

ORACLE®

COMMERCE

Version 11.3

Platform Upgrade and Migration Guide

Platform Upgrade and Migration Guide

Document Version

OCP11.3 MIGRATIONv1 4/28/17

Copyright

Copyright © 1997, 2017 Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

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

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



Contents

1	Introduction	1
	The CIM Migration Tool	1
	New in Oracle Commerce 11.3	1
2	Migration Overview	2
3	Pre-Migration Task Checklist	3
4	Install Oracle Commerce Platform 11.3	4
	Version Compatibility	5
5	Run the CIM Migration Tool	6
	Migration from Older Releases	6
	Migrating from Releases Before 10.2	6
	Data Migration	6
	The importConfig Utility	6
	Command Line Usage	6
	Utility Results	7
	CIM Command Line Options	7
	Migration Kit Location	7
	Starting CIM Migration	8
	Configuring Datasources	10
	Pre-Validation Checks	12
	Import Initial Data	13
6	Application-Specific Post-Migration Steps	14
	Oracle Commerce Core Commerce Notes	14
	Validate Gift Selection Changes in GWPManger	14
	Tax Status Has Been Expanded	14
	Commerce Profile Tools Changes	15
	Oracle Commerce Core Platform	15
	Process Exception Class Changes	15
	Migrating Bi-Directional Properties	15
	Commerce Profile Request Servlet Changes	17
	Droplet Exception and Droplet Form Exception Changes	17



7	Post-Migration Task Checklist	19
8	Migration Testing	20



1 Introduction

This guide describes how to upgrade from Oracle ATG Web Commerce 10.2 or later versions to Oracle Commerce Platform 11.3. It is written for programmers, DBAs, site administrators, and partners.

Before you begin the migration process:

1. Check the Supported Environments information on the My Oracle Support Web site. Make sure you are running on a supported version of your application server, JDK, database server, and JDBC driver.
2. Review the [Migration Overview](#) chapter.

This guide uses the convention <ATGdir> to represent the installation directory for ATG products. By default, the Windows installation creates the installation directory C: \ATG\ATG11. 3, but this location can vary in your installation.

For detailed information about Oracle Commerce Platform products, refer to the Oracle Commerce Platform documentation on the Oracle Technology Network Web site.

Important: You must upgrade your entire Oracle Commerce Platform installation; you cannot upgrade only portions of it. Oracle does not support running different versions of Oracle Commerce Platform products together. For example, using the 11.3 Oracle Commerce Platform with an older version of Commerce Service Center is not supported. The platform and all applications must have the same release number.

If you are performing a migration over more than two releases, you need to review any exported scripts and logs for any errors after each migration.

The CIM Migration Tool

Much of the migration process is performed by the Configuration and Installation Manager (CIM) Migration Tool, which automates schema and data migration based on your current installation and installed migration kits.

New in Oracle Commerce 11.3

For information on the features in this release, refer to the *Oracle Commerce Version 11.3 What's New Reference* knowledge article on My Oracle Support.



2 Migration Overview

To migrate from Oracle ATG Web Commerce 10.2 or later to Oracle Commerce Platform 11.3, you perform the following steps:

1. [Pre-Migration Task Checklist](#).
2. [Install Oracle Commerce Platform 11.3](#).
3. [Run the CIM Migration Tool](#).
4. Complete [Application-Specific Post-Migration Steps](#).
5. Complete [Migration Testing](#).

Each of these steps is detailed in the following chapters.



3 Pre-Migration Task Checklist

Before starting migration, complete these tasks:

1. Back up your Oracle Commerce Platform database.
2. Back up all of your `Publ i shi ng` and `Publ i shi ngAgent` directories --for example, these CIM-generated directories under `<ATG11di r>/home/servers/`:
 - `atg_publ i shi ng_lockserver/Publ i shi ng`
 - `atg_product i on_lockserver/Publ i shi ngAgent`You should also back up the `<ATG11di r>/home/Publ i shi ng` directory.
3. If your installation includes ATG Content Administration, complete all projects.
4. Process all reporting event logs. On the data warehouse load server, execute the `LoadAl l Avai l abl e` method on all loaders in `/atg/report i ng/datawarehouse/loaders/`.
5. Shut down all Oracle Commerce servers.
6. Shut down any search engines your environment uses.
7. Remove Oracle Commerce artifacts, such as server instances, EAR files, and data sources, from the application server.

