Oracle® Fusion Middleware
Migrating Oracle Business Process Management Projects

12*c* (12.2.1.4.0)

E95695-01

September 2019

ORACLE®

Oracle Fusion Middleware Migrating Oracle Business Process Management Projects, 12*c* (12.2.1.4.0)

E95695-01

Contents

4   Understanding Project and Process Migration

5   Understanding Catalog Migration

6   Understanding Business Process Language Migration

7   Understanding BPM API Migration

8   Limitations and Known Issues

Preface

Migrating Oracle Business Process Management Projects describes migration tools, migration techniques, and provides reference material and best practices to enable you to migrate Oracle Business Process Management 10gR3 projects to Oracle Business Process Management 12*c*.

Audience

This guide is intended for administrators and analysts to learn how to migrate Oracle Business Process Management (Oracle BPM) 10*g*R3 projects to Oracle BPM 12*c*.

The information in this guide assumes that you have in-depth knowledge of Oracle BPM 10*g*R3 and 12*c* projects and system infrastructure.

Documentation Accessibility

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`.

Accessible Access to Oracle Support

Oracle customers who 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.

Related Documents

Refer to the Oracle Fusion Middleware library on the Oracle Help Center for additional information.

- For Oracle Business Process Management information, see Oracle Business Process Management.

- For Oracle SOA Suite information, see Oracle SOA Suite.

- For versions of platforms and related software for which Oracle products are certified and supported, review the Certification Matrix on OTN.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

ORACLE®

1

Introduction

Get an overview of the migration utility and Oracle BPM Suite 12c.
This chapter includes the following sections:

- About BPM Migration Paths
- Introduction to the BPM Migration Utility
- BPM 12c Applications
- How to Read This Guide

1.1 How to Read This Guide

This guide provides the information needed to perform the following tasks:

- Learn about migration (this chapter)
- Learn about and use the BPM migration utility and the BPM OU migration script

  See Migrating a BPM Project

- Understand business processes migration and complete the migration using additional manual steps

  See Understanding Project and Process Migration

- Understand business catalog migration and complete the migration manually as needed

  See Understanding Catalog Migration

- Troubleshoot issues related to limitations and learn about known issues

  See Limitations and Known Issues

1.2 About BPM Migration Paths

This topic summarizes BPM migration paths between BPM 10gR3 and BPM 12c.

This guide focuses specifically on migrating your BPM Studio projects from BPM version 10gR3 to BPM version 12c. Oracle BPM Studio is provided as a plug-in that runs in an Oracle JDeveloper environment and allows you to model and implement business processes.

Migration to targets other than BPM 12c are outside the scope of this document.

To migrate from a release previous to BPM 10gR3 to BPM 12c, you must first migrate to BPM 10gR3, and then to BPM 12c.

Migrating projects from BPM 11g to BPM 12c is accomplished by opening your BPM 11g projects in the 12c version of BPM Studio.

A discussion of how to migrate live data and in-progress processes from BPM 11g to BPM 12c is outside the scope of this document.

See Preface for information on contacting Oracle Support to assist with your migration.

1.3 BPM Migration Overview

This topic describes, in general, project migration feasibility and approach.

There are differences in the architecture, tools, applications, functionality and internal implementation between BPM 10gR3 and BPM 12c.

Some 10gR3 elements can be migrated successfully to 12c without changes, and some elements will require a degree of manual intervention up to and including complete replacement with new 12c constructs.

1.4 Introduction to the BPM Migration Utility

This topic provides an introduction to the BPM migration utility, its capabilities, and the output artifacts it creates.

The BPM migration utility is an Ant based utility that migrates a BPM 10gR3 project to a BPM 12c project. The BPM migration utility creates:

*   A BPM 12c `.exp` file

    An exported BPM 12c BPM project that you can import into Oracle BPM Studio or Oracle Business Process Composer

*   A migration report

    Contains details on the migration of each component. Lists components that were not migrated, warnings and errors.

The BPM migration utility migrates all mappable project preferences, process designs, compatible catalog components and organization. It generates a BPM 12c project with the same name and structure from the original BPM 10gR3 project.

For details on the usage and the arguments it requires, see Using BPM Migration Utilities to Migrate Projects.

The migration of a BPM 10gr3 project to BPM 12c consists of the following:

*   Design Time Artifacts

    Most of the design time artifacts are automatically migrated by the BPM migration utility. You must perform manual tasks on some artifacts in order to complete the migration. These tasks are described in Understanding Project and Process Migration and Understanding Catalog Migration .

*   Organization

    The OU migration script automatically migrates BPM 10gr3 organizations to their equivalent in 12c. See About the BPM OU Migration Script.

1.5 BPM 12c Applications

Oracle BPM Suite 12c provides the following applications to model, implement, run, and manage processes and tasks:

*   Oracle Business Process Management Studio

Oracle BPM Studio is a component of the Oracle BPM Suite that enables process developers to create process-based applications and to model and implement business processes. BPM Studio is provided as a plug-in that runs in an Oracle JDeveloper environment.

- Oracle Business Process Composer

  Oracle Business Process Composer is a web-based, collaborative application that enables business analysts to model business processes and perform the basic implementation of some of the artifacts used in the business process.

- Oracle Business Process Management Workspace

  Oracle Business Process Management Workspace is a customizable web-based interface enabling users to access and manage tasks and process instances according to their roles and responsibilities. .

- Oracle Business Process Management Server

  The Oracle Business Process Management Server provides a runtime environment for running business processes. It provides native support for both BPMN 2.0 processes. For more information, see Administering Oracle BPMN Process Service Components and Engines in *Administering Oracle SOA Suite and Oracle Business Process Management Suite*.

2
BPM Migration Utilities

Get an overview of the utilities that are available to assist with the migration of BPM 10gR3 projects to BPM 12c.

This chapter includes the following sections:

- About the BPM Migration Utilities
- BPM Migration Utility Supported Platforms
- BPM Migration Environment Pre-Requisites
- Preparing for the BPM Migration Utility
- BPM Migration Utility Options
- Generating Migration Statistics for Components in Scripts
- Running the BPM Migration Utility
- BPM Migration Utility Outputs
- Accessing the BPM Migration Utility Output
- About the BPM OU Migration Script
- Preparing for the BPM OU Migration Script
- BPM OU Migration Script Options
- Running the OU Migration Script

2.1 About the BPM Migration Utilities

Oracle BPM 12c includes two utilities to facilitate migration.

These are:

- BPM migration utility
- BPM OU migration script

The BPM migration utility is an ant script that migrates BPM processes, screenflows and business catalog artifacts from 10gR3 to 12c. It creates a BPM 12c export project file that you can import into 12c. It also creates a migration log that you can use to troubleshoot problems that occurred during the migration process.

The BPM OU Migration Script is an ant script that migrates organization information such as roles and calendar rules from a BPM 10gR3 environment to BPM 12c.

2.2 BPM Migration Utility Supported Platforms

The BPM migration utility is supported on a number of platforms.

Supported platforms include:

- Oracle Enterprise Linux 5.8 (64 bit)

- Oracle Enterprise Linux 6.4 (64 bit)

- Windows 7 Professional (64 bit)

- Windows 8.1 Enterprise (64 bit)

2.3 BPM Migration Environment Pre-Requisites

The BPM 10gR3 source environment must include a number of packages to run the BPM migration utility and the BPM OU migration script.

The packages required to be present in your 10gR3 environment include:

- Java SE Development Kit

    http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

    Use a JDK version supported by your target BPM 12c. See the *Oracle Fusion Middleware Supported System Configurations* information at http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html. Download the appropriate certification matrix spreadsheet for your product. Use the matrix to confirm the version(s) of JDK supported for your BPM 12c target based on processor, OS version, and Fusion Middleware products.

    See *Installing SOA Suite and Business Process Management Suite Quick Start for Developers* for additional information.

