Oracle® Transportation Management

Migration Guide

Release 6.4.3

Part No. E92129-02

February 2018

ORACLE®

Copyright Notice

Contents

Tables

Send Us Your Comments

Oracle® Transportation Management Migration Guide, Release 6.4.3

Part No. E92129-02

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

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

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

- Electronic mail: otm-doc_us@oracle.com

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

Preface

This 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
12/2017	-01	Modified migration section "Migrating Oracle Transportation Management 6.3 Database to 6.4.3" for 6.4.3
		Added text in "Pre-migration tasks" and " Pre-migration tasks"
		Modified text for Oracle Wallet migration for 6.4.3 in section "Versions Affected"
		Removed following section as per bug# 26929366
		Upgrading from Version 3.7 to 4.0 Upgrading from Version 4.0 to 4.5 Upgrading from Version 4.5 to 5.0 Upgrading from Version 5.0 to 5.5 Upgrading from Version 5.5 to 6.3 Converting Order Releases to a Multi-tier Structure for Customers Who Are Migrating from Version 5.5.CU6 or Earlier
		Added sentence to apply latest roll up of the current version before starting migration in "Migrating Oracle Transportation Management 6.0 Database to 6.3"
		Added following sections:
		• Upgrading to Oracle Business Intelligence Enterprise Edition 12c Migrating Fusion Transportation Intelligence Metadata in OBI EE 12c
		Removed references to APP_CUSTOM and WEB_CUSTOM property sets. All properties should be written to the CUSTOM property set.
		Added a section to Verify Password Rule migration.
		Where appropriate, replaced the words "custom", "customization" "customizing", "customized" with one of the following "configuration", "configure", "edit", "update", "user-defined", or "customer-defined" per Fusion requirements.
		Added FTI properties requirement for migration in the "Upgrading the Oracle Fusion Transportation Intelligence Database"
02/2018	-02	Updated to correct migration script DBPATCH.SH error.

Date	Document Revision	Summary of Changes

1. Migrating Credentials to Oracle Wallet

Overview

As of 6.4.2, Oracle Transportation Management stores credential information in an Oracle wallet. This ensures secure data, such as passwords, are placed into an encrypted store, and not visible to an end user.

Prior to 6.4.2, only OTM user passwords were stored securely[1]. User authentication did not require decryption but relied on message digest comparison instead. Any integration requiring authentication with text credentials stored credentials as plain text or Base-64 encoded. This included credentials for:

- Creating database connections
- Sending integrations to downstream systems
- Invoking external distance or rating engines
- Running reports on a BI Publisher server
- Storing documents in a Content Management System

With 6.4.3, the database columns and properties holding these credentials are modified to point to a wallet key. Accessing or modifying a credential value uses the stored wallet key to read or update the value in the wallet.

Versions Affected

All customers migrating must move their credentials to Oracle Wallet. These credentials may have already been moved in release 6.4.2, so migration to Oracle Wallet may not be necessary if migrating from 6.4.2 or higher version.

Migration Steps

After migrating the database to 6.4.3,

1. Run the Wallet Migration script against the database, making sure you are logged into the application server machine as the installation user. For Unix installations, run:

   ```
   cd $GLOG_HOME/glog/oracle/script8
   ./migrate_passwords.sh –db
   ```

   For Windows installations, open a command prompt and run:

   ```
   cd /d %GLOG_HOME%\glog\oracle\script8
   migrate_passwords.cmd –db
   ```

   This moves passwords from the following tables into the wallet:
   - CONTENT_MANAGEMENT_SYSTEM
   - DATA_SOURCE
   - EXTERNAL_SYSTEM
   - IPP_PRINTER

---

[1] Via an encrypted message digest in the GL_USER table.

- REPORT_SYSTEM
- WEB_SERVICE_ENDPOINT

2. Review the custom section of each **glog.properties** file. These are properties defined between the **# Custom Properties – Beginning** and the **# Custom Properties – End** lines.

   a. Any properties listed below should be removed from the property file. These reflect properties staged as part of the 6.4.3 installation. The installer prompted for these values and wrote them to the wallet. There is no need to override them in a properties file or properties set.
      - otm.db.password
      - glog.database.archive.password
      - glog.database.fti.password
      - glog.database.load.password
      - glog.signedServlet.password
      - glog.web.systemPassword
      - glog.appSystem.password
      - glog.appGuest.password
      - weblogic.system.password

   b. Any properties listed below should be moved from the property file to the CUSTOM property set. For each property, append an instruction to the property set and set the property's key and value. The Property Set screen automatically writes property values to the wallet.
      - alk.api_key
      - glog.bip.externalFarm.password
      - glog.ExternalDistanceEngine.PCMilerWS.Account
      - glog.ExternalDistanceEngine.PCMilerWS.Password
      - glog.ExternalDistanceEngine.PCMilerWS.Username
      - glog.ExternalDistanceEngine.PCMilerWS.AuthorizationKey
      - glog.ExternalDistanceEngine.PCMilerRailWS.AuthorizationKey\
      - glog.odi.password
      - glog.RatingEngine.RatewareXL.License
      - glog.RatingEngine.RatewareXL.Password
      - glog.RatingEngine.RatewareXL.Username
      - glog.security.oam.password
      - here.app_code
      - here.app_id
      - intelliroute.password

   c. Any properties matching the following query should be moved to the CUSTOM property set as in b).

```
select value from DISTANCE_EXTERNAL_ENGINE_PARMS
    where parm='PASSWORD' and (
      value like 'glog.ExternalDistanceEngine.%.Password' or
      value like 'glog.ExternalDistanceEngine.%.Account' or
      value like 'glog.ExternalDistanceEngine.%.Username' or
      value like 'glog.ExternalDistanceEngine.%.AuthorizationKey'
    )
union
select value from SERVICE_EXTERNAL_ENGINE_PARM
    where parm='PASSWORD' and (
      value like 'glog.RatingEngine.%.Password' or
      value like 'glog.RatingEngine.%.Account' or
      value like 'glog.RatingEngine.%.Username'
    )
```

Note that 6.4.3 does not support plain text or Base64-encoded passwords stored in property files. If your custom properties section has a password, it will be ignored: the wallet always takes precedence, even if a password has not been moved to it.

For more information on managing the Oracle Wallet, please review the Security Guide.

2. Migrating the Database

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.

Migrating Oracle Transportation Management 6.0 Database to 6.3

You must install and apply latest Roll up of your current release before starting migration, for example if you are migrating from 6.2.x then install and apply 6.2.10 before installing 6.3.x.

You must install latest Roll Up release (6.3.x) of 6.3 as per Installation guide's section "INSTALLING OTM" of Roll Up release. Proceed to below listed steps after successful installation of latest Roll Up release (6.3.x).

> **Note**: The following steps require database version 11g, do not upgrade database to version 12c until successful completion of these steps.

Pre-migration tasks

1. On the application server, change to directory `<otm_install_path>/glog/oracle/script8`.
2. In SQL*PLUS as the GLOGOWNER user, run the following script to check for missing data. The script will update data if default value exists or notifies the user if it does not exist. For the reported columns, populate column data with the appropriate value prior to proceeding with rest of the migration.

    ```
    @pre_migrate_60.sql
    ```

3. Oracle Transportation Management database objects must be in **VALID** status prior to starting the migration. Also, all Primary Key, Unique Key, Foreign key, and check constraints must be in **ENABLED** and **VALIDATED** status prior to starting the migration.

    Run the following query on the database to identify INVALID objects. If there are INVALID objects, take necessary action to make them VALID.

    ```
    select * from dba_objects where owner in ('GLOGOWNER','REPORTONWER',
    'ARCHIVE') and status <> 'VALID' and object_name NOT IN
    ('CREATE_TABLE','PRIMARY_KEY','FOREIGN_KEY','CHECKC','CREATE_INDEX','CREATE_
    SEQUENCE','ALTER_TABLE');
    ```

    Run the following query on the database to identify DISABLED or NOT VALIDATED constraints. If there are DISABLED or NOT VALIDATED constraints, take necessary action to make them ENABLED and VALIDATED.

