

DOAG BI
München
3. April 2014

Migrating OWB to ODI

BASEL BERN LAUSANNE ZÜRICH DÜSSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MÜNCHEN STUTTGART WIEN

1

2014 © Trivadis

DOAG BI - Migrating OWB to ODI
03.04.2014

trivadis
makes IT easier. ■ ■ ■

Migrating OWB to ODI

1. Introduction

2. Why a migration?

3. The Migration Utility

4. Prerequisites

5. Configuration

6. Execution and limitations

7. Planning

8. Conclusion

Introduction

- Martin de Gooijer
- 20 Years Oracle experience
 - 11 Years Oracle Netherlands
 - Since 2005 with Trivadis
- Trainer for OWB Course
- Participant ODI 12c and OWB Migration Utility Beta Program



Why a migration: Statement of Direction from Oracle (partly)

No major enhancements are planned for Oracle Warehouse Builder beyond the OWB 11.2 release. OWB 11.2 continues to be available and supported by Oracle, and patches and bug fixes will continue to be offered at regular intervals. Oracle will continue to support OWB 11.2 for the full lifetime of Oracle Database 11g in accordance with Oracle's Lifetime Support Policies for Oracle Database releases. Future database releases beyond Oracle Database 12c Release 1 will not be certified with OWB 11.2.

Oracle will support a phased migration from OWB to ODI


ODI 12c enables easier mapping between OWB 11gR2 concepts and objects and their ODI counterpart. An update to OWB 11gR2 will be released as well which will include a migration utility that automatically translates OWB objects as well as physical mappings into their ODI equivalents. This migration utility is intended as an aid in the migration process – it will not translate all types and variants of OWB artifacts - so customers should plan for manual effort as well.



2014 © Trivadis

DOAG BI - Migrating OWB to ODI
03.04.2014

Why a migration

1. OWB is desupported after Oracle DB 12.1
2. Years of development have been invested in OWB Development
3. Manually rebuilding (using other tools) takes to long
4. *It is possible!!!!* 

The Migration Utility – Expectation

- One-Click-Migration
- Fancy GUI: select easily what I want to migrate
- No failures, everything is migrated without a problem
- Afterwards, Mapping in ODI looks exactly the same as OWB
- Mappings work exactly the same

The Migration Utility - Reality

- Command-Line tool, no fancy windows!
- Performs a Repository to Repository migration
 - OWB and ODI Repository must be available and accessible
- Patch to an existing OWB installation
 - Initially only on top of 11.2.0.4, now also directly from 11.2.0.3
- Migrating a MDL-File directly to ODI is not possible!
- Several limitations exist
- A lot of OWB Mappings can be migrated without issues
- Migration runs quite fast



The Migration Utility

- Highest level of the migration utility is a Workspace
- Lower levels can be specified (see Configuration file)
- Test runs are possible: this will execute the same programs as the actual migration without writing the information in the ODI repository
- Logging detail can be specified

```
*****  
* Oracle Warehouse Builder - Migration Utility - Log  
* Created: 10/28/13 8:13 AM  
* Migration Report Style - RUN  
*  
* OWB Release:11.2.0.4.0 - OWB Repository:owb_owner/localhost:1521:orcl - OWB Workspace:OWB_OWNER.OWB_OWNER  
*  
* ODI Release:12.1.2 - ODI Master Repository:ODI/jdbc:oracle:thin:@localhost:1521:orcl - ODI User/Work Repository:SUPERVISOR/WORK_MIGR  
*  
* Log File: /home/oracle/log/Migrate.log  
*****  
  
Migration started at 10/28/13 8:13 AM Central European Time  
  
*****  
START MIGRATE PROJECT TEST_MIGR.  
----START MIGRATE MODULE MAP_MOD.
```


The Migration Utility

OWB Object	ODI Object
Project	Project
Module	Module
Mapping	Mapping
Sequence	Native Sequence
Tables & Views	Models: Tables & Views
Logical Locations	Logical Schemas
Physical Locations	Datastores

The Migration Utility – Prerequisites: MOS note **1503877.1**

The Migration Utility is:

- A command-line tool that enables you to migrate Design-Time Metadata from **OWB 11.2.0.3** or **OWB 11.2.0.4** to ODI 12c.
- Supported on Windows and Linux 64-bit x86 systems only.
- Provided in [Patch 17830453](#) for your OWB 11.2.0.4 installation.
- provided in [Patch 17224695](#) for your OWB 11.2.0.3 installation.