- Apache Ant (TM) version 1.8.4

    http://archive.apache.org/dist/ant/binaries/apache-ant-1.8.4-bin.zip

- Oracle Business Process Management Suite 10gR3

- JDeveloper 12c with Oracle BPM Studio 12c plug-ins. Obtain these components using the BPM Suite Quick Start Installer. See http://www.oracle.com/technetwork/middleware/bpm/downloads/index.html

To view, edit, deploy and run the migrated project you must have an environment with Oracle Business Process Management Suite 12c installed. For information on how to install Oracle BPM Suite, see *Installing and Configuring Oracle SOA Suite and Business Process Management*.

2.4 Preparing for the BPM Migration Utility

Before running the BPM migration utility, ensure that your projects and environment are properly prepared.

Latest BPM 10gR3 version - the project to be migrated must be validated using BPM 10gR3 (10.3.3) with the latest hotfix applied (build 101298 at this writing)

No compilation errors - the project to be migrated must be free of compilation errors

Project dependencies - all dependent projects (that is, projects that work together and have **Project Preferences > Dependency > Use Project Dependency** checked) must be located under the same folder.

Decompressed filenames - some decompression tools render decompressed filenames incorrectly by using invalid characters. Always use BPM Studio 10gR3 to expand imported project .exp files. The migrator project location parameter must point to the project root folder created by BPM Studio 10gR3. For example `C:\OracleBPM \SampleProject`.

Non-English characters - replace foreign language characters that might render as invalid characters (such as 'è' being rendered as '?') with English language replacements (such as 'e'). Non-English characters within BPM file metadata is acceptable.

BPM 5.5 and 5.7 file metadata - The BPM migration tool does not support BPM 5.x metadata within certain files (such as .xpdl and .xsdl files). The Ant console output or the migration report indicates the names of these files. To update the file metadata, edit the affected file using BPM Studio 10gR3, make an immaterial change to the activity (for example, move it slightly), and save the project.

An example migration report error message from invalid file metadata is:

```
SEVERE: path/to/process_name.xpdl is a legacy file format and cannot be
processed. Only 10.3 format is supported. In order to migrate it please
edit and save the file (Process, Screenflow or Procedure) using Studio
10.3, export the project and re-run the migration.
```

Enumerations - Enumerations specified within introspected jars must be specified using a specific Ant parameter when you run the BPM migration utility. For example:

```
ant -lib myenum.jar mgrate10-12...
```

Plumtree - if Plumtree components are used in a script, use the ant -lib parameter to provide the location of the distributed edk jar (fuego.edk-10.3.3.jar). For example:

```
ant -lib route/to/fuego.edk-10.3.3.jar migrate10-12...
```

Web services - ensure that called Web services (.wsdl files) within a Web service are reachable

Schemas - ensure that xml schemas (.xsd files) called within a schema file are reachable

2.5 BPM Migration Utility Options

Understand the command line options for the BPM migration utility.

The BPM migration utility command line has the following syntax:

```
ant migrate10-12 -DprojectLocation=<source> -Ddestination=<target> [-
Doption=-L<log_level>] [-DcustomXMLmappingDirs=<mappingdirs>]
```

where

*   **source** - specifies the location of the BPM 10gR3 project to migrate
*   **target** - the destination for the migrated project and report files

- **log_level** - detail level for logged messages. Messages from the specified level and higher are logged. If you do not specify a log level, the default level is -L2, which logs messages from L2, L3, L4 and L5.

  Log levels are:

  - -L1: Detailed info messages

  - -L2: Info messages

  - -L3: Warning messages

  - -L4: Severe issues messages

  - -L5: Fatal error message

- **mappingdirs** - a list of directories containing custom XML mappings to use when migrating Business Process Language scripts, separated by '**:**'. For example, dir1:dir2:dir*n*.

  An example command line invocation is:

  ```
  ant migrate10-12 -DprojectLocation=/user/migrationTests/input/samples/
  ExpenseReimbursement -Ddestination=/user/migrationTests/output/samples/
  export1
  ```

  To obtain more detailed output, use the ant -d and -v command line parameters and a higher log level:

  ```
  ant -d -v migrate10-12 -DprojectLocation=/user/migrationTests/input/
  samples/ExpenseReimbursement -Ddestination=/user/migrationTests/output/
  samples/export2 -Doption=-L1
  ```

  If you run the BPM migration utility with incorrect command-line arguments it prints a usage statement to the terminal. This statement describes what the usage statement should contain and the conventions you must use. For example:

  ```
  ant migrate10-12 -DprojectLocation=source -Ddestination=target (-
  Doption="options") (-DcustomXMLmappingDirs=customXmlDirs)
  ```

  For information using ant parameters to generate migration statistics, see Generating Migration Statistics for Components in Scripts.

2.6 Generating Migration Statistics for Components in Scripts

You can generate migration statistics for components used in scripts.

To generate migration statistics for components used in scripts, add the parameter `-Doracle.bpm.migration10to12.statistics=true` to the `ANT_OPTS` environment variable before running the BPM migration utility.

This parameter causes a CSV file named `<project_name>_mapperStats.csv` to be created in the BPM migration utility output path.

This file contains the following columns:

- Source

  Possible values: predefined component, user defined component

- Component

  Specifies the full name of the component

- Component Type

  Specifies the technology used

- Member Type

  Possible values: type, method, attribute (read/write)

- Quantity

  Indicates the relative usage of the component

2.7 Running the BPM Migration Utility

Use the BPM migration utility migrates projects from BPM 10gR3 version to BPM 12c.

See BPM Migration Utility Supported Platforms and Migration Environment Pre-Requisites for environment and pre-requisite information.

See Preparing for the BPM Migration Utility for steps to take prior to running the BPM migration utility.

To run the BPM migration utility

1. Access the command line (LINUX) or command prompt (Windows).

2. Navigate to `$JDEV_HOME/soa/plugins/jdeveloper/bpm/`.

3. Run the BPM migration utility: `ant migrate10-12 -DprojectLocation=<source> -Ddestination=<target> [-Doption=-L<log level>] [-DcustomXMLmappingDirs=<mappingdirs>`

   For example: `ant migrate10-12 -DprojectLocation=/home/myUser/ OracleBPMWorkspace/Sample103Project -Ddestination=/home/myUser/ migTests/FirstTest`

   For a detailed description of BPM migration utility parameters and options, see BPM Migration Utility Options.

   > **Note:**
   >
   > The destination directory must not exist. In order to avoid overwriting already migrated projects, if the destination directory exists, the BPM migration utility will not run.

4. Navigate to the output directory and verify that the following files exist:

   - **migrated project file**: a file named the same as the migrated project, with an .exp extension

   - **migration report**: two files named the same as the migrated project, one with an .xml extension and one with an .xsl extension

   After running the BPM migration utility, your processes are migrated to BPM 12c.

The BPM migration utility creates an exported BPM project that you can open with Oracle Business Process Management 12c.

For a detailed description of how the BPM migration utility migrates each of the elements of the BPM project, see the following:

- Understanding Business Process Language Migration
- Understanding Catalog Migration

For more information on how to view the migration report, see Accessing the BPM Migration Utility Output.

2.8 BPM Migration Utility Outputs

The BPM migration utility creates several files in the target directory when it executes.

If the project `myproject` is migrated to the target directory `migrateoutput`, the following files are created:

- **myproject.exp** - BPM 12c format migrated project file
- **myproject.xml** - migration report data
- **report.xsl** - migration report format schema
- **myproject_mapperStats.csv** - (optional) comma separated statistics file
- **myproject_projectStats.csv** - (optional) comma separated statistics file

See BPM Migration Utility Options and Generating Migration Statistics for Components in Scripts for details on BPM migration utility parameters.

2.9 Accessing the BPM Migration Utility Output

The BPM migration utility generates a migration report that describes the results of the migration process and indicates the migration status of each of the artifacts in the source BPM project. Use the report to know where you must make additional manual changes to complete the migration.

To view the migration report:

1. Navigate to the target directory specified with -Ddestination when you ran the BPM migration utility.

2. Open the file named the same as the migrated project, with an .exp extension in your web browser

For more information about the migration report, see Understanding the BPM Migration Utility Migration Report.