    ```
    select * from dba_constraints where ( status <> 'ENABLED' or validated <>
    'VALIDATED' ) and owner in ('GLOGOWNER','REPORTOWNER','ARCHIVE') and
    constraint_type in ('C','P','U','R');
    ```

    ```
    validate_constraints.sql script is provided to enable constraints if it is
    possible to enable constraints without changing any data. The script is
    located in the <otm_install_path>/glog/oracle/script8 directory. Run this
    script as the respective schema owner to enable constraint in the schema.
    ```

    Note: This script does not fix data, Data modification may be require to enable constraints.

4. Migration requires all privileges and roles granted by Oracle Transportation Management. If any privilege or role has been revoked from Oracle Transportation Management database users, restore it as provided by Oracle Transportation Management prior to starting the migration.

5. Migration fails if there are materialized view logs on tables owned by GLOGOWNER or REPORTOWNER. Run following query to check.

    ```
    select log_owner,master from dba_mview_logs where log_owner in
    ('GLOGOWNER','REPORTOWNER')
    ```

    If there are materialized view logs then Log into database as appropriate user and issue SQL similar to below (drops materialized view log of yard table). If there are, materialized views based on these logs then refresh them completely or drop them before dropping materialized view logs.
    ```
    drop materialized view log on yard;
    ```

6. Grant EXEMPT ACCESS POLICY privilege to database user GLOGOWNER. As SYSDBA, execute grant exempt access policy to glogowner.

7. Shut down all processes running against the database including Oracle Transportation Management application and web servers.

8. A database backup or database level flashback restore point should be created prior to starting the migration. It may be required to restore the database to pre-migration state in case of a problem during migration, depending on changes made on the database and errors.

Upgrading 6.0 database to 6.3

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 6.3:

    ```
    @create_mobilcomm_tablespaces.sql
    ```
    Note: This script creates two new tablespaces MSG_PART_TBS1 and MSG_LOB_TBS1. If this procedure has been already ran as part of previous migration then skip this step.

3. Apply database changes by running dbpatch_60 process. Successful completion of this script will bring the database up to version 6.3.0. 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_<version>_<dbname>_<timestamp>. log , update_content_<version>_<dbname>_<timestamp>. log , or any other generated log file) to ensure they are error free before proceeding to the next step. Look for errors by searching key words like "ORA-", "Caught exception","SP2-", "PLS-","IMP-","Warning:"," Caught exception", "SQL* Loader-" ," SQLException", "SQLRecoverableException" or "<Error>" within the log file.

6. If an error occurs while applying particular patch in the migration, the database rolls back changes made by that patch. It will, however, commit changes once a whole migration patch has been successfully applied, and mark patch as "applied". Migration will not reapply that patch again in case of migration rerun. 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.

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).

Verifying Password Rules

Prior to Oracle Transportation Management version 6.4.3, password rules used, by default, Apache Jakarta regular expressions. As of 6.4.3, these rules exclusively use Java regular expressions. The primary difference between these engines is the definition of POSIX character classes. Data migration automatically maps Jakarta POSIX classes to Java POSIX classes.

Given the criticality of password rules, however, we recommend that you verify any password rules you've entered for users.

Changing Application Server Default Users Passwords

After the upgrade, the system and guest passwords should be changed using the instructions provided in the "Oracle Application Server Default Users" section of the Security Guide. This will allow utilization of more robust encryption logic provided in 6.3. If this step is not performed, then warning messages could be seen in application console logs.

If you have additional application users that do not actually manually log into the application, then the passwords for these users should be changed for the utilization of the more robust encryption logic provided in 6.3. These application users could include users that are used in automated scripts, recurring process, or some kind of scheduled diagnostic collection.

Upgrading the Oracle Fusion Transportation Intelligence Database

If Oracle Fusion Transportation Intelligence Historical Database (HDOWNER schema) is installed on Replicated Operational Database, make sure that the ROD database table structure is also upgraded to match with OLTP prior to proceeding. Change to directory `<otm_install_path>/glog/oracle/script8/advanced_analytics`.

1. SQL*Plus log in as HDOWNER
2. Run `@aa_dbupdate_60.sql` to update the database with all the new tables and columns to 6.3.
3. Enter the hdowner 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_<dbsid>_<timestamp>.log` file (located in the same directory as source) that there are no errors. Look for errors by searching key words like "ORA-", "Caught exception","SP2-", "PLS-","IMP-","Warning:"," Caught exception", "SQL* Loader-" ," SQLException", "SQLRecoverableException" or "<Error>" within the log file. 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_<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.

Creating Oracle Data Integrator Database Users

Now, you create the following Oracle Data Integrator users required for FTI. When you run the `create_odi_users.sql script`, you will be prompted for the password for each user listed below.

*   ftiodimaster
*   ftiodiwork
*   ftiodistage

To create these users, complete the following:

1. Log into the database (as per your deployment architecture discussed above) where you want to create the ODI related repositories for Oracle FTI.

---

2. Using SQL*Plus, as user SYS, run the following script from the directory:

`<otm_install_path>\glog\oracle\script8\advanced_analytics\create_odi_user
s.sql`

3. You see an explanation of what the script will do followed by some questions:

   a. Enter **SYS password** to login as SYSDBA.

   b. Enter the **Connection ID** of the database to which you are logging in as SYSDBA.

   This will typically be the same database on which you installed the HDOWNER schema.

   c. Enter **password** for user **ftiodimaster**.

   Make note of this password because you will need it later in the FTI upgrade process.

   d. Enter **password** for user **ftiodiwork**.

   Make note of this password because you will need it later in the FTI upgrade process.

   e. Enter **password** for user **ftiodistage**.

   Make note of this password because you will need it later in the FTI upgrade process.

Migrating Oracle Transportation Management 6.1, 6.2 Database to 6.3

Follow the instructions in the **Migrating Oracle Transportation Management 6.0 Database to 6.3** section to migrate Oracle Transportation Management 6.1, 6.2 to 6.3.

Migrating Oracle Transportation Management 6.0, 6.1, 6.2 Database to 6.4.3

1. Migrate Oracle Transportation Management database to 6.3 by following instructions in the section **Migrating Oracle Transportation Management 6.0 Database to 6.3**.

2. Migrate Oracle Transportation Management database from 6.3 to 6.4.3 by following instructions in the section **Migrating Oracle Transportation Management 6.3 Database to 6.4.3**.

Migrating Oracle Transportation Management 6.3 Database to 6.4.3

Any Oracle Transportation Management migration 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 6.3 you must complete each previous version's migration instructions. You cannot skip any migration steps. Please go through all upgrade steps to 6.3 prior to starting the 6.3 to 6.4.3 migration.

The structure of the database to be migrated should be consistent with the Oracle Transportation Management version 6.3 database specification; otherwise, the migration will fail.

You must install Oracle Transportation Management 6.4.3 as per Installation Guide's section "INSTALLING OTM" of 6.4.3 release. Proceed to below listed steps after successful installation of 6.4.3.

Note: Oracle Transportation Management 6.4.3 requires database version 12c. The database must be version 12c with latest recommended database patches applied. Also the database client must be version 12c (Administrator install) on all application servers. An 11g database may be upgraded to 12c after migrating to 6.3.0, if migrating from earlier version than 6.3.0.

New Users

Migration to release 6.4.3 creates following users.

Database: GLOGOAQ
Application: otmSystem (replaces system user)
            FA.ADMIN

Edition enabled Database Schemas

As part of the 6.4.3 Release, all Oracle Transportation Management Database Schemas are going to be enabled with the database feature Edition Based Redefinition. This is necessary to provide future support for zero downtime patching. The OTM release 6.4.3 uses a new database edition "OTM643" as a default edition instead of the database default edition ORA$BASE. All of the OTM schemas have been editioned enabled except for GLOGOAQ schema, which contains Oracle Advanced Queue related objects.

