENT 5.5 CONSOLIDATED UPDATES	1-15
MIGRATING ORACLE TRANSPORTATION MANAGEMENT 5.5 DATABASE TO 6.2	1-15
CONVERTING ORDER RELEASES TO A MULTI-TIER STRUCTURE	1-20
MIGRATING ORACLE TRANSPORTATION MANAGEMENT 6.0 DATABASE TO 6.2	1-21
MIGRATING ORACLE TRANSPORTATION MANAGEMENT 6.1 DATABASE TO 6.2	1-23
2. MIGRATING FUSION TRANSPORTATION INTELLIGENCE	2-1
MIGRATING HISTORICAL DATABASE (HD)	2-1
MIGRATING THE ORACLE FUSION TRANSPORTATION INTELLIGENCE METADATA	2-1
MIGRATING THE ORACLE FUSION TRANSPORTATION INTELLIGENCE WEB CATALOG	2-1
MIGRATING THE ORACLE FUSION TRANSPORTATION INTELLIGENCE ETL	2-2
MIGRATING THE ETL FROM 6.2 TO 6.2.1	2-2
MIGRATING ORACLE TRANSPORTATION MANAGEMENT PROPERTIES	2-2
NEW VPD PROFILE FOR ORACLE FUSION TRANSPORTATION INTELLIGENCE USERS	2-4

Send Us Your Comments

Oracle Transportation Management Migration Guide, Release 6.2

Part No. E20095-04

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 manual is for administrators who are responsible for migrating the Oracle Transportation Management databases at your site. This manual provides step-by-step installation instructions for migrating your Oracle Transportation Management databases from one version of Oracle Transportation Management to another.

Change History

Date	Document Revision	Summary of Changes
11/2010	-01	Initial release
2/2011	-02	Added the Migrating the ETL from 6.2 to 6.2.1 section to chapter 2.
12/2011	-03	Added the Migrating Oracle Transportation Management Properties section to chapter 2.
10/2012	-04	On page 1-3, updated the script in the Resetting DB Sequences section to be "exec domainman.reset_sequence" instead of "exec domainman.reset_sequences.sql".

1. Database Migration

General Note on Migrations

Prior to migrating your data it is very important that you have the latest bug fixes in place. Please see the section titled "General Instructions on Upgrading Oracle transportation Management from One Release to Another" in the Administration Guide.

Upgrading from Version 3.7 to 4.0

Note: Prior to its acquisition by Oracle, the Oracle Transportation Management product (circa release 5.0) was known as the Global Command & Control Center, or GC3. The migration instructions prior to release 5.0 of Oracle Transportation Management will refer to the product by its old name of GC3.

Note: If you are migrating from a version prior to 3.7, refer to the GC3 4.5 Administration Manual.

Any new GC3 installation should be tested in a controlled environment before migrating your production database. This ensures that any potential problems or incompatibilities don't affect your production GC3 instance(s).

Note: If you are migrating from a version earlier than 3.7, you must complete each previous version's migration instructions. You cannot skip any migration steps. Please go through all upgrade steps to 3.7 prior to starting the 3.7 to 4.0 migration.

IMPORTANT! GC3 4.0 relies on Oracle 9i, so this upgrade *must* occur prior to the upgrade of GC3. If the Oracle database upgrade needs to be run at a separate time due to time constraints, then we recommend upgrading to Oracle database 9i first, and then upgrading to GC3 version 4.0 during back-to-back weekends.

Follow the installation instructions and install the new 4.0 GC3 components into new directories. Once this is complete, you should compare your old glog.properties file with the new glog.properties file and migrate any customized settings. Finally, copy any customer-specific glog.properties files to the new GC3 instance and ensure that the new glog.properties file references these files. Customer-specific properties files usually follow the format glog.<company_name>.properties.

The following topics outline the procedures for migrating a GC3 3.7 database to 4.0. The structure of the database to be migrated should be consistent with the GC3 3.7 database specification; otherwise, the migration will fail.

To complete these procedures you need the following DB passwords:

- system password
- glogowner password
- reportowner password

In addition, you will be creating tablespaces for LOB columns. For this step, you will need to know the directory in which the datafiles will exist and a default initial size for the LOB tablespaces.

Before you begin the migration, shut down all processes running against the database and shut down the GC3 application. Always create a full backup of the database before beginning any GC3 migration.

Applying GC3 3.7 Service Releases

You must apply the latest GC3 3.7 Service Release before you continue with the 4.0 migration. You must also install the latest GC3 Service Release to your GC3 4.0 installation to ensure that any known migration issues have been patched and resolved.

1. Log in as **GLOGOWNER** and run **dbpatch_37.sql** (which resides in the 3.7 directory, (<otm37_install_path>/glog/oracle/script8/).

Note: Do not continue with the upgrade until the dbpatch log is completely clean. Contact Technical Support if you have any questions or concerns.

Important: Ensure that your environment is setup correctly by running:

On UNIX: ". <gc3_install_path>/install/gc3env.sh"

On Windows: "<gc3_install_path>\install\gc3env.cmd"

Updating GLOGOWNER Grants

1. Go to the script8 directory for 4.0 (<otm40_install_path>/glog/oracle/script8).
2. Log on as **SYSTEM**.
3. Run 40_mig_grants.sql, which directly grants GLOGOWNER the ability to create and drop public synonyms (versus through a role).

Adding Tablespaces for LOB columns

1. As the SYSTEM user, run create_lob_tablespace.sql. You are prompted for the directory in which to store the datafiles and the initial size for the tablespaces.

Updating the Structure

1. Run @dbupdate_40.sql to update the database with all the new tables and columns.
2. Enter the glogowner password, reportowner password, and database connection when prompted.
3. After the process has run, verify in the dbupdate_40_<dbsid>_<timestamp>.log file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and then rerun the dbupdate_40 process again, without harm. Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the dbupdate_40 log is clean.**

Updating Data Content

1. Update the PUBLIC data by running update_content. This process is run at the host command line, rather than from within SQL*Plus. Two command scripts have been provided; the script you use is dependent on your operating system:
UNIX shell script:
./update_content.sh <otm_install_path>/glog/config
dbaglogowner dbareportowner V40
OR
DOS command script:
update_content <otm_install_path>\glog\config dbaglogowner dbareportowner V40
2. Review the log file called update_content_v40_<timestamp>.log for errors (located in the same directory as the SQL script). Search for errors beginning with "ORA-" or "<Error>" within the log file.

This procedure migrates the data content into the table structures for the latest enhancements.

1. In SQL*Plus, as user GLOGOWNER run:
`@dbmigrate_40.sql`.
2. Enter the password and database connection string when prompted.
Note: The script might run for several hours depending on the amount of data to be processed.
3. After the process has run, verify in the `dbmigrate_40_<dbsid>_<timestamp>.log` file (located in the same directory as the SQL script) that there are no errors.
4. If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support.