2.10 About the BPM OU Migration Script

Use the BPM Organizational Unit (OU) migration script to migrate artifacts from BPM 10gR3 to BPM 12c.

Artifacts migrated by the OU migration script include:

- Organizational Units

- Calendar Rules

- Holiday Rules

- Application Roles

- Parametric Roles

- Business Parameters

2.11 Preparing for the BPM OU Migration Script

Before running the BPM OU migration script, you must perform some preparatory tasks.

Working folder - create a working folder to use when running the script.

Copy directory.xml to working folder - navigate to `Oracle/Middleware/<Oracle_Home>/soa/plugins/migration/bpm` and copy `directory.xml`. Place a copy of this file in your working folder.

directory.xml configured - you must edit `directory.xml` to specify server and connection information as follows:

- edit the `<serverURL>t3://localhost:7001</serverURL>` line, replacing localhost with the IP address of your SOA/BPM 12c server, and the port number 7001 with the port number of your SOA/BPM 12c server.

- edit the `<admin>` attribute to specify the 10gR3 WebLogic server Admin account username

- edit the `<adminPass>` attribute to specify your 10gR3 WebLogic server Admin account password

Unzip organizationmigration file to working folder - navigate to `$BPM_HOME/ou-migration` and unzip `oracle.bpm.bpm-services.organizationmigration.jar` to your working folder.

BPM 10gR3 components running - the BPM 10g Directory Service database (Fuego Directory Interface (FDI)) and LDAP server must be running

BPM 12c running - the BPM 12c environment must be running

12c_Config.xml configured - you must edit `12c_Config.xml` (located in `Oracle/Middleware/<Oracle_Home>/soa/plugins/migration/bpm`) to specify server and connection information as follows:

- edit the `<serverURL>t3://localhost:7001</serverURL>` line, replacing localhost with the IP address of your SOA/BPM 12c server, and the port number 7001 with the port number of your SOA/BPM 12c server.

- edit the `<adminUserLogin>` attribute to specify the 12c WebLogic server Admin account username

- edit the `<adminUserPassword>` attribute to specify the 12c WebLogic server Admin account password

LDAP connectivity - configure your SOA/BPM 12c environment to be able to connect to the BPM 10gR3 LDAP server.

2.12 BPM OU Migration Script Options

Get an overview of the command line options for the BPM Organizational Unit (OU) migration script.

The BPM OU migration script command line has the following syntax:

```
ant -f ant-organization-migrator.xml -Dfmwhome=<path_to_fmwhome> -
Dalbpm.directory.file=<path_to_10g_directory_xml> -
Dconfig.file=<path_to_12c_configfile_xml> -
Dfuegojarpath=<path_to_10g_fuego> -Dentity.type=<entity> [-
Dversion=<target_jar_version>
```

where

- **path_to_fmwhome** - specifies the path to the BPM 12c fmwhome directory
- **path_to_10g_directory_xml** - specifies the location of the BPM 10gR3 `directory.xml` file. This can be a working folder where you have placed a copy of the file.
- **path_to_12c_configfile_xml** - specifies the location of the BPM 12c `12c_configfile.xml` file. This file must be edited to specify your BPM 12c server and login information. See Preparing for the BPM OU Migration Script.
- **path_to_10g_fuego** - the path to the BPM 10gR3 `fuegopapi.client-10.3.3.jar` file. This file is typically located under `Oracle/Middleware/<Oracle_Home>/soa/plugins/jdeveloper/bpm/libraries`.
- **entity** - specify the organizational elements to be migrated. Possible values are `calendar`, `orgunit`, `approle`, `parametricrole` and `all`. If this parameter is left blank, then the value `all` is used.
- **target_jar_version** - optional. Specify the target 12c version, for example `12.1.3`. The default if unspecified is `12.1.3`.

An example command line invocation to migrate only calendar rules is:

```
ant -f ant-organization-migrator.xml -Dfmwhome=/oracle/fmwhome -
Dalbpm.directory.file=/mytempworkingfolder -Dconfig.file=/oracle/product/
middleware/oracle_home/soa/plugins/migration/bpm -
Dfuegojarpath=OraBPMStudioHome/client/papi/lib -Dentity.type=calendar
```

If you run the BPM migration utility with incorrect command-line arguments it prints a usage statement to the terminal. This statement describes what the usage statement should contain and the conventions you must use.

2.13 Running the OU Migration Script

Use the Organizational Unit (OU) migration script to migrate organizational unit artifacts from BPM 10gR3 to BPM 12c.

See About the BPM OU Migration Script and BPM Migration Environment Pre-Requisites for environment and pre-requisite information.

See Preparing for the BPM OU Migration Script for steps to take prior to running the OU migration script.

To run the OU migration script:

1. Access the command line (LINUX) or command prompt (Windows).

2. Navigate to `$JDEV_HOME/soa/plugins/jdeveloper/bpm/`

3. Run the OU migration script:

   ```
   ant -f ant-organization-migrator.xml -Dfmwhome=<path_to_fmwhome> -
   Dalbpm.directory.file=<path_to_10g_directory_xml> -
   Dconfig.file=<path_to_12c_configfile_xml> -
   Dfuegojarpath=<path_to_10g_fuego> -Dentity.type=<entity> [-
   Dversion=<target_jar_version>
   ```

   For example:

   ```
   ant -f ant-organization-migrator.xml -Dfmwhome=/oracle/fmwhome -
   Dalbpm.directory.file=/mytempworkingfolder -Dconfig.file=/oracle/
   product/middleware/oracle_home/soa/plugins/migration/bpm -
   Dfuegojarpath=OraBPMStudioHome/client/papi/lib -Dentity.type=calendar
   ```

For a detailed description of OU migration script parameters and options, see BPM OU Migration Script Options

The OU migration script migrates and creates the organizational unit elements in the target BPM 12c environment.

An XML migration report named migr_report.xml is created at the location from which you invoked the OU migration script. This report indicates which elements are migrated successfully and which are not, along with the reason for the failure to migrate.

3

Using BPM Migration Utilities to Migrate Projects

Get a high level overview of how to use the output from the BPM migration utility and the OU migration script to migrate Oracle BPM 10gR3 project to Oracle BPM 12c. This chapter includes the following sections:

- Introduction to BPM Project Migration
- Running the Migration Utilities
- Understanding the BPM Migration Utility Migration Report
- Understanding Migration Statistics

3.1 Introduction to BPM Project Migration

The BPM migration utility helps you migrate an Oracle BPM 10gR3 project to an Oracle BPM 12c project.

The BPM migration utility creates an exported BPM project file and a migration report as well as mapper and project statistics `.csv` files. The migration report contains the details of the migration for each of the elements of the BPM project. If a problem occurs during migration, the BPM migration utility logs the details of the problem to the migration report.

Much of the migration is automatic, but some project elements require that you make modifications in the BPM 12c project created using the BPM migration utility in order to complete the migration. This guide contains detailed information explaining which project elements require manual migration work.

3.2 Running the Migration Utilities

BPM 12c includes two command line utilities - the BPM migration utility (to migrate projects) and the OU migration script (to migrate organizational unit information).

The BPM migration utility creates a BPM12c project export file that you can import into BPM 12c, and a migration report that you can use to troubleshoot problems and identify areas where additional manual steps are required to complete the migration.

> **✎ Note:**
>
> When you migrate projects, all your existing 11g projects are placed in the target location that you define in the BPM migration utility command line. Your existing 11g projects get a new permission `BPMPAMDEFAULTGROUP/None`. As the permission is `None`, the projects continue to remain private.
>
> However, when you create a new project in the target location, the project is shared with the migration group `BPMPAMDEFAULTGROUP` with `Editor`permission. Change the project permission from `BPMPAMDEFAULTGROUP/Editor` to `BPMPAMDEFAULTGROUP/None` to keep your new projects private in the target location.