All of the OTM tables have been converted into Editioned Views. Original tables have been renamed to the table name with the "_T" suffix to facilitate this conversion. After a successful migration to 6.4.3, OTM will use the Editioning view seamlessly. Customer-defined queries, saved queries, and reports will not be affected by this change; they will seamlessly use Editioned Views instead of tables.

> **Note:** Any view, trigger, package, procedure, function, type, VPD policy or synonym will be dropped during 6.4.3 migration if the Oracle Transportation Management does not provide it and if it exists on the OTM database schema. In addition, configurations done to the OTM-provided view, trigger, package, procedure, function, type, VPD policy, or synonym will be lost and it will be replaced with the OTM definition. You must store copies of such objects on the file system and recreate them after migrating to 6.4.3. Database object on the customer-defined database schema may be affected if it references any OTM database object.
>
> Any future configurations should use the Editioned View name in the customer-defined code for referencing the OTM table. You may not add non-edition objects which depend on the edition object provided by the Oracle Transportation Management.

Refer to "Using Edition-Based Redefinition" in Oracle Database Development guide.

Pre-migration tasks

1. Ensure that following database init parameters are set to sufficient values for Oracle Transportation Management application. It is recommended that values are set to 3000 or higher.

    ```
    Open_cursors = 3000
    Processes = 3000
    ```

    Run following query on the database to find values.

    ```
    select name, value from v$parameter where name in
    ('open_cursors','processes');
    ```

    Check with the DBA if values are lower than 3000.

2. Review agents that had used PRINT DOCUMENT or GENERATE DOCUMENT. As PRINT DOCUMENT worked on a Report Set, it should be replaced by a PREPARE REPORT action for each report in the Report Set. GENERATE DOCUMENT was available only to DOCUMENT agents. It would generate a report for an existing document wrapper. This concept was replaced by the GENERATE STANDARD DOCUMENT action on various business objects.

3. Review reports that specify "Both" for the Report Database field. You may want to copy these reports before migration and explicitly set one to be "On-Line Database", and one to be "Off-Line Database". This is not required, but may simplify the report upgrade process.

4. Run the following script to perform pre-migration checks. The script will check invalid objects, disabled constraints, missing database roles, deprecated data or any other checks required prior to starting migration. Check log file pre_migration_check_<dbsid>_<timestamp>.log and make sure that all checks are passed prior to starting migration.

   Two command scripts are provided and the script you use is dependent on your operating system.

   On the application server, change to directory <otm_install_path>/glog/oracle/script8

   ```
   UNIX shell script:
   ./pre_migration_check.sh
   ```

   or

   ```
   Windows command line script:
   pre_migration_check.cmd
   ```

5. Shut down all processes running against the database including Oracle Transportation Management application and web servers.

6. **A database backup or database level flashback restore point should be created prior to starting the migration. It may be required to restore the database to pre-migration state in case of a problem during migration, depending on changes made on the database and errors.**

Upgrading a 6.3 Database to 6.4.3

New Archive schemas were introduced in Release 6.4.2. Migration to 6.4.3 requires a one-time setup of the new archive schemas if migrating from Release earlier than 6.4.2. Run the following query to find out if new archive schemas have been set up. Log into database as SYSDBA and execute:

```
select username from dba_users where username in
('ARCHIVE_C','ARCHIVE_C_USER');
```

If it returns two rows as shown below, then the new archive schema is already set up and there's no need to execute the Archive Setup step.

```
USERNAME
-----------------------------
ARCHIVE_C
ARCHIVE_C_USER
```

If it does not return any row as shown above, then execute Archive setup step.

Archive Setup

This step creates two tablespaces (ARCH_C_TBS and ARCH_C_LOB_TBS) and two schemas ARCHIVE_C and ARCHIVE_C_USER.

1. On the application server, change to directory `<otm_install_path>/glog/oracle/script8`.
2. 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:

   `./archive_c/setup_archive_c.sh`

   or

   Windows command line script:

   `.\archive_c\setup_archive_c.cmd`
3. `Enter database connection id, SYSDBA user password, DBA user name who has privilege to alter database users, and its password, ARCHIVE_C user password , ARCHIVE_C_USER password`
4. Check log file archivec_setup_<dbname>_<timestamp>.log to ensure they are error free before proceeding to the next step. Look for errors by searching key words like "ORA-", "Caught exception","SP2-", "PLS-","IMP-","Warning:"," Caught exception", "SQL* Loader-" ," SQLException", "SQLRecoverableException" or "<Error>" within the log file.

Migration Steps

1. On the application server, change to directory `<otm_install_path>/glog/oracle/script8`.
2. Apply database changes by running `dbpatch` 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.sh`
   or

   Windows command line script:
   `dbpatch.cmd`
3. Enter the starting (migrating from) version of the Oracle Transportation Management, database connection ID, glogowner password, glogdba password, reportowner password, archive_c password, DBA user name who has privilege to alter database users, and its password, SYSDBA user password, and property file directory location, GLOGOAQ password and otmSystem user password when prompted.

   Note: Enter 6.3.0 for the starting version prompt if the starting version of the migration is 6.3.0 or earlier versions like 6.2.x or 5.5.x.

To Run in Non-interactive Mode

Running in non-interactive mode is accomplished by adding the twelve arguments when executing the script in the order as specified below.

```
./dbpatch.sh  <Database TNS id> <GLOGOWNER password> <GLOGDBA password>
<REPORTOWNER password>
      <ARCHIVE_C password> <Property file path> <SYSDBA password>
      <Starting version of OTM (like 6.3.4)>
      <DBA user name, other than SYS with privileges to create users (like
SYSTEM)>
      <password for this DBA user>
```

```
        <DIR_XML_USER password (if user doesn't exists then type in new
password>
        <password for application user FA.ADMIN (if user doesn't exists then
type in new password>
<password for database user GLOGOAQ (if user doesn't exists then type in new
password>
<password for application user otmSystem(if user doesn't exists then type in
new password>
```

Sample command:

```
./dbpatch.sh otmdb glogpw glogdbapw reportpw archivec_pwd ../../config
sysdbapw 6.3.4 system systempw newdirxmlpw faadminpw glogoaqpw otmSystempw
```

Check the log files (dbpatch_<version>_<dbname>_<timestamp>. log
,update_content_<version>_<dbname>_<timestamp>. log or any other generated log
file) to ensure they are error free before proceeding to the next step. Look for errors by
searching key words like "ORA-", "Caught exception","SP2-", "PLS-","IMP-","Warning:","
Caught exception", "SQL* Loader-" ," SQLException", "SQLRecoverableException" or
"<Error>" within the log file.

Note: If the migration is abruptly ended due to error and starting migration version is
6.3.x, some log files will be located in the **<glog_home>/glog/oracle/script8.63x**
directory.

If the migration is abruptly ended due to error and starting migration version is 6.4.0 or
6.4.1, some log files will be located in the **<glog_home>/glog/oracle/script8.642**
directory.

4. If an error occurs while applying particular patch in the migration, the database rolls back
    changes made by that patch. It will, however, commit changes once a whole migration patch
    has been successfully applied, and mark patch as "applied". Migration will not reapply that
    patch again in case of migration rerun. If an error occurs partway through the process, then
    you can fix the problem and rerun the process without harm. It may be required to restore the
    database to pre-migration state in case of a problem during migration, depending on changes
    made on the database and errors. 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.

5. **Optional Step**

    Execute the following command to install database listener job which enables processing of XML
    transmissions via OAQ.

    Note: Do not install the database listener job if OAQ is not being use for processing of XML
    transmissions.

Log into database as GLOGOWNER using SQL PLUS.

```
execute pkg_queue_management.install_queue_listener
```

Refer to the Integration Guide for details on the setup for use of Oracle Advanced Queuing.