Resetting DB Sequences

1. In SQL*Plus, as user **SYS** run:
`exec domainman.reset_sequences`
2. Go to the `<gc3_install_path>\glog\oracle\script8` directory on the GC3 application server. In SQL*Plus, as user **SYS** run:
`@analyze_tables.sql`
Running this script will enable you to take advantage of the latest indexes. The script might run for several hours.
3. Go to the `<otm_install_path>\glog\oracle` directory on the GC3 application server. In SQL*Plus, as user **GLOGOWNER** run:
`@insert_security_roles.sql`

The remaining topics are not critical to the upgrade but provide helpful information.

Reviewing Obsolete Tables and Columns

When a table or column becomes obsolete during an upgrade, the table/column is renamed with `XX<version object became obsolete>_<original name of table/column>`.

To generate SQL for dropping obsolete objects, run `@gen_obsolete_objects.sql` as **GLOGOWNER**.

Note: It will not include objects that just became obsolete in the current version, as a safety precaution. This script will generate an SQL script called `drop_obsolete_objects.sql`, which you can review and run at a convenient time.

Verifying Saved Queries

After the upgrade, some of your site's saved queries may no longer be valid due to changes in table structure. Run the following to verify the saved queries:

```
@validate_saved_query.sql
```

If there are invalid saved queries, you can decide what to do with them (remove the records or modify them to correct the syntax). Contact Technical Support if you need assistance.

Upgrading from Version 4.0 to 4.5

Note: Prior to its acquisition by Oracle, the Oracle Transportation Management product (circa release 5.0) was known as the Global Command & Control Center, or GC3. The migration instructions prior to release 5.0 of Oracle Transportation Management will refer to the product by its old name of GC3.

Any new GC3 installation should be tested in a controlled environment before migrating your production database. This ensures that any potential problems or incompatibilities don't affect your production GC3 instance(s).

Note: If you are migrating from a version earlier than 4.0, you must complete each previous version's migration instructions. You cannot skip any migration steps. Please go through all upgrade steps to 4.0 prior to starting the 4.0 to 4.5 migration.

Follow the installation instructions and install the new GC3 4.5 components into new directories. Once this is complete, you should compare your old glog.properties file with the new glog.properties file and migrate any customized settings. Finally, copy any customer-specific glog.properties files to the new GC3 instance and ensure that the new glog.properties file references these files. Customer specific properties files usually follow the format glog.<company_name>.properties.

The following topics outline the procedures for migrating a GC3 4.0 database to 4.5. The structure of the database to be migrated should be consistent with the GC3 4.0 database specification; otherwise, the migration will fail.

Applying GC3 4.0 Service Releases

You must apply the latest GC3 4.0 Service Release before you continue with the 4.5 migration. You must also install the latest GC3 Consolidated Update to your GC3 4.5 installation to ensure that any known migration issues have been patched and resolved.

1. Log in as **GLOGOWNER** and run dbpatch_40.sql (which resides in the 4.0 directory, (<gc3_40_install_path>/glog/oracle/script8/).

Note: Do not continue with the upgrade until the dbpatch log is completely clean. Contact Technical Support if you have any questions or concerns.

Important: Ensure that your environment is setup correctly by running:

On UNIX: ". <gc3_install_path>/install/gc3env.sh"

On Windows: "<gc3_install_path>\install\gc3env.cmd"

Implementing 4.5 components within 4.0 (optional)

To reduce migration time, we have provided scripts that add new 4.5 tables and columns (and in some cases populate) to a 4.0 environment. We recommend testing the overall timeframe of the migration without the preupdate/premigrate steps. If you determine downtime is too long, then you can use these scripts to reduce downtime.

Since these steps can be performed while 4.0 is running, the overall downtime of the 4.5 migration is reduced. However, this script should be run during off-peak hours, when heavy data loads are not running. You can also choose to run this script while the system is down in the 4.0 environment. If you plan to run these scripts, it is recommended to run these steps in the weekend prior to the 4.5 upgrade.

1. As **GLOGOWNER**, run:
@preupdate_45.sql

If the pre-update script cannot obtain access to a table for a new column, you may see a *resource busy* message within the preupdate_45 log file. You can either rerun pre-update at another time once the table is available (which will only apply the failed procedure(s)), or wait until migration downtime. The step for running dbupdate_45.sql will automatically attempt to add the column at that time.

The `ss_status_history` and `order_release` tables have new columns that need to be populated as part of the migration. By running the following, three database triggers will be created to maintain the data within the 4.0 environment. Immediately following the creation of the triggers, the fields will be populated. These triggers will remain in place until your database has been fully upgraded to 4.5.

2. As GLOGOWNER, run:

```
@premigrate_45.sql
```

If your site chooses not to run these scripts in the 4.0 environment, the `dbupdate_45` and `dbmigrate` scripts will automatically add these changes that were not implemented during the `preupdate/premigrate` phase. The rest of the steps are all mandatory, and will be performed as part of the downtime 4.5 migration.

Updating GLOGOWNER Grants

1. Go to the `script8` directory for 4.5 (`<gc3_45_install_path>/glog/oracle/script8`).
2. Log on as SYSTEM.
3. Run `45_mig_grants.sql`.

Updating the Structure

1. Run `@dbupdate_45.sql` to update the database with all the new tables and columns.
2. Enter the `glogowner` password, `reportowner` password, and database connection when prompted.
3. After the process has run, verify in the `dbupdate_45_<dbsid>_<timestamp>.log` file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and then rerun the `dbupdate_45` process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the `dbupdate_45` log is clean.**

Updating Data Content

1. Update the PUBLIC data by running `update_content`. This process is run at the host command line, rather than from within SQL*Plus. Two command scripts have been provided; the script you use is dependent on your operating system:
UNIX shell script:

```
./update_content.sh <gc3_install_path>/glog/config V45
```


or
DOS command script:

```
update_content <gc3_install_path>\glog\config V45
```
2. Review the log file called `update_content_v45_<timestamp>.log` for errors (located in the same directory as the SQL script). Search for errors beginning with "ORA-" or "<Error>" within the log file.

This procedure migrates the data content into the table structures for the latest enhancements.

1. In SQL*Plus, as user **GLOGOWNER** run: `@dbmigrate_45.sql`.
2. Enter the password and database connection string when prompted.
Note: The script might run for several hours depending on the amount of data to be processed.
3. After the process has run, verify in the `dbmigrate_45_<dbsid>_<timestamp>.log` file (located in the same directory as the SQL script) that there are no errors.

4. If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support.
5. Go to the <gc3_install_path>\glog\oracle\script8 directory on the GC3 application server. In SQL*Plus, as user **SYS** run:

```
@gather_table_stats.sql
```

 Running this script will enable you to take advantage of the latest indexes. The script might run for several hours.
6. Go to the <gc3_install_path>\glog\oracle directory on the GC3 application server. In SQL*Plus, as user GLOGOWNER run:

```
@insert_security_roles.sql
```

The remaining topics are not critical to the upgrade but provide helpful information.

Migrating Audit Trail Information (optional)

Audit trail information has been redesigned in 4.5. GC3 provides an optional migration script that can be run if your site would like to keep its historical audit trail information. This can be run while 4.5 is up-and-running.