Note: In addition to the tasks listed here, some applications may have their own requirements. Before starting the migration, review the instructions for each installed application, under [Application-Specific Post Migration Steps](#).

4 Install Oracle Commerce Platform 11.3

The next steps in the migration process are to install Oracle Commerce Platform 11.3, obtain the resources required for the migration process, and configure your environment.

1. Install Oracle Commerce Platform 11.3 as directed in the [Installation and Configuration Guide](#).
2. If you are migrating from releases earlier than 10.2, download the required migration kits for each of those releases and migrate them separately until you have migrated to release 10.2.

If you are migrating from 10.2, or later, you only need to download the `OCPlatform113MigrationKit`.

3. Copy the files to `<ATGdir>` and unpack the kit, which creates the `<ATGdir>\migration` directory.
4. **Note:** If you use both the Oracle Commerce Platform and the Oracle Commerce Service Center modules, you must unpack both kits before starting your schema migration since both the modules are migrated in one step. This note is only applicable if you are upgrading from versions older than 11.2. If you are migrating from version 11.2 to 11.3, there is no migration kit for Oracle Commerce Service Center.
5. If you are using an MSSQL database, delete all MSSQL script files that contain international data. For example:

```
<ATGdir>\migration\11.0to11.3\ARF\DW\base\sql\db_components\mssql\
arf_rpd_int18n_int_XXX.sql
```

```
<ATGdir>\migration\11.2to11.3\ARF\DW\Endeca\sql\db_components\mssql\
arf_search_datawarehouse_int18n_int_XXX.sql
```

```
<ATGdir>\migration\11.2to11.3\DCS\DW\sql\db_components\mssql\
arf_dcs_int18n_int_XXX.sql
```

Alternatively, you can leave a single script file for each language, assuming that your database is capable of supporting that language.

6. Copy your existing configuration files and application modules to the Oracle Commerce Platform 11.3 installation. Update application code as necessary to conform to new Oracle Commerce Platform 11.3 functionality. For more information, see the [Application-Specific Post Migration Steps](#) chapter.

Note: Commerce Reference Store and the reference applications do not migrate.
7. Copy the directories listed from your older (10.2-11.2) installation into your Oracle Commerce Platform 11.3 installation:



```
/home/Publishing/versionFileStore  
/home/servers/atg_svcagent_lockserver/publishing  
/home/servers/atg_production_lockserver/PublishingAgent  
/home/servers/atg_svcagent_lockserver /PublishingAgent  
/home/servers/atg_staging/PublishingAgent
```

8. If you use a Publishing Web Agent server, copy `/home/PublishingWebAgent` from your older (10.2-11.2) installations to your Oracle Commerce Platform 11.3 installation.

Version Compatibility

Before you begin installation of Oracle Commerce Platform 11.3, read the Oracle Commerce Supported Environments document in the My Oracle Support knowledge base (<https://support.oracle.com/>).



5 Run the CIM Migration Tool

The CIM Migration Tool is a utility that automates schema and data migration based on migration kits you install.

Migration from Older Releases

With this version of the CIM Migration Tool, you can migrate from older versions (10.2 and later), migrating sequentially one release at a time.

Migrating from Releases Before 10.2

If you are migrating from releases earlier than 10.2, download the related kits and perform migrations necessary to bring your installation to the 10.2 release using the instructions in the related migration guides. After you have migrated to 10.2, follow this guide to complete the migration to 11.3.

Data Migration

Data migration is not part of the Oracle Commerce Platform 11.1 to Oracle Commerce Platform 11.2 migration and running the CIM data migration option will not perform a data migration action. Review the Migration documentation for each release for more details.

The importConfig Utility

Before running the CIM Migration Tool, you can use the `importConfig` utility to identify which products and add-ons were installed in your previous installation. Otherwise you will need to manually select them in the CIM Migration tool. Before running this utility, verify that in your environment `$DYNAMO_HOME` reflects your ATG11.3 installation folder.