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 Database

If Oracle Fusion Transportation Intelligence Historical Database (HDOWNER schema) is installed on Replicated Operational Database, make sure that the ROD database table structure is also upgraded to match with OLTP prior to proceeding.

The migration process calls the CSV load utility. This utility obtains the details of an Oracle FTI database from the property files and property sets. Before invoking the SQL script,

1. Set the following properties in the **CUSTOM** property set:

Table 2-1

Key	Value	Description
glog.database.fti.password		Oracle FTI database password

2. Also set the following properties in the **glog.properties** file**:**

Table 2-2

Key	Value	Description
glog.database.fti.dbserver	ftihost.mycompany.com	Fully qualified domain name of the Oracle FTI database
glog.database.fti.port	**1521**	Port that Oracle is using on the Oracle FTI database server
glog.database.fti.sid		Oracle FTI database service name

See the **Custom Properties** section of the Administration Guide for more information.

Change to directory `<otm_install_path>/glog/oracle/script8/advanced_analytics`.

1. Run the following script to perform pre-migration checks. The script will check invalid objects, disabled constraints, missing database roles, deprecated data or any other checks required prior to starting migration. Check log file `pre_migration_aa_check_<dbsid>_<timestamp>.log.` and make sure that all checks are passed prior to starting migration.

   Two command scripts are provided and the script you use is dependent on your operating system:
   UNIX shell script:

   `./pre_migration_aa_check.sh`

   or

   Windows command line script:
   `pre_migration_aa_check.cmd`

2. Apply database changes by running `aa_dbpatch` 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:

   `./aa_dbpatch.sh`
   or

   Windows command line script:
   `aa_dbpatch.cmd`

3. Enter the starting (migrating from) version of the Oracle Transportation Management, database connection for Oracle Fusion Transportation Intelligence database, hdowner password, sys password, property file location, database connection for OLTP database , and glogowner password for OLTP database when prompted. Note: Enter 6.3.0 for the starting version prompt if starting version of the migration is 6.3.0 or earlier versions like 6.2.x or 5.5.x.

   After the process has run, verify in the `aa_dbpatch_<version>_<dbsid>_<timestamp>.log,` `aa_update_content_<dbsid>_<timestamp>.log` files or any other generated log file (located in the same directory as source) that there are no errors. Look for errors by searching key words like "ORA-", "Caught exception","SP2-", "PLS-","IMP-","Warning:"," Caught exception", "SQL* Loader-" ," SQLException", "SQLRecoverableException" or "<Error>" within the log file.

   Note: If the migration is abruptly ended due to error and starting migration version is 6.3.x, some log files will be located in the **<glog_home>/glog/oracle/script8.63x/advanced_analytics** directory.

   If the migration is abruptly ended due to error and starting migration version is 6.4.0 or 6.4.1, some log files will be located in the **<glog_home>/glog/oracle/script8.642/advanced_analytics** directory.

If the solution to the error is apparent, then you can fix the problem and then rerun the aa_dbpatch process again, without harm (you may also want to rerun to see if it automatically resolves your problem). It may be required to restore the database to pre-migration state in case of a problem during migration, depending on changes made on the database and errors. 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_dbpatch and aa_update_content logs are clean.**

Migrating Oracle Transportation Management 6.4 Database to 6.4.3

Follow the instructions in the "Migrating Oracle Transportation Management 6.3 Database to 6.4.3" section to migrate Oracle Transportation Management 6.4 database to 6.4.3.

3. Migrating Workbench Layouts

Overview

If you are using Workbench layouts created in either 6.4, 6.4.1, or 6.4.2, you need to run the workbench layout migration. The Workbench Layout migration upgrades workbench layouts to the 6.4.3 format.

Versions Affected

The Workbench layout migration is required for installations running on version 6.4.0, 6.4.1 and 6.4.2 only. Since Workbench layouts were introduced in version 6.4, no migration is needed for earlier versions of Oracle Transportation Management.

Upgrading Workbench Layouts

After the OLTP database has been migrated to 6.4.3, complete the following:

1. Go to the `<GLOG_HOME>/install/workbench/` directory.
2. Copy two XML files (`mdsParamsExport.xml` and `mdsParamsImport.xml`) to the `<GLOG_HOME>/temp` directory.

   For both of these XML files, update the "jdbc-password" to access the Metadata repository (MDS repository) used by the application. All other properties are updated by the installer.

   > **Note**: The value of "jdbc-userid" and "jdbc-password" elements in `mdsParamsExport.xml` should be the **6.4.2 MDS schema** (or version of OTM you are migrating from) name and its password.

   Check with the administrator to get credentials for the `Metadata repository`.

3. The directory `<GLOG_HOME>/temp` should contain the following files:
   - Modified version of mdsParamsExport.xml
   - Modified version of mdsParamsImport.xml

4. Run the workbench migration script from the `<GLOG_HOME>/install/workbench` directory.

   For UNIX:

   ```
   ./migrate_workbench.sh from642
   ../../temp/mdsParamsExport.xml  ../../temp/mdsParamsImport.xml
   ```

   For Windows:

   ```
   migrate_workbench.bat from642
   ..\..\temp\mdsParamsExport.xml  ..\..\temp\mdsParamsImport.xml
   ```
   When migrating directly from 6.4.0 or 6.4.1 to 6.4.3, change the argument in the above command to "from640" or "from641" instead of "from642".

   Note:
   - At the end of the migration, input files located in the `temp` directory are deleted by the script.

- Please check the log file (migrate_workbench_<date/time>.log) in the <GLOG_HOME>/install/workbench directory if there is any error.

4. Migrating Configurations

If you are upgrading from a previous version of Oracle Transportation Management and have any configurations in the system other than customer-defined properties in the glog.properties file or instructions in a property set, you must copy those configurations from your old instance to your new instance. See the "Instructions on Upgrading" section in the Oracle Transportation Management Administration Guide for more details on what exactly needs to be copied. In order to facilitate this copying, a migration script can be generated based on your specific setup by following the directions below.

After the Oracle Transportation Management code has been installed and patched with the latest Roll-Up code and the database migrations have been completed, you will need to create the configuration migration script. This script is either a shell script or a batch file depending on which operating system you are installing on (UNIX or Windows, respectively). Once this script is created it should be reviewed and adapted as needed for your specific setup, then it should be run. This must be done on every application and web server being upgraded.

In order to migrate configurations from a prior release, the following resources must be accessible from the new install:

- the prior release's database
- the prior release's code installation (or a copy of it) with the database connection information properly configured

The following abbreviations will be used in the instructions below:

- $NEWOTM: the directory where the new version of Oracle Transportation Management is installed
- $OLDOTM: the directory where the previous version of Oracle Transportation Management is installed

To create and run the configurations migration script, do the following:

1. Log in as the user that the new version will use.
   a. This user will need full read access to $OLDOTM directory structure.
2. Change directory to `$NEWOTM/install`
3. Create the migration script based on your specific setup.
   a. Execute the following script (depending on operating system) to run a Java program to create the configuration script:
      i. [**Unix**] `./create_migration_script.sh $OLDOTM [<script name>]`
      ii. [**Windows**] `create_migration_script.cmd $OLDOTM [<script name>]`
   b. This Java program will read the database to determine potential locations of configured files (see the "Instructions on Upgrading" section in the Oracle Transportation Management Administration Guide for more details).
   c. The script creation program takes one mandatory parameter and one optional parameter. The mandatory parameter is the value of `$OLDOTM;` the optional parameter is the name of the script to be generated (default value is `migrate_customizations.sh` or `migrate_customizations.cmd`, depending on the operating system).
4. Carefully review the generated script to make sure it is appropriate for your specific setup (see the "Instructions on Upgrading" section in the Oracle Transportation Management Administration Guide for more details).
5. Once the script is reviewed, run the generated script while still in the `$NEWOTM/install` directory.

a. Assuming the default script name was used, this would be one of the following, depending on operating system:

    i. **[Unix]** `./migrate_customizations.sh`

    ii. **[Windows]** `migrate_customizations.cmd`

6. Test the various configurations out thoroughly before rolling into production.

5. 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 6.3.