The OU migration script migrates Organizational Units, Calendar and Holiday Rules, Roles, and Business Parameters.

These utilities also create reports which show you elements that could not be migrated and the reason for the failure. Manual intervention is required to fix the problem or re-create elements manually in the target system.

See About the BPM Migration Utilities for complete details on what the utilities do, what the prerequisites are for the utilities, and how to run them.

3.3 Understanding the BPM Migration Utility Migration Report

The BPM migration utility creates a migration report in the target directory.

The migration report has the same name as the migrated project, with an `.exp` extension. It contains the following information:

- **Date**

  The first line contains timestamp that indicates when the BPM migration utility was run.

- **Business Catalog Migration Messages**

  The initial messages describe the migration of each of the artifacts in the business catalog.

- **Process Design Migration Messages**

  Process design migration messages describe the migration of each of the artifacts in the process designs contained in the source BPM 10gR3 project.

Messages are tagged with the following categories:

- **INFO**

  Describes that an artifact was successfully migrated.

- **WARNING**

  Describes a limitation found when migrating an artifact.

- **SEVERE**

Describes specific features from an artifact that were not migrated.

- **FATAL**

  Describes problems that prevented the migration of an artifact. It can also mean that the migration process was not completed.

See BPM Migration Utility Options for more details.

3.4 Understanding Migration Statistics

The BPM migration utility generates the report by default, without requiring any command line switches or parameters.

The BPM migration utility generates a report that includes information about the migration status of BPM artifacts. This report provides information about all the types included in the original 10gR3 business catalog so that you are aware of types with more usages. Use this information to plan your migration, taking into account the BPM migration utility limitations.

The report contains the following information about BPM processes and the business catalog:

- Artifact Type
- Artifact Type Name
- Number of Occurrences in the 10gr3 Project

The migration utility saves this information in a `.csv` file. The file name is composed of the project name concatenated with the suffix `_projectStats`, and has the `.csv` extension.

The report contains the following columns:

- Group

  Indicates the group of components. For example: CATALOG

- Element:

  Indicates the specific component within a group. For example: SQL (part of the CATALOG group)

- Quantity

  Indicates the usages in the project.

When viewing the report, consider the following:

- Web service usage is counted by WSDL, not by port type
- Enumerations are divided between XML, and NO-XML.

  XML: imported from an XML schema

  NO-XML: created manually

- Presentations are counted individually
- XML types are counted individually

  In the example below, an unknown number of schemas define a total of 4 types.

Example Report

Group	Element	Quantity
Catalog Components	Business Object Exception	1
Catalog Components	Web Service	1
Catalog Components	XML Enumeration	1
Catalog Components	CUnit Suite	1
Catalog Components	Business Object Presentation	3
Catalog Components	BPM Object	3
Catalog Components	PUnit Suite	1
Catalog Components	Enumeration	1
Catalog Components	XML Type	4
Catalog Components	Module	2

Process Modeling

The following elements are included in the Process Modeling group:

*   All types of processes

    Includes: screenflows, processes, procedures, lanes
*   Activities

    Includes: both interactive and automatic activities implemented with methods, components and procedures
*   Transitions

    Includes: unconditional, due, due schedule, compensate, message based, dependency, any, business rule, exception and conditional transitions
*   Other

    Includes web resources, views, custom JSPs and presentations

4

Understanding Project and Process Migration

Get an overview of project artifact and BPM process component migration.
This chapter includes the following sections:

- Project Artifacts Migration
- BPM Activities Migration
- Migrating Transitions
- Exception Migration
- Argument and Argument Mapping Migration

4.1 Project Artifacts Migration

The BPM migration utility migrates BPM project artifacts using mapping.

The BPM migration utility migration mapping is described in the following table:

Table 4-1   Project Artifact Mapping

10gr3 Artifact	12c Artifact
Project Properties	Only configured languages migrated.
Languages	Migrated to the Languages preference in Project Preferences.
Current languages	Migrated to the Current Language preference in Project Preferences
Project Variables	Project data objects
Business Indicators	Business indicators
Process Variables	Process data objects
Process Properties	Migrates the Generate Events and Author properties.
Procedures	BPMN process
Documentation	Migrates i18n process documentation.
	Images, font type, font style (bold, italic, underline, etc) and font color are not migrated.
Organization	Migrates roles used in the BPMN processes.
Lanes	Lanes
Layout	Migration not supported. BPM 12c uses an automatic layout.
Simulation Models	Migration not supported.
Presentations	Not migrated

Table 4-1   (Cont.) Project Artifact Mapping

10gr3 Artifact	12c Artifact
Project Dependency	Project dependencies and base project properties are not migrated.
	The catalog and roles migration supports project dependency. To use this functionality, you must copy the base project to the same directory in which the dependent project is stored.

4.2 BPM Activities Migration

The BPM migration utility migrates the BPM process activities using the mapping.

The BPM migration utility migration mapping is described in the following sections:

- Events Migration
- Activities Migration
- Gateway Migration
- Migrating Other Process Artifacts

For more information on the limitations and known issues when migrating BPM activities, see BPM Activities Migration Known Issues.

4.2.1 Events Migration

The BPM migration utility uses the following mappings to migrate events from BPM 10gr3 processes to BPM 12c processes:

Table 4-2   BPM 10gr3 Activity to BPM 12c Events Mapping

10gr3 Artifact	12c Artifact
Begin	Start message catch event.If the begin event contains multiple argument sets, they are migrated to multiple BPMN message start events, one for each argument set. For more information, see Argument and Argument Mapping Migration.
End	End event.
	For information about argument and argument mapping migration, see Argument and Argument Mapping Migration.
Boundary events	Only activities support boundary events. The boundary events in artifacts that are migrated as BPMN events, are not migrated.
	See Migrating Transitions.
Message wait that allows interruptions	Event subprocess with a message start event.
	In some cases, notification wait events that allow interruptions may not have a mapping. If the notification wait has more than one outbound transition it is mapped to multiple event subprocesses.
	See Argument and Argument Mapping Migration.

Table 4-2   (Cont.) BPM 10gr3 Activity to BPM 12c Events Mapping

10gr3 Artifact	12c Artifact
Message wait that does not allow interruptions	Catch message event.
	In some cases, notification wait events that do not allow interruptions may not have a mapping. If the notification wait has multiple argument sets, the utility migration creates a catch event for each of the argument sets.
	See Argument and Argument Mapping Migration.
Send message	Intermediate message throw event
Timer	Intermediate timer catch event
Compensate	Migrated as an abstract activity with no implementation. You must decide how to implement it, or remove it.
Runtime Settings Advanced Settings Custom Images	Not migrated.

4.2.2 Activities Migration

The BPM migration utility uses the following mappings to migrate activities from BPM 10gr3 processes to BPM 12c processes:

Table 4-3   BPM 10gr3 Activity to BPM 12c Activities Mapping

10gr3 Artifact	12c Artifact
Automatic with no implementation	Abstract activity
Automatic implemented with a component	Service task
Automatic implemented with a method	Script task
Automatic implemented with a procedure	Service task with a process call implementation. Arguments and arguments mappings are migrated.
Automatic with a loop	Loop condition, evaluation order and condition are not migrated.
Interactive with no implementation	User task implemented with an empty human task. You must implement the human task to complete the migration.
Interactive that invokes a screenflow	Service task implemented with a process call. The service task does not support role definition, the role assignment is in the process, so the original role may change after migration.
Interactive implemented with: • Component • Method • External • Input • Display	User task implemented with an empty human task. You must implement the human task to complete the migration.
Interactive with Optional Tasks	Not migrated.
Interactive with PUnit Tasks	Not migrated.