Note: Your previous installation directory must be available to use the `importConfig` utility.

Command Line Usage

From the command line, run the following script:

```
<ATGdir>\bin\importConfig.sh|bat [previous platform location]
```



Utility Results

When you run the `importConfig` utility, the following file is created:

```
<ATGdir>\CIM\data\import.xml
```

When the CIM Migration Tool runs, `import.xml` provides information on products and add-ons from previous installations that are listed in the migration information section.

CIM Command Line Options

Start the CIM Migration Tool from the command line. The following options are available:

```
cim.sh|bat [-m MODULELIST] [-debug] [-record] [-noencryption] [-batch
inputFilePath] [-migration] [-migrationpath migrationKitsBasePath]
```

Other options

Option	Description
<code>-migration</code>	Begins the migration process.
<code>-debug</code>	Enables debug mode.
<code>-record</code>	Records the CIM options you selected and at the end of the migration, saves the recorded file to a location you specify.
<code>-noencryption</code>	Works with record option. Normally the CIM options are recorded in encrypted form. If you use this option, they are written a readable form.
<code>-batch</code>	Runs the recorded file you specify, completing the migration automatically, without any manual intervention.
<code>-m</code>	Specifies specific modules, such as DAS, DPS, etc.

Migration Kit Location

The default path for the migration kits is, for example:

```
<ATGdir>\migration\11.2to11.3
```

If you have unpacked the migration kit or kits to `<ATGdir>\migration\11.2to11.3` use the unqualified CIM command:

```
cim.sh|bat -migration
```

Otherwise, specify the kit location. For example:



```
ci m.sh|bat -mi grati on -mi grati onpath \oracl e\mi grati on
```

Starting CIM Migration

When you start CIM with the `-mi grati on` switch, the `Mi grati on Uti li ty` task appears on the menu, for example:

```
=====CIM MAIN MENU=====
enter [h]Help, [q]Quit to exit
```

Choose the task you want to perform:

```
[R] Set the Administrator Password - Done (pending database import)
*[A] Select Application Server
[M] Migration Utility
[C] Custom CIM Plugin Launcher
```

Note: When the `-mi grati on` switch is used, the tasks `Product Selecti on` and `Database Configurati on` do not display.

Choose `M` to start the migration process. After you select `Mi grati on Uti li ty`, CIM displays the all the versions that can be migrated to the current release, for example:

```
-----MIGRATION SOURCE SELECTION-----
enter [h]Help, [m]Main Menu, [q]Quit to exit
```

Choose the version you are migrating from:

```
[1] 10.2
[2] 11.0
[3] 11.1
[4] 11.2
```

Select one >

After making your selection, the following menu displays:

```
-----PRODUCT SELECTION-----
enter [h]Help, [m]Main Menu, [q]Quit to exit
```

```
* [P] Product Selection
```

```
>
```

Choose all the products and add-ons that you have installed, then select `Product Selecti on` in menu and then select `Conti nue` when the `WARNING` message displays.



Note: Partial migration may cause the software to malfunction and is not permitted. You must migrate all modules.

If you have run the `importConfig` utility before running `cim`, the products and add-ons from the previous installation are selected by default in the available options.

CIM then displays a list of the installed products and add-ons that can be migrated, for example:

```
-----PRODUCT SELECTION-----
enter [h]Help, [m]Main Menu, [q]Quit to exit
```

Select product you wish to configure by entering the corresponding item number.

(Searching for products... done.)

Migration information: following products were found in previous installation:

Oracle Commerce Core Commerce

Content Administration

Choose one of the following options: (* = Currently selected)

- [1] Oracle Commerce Platform -
Includes, optionally, data warehouse components
- [2] Oracle Commerce REST -
RESTful Web Services
- [3] Oracle Commerce WebCenter Sites Extensions -
Includes Commerce Platform and Commerce CAS Reader.
- [4] Oracle Commerce Site Administration -
Includes Commerce Platform and Content Administration
- [5] Oracle Commerce Platform-Guided Search Integration -
Includes Commerce Platform. Select this option when Commerce Guided Search is used.
- *[6] Content Administration -
Includes Oracle Commerce Platform. Optional: Preview
- *[7] Oracle Commerce Core Commerce -
Includes Commerce Platform and Content Administration. Optional: data warehouse components, Preview and Merchandising UI
- [8] Oracle Commerce CAS Reader -
Includes Commerce Platform. Select this option when Commerce Platform is used to import data to Commerce CAS.
- [9] Oracle Commerce Service Center -
Agent-facing commerce application

```
[D] Done
Select one or more >
```