The migration process for Fusion Transportation Intelligence should follow this sequence:

1. Migrating Fusion Transportation Intelligence Historical Database (HD)
2. Upgrading to Oracle Data Integrator 12c
3. Migrating Fusion Transportation Intelligence ETL in Oracle Data Integrator 12c
4. Upgrading to Oracle Business Intelligence Enterprise Edition 11g
5. Migrating Fusion Transportation Intelligence Metadata in OBI EE 11g
6. Migrating Fusion Transportation Intelligence Dashboard Reports in OBI EE 11g

Migrating Fusion Transportation Intelligence Historical Database (HD)

Pre-Requisites:

Ensure the following steps were done successfully before attempting the migration of the FTI Historical database:

1. Successful migration of the Oracle Transportation Management database (glogowner schema).
2. If the ROD is in use, then the successful migration of the ROD database (glogowner schema).

Migration Steps:

Refer to the **Upgrading the Oracle Fusion Transportation Intelligence Database** section for your corresponding Fusion Transportation Intelligence version available under the Migrating the Database chapter (above) for steps to migrate your existing Fusion Transportation Intelligence historical database and its data content.

Upgrading to Oracle Data Integrator 12c

Pre-Requisites:

1. Export the Master and Work repositories as .zip files. In other words, make a backup of the following Fusion Transportation Intelligence 6.2 specific Oracle Data Integrator database schemas:
   - ftimaster or ftiodi master
   - ftiwork or ftiodi master

Upgrade Steps:

The next step in the migration process is to upgrade Oracle Data Integrator (ODI) to 12c. Refer to the installation and migration documentation for Oracle Data Integrator before attempting to perform the upgrade.

Migrating Fusion Transportation Intelligence ETL in Oracle Data Integrator 12c

Prerequisites:

Ensure the following steps were completed before attempting the migration of the Fusion Transportation Intelligence ETL in Oracle Data Integrator 12c:

1. The Fusion Transportation Intelligence Master and Work repositories are backed-up.
2. Export the master and work repositories.

Migration of Customer-defined ETL:

1. Oracle Data Integrator 12c is installed. Please refer to the **Installing Oracle Data Integrator (ODI)** section in the Install Guide to install ODI 12c.
2. Upgrade ODI repositories by running `<ODI_12C_Home>\oracle_common\upgrade\bin\ua.bat`.
3. Configure ODIC. Please refer to the **Configuring ODI for Oracle FTI ETL** section in the Install Guide.

   Note: Import master and work repositories using Synonym INSET_UPDATE mode.

4. If you have added customer-defined ETL packages to the Fusion Transportation Intelligence Work repository in version 6.2 or 6.3, perform a manual merge of your customer-defined ETL code into the Fusion Transportation Intelligence 6.4 ETL package in ODI 12c.

   Note: ODI does not provide an out-of-the box utility for merging the Work repository, so it must be done manually.

5. If you have customer-defined ETL code in a PL/SQL package then you can do one of the following:

   - Manually re-implement your customer-defined ETL functionality in the Fusion Transportation Intelligence Work repository in ODI.

     Or

   - Plug in your customer-defined PL/SQL package within the Fusion Transportation Intelligence Work repository in ODI. Refer the FTI Reference Guide "Configuring the Fusion Transportation Intelligence ETL" for more documentation.

Post Migration Steps:

1. Perform a thorough sanity test of the Fusion Transportation Intelligence 6.4 ETL on ODI 12c to ensure stability of the ETL process.
   a. Go to **Operator**.
   b. Select **Load Plan and Scenarios**.
   c. Select the **FTI_LOAD Version-001** scenario.
   d. Right click and select **Execute**.
   e. Select **Global** context and click **ok**.
2. Once stable, you can drop the following backed-up Fusion Transportation Intelligence 6.2 ODI schemas, if they exist:
   - ftimaster
   - ftiwork
   - ftistage

Note: The above step is required only if you are migrating from version 6.2.

Migrating Targets

This has to be run immediately after the W_TARGETS_F table is created or when W_TARGETS_F table is empty. This is a one-time process to migrate the data from F_TARGETS to W_TARGETS_F. Execute the procedure in the following location in ODI:

Note: This is only required if you are migrating from versions 6.3.x or before to 6.4.x.

1. Go to ODI Studio > Designer > FTI_ETL > FTI_ETL > Procedures > 64_DATA_MIGRATION_W_TARGETS_F.
2. Right click and execute the procedure by selecting the content parameter as 'Global'.

Inserting Default Records for Dimensions

While migrating to 6.4.3 you need to run the following procedure in the ODI to insert the default records into all dimensions. For fresh install this is run as part of initial ETL; but for migration it has to be done manually. This will load default records with UNKNOWN value. This was changed from UN-KNOWN in 6.3.x versions.

1. Log on to ODI studio.
2. Go to the **Designer** tab.
3. Expand **FTI_ETL Project > FTI_ETL folder > Procedures**.
4. Select **INSERT_DEFAULT_RECORDS** procedure, right-click, and click **Run**.
5. Make sure to select the correct context (Global) and click **OK**.
6. Wait for the dialog box that indicates execution started to display, and then click **OK**.

Upgrading to Oracle Business Intelligence Enterprise Edition 12c

Prerequisites:

1. Create a backup of the following Fusion Transportation Intelligence version components in OBI EE:
    *   `advanced_analytics.rpd` (Fusion Transportation Intelligence metadata file)

        Note: Rename it to `advanced_analytics_bk_<Time Stamp>`. The file is located in the path `<OBIEE Root Dir>/server/Repository/`.

    *   `aa Webcat` folder (containing Fusion Transportation Intelligence dashboard reports, prompts, etc.). This file is located in the path `<OBIEE Root Dir>/OracleBIData/web/catalog/`.

Upgrade Steps:

Upgrade Oracle Business Intelligence Enterprise Edition (OBI EE) from version 11g to 12c. Refer to the Migrating from Oracle BI 11g to 12c section in *Fusion Middleware Migration Guide for Oracle Business Intelligence* for migrating OBIEE 11g RPD and Web catalog.

Migrating Fusion Transportation Intelligence Metadata in OBI EE 12c

Pre-requisites:

*   You have an 11g certified release (11.1.1.7, 11.1.1.8, and 11.1.1.9) installed.

5-3

- You have installed and configured the 12c software as per the instructions in *Oracle Fusion Middleware Installing and Configuring Oracle Business Intelligence Installation Guide*.
- You have file system permission on both the 11g and 12c systems.
- You have stopped the 12c BI instance in case it was running. The BI instance should not be running before migrating content into the new 12c instance.
- You have the JDK version 1.8.0_101 installed on your system.
- You have stopped OBIS1 and OBIPS1.

Overview:

Generating the BI Migration Tool (bi-migration-tool.jar)

The BI Migration Tool (bi-migration-tool.jar) is self-executing and self-contained. You must generate the BI Migration Tool jar file by using the BI Migration Script (migration-tool.sh| cmd). The BI Migration Script is made available after you configure the 12c domain. This step packages the components of the BI Migration Tool into a single, self-executing jar file, so that it can be easily transported on to an 11g system. To do this, you must have access to a 12c system with a configured domain. You must set up the 12c domain to provide sufficient infrastructure to run the BI Migration Tool and allow it to repackage itself. The 12c domain can be empty at this stage.

Creating the Export Bundle

The export bundle is a.jar file and consists of the metadata information from the 11g Oracle home.

Importing with the BI Migration Script (migration-tool.sh| cmd)

You must import the export bundle using the BI Migration Script (migration-tool.sh| cmd). The BI Migration Script automatically determines the Oracle home and the Domain home directories.