Table 4-3   (Cont.) BPM 10gr3 Activity to BPM 12c Activities Mapping

10gr3 Artifact	12c Artifact
Decision	User task implemented with an empty human task.
	You must implement the human task to complete the migration.
Interactive Component Call	User task implemented with an empty human task.
	You must implement the human task to complete the migration.
Interactive Component Call implemented with: • Component • Input • Display • BPM Object Interactive Call	User task implemented with an empty human task. You must implement the human task to complete the migration.
Group	Embedded subprocess
	There is no mapping for BPM 10gr3 transactional property.
Subflow	Abstract activity with no implementation. The activity names have the suffix "was screenflow" to identification. You must decide how to implement them.
	For information about argument and argument mapping migration, see Argument and Argument Mapping Migration.
Screenflow	BPMN process
Subscreenflow	Service task with process call implementation. Dynamic process invocation is not supported
Process Creation	Send task with no implementation. Dynamic process invocation is not supported.
	Runtime settings are not migrated.
Termination Wait	Receive task with no implementation.
	Runtime settings are not migrated.
Grab	Does not require migration. The alter flow feature is available from Process Workspace for any BPM process.
	Migrated to an abstract activity with no implementation. This activity is not connected to the process flow.
Grab All	Does not require migration. The alter flow feature is available from Process Workspace for any BPM process.
Global Creation	Start event with no implementation, followed by a default sequence flow to an initiator user task with no implementation defined, followed by a default sequence flow.
	If the global creation activity contains a script, a script task with the script is added after the last default sequence flow.
	If the global creation activity invokes a screenflow, a service task that calls the migrated screenflow is added after the last default sequence flow.
	The autocomplete and runtime settings are not migrated.
Global automatic	Start event followed by a script task and an end event of type none. Runtime settings are not migrated.
Global Automatic | Polling by Interval	Timer start event.
Global Automatic - Triggered by Events	Message start event.

Table 4-3   (Cont.) BPM 10gr3 Activity to BPM 12c Activities Mapping

10gr3 Artifact	12c Artifact
Global Automatic - Automatic Schedule	Message start event.
Global automatic - JMS Listener	Message start event.
Global automatic - JMS/JMX	Message start event.
Global Interactive	Not migrated.
Runtime Settings Advanced Settings Custom Images	Not migrated.

4.2.3 Gateway Migration

The BPM migration utility uses the following mappings to migrate gateways from BPM 10gr3 processes to BPM 12c processes:

Table 4-4   BPM 10gr3 Activity to BPM 12c Gateway Mapping

10gr3 Artifact	12c Artifact
Conditional Gateway	Exclusive Gateway
Split	Parallel Gateway. The Generate Copies property is not migrated.
Or Split	Inclusive Gateway.
Multiple Gateway	Multi-instance subprocess. You must complete the migration manually. For more information, see Understanding Multiple Gateway Migration.

4.2.3.1 Understanding Multiple Gateway Migration

The BPM migration utility migrates multiple gateways to a multi-instance subprocess with the following characteristics:

- The type of the multi-instance subprocess is parallel.
- The loop characteristics tab has Loop cardinality selected implemented with a simple expression. When you check the migrated process a warning appears asking you to complete the multi-instance implementation expression.

With multiple gateways, you must perform additional manual steps in order to complete the migration. For more information, see Completing Multiple Gateway Migration.

4.2.3.2 Completing Multiple Gateway Migration

After running the BPM migration utility, you must manually complete the migration of multiple gateways by implementing or adapting the following:

- Loop Characteristics
- Creation Type

- Expressions and Scripts

To replicate BPM 10gr3 functionality using a multi-instance subprocess:

1. Set the required value or expression.

2. If the Split-N was configured with the concurrent amount of copies set t1', then configure the multi-instance subprocess as sequential instead of parallel.

3. Create the variables in the subprocess and map them to the process variables.

4. Edit the script code to keep the mapping behavior.

5. If the multiple gateway uses data objects, then migrate all the copy instances.

6. If the BPM 10gr3 process contained split/join activities implemented with scripts, then add a script task as the first and last flow objects in the BPM 12c process. Then migrate the script implementation. The migration utility migrates the script implementation as a comment so that you can write the new script based on the BPM 10gr3 implementation.

4.2.4 Migrating Other Process Artifacts

The migration utility uses the following mappings to migrate process artifacts from BPM 10gr3 processes to BPM 12c processes:

Table 4-5   BPM 10gr3 Activity to BPM 12c Process Artifacts Mapping

10gr3 Artifact	12c Artifact
Lane	BPMN Lanes
Note	Text annotation. The location of the note changes because of the different process structure. You must relocate the note to complete the migration.
Measurement mark	Not migrated.
Documentation	Process notes documentation and process notes use case documentation are migrated to the same artifacts 12c. Images and font type, font style and font color are not supported.
Folders	Folders containing processes or screenflows are migrated to folders in the 12c BPM project.
Business Rule	Not migrated.
Process Interface	Not migrated.

4.3 Migrating Transitions

The BPM migration utility migrates BPM 10gr3 transitions to BPM 12c sequence flows using mapping.

The BPM migration utility migrates BPM 10gr3 transitions to BPM 12c sequence flows using the following mapping:

Table 4-6   BPM 10gr3 Transition to BPM 12c Sequence Flow Mapping

10gr3 Artifact	12c Artifact
Unconditional transition	Default sequence flow.
Conditional transition	Migrated as a BPMN Gateway and a Conditional Sequence Flow. The condition is migrated if the expression is supported in 12c simple expressions. Otherwise the condition appears as a comment in the simple expression.
Due transition	Sequence flow with an interrupting boundary timer event. For more information, see Understanding Due Transition Migration.
Message transition	Default sequence flow.
Exception transition	Catch error boundary event with a default sequence flow.
Business rule transition	Conditional sequence flow with an empty condition. For more information, see Understanding Business Rule Transition Migration.
Compensate transition	Not migrated.

4.3.1 Understanding Due Transition Migration

The BPM migration utility migrates due transitions to a catch timer boundary event followed by a default sequence flow.

If the due transition contains an interval expression or a constant expression, this expression must be compatible with the supported expressions in BPM 12c. If the expression is not compatible, a message error appears and you must manually correct it to complete the migration.

Only interval expressions and constant expression are migrated.

4.3.2 Understanding Business Rule Transition Migration

The BPM migration utility migrates each business rule transition to a conditional sequence flow with an empty condition.

The BPM 10gr3 business rule code appears as a comment in the empty condition to be used to manually code the new business rule. For more information, see Completing Business Rule Transition Migration.

4.3.3 Completing Business Rule Transition Migration

Business rule transitions are migrated as a conditional sequence flow with an empty condition. The empty condition contains the commented code from BPM 10gr3 to be used as a guide for the new implementation.

After migrating a business transition with the migration utility you must manually code the business rule and configure the condition in the conditional sequence flow to use it.

4.4 Exception Migration

The BPM migration utility migrates exception handling using mapping.

The BPM migration utility mapping is described in the following table:

Table 4-7   BPM 10gr3 Transition to BPM 12c Sequence Flow Mapping

10gr3 Artifact	12c Artifact
Exception Transition	Error catch boundary event with a default sequence flow
Process Exception	BPMN Event Subprocess. Each exception transition is migrated to a different event subprocess.
Compensate	Sequence flow with an interrupting boundary timer event.
Skip and back action	Skip and back actions are supported using the predefined variable action. For more information, see "Configuring Catch Events to Recover from an Exception" in *Developing Business Processes with Oracle Business Process Management Studio*.

4.5 Argument and Argument Mapping Migration

The BPM migration utility migrates arguments and argument mappings:

The BPM migration utility migrates arguments and argument mappings for the following artifacts:

- Begin
- End
- Message Wait
- Interactive with Screenflow Call Implementation
- Subflow

BPM 10gr3 arguments are migrated to BPM 12c arguments. BPM 10gr3 argument mappings are migrated to BPM 12c data associations.

The migration utility migrates arguments only when the variables, expressions and data types used for those arguments are supported or have an equivalence in BPM 12c.

For interactive activities implemented with a screenflow call, the migration utility uses the input/output arguments of the main task.

Multiple argument sets are not supported in BPMN. For some particular cases multiple argument sets are mapped to a different structure that models the same situation. For more information, see begin activities and notification wait activities that allow interruption in BPM Activities Migration.