In this menu and those that follow, select the products and add-ons you want to migrate and the migration tool will update the schema and import data. When your selection is complete, the Migration option displays:

```
-----PRODUCT SELECTION-----
enter [h]Help, [m]Main Menu, [q]Quit to exit
```

```
[P] Product Selection - Done
*[C] Migration Selection
```

```
>
```

When the Product Selection is complete, choose option C to proceed with the migration process. The following screen displays:

```
-----MIGRATION SELECTION-----
enter [h]Help, [m]Main Menu, [q]Quit to exit
```

Choose the Migration task you want to perform:

```
*[1] 10.2 -> 11.0
[2] 11.0 -> 11.1
[3] 11.1 -> 11.2
[4] 11.2 -> 11.3
```

```
Select one >
```

After selecting a migration step, the following screen displays:

```
-----DATABASE MIGRATION SELECTION-----
enter [h]Help, [m]Main Menu, [q]Quit to exit
```

```
*[D] Database Migration
[C] Cancel
```

```
>
```

Choose D to begin the Data Migration process, which is described in the following section.

Configuring Datasources

In the next step, you choose and configure the named datasource for each server you are using. The following example shows the servers that may be migrated by choosing the related datasource:

```
*[1] Reporting Data Warehouse
[2] Reporting Loader
```



- [3] Publishing
- [4] Production Core
- [5] Agent Datasource
- [D] Done

Select one >

Select the server you want to migrate, for example, Reporting Loader:

-----CONFIGURE DATASOURCE REPORTING LOADER-----

enter [h]Help, [m]Main Menu, [q]Quit to exit

- [C] Connection Details
- [T] Test Connection
- [E] Export Database Migration Script
- [S] Migrate Schema
- [D] Migrate Data
- [X] Export Database Cleanup Script
- [L] Cleanup Schema
- [0] Configure Another Datasource

After you have provided connection details, you may optionally test the connection by choosing Test Connection. Once you verify that you can connect to the database schema, for each server complete the following steps:

1. **Export Database Migration Scripts.** You must complete this step because the exported scripts are used to migrate the schema. After exporting the migration scripts, review the scripts and customize them, if needed for your environment.
2. **Migrate Schema.** After customizing the scripts, execute them by selecting the Migrate Schema option. When this completes, review the `ci m. l og` that has been created in `<ATGdir>/ci m/l og` for any errors.
3. **Migrate data.** Complete this step if data needs to be migrated.
4. **Export Database Cleanup Scripts.** These scripts list the tables that are no longer needed and would be dropped if you run Cleanup Schema. Review the exported scripts before running Cleanup Schema and make a copy the contents of the listed tables if you need to save the data. By running this menu option, the CIM Migration Tool may pick up either `dynamic_metadata_ddl_cleanup.sql`, `versioned_dynamic_metadata_ddl_cleanup.sql` or both depending on your configuration. If a script is not exported, you do not have to run the Cleanup Schema option.
5. **Cleanup Schema.** Execute the cleanup scripts by choosing Cleanup Schema. Execute this step only when you are ready to drop the tables that are no longer needed. It is not necessary to carry out this step before moving to migrate another server and datasource.