Steps:

Generating the BI Migration Tool (bi-migration-tool.jar):

Refer to Fusion Middleware Migration Guide for Oracle Business Intelligence and follow the steps mentioned under Generating the BI Migration tool. This will create the bi-migration-tool.jar file which you have to place in the machine running the 12c version.

Creating the Export Bundle:

Refer to Fusion Middleware Migration Guide for Oracle Business Intelligence and follow the steps mentioned under Creating the Export bundle. The end point of this step is the export bundle (.jar) file which contains 11g data (advanced_analytics.rpd and aa web catalog).

Contents of the Export Bundle:

To view the contents of the export bundle just created, unzip the <export_bundle_name>.jar file. The export bundle has a ton of scripting files and a ton of class files which are either unreadable due to binary-ness or just not good reading.

One of the folders is the PluginData where we find our Business Intelligence Archive (BAR) file. It's a new feature for OBIEE 12c. The metadata.bar file contains the 3 main artifacts for our upgrade. To view them, unzip the bar file.

The first artifact is our system-jazn-data.xml file containing our Application Roles and Policies from our source OBIEE 11g system.

Next is our OBEE 11g FTI web catalog file, live.cat (which is the name for the aa catalog file from 11g), in catalog archive format along with a customer-defined messages zip file, msgdb.zip.

The third artifact is our OBIEE 11g FTI RPD file, live.rpd (which is the name for advanced_analytics.rpd file from 11g), upgraded to the 12c version.

Importing with the BI Migration Script (migration-tool.sh| cmd):

Refer to Fusion Middleware Migration Guide for Oracle Business Intelligence and follow the steps mentioned under Importing the BI Migration Script (migration-tool.sh| cmd). This step uploads the rpd and catalog files migrated in the export bundle to the 12c environment.

> Note: This step replaces the RPD and catalog files that have been shipped with FTI release 6.4.3 which may have been configured to the 12c environment. So in order to keep the changes from 6.4.3 as well as bringing in the customer-defined changes from 11g migration, we need to merge the two RPDs.

Merging the Migrated RPD with 6.4.3 RPD:

In this step, we need to merge the advanced_analytics.rpd from 6.4.3 release with the advanced_analytics.rpd that was migrated from the 11g environment. The 11g rpd file is the same as the live.rpd file present in the export bundle.

Merging can be done using the merge option available in the Administration tool. For detailed steps, refer to the section *"Merging Repositories" -"Performing Full Repository Merges without a Common Parent"* in the *Oracle Business Intelligence Enterprise Edition 11g* documentation.

Once the merging is done, we are ready to upload the final RPD (which contains 6.4.3 changes as well as customer-defined changes from previous versions) to the OBIEE 12c. To do that, run the following command:

1. Navigate to <OBIEE 12c INSTALL PATH>/user_projects/domains/bi/bitools/bin.
2. In the command line, give ./datamodel.sh uploadrpd -I advanced_analytics.rpd -SI ssi -U weblogic -P <weblogic password>.
3. The datamodel.sh command is used to upload the rpd.
4. The **uploadrpd** parameter denotes that the RPD is being uploaded to the system.
5. The name of the RPD (the merged one) is given for the –I parameter.
6. The default service instance name is 'ssi' which is given for the –SI parameter.
7. –U and –P require the weblogic username and password to be specified (set at the time of configuration).
8. While executing the command, it asks for the RPD password. The default password for the shipped RPD is 'transportationintelligence1'. If you have changed it, use the changed password.

After this, you can restart the 12c servers to see the migrated metadata and dashboards.

Upgrading to Oracle Business Intelligence Enterprise Edition 11g

Prerequisites:

1. Create a backup of the following Fusion Transportation Intelligence version components in OBI EE:
   - `advanced_analytics.rpd` (Fusion Transportation Intelligence metadata file)

> **Note:** Rename it to `advanced_analytics_bk_<Time Stamp>`. The file is located in the path `<OBIEE Root Dir>/server/Repository/`.

- `aa Webcat` folder (containing Fusion Transportation Intelligence dashboard reports, prompts, etc.). This file is located in the path `<OBIEE Root Dir>/OracleBIData/web/catalog/`.

Upgrade Steps:

Upgrade Oracle Business Intelligence Enterprise Edition (OBI EE) from version 10g to 11g. Refer the installation and migration documentation available for OBIEE 11g before upgrading.

Migrating Fusion Transportation Intelligence Metadata in OBI EE 11g

Follow these steps to migrate from an existing Oracle Fusion Transportation Intelligence metadata to the Oracle Fusion Transportation Intelligence version 6.4 metadata on an Oracle Business Intelligence Enterprise Edition server.

Prerequisites:

Ensure the following steps were completed before attempting the migration of the Fusion Transportation Intelligence ETL in ODI 12c:

- Back up the Fusion Transportation Intelligence 6.2 version of metadata and `aa Webcat` folder

Migration of Standard Metadata in OBIEE 11g

If you have not configured the metadata, you can directly deploy the Fusion Transportation Intelligence 6.4 shipped standard metadata into OBI EE 11g.

Refer to the section "Copy RPD/Webcat/Images to OBIEE Server for Deployment" in the Install Guide for details on deploying the standard metadata.

Migration of Customer-defined Metadata in OBI EE 11g

If you have configured the standard metadata, complete these steps:

1. Copy the Oracle Fusion Transportation Intelligence metadata file (`advanced_analytics.rpd`) from the directory: `<otm_install_path>/fti`.
2. Upgrade the customer-defined Fusion Transportation Intelligence metadata file (`advanced_analytics.rpd`) and web catalog file (`aa`) which were originally compatible with version OBI EE 10g to OBI EE 11g. To achieve this, use the OBI EE 11g Upgrade Assistant available in the following paths:
   - Windows: $OBIEE11G_HOME$\Oracle_BI1\bin\ua.bat
   - Linux: $OBIEE11G_HOME$\Oracle_BI1\bin\ua

   Note: Refer to the section "Upgrade the Oracle BI Repository and Catalog" in the OBI EE 11g documentation for using the Upgrade Assistant.

   Note: For upgrading OBIEE catalog from 11.1.1.7 to 11.1.1.9, see the section "5.15.3 Upgrading Oracle BI Presentation Catalogs" in the Oracle Fusion Middleware Patching Guide.

3.   Merge all your customer-defined metadata changes (from your backed up metadata file) to the new `advanced_analytics.rpd` file using the merge option available in the Administration tool.

     For detailed steps, refer the section "Merging Repositories" -"Performing Full Repository Merges without a Common Parent" in the Oracle Business Intelligence Enterprise Edition 11g documentation.

     Merging basically requires three repository files:
     o   Current Repository: 6.3 Standard repository file,
     o   Modified Repository: Customer-defined repository file, and
     o   Original Master Repository: Dummy, Blank repository file.

4.   Add all customer-defined columns to the new subject area Transportation Intelligence in order use them in the reports. Refer the FTI Reference Guide for the detailed steps on how to add customer-defined columns into the new subject area.

5.   Deploy the `advanced_analytics.rpd` in the OBI EE using Enterprise Manager (EM) console. Refer to the "Deploying RPD and Web Catalog" section in the Installation Guide for the specific instructions.

6.   Stop and re-start the Oracle Business Intelligence Server and Oracle Business Intelligence Presentation Server services using Enterprise Manager (EM).

     Note: While deploying the RPD, if you encounter an error, you can view the log messages in the Enterprise Manager (EM) under the Diagnostic tab.

Migrating Fusion Transportation Intelligence Dashboard Reports in OBI EE 11g

Prerequisites

Complete the following steps before attempting the migration of the Fusion Transportation Intelligence dashboard reports in OBI EE 11g:

1.   Successfully upgrade OBI EE to version 11g.
2.   Perform a sanity test to verify OBI EE's stable operation.
3.   Successfully migrate Fusion Transportation Intelligence metadata to version 6.4 on OBI EE 11g.
4.   Back-up the Webcat folder `aa` (containing all Fusion Transportation Intelligence dashboard reports). This file is available in the directory `<OBIEE Root dir>/OracleBIData/web/catalog`.