5

Understanding Catalog Migration

Understand the migration of the business catalog, the predefined catalog, and introspected components.
This chapter includes the following sections:

- Business Catalog Migration
- BPM Objects Migration
- Predefined Catalog Migration
- Introspected Components Migration
- Project Dependency Migration

5.1 Business Catalog Migration

The BPM migration utility migrates the components in the business catalog using mapping.

The BPM migration utility mapping is described in the following table:

Table 5-1   Business Catalog Mapping

10gr3 Artifact	12c Artifact
Module	Module
BPM object	Business object For more information, see BPM Objects Migration.
Business exception	Business exception For more information, see Business Exceptions Migration.
Introspected object	See Introspected Components Migration
Test object	Not migrated.
Enumeration	Enumeration

The migration of the following elements is limited:

- Scripts that use the Plumtree component
- Projects using enumerations introspected from `JAR` files
- Project dependencies

For more information on these limitations, please see Limitations.

5.1.1 Business Exceptions Migration

The BPM migration utility migrates exceptions in the same way it migrates business objects. However, BPM 12c exceptions allow only one attribute named `errorInfo`, so

the BPM migration utility creates an exception with the name from the BPM 10gR3 exception and with only one attribute. It also creates a business object that contains all the attributes from the BPM 10gR3 exception and names it by concatenating the name of the original exception and the suffix `ExceptionPayload`. The `errorInfo` attribute in the migrated BPM 12c exception uses this business object as its type.

5.2 BPM Objects Migration

Understand how BPM migration utility migrates BPM objects.

The BPM migration utility migrates BPM objects in the following way:

Table 5-2   BPM Object Migration

BPM Object Element	12c Migration
Name	Migrated
Attributes	Partially migrated. For more information, see Table 5-3.
Methods	Migrated to BPM Scripting.
	For more information, see Understanding Business Process Language Migration .
Groups	Not migrated
Presentations	Partially migrated.
	For more information, see BPM Object Presentations Migration.
Inheritance	Migrated
	Only the migration of inheritance between Business Objects is supported. For more information see Known Issues.
Comparable property	Not migrated
Abstract property	Not migrated

5.2.1 BPM Object Attributes Migration

The BPM migration utility migrates BPM object attributes in the following way:

Table 5-3   BPM Object Attributes Migration

10gr3 Attribute Element	12c Migration
Name	Migrated.
Documentation	Migrated.
	Images, font type, font style (bold, italic, underline, etc) and font color are not migrated.
Type	Migrated for most types. For a detailed description, see Table 5-4.
Overridden access	Not migrated.
Valid values	Not migrated.
Primary key	Not migrated.
Check expression	Not migrated.

Table 5-3   (Cont.) BPM Object Attributes Migration

10gr3 Attribute Element	12c Migration
Required expression	Not migrated.

The BPM migration utility migrates the BPM object attribute types in the following way:

Table 5-4   Attribute Type Migration

Type	12c Migration
Simple type	Migrated except for Any, which are migrated to an invalid type. In this case you must choose a valid type manually after you run the BPM migration utility.
BPM object type	Migrated
Java/Predefined catalog type	Not migrated
RegExp	Not migrated. Attributes are created with the type invalid type. You must complete the migration manually.
Array	Simple and BPM object type arrays are migrated.
	Business catalog predefined type arrays are migrated as arrays of type Any.
AssociativeArray	Not migrated. Attributes are created with the type invalid type. You must complete the migration manually.
Iterator	Not migrated. Attributes are created with the type invalid type. You must complete the migration manually.
Virtual	Not migrated. Attributes are created as regular attributes.

5.2.2 BPM Object Presentations Migration

Each BPM 10gR3 Object Presentation is migrated to a BPM 12c Human Task.

The payload of the Human Task contains the attributes that appear in the BPM 10gR3 BPM Object Presentation. Attribute types that are not supported for migration, such as BPM Groups, are not included in the Human Task payload.

The BPM migration utility generates a Task Form Wizard context file for each Human Task. This file includes layout and reference information. Use the Task Form Wizard to create a form so that you can use the layout and reference information used to design the BPM Object Presentation.

The BPM migration utility also migrates the 10gR3 calls to BPM Object Presentations to uses of the migrated Human Tasks.

The result of migrating a BPM Object Presentation is:

- a Human Task

  The Human Task is named by concatenating the following:
  BusinessObjectFullName + "_" + PresentationName + ".xml"

- a FormWizardInfo file

This file contains the information about the attributes used in the form, and the layout of the form.

It is stored in the HumanTask directory using a name that results of concatenating the following: "TaskFormUiCtx_" + BusinessObjectName + "_" + PresentationName + ".xml"

- a payload file

The name and location of this file is defined by the Human Task API.

To use the Task Form Wizard with the FormWizardInfo files, open the Human Task editor and select the Auto-Generate Task Form option. This option is available if a FormWizardInfo file that matches the name of the Human Task exists.

5.3 Predefined Catalog Migration

BPM 12c does not support a predefined catalog. When possible, the migration utility migrates the usages of methods from predefined components to their equivalent in BPM 12c.

Table 5-5 lists all the component groups (Fuego blocks) in the predefined catalog and, where possible, describes how to obtain a similar behavior in BPM 12c.

Table 5-5   Predefined Catalog Migration

Component	12c Migration	Details
Plumtree	Not available	Not Available
Java	Available from BPM scripting	Usages in script tasks and business methods are migrated to similar Java method invocations that are available to use from Groovy. Only methods that receive and return simple types or business objects are migrated.
		Migration is not available for methods that return Java types.
		There is no migration for the component `java.net.InetAddress.` If the methods you are migrating contain calls to methods from this component, the migration fails with a compilation error. The suggested workaround is to comment the lines of code in the 10gR3 project that use this component and after you complete the migration, uncomment the code and manually migrate them.
Fuego – XML – XMLObject	Partial	XML definitions are migrated to XSD files.
		Native usages (declaration, variables assignment, and so on) are migrated to BPM Scripts.
		You can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – XML – DynamicXML	Not available	Use native XPATH support for transformations and conditions.

Table 5-5　(Cont.) Predefined Catalog Migration

Component	12c Migration	Details
Fuego – Rules	Not available	For complex business rules, use Oracle Business Rules.
		For simple business rules, you can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – Social	Not available	Use UMS adapter.
		You can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – Test	Not available	Use the debugger in Oracle BPM Studio. For more information, see *Debugging a BPM Project* in *Developing Business Processes with Oracle Business Process Management Studio*.
Fuego – RQL	Not available	You can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – JMX	Not available	You can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – UI	Not available	-
Fuego – Web Services	Partial	Web services definition are migrated to WSDL files. Oracle BPM Suite and Oracle SOA Suite provide native support for the invocation of web services.
Fuego – PAPI	Manual migration	-
Fuego – FDI	Not available	Not available
Fuego – Util	Not available	You can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – Lib	Not available	Part of this information is supported through predefined variables in the BPM 12c. Also consider using BPM scripting and migration mapper framework to create custom extensions.
Fuego – Lang	Not available	-
Fuego – Net	Not available	-
Fuego – NetX	Not available	-
Fuego – Msg	Not available	-
Fuego – Auth	Not available	-

ORACLE®

Table 5-5   (Cont.) Predefined Catalog Migration

Component	12c Migration	Details
Fuego – Bis	Not available	-
Fuego – Chart	Not available	For charting and reporting functionality, use Oracle BAM.
Fuego – Corba	Not available	You can extend the scripting mapping framework and script custom libraries to add custom components that replace this functionality. Then you can migrate the usages of the fuego block components to the custom components you created.
Fuego – EJB	Not available	-
Fuego – IO	Not available	-
Fuego – SQL	Not available	Use database adapters. .
Fuego – COM	Not available	For reporting, use Oracle Business Intelligence Publisher to generate Microsoft Excel and Microsoft Word reports.
Fuego – Dynamic	Not available	-

5.4 Introspected Components Migration

BPM 12c does not support component introspection, but it does support integration with external components using either Oracle BPM Suite or Oracle SOA Suite features.