Note that in the Publishing module, the following tables have been renamed, and are no longer required. The tables have been preserved in case they contain data required by other applications. All foreign keys, primary keys and indexes have been dropped from these tables. Note that some of these tables may not exist in your environment:



Original Table	Renamed Table
dcscat_chldcat	dcscat_chldcat_del
dcscat_chldprd	dcscat_chldprd_del
dcscat_subcats	dcscat_subcats_del
dcscatfol_chld	dcscatfol_chld_del
dcscchildfol_cat	dcscchildfol_cat_del
dcscchildfol_pl	dcscchildfol_pl_del
dcscprd_chldsku	dcscprd_chldsku_del
dcscroot_cats	dcscroot_cats_del
dcscroot_subcats	dcscroot_subcats_del
srchcfg_fol_chldcfgs	srchcfg_fol_chldcfgs_del
srchcfg_fol_chldfol	srchcfg_fol_chldfol_del
srchcfg_rpset	srchcfg_rpset_del
srchcfg_synlnk	srchcfg_synlnk_del

If you are migrating over more than two releases, start with the oldest version and migrate to the next newest version till you reach the current release. After each migration, review any exported scripts and review the CIM logs for any errors. The main purpose of the exports are described in the following section, which will help you understand which DDLs are executed as part of each migration. This allows you to make any changes necessary for your environment before you begin the migration process. If you are migrating from versions earlier than 11.2, please refer to the appropriate version of the Migration Guide.

Pre-Validation Checks

When migrating from 11.2 to 11.3, you may need to execute the following pre-validation checks.

DAF Search Module

If you have the DAF Search module in your installation, please execute the following:



```
<ATGdir>/migration/11.2to11.3/Pre-Verification/DAF/Search/Versioned/sql/db_components/DBMS_PLATFORMVersioned-refinement_ddl_verification.sql
```

Note: The *DBMS_PLATFORM* can be oracle, mssql or db2.

Important: If any of the SQLs in this file return rows, contact Oracle Core Commerce Support and do not continue the migration process. If the script reports `ORA-00942: table or view does not exist`, it indicates that this module does not exist in your installation and you can ignore this step.

Import Migration Data

After you have migrated schemas, the Import Migration Data task imports migration data for the applications you are configuring.

Data migration is not part of the Oracle Commerce Platform 11.1 to Oracle Commerce Platform 11.2 migration. Running the CIM Data Migration option will not migrate data from 11.1 to 11.2.

However, if you are migrating from earlier versions to 11.1, data migration is performed by the CIM tool. Refer to the Migration documentation for each release for additional details. Data migration from 11.2 to 11.3 is performed by the CIM Migration Tool.

Import Initial Data

Once all of your migration steps for each selected release are successful, the menu item **Import Initial Data** displays on the **Migration Utility** menu, for example:

Choose the Migration task you want to perform:

- *[1] 10.2 -> 11.0
- [2] 11.0 -> 11.1
- [3] 11.1 -> 11.2
- [4] 11.2 -> 11.3
- [I] Import Initial Data

Select one >

Select the **Import Initial Data** menu option to execute the `startSQLRepository` imports required for migration. You should run these imports in the required order.



6 Application-Specific Post-Migration Steps

After completing all tasks described in previous sections, review these migration steps for specific applications, beyond the migration steps described in previous chapters.

Any applications not listed here can be assumed to have no additional migration issues.

Oracle Commerce Core Commerce Notes

Information relating to Core Commerce is discussed in the following section.

Validate Gift Selection Changes in GWPManger

The Boolean `pReplaceExistingSelection` parameter has been replaced with the `CommerceItem pItemToReplaceExistingQuantity` parameter.

If you have overridden the original `validateGiftSelection` method, you must update your code to pass in the new set of expected parameters. These include:

- `CommerceItem pItemToReplaceExistingQuantity`
- `Boolean pReplaceRemovedQuantity`
- `Boolean pReplaceFailedQuantity`

The following is the structure required:

```
protected void validateGiftSelection(RepositoryItem pOrderMarker,
    String pProductId, String pSkuId, long pQuantity, double
    pQuantityWithFraction, CommerceItem
    pItemToReplaceExistingQuantity, boolean pReplaceRemovedQuantity,
    boolean pReplaceFailedQuantity, Map pExtraParameters)
```

Tax Status Has Been Expanded

The `/atg/payment/tax/TaxStatus` interface has been expanded to include the following declarations:

```
public boolean getIsTaxIncluded();
public double getMiscTax();
public Map<String, Double> getMiscTaxInfo();
public Map<String, Double> getDetailedTaxInfo();
```

If you have implemented the `TaxStatus` interface, you will need to provide implementations of the new methods.