     Note: The backup step is very important in case a roll-back is required if the report migration fails.

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

Migration of Standard Dashboard Reports

If you have not configured the webcatalog (dashboard reports, prompts etc), you can directly deploy the 6.4 shipped standard webcatalog into OBI EE 11g.

Refer to the section "Copy RPD/Webcat/Images to OBIEE Server for Deployment" in the Install Guide for deploying the standard web catalog.

Migration of Customer-defined Dashboard Reports

While migrating the Fusion Transportation Intelligence customer-defined reports from 6.3 or later versions to version 6.4.3, complete the following steps:

1. Take a backup of webcatalog before migration.
2. Complete migration.
3. Copy the customer-defined artifacts (reports and dashboards) to the new location.

In version 6.3 and later, Fusion Transportation Intelligence provides a reorganized metadata (subject area). You must modify all Fusion Transportation Intelligence standard and customer-defined dashboard reports that were used with versions 6.2 and earlier to use the new subject area. This section details the steps to migrate Fusion Transportation Intelligence dashboard reports from the deprecated 6.2 Fusion Transportation Intelligence subject area to the new Transportation Intelligence subject area (available in the 6.3 and later releases).

> **Note**: This documentation is only necessary if you are migrating from Fusion Transportation Intelligence release 6.2 or earlier to 6.4.

> **Note**: The deprecated subject area of Fusion Transportation Intelligence is hidden by default. To view this subject area, see the online help topic Report Common Properties and set the property SHOW_FTI_SUBJECT_AREA to true.

You can migrate the Fusion Transportation Intelligence Reports using the Catalog Manager utility available within the OBI EE (metadata repository) Administration tool. To enable this migration, Fusion Transportation Intelligence ships a Subject Area Mapper XML file (explained in a later section below) that can be used in the Catalog Manager to migrate your Fusion Transportation Intelligence reports. Below are some notes on Subject Area Mapper XML file:

*   The following two XML files are provided to help you migrate reports from the old subject area to the new subject area. These files are available at `<otm_install_path>/fti`.
    *   **subj_area_mapper.xml:** This replaces old subject area Fusion Transportation Intelligence with new subject area Transportation Intelligence in all the reports.
    *   **subj_area_column_mapper.xml:** This replaces all old subject area column names in the standard reports with new subject area column names.

*   These XML files are merely utilities to help migrate Fusion Transportation Intelligence reports (standard or customer-defined) that are based **only** on the **standard columns** available in the Fusion Transportation Intelligence subject area of the 6.2 release.

    If you have Fusion Transportation Intelligence reports based on customer-defined columns in your Fusion Transportation Intelligence subject area, then you should use the `subj_area_column_mapper.xml` file as a base and manually add the relevant mapping of your customer-defined columns. See the section "Adding Customer-defined Columns to Subject Area Mapper XML" for more details. And then, use the modified `subj_area_mapper.xml` file to migrate your Fusion Transportation Intelligence reports.

*   The steps listed here along with the `subj_area_column_mapper.xml` are broad guidelines to assist you in migrating all of your Fusion Transportation Intelligence reports without having to rewrite them. This is not an automated tool. Given the complex configurations with each Fusion Transportation Intelligence installation, it is strongly recommended that you review your configurations and test the migration in a test environment before performing the migration on your production environment.

It is important to remember that this migration is a manual process and by no means an automated process. The table below explains the relevance to perform migration of Fusion Transportation Intelligence reports from 6.2 to 6.3 subject areas:

Table 5-1

Customer Scenario	Include Customer-defined Columns in Subject Area Mapper XML	Execute FTI Reports Migration Using Catalog
Fresh Installation of Fusion Transportation Intelligence 6.3	Not Required	Not Required
Upgrade from Fusion Transportation Intelligence 6.2 with: • Customer-defined reports • No customer-defined metadata columns	Not Required	Required
Upgrade from Fusion Transportation Intelligence 6.2 with: • Customer-defined reports • Customer-defined metadata columns	Required	Required
Upgrade from Fusion Transportation Intelligence 6.1 or earlier releases	NOT SUPPORTED	NOT SUPPORTED

Migrating the Fusion Transportation Intelligence Dashboard Reports Using Catalog Manager

To migrate the Fusion Transportation Intelligence dashboard report using the OBI EE Catalog Manager, complete the following:

1. Launch the **OBI EE Catalog Manager**.
2. Open the Fusion Transportation Intelligence webcat folder **aa** in offline mode.
3. In the Catalog Manager, in the **tools** menu, go to the **XML Search and Replace**.
4. Browse to and select your **subj_area_mapper.xml** file to change the subject area in the Import from File text box.
5. Click **OK** to execute and complete the process.
6. Browse to and select your **subj_area_column_mapper.xml** file to change columns in the Import from File text box.
7. Click **OK** to execute and complete the process.
8. Place the modified catalog **aa** in a specified path as provided in the OBI EE Enterprise manager-Fusion Middleware control.
9. **Start** the OBI EE services.
10. Perform the post migration steps listed below to validate the migration.
11. Set the correct OBI EE privileges for the Transportation Intelligence subject area. This is required so that FTI users can access the subject area.
    a. In OBI EE, go to **Administration > Manage Privileges**.

b. For the entry of Subject Area: "Transportation Intelligence", select **Authenticated User**.

Post-migration Steps

The XML file that the Catalog Manager has processed might not have successfully migrated all report columns owing to incorrectly mapped customer-defined columns in the Subject Area Mapper XML. You should complete the following steps to test for successful migration.

1. Test all Fusion Transportation Intelligence reports after migration and specifically identify the report columns that failed the migration.
2. Once identified, correct the mapping in the Subject Area Mapper XML (as explained above) and re-run the migration steps.

Adding Customer-defined Columns to Subject Area Mapper XML

This section explains the steps to follow when the 6.2 version of the Fusion Transportation Intelligence subject area (metadata) has customer-defined columns and customer-defined reports were developed using these columns.

Pre-requisites

- Successful migration of all customer-defined FTI metadata changes in version 6.2 to version 6.3.
- Columns, which are included in your current reports but are not included in the 6.3 subject area, cannot be automatically removed from your reports. You will have to find these reports during testing and manually remove any columns that are no longer supported in the 6.3 subject area. To determine if a column in your current report is not included in the 6.3 subject area, use the Column Mapping tab of the spreadsheet referred to below.

Adding Customer-defined Columns in Subject Area Mapper XML:

1. Take the customer-defined columns from 6.2 metadata in the format "table"."Column".
2. Find the new corresponding customer-defined columns in 6.3 metadata under the subject area Transportation Intelligence.
3. Create xml line with xml tags with the old and the corresponding new customer-defined columns as shown in the example below:

```
<action command="textReplace" oldValue="&quot;Period&quot;.&quot;Day&quot;"
newValue="&quot;- Shipment Period Dimensions&quot;.&quot;Date&quot;"
ignoreCase="true"/>
```

In this example, the old column is

```
&quot;Period&quot;.&quot;Day&quot;
```

And the new column is

```
&quot;- Shipment Period Dimensions&quot;.&quot;Date&quot;
```

Various action commands can be used to migrate the reports. The action commands can be:

- o textReplace
- o renameColumn
- o renameSubjectArea

The detailed descriptions for action commands can be found in "Oracle Business Intelligence Presentation Services Administration Guide" under the section "Managing Presentation Catalogs Using Oracle BI Catalog Manager" under the heading "Searching for and Replacing Presentation Catalog Text Using Catalog Manager".

In this example, the action command 'textReplace' is sufficient to migrate the reports from the old subject area to the new subject area.

4. After adding the XML lines for all the customer-defined columns, add the XML lines to the Subject Area Mapper XML to change columns.