Table 5-6 shows how introspected components from BPM 10gR3 are migrated to BPM 12c.

Table 5-6   Introspected Components Migration

Technology	12c Migration
Business Object	Automatic migration.
COM	Not available.
.NET	Not available.
SQL	Not available. You can use the stack level database adapter in the BPM 12c.
SQL Query	Not available.
EJB	Not available in BPM 12c. You can add EJBs using Oracle SOA Suite.
Enumeration	Automatic migration.
XML Schema	The migration utility uses introspected schema metadata to create XSD files in the migrated project. For more details, see Introspected XML Schemas Migration.
JNDI	Not available.
JMX	Not available.

Table 5-6   (Cont.) Introspected Components Migration

Technology	12c Migration
JPD	Not available. You can use web services integration to reproduce this functionality.
Java	You must migrate libraries manually. Invoke the methods and classes they contain from BPM scripts. For more information, see "Writing BPM Scripts" in Developing Business Processes with Oracle Business Process Management Studio.
Oracle Service Bus	The migration utility creates the WSDL definition files.
SAP	Not available.
Corba	Not available.
Web Services	WSDLs are migrated with the same limitations described for XML schemas, see Introspected XML Schemas Migration. The ability to invoke web services from BPM scripts is not available in BPM 12c, so there is no migration for the Business Process Language code that invokes the web services. You can manually integrate web services using services tasks.

5.4.1 Introspected XML Schemas Migration

The BPM migration utility migrates XML schemas in the following way:

- XML Schemas are copied to the `SOA/Schemas` directory in the BPM Project.
- WSDL Definitions are copied to the `SOA/WSDLs` directory in the BPM Project.

Import Tags

Instrospected XML schemas or a WSDL definitions might contain import tags, resulting in additional files that correspond to the imported schemas.

The introspected metadata does not store the original file names. The BPM migration utility names these files using the name of the container module of the introspected component in 10gR3, followed by correlative numbers in case there is more than one file.

If this happens, the migration utility adds a message to the log, warning you to rename the files. When you rename the files, ensure you also rename the references to those files.

Namespace Support

The BPM 10gR3 supports schemas without a namespace definition, but the BPM 12c does not. If you must migrate schemas that do not contain a namespace definition do one of the following:

- Add a namespace to the original schemas and re-introspect them before running the migration process.
- Copy the files manually to the migrated project. Ensure that the import tags point to the right locations.

Also, modify any embedded schemas contained in the WSDL definitions.

Protected Namespaces

The following namespace is used internally in the 12c version, if your 10gr3 project uses this namespace make sure you change it before running the migration: http://xmlns.oracle.com/bpel/workflow/

File Name Validation

BPM 12c requires valid XML schemas and WSDLs definitions. If the BPM migration utility finds invalid names during migration, it logs a warning message and continues with the migration. The invalid names might come from invalid constructs that were ignored during 10gR3 introspection.

You must rename these files in BPM 12c to avoid compilation issues.

5.5 Project Dependency Migration

The BPM migration utility does not migrate the project dependency property.

If a project has a dependency with another project, the BPM migration utility migrates the business catalog components from the source project to the business catalog of the migrated 12c BPM project.

The migrated catalog contains the components from both the base project and the dependant project from the original BPM 10gR3 project.

6

Understanding Business Process Language Migration

Understand the migration of BPM 10gR3 Business Process Language scripts. This chapter includes the following sections:

- Introduction to Business Process Language Code Migration
- Business Process Language Control Flow Statements Migration
- Business Process Language Basic Types Migration

6.1 Introduction to Business Process Language Code Migration

BPM 12c supports standard Groovy scripts.

For more information, see chapter "Writing BPM Scripts" in *Developing Business Processes with Oracle Business Process Management Studio*.

The following elements have migration limitations:

- Scripts using the Plumtree component
- Project using enumerations introspected from JAR files
- Project Dependency

For more information on these limitations, please see Limitations.

Table 6-1 shows how different BPM 10gR3 Business Process Language components are migrated to BPM 12c scripting.

Table 6-1   Business Process Language Code Migration

Functionality	12c Migration
Business Process Language	BPM Scripting support in 12c is based on Groovy
Business Objects	Partially migrated. Attributes of type RegExp, Attributes : valid values, and Attributes : check expression are not migrated and require scripting to be implemented.
Skins (PBL, Java, VBasic)	Migrated to Groovy script functionality
Scripting Language Exception Handling	Migrated on a best efforts basis. See Business Process Language Control Flow Statements Migration
Simple Types	Migrated. See Business Process Language Basic Types Migration
Complex Types	Java types, business objects and XML schemas are migrated. You must migrate dependent Java libraries manually. Schemas / XSDs are inferred to the extent used in the project and migrated.

Table 6-1   (Cont.) Business Process Language Code Migration

Functionality	12c Migration
Activities (Automatic and Method)	Migrated as scripting tasks.
Multiple Gateways with scripting code	Migrated as scripting tasks.
Language Structure	See Business Process Language Control Flow Statements Migration
Embedded SQL	Not available
Web Services Invocations	Not available
Predefined Variables	BPM 12c supports a new set of predefined variables. For more information, see "Handling Information in Your Process Design" in Developing Business Processes with Oracle Business Process Management Studio.

6.2 Business Process Language Control Flow Statements Migration

BPM Scripting is based on standard Groovy and uses its control flow statements.

The BPM migration utility uses the control flow statement mappings shown in Table 6-2 to migrate Business Process Language scripts from BPM 10gR3 to Groovy scripts in BPM 12c.

Table 6-2   Control Flow Statements Migration

Business Process Language	Groovy
case/when/else	switch/case/default
name do end	name: {}
for/in/do	for(){}
for each/in	for(){}
for each in where	for(){if(){}}
if/then/elseif/else	if(){}elseif{}else{}
while	while(){}
is	instanceof
label:statement	label:statement
(mapped ref example) intervals[].hours	intervals.collect() {XmlDuration it -> it.hours}
do/on excep/on exit	try{}catch(){}finally{}

6.3 Business Process Language Basic Types Migration

Business Process Language basic types in the BPM 10gR3 scripts are migrated to standard Java types.

Table 6-3 shows the mapping used to migrate Business Process Language basic types.

Table 6-3   Business Language Basic Types Migration

Business Process Language	BPM Scripting
Bool	java.lang.Boolean
Fuego.Internal.Bool	java.lang.Boolean
Fuego.Internal.Int	java.lang.Integer
Fuego.Internal.Real(32)	Float
Fuego.Internal.Real(64)	java.lang.Double
Fuego.Internal.Real	java.lang.Double
Fuego.Internal.RegExp	java.util.regex.Pattern
Fuego.Internal.Str	java.lang.String
Fuego.Internal.Interval	com.oracle.scripting.lib.xml.datatype.XmlDuration
Fuego.Internal.Time	com.oracle.scripting.lib.xml.datatype.XmlCalendar
Interval	com.oracle.scripting.lib.xml.datatype.XmlDuration
Int	java.lang.Integer
Fuego.Internal.Logger	com.oracle.scripting.logger.ScriptLogger
Real(32)	java.lang.Float
Real(64)	java.lang.Double
Real	java.lang.Double
String	java.lang.String
Time	com.oracle.scripting.lib.xml.datatype.XmlCalendar
Type[] (indexed array)	java.util.List<Type>
TypeA[TypeB] (associative array)	java.util.Map<TypeB, TypeA>
TypeA[ordered TypeB] (sorted associative array)	java.util.SortedMap<TypeB, TypeA>
Any (only for scripting)	java.lang.Object

6.4 Business Process Language Post-Migration Activities

After migrating Business Process Languages elements, you must perform a few additional activities.