Commerce Profile Tools Changes

The `CommerceProfileTools` properties `getProfileAddress` and `removeProfileRepositoryAddress` have been modified to return or remove the primary address when 'My Home Address' is passed in as a parameter.

If you contained 'My Home Address' as a map key within `secondaryAddresses`, you will now have the primary address returned or removed.

Oracle Commerce Core Platform

Issues relating to Core Platform are discussed in the following section.

Process Exception Class Changes

The `atg.process.ProcessException` class extends the `ContainerException` class, which now includes a `String` `errorCode` property. The `mErrorCode` integer member variable has been removed.

If you have any code that requires a `processException` on `errorCode` to be an integer, you will have to modify the `errorCode` to treat it as a string. Note that the `setErrorCode(int)` still exists, although the integer value is now stored in the exception as a string.

Migrating Bi-Directional Properties

Some versioned relationships can be modified so that the extra asset version column is no longer needed. These bi-directional properties are defined on two different item descriptors that reciprocate each other and are stored in a single table. For example, this could be a list of products in a category and parent categories on a product. Use of these properties requires a data migration, as there are changes to the repository definition.

Note: Not all relationships can be migrated. The property must be a LIST-SET or SET-SET many-to-many property with one of its SET sides configured for read-only (or `writable="false"` in the repository definition).

The following is an example of a LIST-SET property from the product catalog with the `productparentCategories` set to read-only.

```
<item-descriptor name = "category" >
  <table type = "primary" name = "dcs_category" >
  </table>
  <table name = "dcs_cat_chldprd" multi-column-name = "sequence_num"
    type = "multi" id-column-name = "category_id" shared-table-sequence = "1" >
    <property name = "fixedChildProducts" data-type = "list"
      column-name = "child_prd_id" component-item-type = "product"
      writable = "true" >
    </property>
  </table>
</item-descriptor>
<item-descriptor name = "product" >
  <table type = "primary" name = "dcs_product " >
```

```

</table>
<table name = "dcs_cat_chldprd" type = "multi" id-column-name = "child_prd_id" >
  <property name = "parentCategories" data-type = " set "
    column-name = "category_id" component-item-type = "category"
    writable = "false" shared-table-sequence = "2" >
  </property>
</table>
</item-descriptor>

```

The existing table definition would be as follows:

```

CREATE TABLE dcs_cat_chldprd
(
  sec_asset_version NUMBER ( 19 ) NOT NULL ,
  asset_version     NUMBER ( 19 ) NOT NULL ,
  category_id       VARCHAR2 ( 40 ) NOT NULL ,
  sequence_num      INTEGER NOT NULL ,
  child_prd_id     VARCHAR2 ( 40 ) NOT NULL ,
  CONSTRAINT dcs_cat_chldprd_p PRIMARY KEY (category_id , sequence_num ,
  asset_version , sec_asset_version)
) ;
CREATE INDEX ct_chldprd_cpi_idx
ON dcs_cat_chldprd (child_prd_id)
CREATE INDEX ct_chldprd_cid_idx
ON dcs_cat_chldprd (category_id) ;

```

To perform a migration of a table with a sec_asset_version column:

1. Create a temporary table identical to the table you want to migrate. However, exclude the sec-asset-version column. The following example, used throughout these instructions, shows how you might create a temporary table named dcs_cat_chldprd_temp. Make sure that you rename the primary key constraint.

```

CREATE TABLE dcs_cat_chldprd_temp
(
  asset_version     NUMBER ( 19 ) NOT NULL ,
  category_id       VARCHAR2 ( 40 ) NOT NULL ,
  sequence_num      INTEGER NOT NULL ,
  child_prd_id     VARCHAR2 ( 40 ) NOT NULL ,
  CONSTRAINT dcs_cat_chldprd_temp_p PRIMARY KEY
  (category_id , sequence_num , asset_version)
) ;

```

2. Add an INSERT INTO SELECT statement to copy rows of data from old tables to the new temporary table, joining the SELECT query with the writable side's primary table, or dcs_category in this example:

```

INSERT INTO dcs_cat_chldprd_temp
  (category_id ,
  asset_version ,
  sequence_num ,