You will be prompted for the date range for the auditing information you would like to keep. The smaller the date range, the quicker the process will finish. You can run this process multiple times, if you would like to process sets of small ranges during off-peak hours, rather than processing all records at once.

1. As GLOGOWNER, run @45mig_audit_trail.sql

Reviewing Obsolete Tables and Columns

When a table or column becomes obsolete during an upgrade, the table/column is renamed with XX<version object became obsolete>_<original name of table/column>.

To generate SQL for dropping obsolete objects, run @gen_obsolete_objects.sql as GLOGOWNER.

Note: This will not include objects that just became obsolete in the current version, as a safety precaution. This script will generate a SQL script called drop_obsolete_objects.sql, which you can review and run at a convenient time.

Verifying Saved Queries

After the upgrade, some of your saved queries may no longer be valid due to changes in table structure.

1. Run the following to verify the saved queries:

```
@validate_saved_query.sql
```

If there are invalid saved queries, you can decide what to do with them (remove the records or modify them to correct the syntax).

Upgrading the Replicated Operational Database (ROD)

After the 4.0 to 4.5 upgrade, the Replicated Operational Database (ROD) will not automatically be aware of new or changed GC3 tables. If structural changes occurred, the automatic refresh process will fail until the following steps have been completed.

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

1. On the OLTP (Online Transactional Processing) database in SQL*Plus, as user GLOGOWNER run:

```
@create_mview_logs
```

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

```
@create_logon_triggers.sql
```

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

```
@dbupdate_rod.sql
```

This will first identify broken views caused by columns becoming obsolete. It will then build (or rebuild) the materialized views for any new/modified tables.

Note: This will not automatically add columns that were added during an upgrade. This is because the materialized view would need to be recreated from scratch again, and you may not even need those new columns. Therefore, views should only need to be completely refreshed when a column has been removed, or modified in a way that Oracle Fusion Transportation Intelligence needs a new copy of the data. Once the views are set, it refreshes the views and builds any new indexes.

Upgrading from Version 4.5 to 5.0

Any new Oracle Transportation Management installation should be tested in a controlled environment before migrating your production database. This ensures that any potential problems or incompatibilities don't affect your production Oracle Transportation Management instance(s).

Note: Prior to its acquisition by Oracle, the Oracle Transportation Management product (before release 5.0) was known as the Global Command & Control Center, or GC3. The migration instructions prior to release 5.0 of OTM will refer to the product by its old name of GC3.

Note: If you are migrating from a version earlier than 4.5, you must complete each previous version's migration instructions. You cannot skip any migration steps. Please go through all upgrade steps to 4.5 prior to starting the 4.5 to 5.0 migration.

The following topics outline the procedures for migrating a GC3 4.5 database to Oracle Transportation Management 5.0. The structure of the database to be migrated should be consistent with the GC3 4.5 database specification; otherwise, the migration will fail.

Applying GC3 4.5 Service Releases

Important: Ensure that your environment is setup correctly by running:

```
On UNIX: ". <gc3_install_path>/install/gc3env.sh"
```

```
On Windows: "<gc3_install_path>\install\gc3env.cmd"
```

You must apply GC3 4.5 Service Release dated February 2005 (SR-02.05) or later before you continue with the Oracle Transportation Management 5.0 migration. You must also install the latest Oracle Transportation Management Service Release to your Oracle Transportation Management 5.0 installation to ensure that any known migration issues have been patched and resolved.

1. Log in as **GLOGOWNER** and run dbpatch_45.sql (which resides in the 4.5 directory, (<gc3_45_install_path>/glog/oracle/script8/).

Note: Do not continue with the upgrade until the dbpatch log is completely clean. Contact Technical Support if you have any questions or concerns.

Migrating GC3 4.5 Database to 5.0

Oracle Transportation Management 5.0 had an interim release called 5.0 LA. To migrate a GC3 4.5 database to 5.0 GA, you must first migrate 5.0 LA. The following steps will guide you through the process.

Migrating to 5.0 LA: Updating the Structure

1. On the app server change to directory `<otm_50_install_path>/glog/oracle/script8LA`.
2. In SQL*Plus as user **GLOGOWNER**, run `@dbupdate_50a.sql` to update the database with all the new tables and columns to 5.0LA.
3. Enter the glogowner password, reportowner password, archive password, and database connection when prompted.
4. After the process has run, verify in the `dbupdate_50a_<dbsid>_<timestamp>.log` file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and then rerun the `dbupdate_50a.sql` process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the dbupdate_50a log is clean.**

Migrating to 5.0 LA: Updating Data Content

1. Update the PUBLIC data by running `update_content`. This process is run at the host command line rather than from within SQL*Plus. Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./update_content.sh <otm_install_path>/glog/config V50A
```

or

DOS command script:

```
update_content <otm_install_path>\glog\config V50A
```

2. Review the log file called `update_content_v50A_<timestamp>.log` for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", or "<Error>" within the log file.
3. Load the Oracle Transportation Sourcing (formerly Procurement) PUBLIC data by running `update_procure_content`. This process is run at the host command line, rather than from within SQL*Plus. Two command scripts have been provided; the script you use is dependent on your operating system:

UNIX shell script:

```
./update_procure_content.sh <otm_install_path>/glog/config V50A
```

or

DOS command script:

```
update_procure_content <otm_install_path>\glog\config V50A
```

4. Review the log file called `update_procure_content_v50A_<timestamp>.log` for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", within the log file.

Migrating to 5.0 LA: Data Migration

1. In SQL*Plus as GLOGOWNER, run:
`@dbmigrate_50a.sql`
2. Enter the password and database connection string when prompted.
Note: The script might run for several hours depending on the amount of data to process.
3. After the process has run, verify in the `dbmigrate_50a_<dbsid>_<timestamp>.log` file (located in the same directory as the SQL script) that there are no errors.

If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support.

Migrating to 5.0 LA: Apply 5.0 LA Patch

1. In SQL*Plus as user GLOGOWNER, run:
`@dbpatch_50a.sql`
2. Enter the passwords and database connection strings when prompted.
3. Check the log files (dbpatch log and update_content log) to ensure they are error free before proceeding to the next step.

Migrating to 5.0 GA: Updating the Structure

1. On the Application server change to directory
`<otm50_install_path>/glog/oracle/script8.`
2. In SQL*Plus as **GLOGOWNER**, run
`@dbupdate_50b.sql` to update the database with all the new 5.0 tables and columns.
3. Enter the glogowner password, reportowner password, and database connection when prompted.

After the process has run, verify in the `dbupdate_50b_<dbsid>_<timestamp>.log` file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and rerun the `dbupdate_50b.sql` process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the dbupdate_50b log is clean.**

Migrating to 5.0 GA: Updating Data Content

The Oracle Transportation Management 5.0 database includes Oracle Transportation Sourcing (formerly Procurement) structures and data contents even though you have not purchased the Oracle Transportation Sourcing (formerly Procurement) component license. The database structure has been loaded in the previous step, while the content is loaded below after normal Oracle Transportation Management content loading.

1. Update the Oracle Transportation Management PUBLIC data by running update_content. This process is run at the host command line, rather than from within SQL*Plus. Two command scripts have been provided; the script you use is dependent on your operating system:

UNIX shell script:

```
./update_content.sh <otm_install_path>/glog/config V50B
```

or

DOS command script:

```
update_content <otm_install_path>\glog\config V50B
```

2. Review the log file called update_content_v50B_<timestamp>.log for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", or "<Error>" within the log file.
3. Load the Oracle Transportation Sourcing (formerly Procurement) PUBLIC data by running update_procure_content. This process is run at the host command line, rather than from within SQL*Plus. Two command scripts have been provided; the script you use is dependent on your operating system:

UNIX shell script:

```
./update_procure_content.sh <otm_install_path>/glog/config V50B
```

or

DOS command script:

```
update_procure_content <otm_install_path>\glog\config V50B
```

4. Review the log file called update_procure_content_v50B_<timestamp>.log for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", within the log file.

Migrating to 5.0 GA: Data Migration

1. In SQL*Plus as GLOGOWNER, run:
@dbmigrate_50b.sql
2. Enter the password and database connection string when prompted.
Note: The script might run for several hours depending on the amount of data to process.
3. After the process has run, verify in the dbmigrate_50b_<dbsid>_<timestamp>.log file (located in the same directory as the SQL script) that there are no errors.

If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support.

Running Security Role Script

1. Go to the <otm_install_path>\glog\oracle directory on the Oracle Transportation Management application server. In SQL*Plus as **GLOGOWNER**, run:
@insert_security_roles.sql

Analyzing the Database

1. Oracle Transportation Management databases should be fully analyzed after the 5.0 migration. Oracle Transportation Management provides following analyze script. But of course a DBA can use their own analyze process.

```
@gather_table_stats.sql
```

The remaining steps are not critical to the upgrade but provide helpful information.

Reviewing Obsolete Tables and Columns

When a table or column becomes obsolete during an upgrade, the table/column is renamed with XX<version object became obsolete>_<original name of table/column>.

1. To generate SQL for dropping obsolete objects, run: @gen_obsolete_objects.sql as **GLOGOWNER**.

Note: This will not include objects that just became obsolete in the current version, as a safety precaution. This script will generate a SQL script called drop_obsolete_objects.sql, which you can review and run at a convenient time.

Verifying Saved Queries

After the upgrade, some of your saved queries may no longer be valid due to changes in table structure.

1. Run the following to verify the saved queries:

```
@validate_saved_query.sql
```

If there are invalid saved queries, you can decide what to do with them (remove the records or modify them to correct the syntax).

Upgrading the ROD

After the GC3 4.5 to Oracle Transportation Management 5.0 migration, the Oracle Fusion Transportation Intelligence Replicated Operational Database (ROD) will not automatically be aware of new or changed Oracle Transportation Management tables. If structural changes occurred, the automatic refresh process will fail until the following steps have been completed.

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 **GLOGOWNER**, run:

```
@create_mview_logs
```

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

```
@dbupdate_rod.sql
```

This will first identify broken views caused by obsolete columns. It will then build (or rebuild) the materialized views for any new/modified tables.

Note: This will not automatically add columns that were added during an upgrade. This is because the materialized view would need to be recreated from scratch again, and you may not even need those new columns. Therefore, views should only need to be completely refreshed when a column has been removed, or modified in a way that Oracle Fusion Transportation Intelligence needs a new copy of the data. Once the views are set, it refreshes the views and builds any new indexes.

Upgrading from Version 5.0 to 5.5

Any new Oracle Transportation Management installation should be tested in a controlled environment before migrating your production database. This ensures that any potential problems or incompatibilities don't affect your production Oracle Transportation Management instance(s).

Note: If you are migrating from a version earlier than 5.0 you must complete each previous version's migration instructions. You cannot skip any migration steps. Please go through all upgrade steps to 5.0 prior to starting the 5.0 to 5.5 migration.

The structure of the database to be migrated should be consistent with the Oracle Transportation Management 5.0 database specification; otherwise, the migration will fail.

Custom Help

If you have written custom documentation for Oracle Transportation Management, you can link it to the Oracle Transportation Management help. A file exists for you to customize. You must know some elementary HTML in order to edit the file. By editing the HTML of the file `general/custom_help.htm`, you can add links to any documents that you have written to supplement Oracle Transportation Management's documentation. Alternatively, you can replace that topic with one of the same name that you have created.

Note: If you choose to edit that topic and add your own documentation or links to your own documentation, be careful not to overwrite the file when you upgrade your software.

If you install a consolidated update, that help file will be overwritten when the new help is installed. To avoid overwriting your edited help topic, make a backup of the file before upgrading. Then, replace the newly installed file with the one from your backup.

Applying Oracle Transportation Management 5.0 Consolidated Updates

Important: Ensure that your environment is setup correctly by running:

- On UNIX: `./ <otm_install_path>/install/gc3env.sh`
- On Windows: `"<otm_install_path>\install\gc3env.cmd"`

You must also install the latest Oracle Transportation Management 5.0 Service Release to your Oracle Transportation Management 5.0 installation and apply the Service Release script to your 5.0 database to ensure that any known migration issues have been patched and resolved.

1. Log in as **GLOGOWNER** and run `dbpatch_50b.sql` (which resides in the 5.0 directory, `(<otm_50_install_path>/glog/oracle/script8/)`).

Note: Do not continue with the upgrade until the `dbpatch` log is completely clean. Contact Technical Support if you have any questions or concerns.

Migrating GC3 5.0 Database to 5.5

Updating the Structure

If 5.5 CUs are available, you should install the latest CU before running the following database migration steps. This is important since bugs, if any, in the database migration scripts would get fixed in the Service Releases. Therefore, you will not run into any known issues.

1. On the app server, change to directory `<otm_55_install_path>/glog/oracle/script8`.
2. In SQL*Plus log in as **GLOGOWNER**

3. Run @dbupdate_55.sql to update the database with all the new tables and columns to 5.5.
4. Enter the glogowner password, reportowner password, archive password, and database connection when prompted.
5. After the process has run, verify in the dbupdate_55_<dbid>_<timestamp>.log file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and then rerun the dbupdate_55.sql process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the dbupdate_55 log is clean.**

Updating Data Content

1. Update the PUBLIC data by running update_content. This process is run at the host command line rather than from within SQL*Plus. Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./update_content.sh <otm_install_path>/glog/config V55
```

or

Windows command line script:

```
update_content <otm_install_path>\glog\config V55
```

2. Review the log file called update_content_v55_<timestamp>.log for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", or "<Error>" within the log file. **Do not continue until the update_content log is clean.**

Migrating Data

1. In SQL*Plus as **GLOGOWNER**, run:
@dbmigrate_55.sql.
2. Enter the password and database connection string when prompted.
3. After the process has run, verify in the dbmigrate_55_<dbid>_<timestamp>.log file (located in the same directory as the SQL script) that there are no errors.