Actions to take after migrating Business Process Language elements include:

- If there are Web service invocations in your scripts, use the Web Services adapter
- As an alternative to XSLT transformations, use scripting where complex transformations are needed
- Map complex data types as plain Java objects (POJOs)

7
Understanding BPM API Migration

Learn how to migrate custom clients that use BPM APIs.
This chapter includes the following sections:

- API Migration

7.1 API Migration

Table 7-1 contains a description of the different APIs supported in BPM 10gR3 and details on how to migrate them to the BPM 12c version.

For examples on PAPI-to-BPM12c migration, see the following resources:

- Oracle Technology Network (OTN): http://www.oracle.com/technetwork/index.html
- Oracle SOA Suite Samples and Tutorials

Table 7-1   API Migration

API	Description	MIgration
FDI	Enabled access to the organization information repository directly. This API was not officially supported in BPM 10gR3. Only access through the fuego.fdi block was officially supported. However some customers were given access to this internal API.	Not available.
PAPI	Enabled the interaction with the 10gr3 BPM Engine from a custom Java application.	Migrate the code that uses this API manually.
PAPI-WS	Enabled the interaction with the 10gr3 BPM Engine from a custom web service.	Migrate the code that uses this API manually.
WAPI	Provided a framework to run functions similar to the ones provided by the PAPI API using HTTP requests.	Migrate the code that uses this API manually. Most BPM services provide a web service interface.
Oracle BPM 10.3 Java Interfaces	Java classes and interfaces used to extend BPM functionality.	Not Available

8

Limitations and Known Issues

Know the limitations and issues for migrating BPM 10gR3 projects to BPM 12c projects.
This chapter includes the following sections:

- Limitations
- Known Issues

8.1 Limitations

The following are not migrated when you migrate BPM 10gR3 and BPM 12c.

- Project preferences

  Only languages and current language preferences are migrated.

- Boundary Events attached to Events

  BPMN supports only Boundary Events for Activities. It does not support Boundary Events for Events.

- BPM Activities Runtime Settings
- BPM Activities Advanced Settings
- BPM Activities Custom Images
- Measurement Marks
- Optional Tasks
- PUnit Tasks
- Process Interfaces
- Compensate Transitions
- Business Rules
- Predefined Variables
- Correlations
- Unused Roles
- Groups
- Participants
- Custom JSPs
- Fuego Tag Library
- Layouts
- Views
- Simulations

- WorkSpace Customization

- FDI API

- The usage of some of the component groups from the predefined catalog

  See Predefined Catalog Migration.

- Some introspected components

  See Introspected Components Migration

8.2 Known Issues

Issues that may occur when migrating a project from BPM 10gR3 to BPM 12c.

- Project Dependency

  The project dependency property and the base project properties are not migrated. However, BPM 12c supports dependencies for catalogs and roles. To use dependencies you must store the base project and the dependent project in the same directory. For example, if the path of the base project is `/home/user/baseproject`, then the path of the dependent project must be `/home/user/dependentproject`.

- Parametric Roles

  Parametric roles are migrated as regular roles.

- Business Parameters

  Values specific to an Organization Unit are not migrated, only default values are migrated.

- Connectors

  The 12c version does not support connectors. Connectors are replaced with default sequence flows.

- Documentation

  Images, font type, font style (bold, italic, underline, etc) and font color are not migrated.

- Presentations

  Migrated as empty human tasks. You must implement these human tasks manually to complete the migration.

- Custom JSPs

  Custom JSPs are not migrated. You can move custom JSPs to an ADF project, but additional work is required to re-use them properly with ADF Task Flows. Note the Fuego Tag Library cannot be used in BPM 12c. You must map data using using Human Tasks.

- Business Object Inheritance

  The complex attributes of Business Objects are not migrated. Business Objects that extend from catalogued complex objects are not migrated. Inheritance between Business Objects and business exceptions are not migrated.

- `java.net.InetAddress` method calls

There is no migration for the component `java.net.InetAddress`. If the methods you are migrating contain calls to methods from this component, the migration fails with a compilation error. The suggested workaround is to comment the lines of code in the BPM 10gR3 project that use this component. After you complete the migration, uncomment the code and manually migrate the calls.

- Scripts that use the Plumtree component

  If the BPM 10gr3 project contains a script that uses the Plumtree component, then you must invoke the migration utility ant task using the `-lib` parameter and specify the location of the distributed edk jar (`fuego.edk-10.3.3.jar`).

  For example: `ant -lib route/to/fuego.edk-10.3.3.jar migrate10-12 -DprojectLocation=/home/myUser/OracleBPMWorkspace/Sample103Project -Ddestination=/home/myUser/migTests/FirstTest`

- Project using enumerations introspected from JAR files

  In BPM 10.3 you can add a jar file (external resource) and then catalog its inner types as Java. One of the possible types is the `Enum` type.

  If the BPM 10gR3 project contains an enumeration that was added to the business catalog using a jar file, then you must provide the jar file or the directory that contains the jar files, to the migration utility using the `-lib` parameter with the corresponding path.

  For Example:

  ```
  ant -lib myenum.jar migrate10-12 -DprojectLocation=/home/myUser/
  OracleBPMWorkspace/Sample103Project -Ddestination=/home/myUser/
  migTests/FirstTest
  ```

- Missing Class in Migration Report

  If the migration report indicates that a custom class is missing, you must provide the path to the custom class to the migration utility using the `-lib` parameter with the corresponding class.

- Project dependency

  To migrate a project that depends on another project you must ensure that both projects are stored in the same directory. Otherwise, the components from the base project are not migrated.

- Path issues: missing files or directories.

  Migrating a project from BPM PS6 to BPM 12c might cause an error from the Rules SDK similar to the following:

  ```
  oracle.xml.parser.schema.XSDException: /myjdev/mywork/
  myproject/../../../xsd/BusinessCaseRuleFact.xsd (No such file or
  directory) at oracle.xml.parser.schema.XSDBuilder.build(XSDBuilder.java:
  621)
  ```

  This occurs if the stored schema path is expressed relative to the rules directory instead of the project directory.

  To resolve this error, edit the problematic .rules file and replace all occurrences of '/../../../xsd/BusinessCaseRuleF act.xsd' with 'xsd/BusinessCaseRuleFact.xsd', save the .rules file, and then open the BPM PS6 project in BPM Studio 12c to migrate it.

8.2.1 BPM Activities Migration Known Issues

The following known issues apply when migrating these BPM 10gR3 activities to BPM 12c.

Timer Events

Runtime settings are not migrated.

Compensate

Migrated as abstract activities with no implementation. You must either remove them or decide how to manually implement them.

Automatic Activities

• Activities implemented with Components are migrated as BPMN service tasks with no implementation. You must implement the service tasks manually to complete the migration.

• Loop Conditions are not migrated.

• Evaluation Orders are not migrated.

• Conditions are not migrated.

Interactive Activities

• Screenflow Calls are migrated as BPMN service tasks with a process call implementation. Service tasks do not support a role definition, so the role assignment is in the called process itself. The original role could change.

• PUnit Tasks are not migrated.

• Optional Tasks are not migrated.

The following implementations of an interactive activity are migrated as BPMN user tasks with empty human tasks. You must decide how to implement them manually to complete the migration process:

• With No Implementation

• Component

• Method

• External

• Input

• Display

• Decision

Decision Activity

Decision activities are migrated as BPMN user tasks with empty human tasks. You must decide how to implement them manually to complete the migration process.

Interactive Component Call

Interactive Component Calls are migrated as BPMN user tasks with empty human tasks. You must decide how to implement them manually to complete the migration process:

- Components
- Inputs
- Displays
- BPM Object Interactive Calls

Subflow

Dynamic process invocation is not supported.

SubScreenflow

Migrated as an abstract activity with no implementation. The activity label has the suffix "was subscreenflow". You must decide how to implement this.

Process Creation

Dynamic process invocation is not supported.

Runtime settings are not migrated.

Termination Wait

Runtime settings are not migrated.

Global Creation

Auto-complete and other runtime settings are not migrated.

Global Automatic

Runtime settings are not migrated.