```



```

        child_prd_id)
SELECT DISTINCT t1 . category_id ,
               t1 . asset_version ,
               t1 . sequence_num ,
               t1 . child_prd_id
FROM   dcs_cat_chldprd t1 ,
       dcs_category t2
WHERE  t1 . asset_version = t2 . asset_version
      AND t1 . category_id = t2 . category_id

```

Note that if `shared-table-sequence="2"` was associated with the writable side, `t1.sec_asset_version` would be used instead of `t1.asset_version` in the list of selected columns and the join would then be made against `dcs_cat_chldprd.sec_asset_version`.

3. DROP the original table.

```
DROP TABLE dcs_cat_chldprd ;
```

4. Rename the temporary table with the original table's name.

```
ALTER TABLE dcs_cat_chldprd_temp
  RENAME TO dcs_cat_chldprd ;
```

5. Rename the primary key constraint.

```
ALTER TABLE dcs_cat_chldprd
  RENAME CONSTRAINT dcs_cat_chldprd_temp_p to dcs_cat_chldprd_p
```

6. Create the indices.

```
CREATE INDEX ct_chldprd_cpi_idx
  ON dcs_cat_chldprd (child_prd_id) ;
CREATE INDEX ct_chldprd_cid_idx
  ON dcs_cat_chldprd (category_id) ;
```

7. Remove the `shared-table-sequence` attribute from the XML repository definition for the relationship.
8. Restart the servers.

Commerce Profile Request Servlet Changes

The `dafpeline` has been modified to correctly use the `CommerceProfileRequestServlet`. Please note that this class's `loadOrdersOnAutoLogin` property is now set to `false` by default. This property was set to `false` because a new `CommerceProfilePropertyServlet`, which identifies order and pricing information, loads the order during automatic login.

If you created a local configuration for a `ProfileRequestServlet` to set the `loadOrdersOnAutoLogin` to `true`, you must remove that configuration as the `CommerceProfileRequestServlet` and the `CommerceProfilePropertyServlet` cannot run simultaneously.

Droplet Exception and Droplet Form Exception Changes

Additional constructors have been added to `DropletException` and `DropletFormException`. These constructors handle new `inputErrorCode`:



```
public DropletException (String, String, String)
public DropletException (String, Throwable, String, String)
public DropletFormException (String, Throwable, String, String)
public DropletFormException (String, String, String)
```

These constructors were added to prevent ambiguous references to the `DropletException` and `DropletFormException` constructors by not passing in null for some parameters of the existing constructors. If you use the null constructors, you may encounter the same situation. It is suggested that you do not pass in null for fields that you do not wish to set.



7 Post-Migration Task Checklist

After all migrations are complete, perform the following tasks:

1. If you use ATG Content Administration, it is good practice to launch a full deployment to all workflow targets. This ensures that content is fully synchronized between the asset management server and workflow targets.

Note: Before launching a full deployment, remap the repositories for your site in the Content Administration Console. This step is necessary because a number of repositories (such as Product Catalog and Site) are secured by default in this release.

2. Clear browser caches.
3. Clear the application server cache.

For example, on JBoss, empty the contents of each server directory:

```
/work/jboss.web/local host
```

1. It is good practice to review any existing scenarios and restart them if appropriate.
2. Test the upgraded platform and applications (see [Migration Testing](#)).



8 Migration Testing

To verify the success of your migration production servers, follow these steps:

1. Assemble EAR files for the asset management and production servers.
2. Deploy the EAR files to the application server and start the production and asset management servers.
3. From the Oracle Commerce 11.3 Business Control Center, verify that you can view:
 - All internal users created in Oracle Commerce Platform 11.2, or earlier releases
 - All assets created in Oracle Commerce Platform 11.2, or earlier releases
4. Confirm that the Oracle Commerce Platform 11.3 production server contains all assets that were accessible on the previous releases of Oracle Commerce Platform.
5. Create some test projects and add new assets and edit existing ones. Deploy these assets and verify that the deployment is successful.
6. Confirm that deployed file assets are in the correct location and behave as expected.