If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support.

Applying 5.5 Consolidated Update

If the 5.5 Consolidated Update is available, apply the latest CU to the database.

1. In SQL*Plus as user **GLOGOWNER**, run:
@dbpatch_55.sql
2. Enter the passwords and database connection strings when prompted.
3. Check the log files (dbpatch log and update_content log) to ensure they are error free before proceeding to the next step.

Running Security Role Script

1. Go to the <otm_install_path>\glog\oracle directory on the Oracle Transportation Management application server. In SQL*Plus as **GLOGOWNER**, run:

```
@insert_security_roles.sql
```

2. Shut down your database and then restart it.

Analyzing the Database

1. Oracle Transportation Management database should be fully analyzed after the 5.5 migration.

Oracle Transportation Management provides following analyze script.

```
@gather_table_stats.sql
```

The remaining steps are not critical to the upgrade but provide helpful information.

Reviewing Obsolete Tables and Columns

When a table or column becomes obsolete during an upgrade, the table/column is renamed with XX<version object became obsolete>_<original name of table/column>.

1. To generate SQL for dropping obsolete objects, run: @gen_obsolete_objects.sql as **GLOGOWNER**.

Note: This will not include objects that just became obsolete in the current version, as a safety precaution. This script will generate a SQL script called drop_obsolete_objects.sql, which you can review and run at a convenient time. **Do not run drop_obsolete_objects.sql script right after the database migration to prevent accidental dropping newly obsolete objects.**

Verifying Saved Queries

After the upgrade, some of your saved queries may no longer be valid due to changes in table structure.

1. Run the following to verify the saved queries:

```
@validate_saved_query.sql
```

If there are invalid saved queries, you can decide what to do with them (remove the records or modify them to correct the syntax).

Upgrading the Oracle Fusion Transportation Intelligence ROD

After the 5.0 to 5.5 migration, the Oracle Fusion Transportation Intelligence ROD will not automatically be aware of new or changed Oracle Transportation Management tables. If structural changes occurred, the automatic refresh process will fail until the following steps have been completed.

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 **GLOGOWNER**, run:

```
@create_mview_logs
```

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

```
@dbupdate_rod.sql
```

This will first identify broken views caused by obsolete columns. It will then build (or rebuild) the materialized views for any new/modified tables.

Note: This will not automatically add columns that were added during an upgrade. This is because the materialized view would need to be recreated from scratch again, and you may not even need those new columns. Therefore, views should only need to be completely refreshed when a column has been removed, or modified in a way that Oracle Fusion Transportation Intelligence needs a new copy of the data. Once the views are set, it refreshes the views and builds any new indexes.

Upgrading from Version 5.5 to 6.2

Any new Oracle Transportation Management installation should be tested in a controlled environment before migrating your production database. This ensures that any potential problems or incompatibilities don't affect your production Oracle Transportation Management instance(s).

Note: If you are migrating from a version earlier than 5.5 you must complete each previous version's migration instructions. You cannot skip any migration steps. Please go through all upgrade steps to 5.5 prior to starting the 5.5 to 6.2 migration.

The structure of the database to be migrated should be consistent with the Oracle Transportation Management 5.5 database specification; otherwise, the migration will fail.

Custom Help

If you have written custom documentation for Oracle Transportation Management, you can link it to the Oracle Transportation Management help. A file exists for you to customize. You must know some elementary HTML in order to edit the file. By editing the HTML of the file `general/custom_help.htm`, you can add links to any documents that you have written to supplement Oracle Transportation Management's documentation. Alternatively, you can replace that topic with one of the same name that you have created.

Note: If you choose to edit that topic and add your own documentation or links to your own documentation, be careful not to overwrite the file when you upgrade your software.

If you install a consolidated update, that help file will be overwritten when the new help is installed. To avoid overwriting your edited help topic, make a backup of the file before upgrading. Then, replace the newly installed file with the one from your backup.

Applying Oracle Transportation Management 5.5 Consolidated Updates

You must install the latest Oracle Transportation Management 5.5.06 to your Oracle Transportation Management 5.5 installation. You must also apply all of the Roll Up (RU) patches available for the 5.5.06 to ensure that any known migration issues have been patched and resolved.

1. Log in as **GLOGOWNER** and run `dbpatch_55.sql` (which resides in the 5.5 directory, `<otm_55_install_path>/glog/oracle/script8/`).

Note: Do not continue with the upgrade until the `dbpatch_55 log` is completely clean. Contact Technical Support if you have any questions or concerns.

Migrating Oracle Transportation Management 5.5 Database to 6.2

You must install all of the Roll Up (RU) patches available for 6.2. This is important as bugs, if any, in the database migration scripts would get fixed in the associated Roll Up patches. Therefore, you will not run into any known issues.

Running the Oracle Fusion Transportation Intelligence Data Cleanup Utility

Migration of the Oracle Fusion Transportation Intelligence solution from the version 5.5 to 6.2 will involve automatic clean-up of unsupported data in the TARGET_TYPE, TARGET_VALUE and E_KPI_TARGET_VALUE_TYPE tables. To see the exact data that is cleaned up by the migration script, the aa_user_entered_cleanup.sql SQL script is provided. You should run this script before updating structure of the OLTP database to the Oracle Transportation Management 6.0 version. Perform following steps to run this utility script.

1. On the app server, change to directory <otm_install_path>/glog/oracle/script8.
2. In SQL*Plus log in as **GLOGOWNER** and run @ aa_user_entered_cleanup.sql. Data will be spooled into e_kpi*.csv files.

Updating the Structure

1. On the app server, change to directory <otm_install_path>/glog/oracle/script8.
2. In SQL*PLUS as SYS user run the following script to create new tablespaces required for Oracle Transportation Management 6.2:

```
@create_mobilcomm_tablespaces.sql
```

Check the log file (create_mobilcomm_tablespaces.log) to ensure they are error free before proceeding to the next step.

3. Update the database with all the new tables and columns to 6.2 by running dbupdate_60 process. This process is run at the host command line; Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./dbupdate_60.sh
```

or

Windows command line script:

```
dbupdate_60.cmd
```

4. Enter the database connection, glogowner password, reportowner password, archive password, archive database connection, archive database glogowner password, SYS user password, DBA user name that can alter database users, and its password when prompted.
5. After the process has run, verify in the dbupdate_60_<dbsid>_<timestamp>.log file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and then rerun the dbupdate_60.sql process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the dbupdate_60 log is clean.**

Updating Data Content

1. Update the PUBLIC data by running update_content. This process is run at the host command line rather than from within SQL*Plus. Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./update_content.sh <otm_install_path>/glog/config V60
```

or

Windows command line script:

```
update_content.cmd <otm_install_path>\glog\config V60
```

2. Review the log file called update_content_<timestamp>.log for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", or "<Error>" within the log file.

Note: Do not continue until the update_content log is clean.