5. Execute the Subject Area Mapper XML in the catalog manager as described in the Migration Steps section.

    Note: The Catalog Manager will replace column names exactly as given in these columns; this includes the quotes. If your reports have variations on how the quotes are placed, the replace will fail. For example, your report can have &quot Period&quot. &quot Day&quot, Period. &quot Day&quot, &quot Period&quot.Day, or Period.Day. While you can add all permutations to the XML file, testing of all reports will be needed.

FTI Subject Areas Column Mapping Spreadsheet

Oracle provides the `subject_area_column_reference.xls` file as guide to assist you in the converting your reports to the new subject area. This spreadsheet details the table and column information for the 6.2 and 6.3 subject areas. Use this file as a guide.

Understanding the Column Mapping Spreadsheet

The spreadsheet consists of the following tabs:

- **Description tab**: contains a list and description of all of the tabs in the spreadsheet.

- **FTI tab**: contains the table and column information for the 6.2 subject area. By default, this tab is populated with the information contained in the shipped version of Fusion Transportation Intelligence 6.2. Column A contains the subject area of Fusion Transportation Intelligence. Column B contains the table name. Column C contains the column name. Column D contains a concatenation of columns B and C ("table name"."column name").

    You can replace the data in this tab with table and column information from your configured version of 6.2.

- **TI tab**: contains the table and column information for the 6.3 subject area. By default, this tab is populated with the information contained in the shipped version of Fusion Transportation Intelligence 6.3. See column explanation for FTI tab. TI tab also contains a Tool Tips column, but this is included for informational purposes only.

    You can replace the data in this tab with table and column information from your configured version of 6.2.

- **Column Mapping tab**: lists the old (6.2.x) subject area, including presentation table, presentation column, and column formula and maps it to the new subject area (6.3) including presentation table, presentation column and column formula.

    The column labeled 6.3 Column helps to track any columns that were removed in 6.3. If this column contains a value of No, then that column not longer exists in 6.3.

- **Formula Mapping tab**: contains the formulas that are used to generate the final XML. The Final XML is automatically populated as you configure columns A and C. You will configure these fields as you use the tool.

- **XML tab**: contains a place holder for the final XML. You will populate this tab when you use the tool.

Converting OBIEE Groups to Roles

Note: This section should be completed only after you migrate your OBIEE metadata (advanced_analytics.rpd) and OBIEE web catalog (aa) from an earlier version of Fusion Transportation Intelligence.

In previous versions of Fusion Transportation Intelligence, the Business Intelligence Roles or BI roles were mapped to groups in OBIEE. Since OBIEE no longer uses groups, the BI roles must be mapped to roles. Refer to the section Creating Business Intelligence Roles OBIEE, in the Oracle Transportation Management Installation Guide for Release 6.3.2 for complete details.

To convert the existing BI roles to OBIEE groups complete the step listed in this section. The steps listed here are required in addition to the steps listed in the Oracle Transportation Management Installation Guide.

1. First, complete the section, Creating Business Intelligence Roles OBIEE, in the Oracle Transportation Management Installation Guide for Release 6.3.2.
2. Add a new variable called ROLES in the metadata as follows:
   a. Open metadata **advanced_analytics.rpd** in Administration tool.
   b. Navigate to **manage->Variables**.
   c. Create a new variable called **ROLES** in system session variable.
   d. Add a new initialization block to the variable and name it as **GET_USER_ROLE**.
   e. Add the following source query to the GET_USER_ROLE initialization block:

   ```
   SELECT
   NVL(LTRIM(MAX(SYS_CONNECT_BY_PATH(BI_ROLE_GID,';'))
   KEEP(DENSE_RANK LAST ORDER BY curr),';'),'NO ACCESS')
   FROM
   (SELECT GL_USER_GID,
   BI_ROLE_GID,
   ROW_NUMBER() OVER (PARTITION BY GL_USER_GID ORDER BY BI_ROLE_GID) AS
   curr,
   ROW_NUMBER() OVER (PARTITION BY GL_USER_GID ORDER BY BI_ROLE_GID) -1
   AS prev
   FROM GL_USER_BI_ROLE
   where GL_USER_GID=':USER')
   CONNECT BY prev = PRIOR curr
   AND GL_USER_GID = PRIOR GL_USER_GID
   START WITH curr = 1
   ```

3. Locate the migrated catalog **aa**.
4. Get the catalog in a local folder which is mentioned as **offline catalog path**.
5. Execute the following commands to modify the permissions given to groups to assign it to corresponding roles:
   a. Go to the folder
   ```
   OBIEE_HOME\instances\instance1\bifoundation\OracleBIPresentationServicesC
   omponent\coreapplication_obips1\catalog\aa
   ```
   b. Execute the commands:
      i. ```runcat.cmd -cmd replaceAccountInPermissions -old "DEVELOPER" -oldType group -new "BIAuthor" -newType role -offline <offline catalog path>```
      ii. ```runcat.cmd -cmd replaceAccountInPermissions -old "EXECUTIVE" -oldType group -new "BIConsumer" -newType role -offline <offline catalog path>```

---

iii.

```
iii.  runcat.cmd -cmd replaceAccountInPermissions -old "LSP CUSTOMER" -
      oldType group -new "LSP CUSTOMER" -newType role -offline <offline
      catalog path>

 iv.  runcat.cmd -cmd replaceAccountInPermissions -old "LSP HOST" -
      oldType group -new "LSP HOST" -newType role -offline <offline
      catalog path>

  v.  runcat.cmd -cmd replaceAccountInPermissions -old "OPERATIONAL" -
      oldType group -new "BIConsumer" -newType role -offline <offline
      catalog path>

 vi.  runcat.cmd -cmd replaceAccountInPermissions -old "Presentation
      Server Administrators" -oldType group -new "BIAdministrator" -
      newType role -offline <offline catalog path>

vii.  runcat.cmd -cmd replaceAccountInPermissions -old "SERVPROV" -
      oldType group -new "SERVPROV" -newType role -offline <offline
      catalog path>

viii.  runcat.cmd -cmd replaceAccountInPermissions -old "NO ACCESS" -
       oldType group -new "NO ACCESS" -newType role -offline <offline
       catalog path>
```

6. Remove the deprecated Fusion Transportation Intelligence roles from the WebLogic Enterprise Manager.

   While migrating from OBIEE 10 g to OBIEE 11g, the groups are converted into roles and are saved in the WebLogic server. So you need to remove the following roles: DEVELOPER, EXECUTIVE, PRESENTATION SERVER ADMINISTRATOR, and OPERATIONAL from the WebLogic server.

   a. To remove these roles, complete the following:

      i. Go to http:<obiee_host>:<port>/em.

      ii. Login with your WebLogic user name and password created during the OBIEE installation.

      iii. Go to Farm_bifoundation_domain > Business Intelligence > coreapplication.

      iv. Go to Business Intelligence Instance > Security >Application Roles.

      v. Delete the following roles:

         - DEVELOPER,
         - EXECUTIVE,
         - PRESENTATION SERVER ADMINISTRATOR, and
         - OPERATIONAL.

      vi. Logout from the WebLogic Enterprise Manager.

Reference

For more information, refer to the following OBI EE documentation:

- "Managing Objects in the Oracle BI Presentation Catalog" in Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition.
- "Working with Objects in Catalog Manager" in Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition.
- "Managing Presentation Catalogs Using Oracle BI Catalog Manager" in Oracle Fusion Middleware Presentation Services Administration Guide for Oracle Business Intelligence Enterprise Edition

6. Migrating Global Trade Intelligence

This chapter is applicable only for existing Oracle Global Trade Intelligence (GTI) customers and it details the steps to migrate to the Oracle Global Trade Intelligence version 6.4.

Please refer to the following sections in Migrating Fusion Transportation Intelligence :

- Migrating Fusion Transportation Intelligence Historical Database (HD)
- Upgrading to Oracle Data Integrator 12c
- Migrating Fusion Transportation Intelligence ETL in Oracle Data Integrator 12c