Migrating Data

1. Migrate data to 6.2 by running dbmigrate_60 process. This process is run at the host command line; two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./dbmigrate_60.sh
```

or

Windows command line script:

```
dbmigrate_60.cmd
```

2. Enter database connection ID, glogowner password, and property file location when prompted.
3. After the process has run, verify in the dbmigrate_60_<dbsid>_<timestamp>.log file (located in the same directory as the SQL script) that there are no errors.

Patching the Database

1. Additionally, apply remaining database changes by running dbpatch_60 process. This process is run at the host command line; Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./dbpatch_60.sh
```

or

Windows command line script:

```
dbpatch_60.cmd
```

2. Enter the database connection, glogowner password, reportowner password, archive password, archive database connection, archive database glogowner password, property file location, SYS user password, DBA user name that can alter database users, and its password when prompted.

3. Check the log files (dbpatch log and update_content log) to ensure they are error free before proceeding to the next step.

If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support.

4. Shut down your database and then restart it.

Analyzing the Database

1. Oracle Transportation Management database should be fully analyzed after the 6.2 migration.
2. In SQL*Plus as user **GLOGOWNER**, run the following script to gather statistics.

```
@gather_table_stats.sql
```

The remaining steps are not critical to the upgrade but provide helpful information.

Reviewing Obsolete Tables and Columns

When a table or column becomes obsolete during an upgrade, the table/column is renamed with XX<version object became obsolete>_<original name of table/column>.

1. To generate SQL for dropping obsolete objects, run: @gen_obsolete_objects.sql as **GLOGOWNER**.

Note: This will not include objects that just became obsolete in the current version, as a safety precaution. This script will generate a SQL script called drop_obsolete_objects.sql, which you can review and run at a convenient time. **Do not run drop_obsolete_objects.sql script right after the database migration to prevent accidental dropping newly obsolete objects.**

Verifying Saved Queries

After the upgrade, some of your saved queries may no longer be valid due to changes in table structure.

1. In SQL*Plus as user **GLOGOWNER**, run the following to verify the saved queries:

```
@validate_saved_query.sql
```

If there are invalid saved queries, you can decide what to do with them (remove the records or modify them to correct the syntax).

Upgrading the Oracle Fusion Transportation Intelligence ROD

After the 5.5 to 6.2 migration, the Oracle Fusion Transportation Intelligence ROD will not automatically be aware of new or changed Oracle Transportation Management tables. If structural changes occurred, the automatic refresh process will fail until the following steps have been completed.

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 **GLOGOWNER**, run:

```
@create_mview_logs
```

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

```
@create_mview_logs
```

3. On the ROD database in SQL*Plus as **GLOGOWNER**, run:
`@dbupdate_rod.sql`

This will first identify broken views caused by obsolete columns. It will then build (or rebuild) the materialized views for any new/modified tables.

Note: This will not automatically add columns that were added during an upgrade. This is because the materialized view would need to be recreated from scratch again, and you may not even need those new columns. Therefore, views should only need to be completely refreshed when a column has been removed, or modified in a way that Oracle Fusion Transportation Intelligence needs a new copy of the data. Once the views are set, it refreshes the views and builds any new indexes.

Upgrading the Oracle Fusion Transportation Intelligence (FTI) Database

1. If Oracle Fusion Transportation Intelligence Historical Database (HDOWNER schema) is installed on ROD database, make sure that the ROD database is upgraded. Change to directory `<otm_install_path>/glog/oracle/script8/advanced_analytics`.
2. SQL*Plus log in as **HDOWNER**
3. Run `@aa_dbupdate_60.sql` to update the database with all the new tables and columns to 6.2.
4. Enter the hdowner password, SYS password, database connection for Oracle Fusion Transportation Intelligence database, glogowner password for OLTP database and database connection for OLTP database, Y or N depending on the machine type, and glog property file location when prompted.
5. After the process has run, verify in the `aa_dbupdate_60_<dbsid>_<timestamp>.log` file (located in the same directory as source) that there are no errors. Also verify that `csvone_aa_W_LOCALIZED_STRING_G.log` file does not have any errors like "ORA-", "Caught exception", "SP2-", or "<Error>". If the solution to the error is apparent, then you can fix the problem and then rerun the `aa_dbupdate_60.sql` process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the aa_dbupdate_60 log is clean.**

Updating Oracle Fusion Transportation Intelligence Data Content

The Oracle Fusion Transportation Intelligence data content in the Historical Database includes only the metadata translations.

1. On the OTM server go to the directory `<otm_install_path>/glog/oracle/script8`.
2. Update the data content by running `update_content_aa`. This process is run at the host command line rather than from within SQL*Plus. Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./update_content_aa.sh <otm_install_path>/glog/config
```

or

Windows command line script:

```
update_content_aa.cmd <otm_install_path>\glog\config
```

3. Review the log file called `update_content_aa_<timestamp>.log` for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", or "<Error>" within the log file.

Converting Order Releases to a Multi-tier Structure

Order releases that were planned prior to 6.0 must be converted into multi-tier database structures before they can be further processed by Oracle Transportation Management in 6.0. The primary purpose of conversion is to create order movements and separate shipment ship units from shipment to shipment in order to support the development of more complex planning and execution processes in Oracle Transportation Management.

A new field was added to the ORDER_RELEASE table and is used to indicate if an order is converted or not. The OTM_VERSION is set to NULL for unconverted orders. When an order is converted, the OTM_VERSION is set to '6.0'. If you try to invoke any action on an unconverted order in 6.0, directly on the order, such as unassigned order, or indirectly on any shipment related to the order, such as split/merge shipments or change stop times, you will get a warning and the action will be aborted.

The conversion of orders in 6.0 can be done only through a Java process. In 6.1, a PL/SQL version was added to convert orders planned on single leg shipments with significant performance improvement. Orders on multi-leg shipments will still need to be converted with the Java process.

In 6.1 you can run a validation script as @validate.sql to get a report of ill-structured orders, order movements, and shipments in the database. This script will also report the number of orders that will be converted by the newly added PL/SQL conversion process that you will run next. For the poorly-structured data, Oracle Transportation Management does not provide any tool to fix them, but you can fix them manually. The validation script should be run right after the database migration process. This process is optional.

The next step is to convert orders on single leg shipments using the PL/SQL script by running: @convert_orders.sql. This is a required script. It can be run right after the database migration process or after running the validation script. After running the database script, verify in the convert_<dbname>_<timestamp>.log (located in the same directory as source) that there are no errors.

Orders planned on multi-leg shipments need to be converted using a Java process after running the PL/SQL script convert_orders.sql. This process can be run interleaving with other Oracle Transportation Management processes. However, we recommend it be run independently before any other Oracle Transportation Management processes. If there are a lot of multi-leg shipments left to convert after running the PL/SQL conversion script, it is not necessary to convert all of them at once. But orders which need to be further processed (order being unassigned, related shipments being split, merged, etc.) by Oracle Transportation Management have to be converted. For better management and performance, you should consider converting orders for a short period of time, such as orders created within a quarter or half year at one time. Old orders that are no longer used will not need to be converted or can be converted at a later time.

Conversion process can be invoked at any time, but due to the performance impact, we recommend it to be run after normal business hours.

Converting Order Releases

1. Log in as DBA.ADMIN.
2. Turn on conversion-related logs. There are three log IDs related to conversion: SSUConversion, SSUConversionDetails, and SSUConversionSummary. When you are testing out the conversion process with a small number of orders, you can turn on all three. However, when you convert a large number of orders, we recommend only turn on SSUConversionSummary due to performance concerns.
3. Navigate to Order **Management** > **Process Management** > **Convert Order Releases**.
4. Enter a starting date. All orders with insert_date later than the starting date will be converted.

5. Enter a batch size for converting. The default batch size is 150. It provides the best performance based on testing experience.
6. Monitor your log file to see if there are any failed batches during conversion. Orders in failed batches will be re-attempted one by one once all batches are done and most likely they will be converted successfully at this time. You only need to contact Oracle technical support for further evaluation if any order failed to convert during the second attempt.

Migrating Oracle Transportation Management 6.0 Database to 6.2

You must install all of the Roll Up (RU) patches available for 6.2. This is important as bugs, if any, in the database migration scripts would get fixed in the associated Roll Up patches. Therefore, you will not run into any known issues.

Upgrading 6.0 database to 6.2

1. On the app server, change to directory `<otm_install_path>/glog/oracle/script8`.
2. In SQL*PLUS as SYS user run following script to create new tablespaces required for Oracle Transportation Management version 6.2:

```
@create_mobilcomm_tablespaces.sql
```

3. Apply database changes by running `dbpatch_60` process. This process is run at the host command line. Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./dbpatch_60.sh  
or
```

Windows command line script:
`dbpatch_60.cmd`

4. Enter the database connection, glogowner password, reportowner password, archive password, archive database connection, archive database glogowner password, property file location, SYS user password, DBA user name that can alter database users, and its password when prompted.
5. Check the log files (`dbpatch` log and `update_content` log) to ensure they are error free before proceeding to the next step.
6. If an error occurs during a migration patch, the database changes roll back. It will, however, commit changes once a whole migration patch has been successfully applied, and it will not try to migrate the data associated with that patch again. If an error occurs partway through the process, then you can fix the problem and rerun the process without harm. If you cannot resolve the error yourself, contact Technical Support and send the entire log file (along with any other logs you have from the day) to our Technical Support team.
7. Shut down your database and then restart it.

Analyzing the Database

The Oracle Transportation Management database should be fully analyzed after the 6.2 migration. Oracle Transportation Management provides following analyze script:

```
@gather_table_stats.sql
```

The remaining steps are not critical to the upgrade but provide helpful information.

Reviewing Obsolete Tables and Columns

When a table or column becomes obsolete during an upgrade, the table/column is renamed with `XX<version object became obsolete>_<original name of table/column>`.

To generate SQL for dropping obsolete objects, run: `@gen_obsolete_objects.sql` as GLOGOWNER.

Note: This will not include objects that just became obsolete in the current version, as a safety precaution. This script will generate a SQL script called `drop_obsolete_objects.sql`, which you can review and run at a convenient time. **Do not run `drop_obsolete_objects.sql` script right after the database migration to prevent accidental dropping newly obsolete objects.**

Verifying Saved Queries

After the upgrade, some of your saved queries may no longer be valid due to changes in table structure.

Run the following to verify the saved queries:

```
@validate_saved_query.sql
```

If there are invalid saved queries, you can decide what to do with them (remove the records or modify them to correct the syntax).

Upgrading the Oracle Fusion Transportation Intelligence ROD

After the 6.0 to 6.2 migration, the Oracle Fusion Transportation Intelligence ROD will not automatically be aware of new or changed Oracle Transportation Management tables. If structural changes occurred, the automatic refresh process will fail until the following steps have been completed.

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 GLOGOWNER, run:

```
@create_mview_logs
```

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

```
@create_mview_logs
```

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

```
@dbupdate_rod.sql
```

This will first identify broken views caused by obsolete columns. It will then build (or rebuild) the materialized views for any new/modified tables.

Note: This will not automatically add columns that were added during an upgrade. This is because the materialized view would need to be recreated from scratch again, and you may not even need those new columns. Therefore, views should only need to be completely refreshed when a column has been removed, or modified in a way that Oracle Fusion Transportation Intelligence needs a new copy of the data. Once the views are set, it refreshes the views and builds any new indexes.

Upgrading the Oracle Fusion Transportation Intelligence Database

If Oracle Fusion Transportation Intelligence Historical Database (HOWNER schema) is installed on ROD database, make sure that the ROD database is upgraded. Change to directory `<otm_install_path>/glog/oracle/script8/advanced_analytics`.

1. SQL*Plus log in as HOWNER
2. Run `@aa_dbupdate_60.sql` to update the database with all the new tables and columns to 6.2.
3. Enter the howner password, system password, database connection for Oracle Fusion Transportation Intelligence database, and glogowner password for OLTP database and database connection for OLTP database when prompted.

After the process has run, verify in the `aa_dbupdate_60_<dbid>_<timestamp>.log` file (located in the same directory as source) that there are no errors. If the solution to the error is apparent, then you can fix the problem and then rerun the `aa_dbupdate_60.sql` process again, without harm (you may also want to rerun to see if it automatically resolves your problem). Contact Technical Support if the problem is not resolvable, and send the entire log file (along with any other logs you have from the day) to our Technical Support team. **Do not continue until the `aa_dbupdate_60` log is clean.**

Updating Fusion Transportation Intelligence Data Content

The Oracle Fusion Transportation Intelligence data content in the Historical Database includes only the metadata translations.

1. On the OTM server go to the directory `<otm_install_path>/glog/oracle/script8`.
2. Update the data content by running `update_content_aa`. This process is run at the host command line rather than from within SQL*Plus. Two command scripts are provided and the script you use is dependent on your operating system:
UNIX shell script:

```
./update_content_aa.sh <otm_install_path>/glog/config
```

or

Windows command line script:

```
update_content_aa.cmd <otm_install_path>\glog\config
```

3. Review the log file called `update_content_aa_<timestamp>.log` for errors (located in the same directory as the SQL script). Look for errors by searching key words like "ORA-", "Caught exception", "SP2-", or "<Error>" within the log file.

Migrating Oracle Transportation Management 6.1 Database to 6.2

Follow the instructions in the **Migrating Oracle Transportation Management 6.0 Database to 6.2** section to migrate Oracle Transportation Management 6.1 to 6.2.

2. Migrating Fusion Transportation Intelligence

This chapter is applicable only for existing Oracle Fusion Transportation Intelligence (FTI) customers and it details the steps to migrate to the Oracle Fusion Transportation Intelligence version 6.2

Migrating Historical Database (HD)

Please refer the **Upgrading the Oracle Fusion Transportation Intelligence Database** section for your corresponding Fusion Transportation Intelligence version available under the **Database Migration** chapter (above) for steps to migrate your existing Fusion Transportation Intelligence Historical Database and its data content.

Migrating the Oracle Fusion Transportation Intelligence Metadata

In the Oracle Business Intelligence Enterprise Edition server, to migrate from an existing Oracle Fusion Transportation Intelligence metadata to the Oracle Fusion Transportation Intelligence version 6.2 metadata, please follow the instructions listed below.

1. Stop the Oracle Business Intelligence Server and Oracle Business Intelligence Presentation Server services.
2. Back up your existing `advanced_analytics.rpd` file (Recommended action: rename it to "`advanced_analytics_bk_<Time Stamp>`"). The file is located here:
`<OBIEE Root Dir>/server/Repository/`
3. Copy the Oracle Fusion Transportation Intelligence metadata file (`advanced_analytics.rpd`) from the following directory:
`<otm_install_path>/fti`
4. In the Oracle Business Intelligence Enterprise Edition server, place the new `advanced_analytics.rpd` file in the following directory:
`<OBIEE Root Dir>/server/Repository/`

Note: If you have customized the `advanced_analytics.rpd` file, you will have to manually port your custom changes from the backup file created in Step 2 to the newly deployed `advanced_analytics.rpd` file.

5. Start the Oracle Business Intelligence Server and Oracle Business Intelligence Presentation Server services.

Migrating the Oracle Fusion Transportation Intelligence Web Catalog

In the Oracle Business Intelligence Enterprise Edition server, to migrate from an existing Oracle Fusion Transportation Intelligence web catalog file (or webcat file which contains the canned reports) to the Oracle Fusion Transportation Intelligence version 6.2 webcat please follow the instructions listed below.

1. Stop the Oracle Business Intelligence Server and Oracle Business Intelligence Presentation Server services.
2. Back up your existing "**aa**" webcat folder by renaming it to "`aa_bk_<Time Stamp>`"). The aa webcat folder is located here:
`<OBIEE Data Dir>/web/catalog/`
3. Copy the Oracle Fusion Transportation Intelligence web catalog file (`aa_webcat.zip`) from the directory below.

<otm_install_path>/fti/aa_webcat.zip

4. In the Oracle Business Intelligence Enterprise Edition server, extract the contents of the aa_webcat.zip file to the directory below.

<OBIEE Data Dir>/web/catalog

Note: If you have custom reports or related webcat contents stored under the "aa" folder, you will have to manually copy the relevant files from the backup folder created in Step 2 to the newly created "aa" folder. This is the only solution currently available to retain your existing custom report contents.

5. Start the Oracle Business Intelligence Server and Oracle Business Intelligence Presentation Server services.

Migrating the Oracle Fusion Transportation Intelligence ETL

As Oracle Fusion Transportation Intelligence incorporates an upgraded ETL technology from the PLSQL (aa_load_hd package on HDOWNER schema) to Oracle Data Integrator, there is no migration.

If you have custom ETL code plugged-in to the old aa_load_hd package, you can either re-write your custom code in Oracle Data Integrator or simply plug-in your custom PLSQL code from within Oracle Data Integrator. Refer the Oracle Data Integration developer documentation for detailed steps to achieve this.

Migrating the ETL from 6.2 to 6.2.1

1. Connect to work repository by opening **Designer**.
2. Export the Work repository and keep it safe by selecting **File > Export > Work Repository**.
3. Expand project FTI_FTL and then select FTI_FTL folder, right-click and select **Import > Import Interfaces**.
4. Choose *INSERT_UPDATE* mode and select *POP_INCR_F_SHIPMENT_EVENT.xml*, *POP_INIT_F_SHIPMENT_EVENT.xml* from the <otm_install_path>/fti/etl/odi/ directory and import the interfaces.
5. Regenerate all scenarios by right-clicking on the FTI_ETL project
6. Click on **Generate all Scenarios** and choose *Re-generate* option
7. Click **OK**.

Migrating Oracle Transportation Management Properties

With the release of Oracle Transportation Management 6.2, a new set of Fusion Transportation Intelligence-specific properties have been added to Oracle Transportation Management. To include these properties, follow these instructions:

1. Open the file <OTM Install Dir>/glog/config/glog.properties and locate the FTI properties section containing these lines:

```
## Fusion Transportation Intelligence (formerly Advanced Analytics) - optional
```

2. Include the following properties:

Property Name	Description
glog.fti.dbserver	Use this to specify the Host name of the Database hosting the FTI Historical database

Property Name	Description
<code>glog.fti.database.port</code>	Use this to specify the Port number of the FTI Historical database
<code>glog.database.fti.sid</code>	Use this to specify the SID/Service name (as applicable) of the Database hosting the FTI Historical database
<code>glog.fti.database.schema</code>	Use this to specify the name of the Schema hosting the FTI Historical database. This should typically be HDOWNER unless a different schema name is used. Example : HDOWNER
<code>glog.database.fti.password</code>	Use this to specify the Password of the HDOWNER user in the FTI Historical database Example : HDOWNER
<code>glog.fti.connectstring</code>	Use this to specify the database connection string for the FTI Historical database
<code>glog.odi.master.database.server</code>	Use this to specify the Host name of the Database hosting the ODI Master Repository.
<code>glog.odi.master.database.port</code>	Use this to specify the port number of the Database hosting the ODI Master Repository.
<code>glog.odi.master.database.sid</code>	Use this to specify the SID/Service name (as applicable) of the Database hosting the FTI Historical database
<code>glog.odi.master.database.password</code>	Use this to specify the Password of the FTIMASTER user in the Database hosting the ODI Master Repository Example : FTIMASTER
<code>glog.odi.agent.server</code>	Use this to specify the Host name of the server that is running the ODI Agent.
<code>glog.odi.agent.port</code>	Use this to specify the port number on the server on which the ODI agent is running.
<code>glog.odi.password</code>	Use this to specify the Password of the ODI login user . Example : CHANGEME
<code>glog.odi.work.repository.code</code>	Use this to specify the name of the FTI Work Repository deployed in the ODI server. Note: <code>glog.odi.work.repository.code</code> is the "Work Repository Name" that we gave while creating the work repository. Example: FTI_WORK.

Note: All the property values provided are intended as examples only. You must set values that are valid to your implementation. See the Oracle Transportation Management Online Help for additional information about these properties.

The following properties are deprecated from version 6.2:

- `glog.database.FTI.t2client.driverClassName`
- `glog.database.FTI.t2client.databaseURL`
- `glog.database.FTI.gluser`
- `glog.database.FTI.user`
- `glog.database.FTI.password`
- `glog.database.FTI.server`
- `glog.database.FTI.t2client.pool`

New VPD profile for Oracle Fusion Transportation Intelligence Users

Follow the instructions in the **Mandatory Oracle Transportation Management User Role (VPD Profile) Configuration** chapter in the Oracle® Transportation Management Installation Guide to configure the appropriate Oracle Fusion Transportation Intelligence specific VPD profile to your existing Oracle Fusion Transportation Intelligence users defined in Oracle Transportation Management.