The migration utility requires:

- **For OWB 11.2.0.4**
 - OWB 11.2.0.4 installed
 - OWB [Patch 17830453](#) applied
 - ODI 12.1.2.0.0 installed
 - ODI 12.1.2.0.1 Bundled [Patch 17836908](#) applied
- **For OWB 11.2.0.3**
 - OWB 11.2.0.3 installed
 - OWB Third Cumulative [Patch 16568042](#) applied
 - OWB [Patch 17224695](#) applied
 - ODI 12.1.2.0.0 installed
 - ODI 12.1.2.0.1 Bundled [Patch 17836908](#) applied

Migration Utility - Configuration

- To configure the migration utility a configuration file is used. Oracle provides an example configuration file which can be adapted to suite the individual needs
- When a Configuration Option is not set, the default value is used
- When a Configuration Option is misspelled, this is simply ignored (no error is raised)

```
OWB_WORKSPACE_OWNER=owb_owner
OWB_WORKSPACE_NAME=OWB_OWNER
OWB_URL=localhost:1521:orcl
ODI_MASTER_USER=ODI
ODI_MASTER_URL=jdbc:oracle:thin:@localhost:1521:orcl
ODI_MASTER_DRIVER=oracle.jdbc.OracleDriver
ODI_USERNAME=SUPERVISOR
ODI_WORK_REPOSITORY_NAME=WORK_MIGR
MIGRATION_MODE=RUN
MIGRATION_OBJECTS=*
MIGRATION_REPORT_INCLUDE=ALL
MIGRATE_DEPENDENCIES=YES
STOP_ON_ERROR=FALSE
SPLIT_JOIN_FOR_ANSI=TRUE
MIGRATE_UNBOUND_OPERATOR=TRUE
MIGRATION_LOG_FILE=/home/oracle/log/Migrate.log
```

Migration Utility – Configuration File

The configuration file is divided in several groups:

- OWB:
 - OWB_WORKSPACE_OWNER= <Workspace owner>
 - OWB_WORKSPACE_NAME= <Name OWB Workspace>
 - OWB_URL= <host:port:sid>

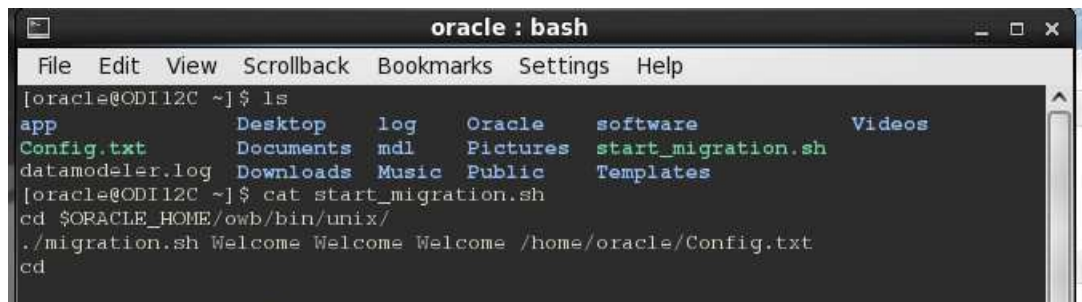
- ODI:
 - ODI_MASTER_USER=ODI
 - ODI_MASTER_URL=jdbc:oracle:thin:@ <host:port:sid>
 - ODI_MASTER_DRIVER=oracle.jdbc.OracleDriver
 - ODI_USERNAME=SUPERVISOR
 - ODI_WORK_REPOSITORY_NAME= <Name Work Repository>

Migration Utility – Configuration File

- Migration settings (optional):
 - MIGRATION_MODE=RUN
 - MIGRATION_OBJECTS=*
 - MIGRATION_REPORT_INCLUDE=ALL
 - MIGRATE_DEPENDENCIES=YES
 - STOP_ON_ERROR=FALSE
 - SPLIT_JOIN_FOR_ANSI_SYNTAX=TRUE
 - MIGRATE_UNBOUND_OPERATOR=TRUE
 - MIGRATION_LOG_FILE=/home/oracle/log/Migrate.log
 - FLUSH_BATCH_SIZE=50

Migration Utility – Executing

- Executing the Utility is easy:
 - Prepare the Configuration utility: create the configuration file
 - Starting the utility:
 - `$ORACLE_HOME/owb/bin/unix/migration.sh <PW OWB> <PW Supervisor> <PW ODI User> /home/oracle/Config.txt`

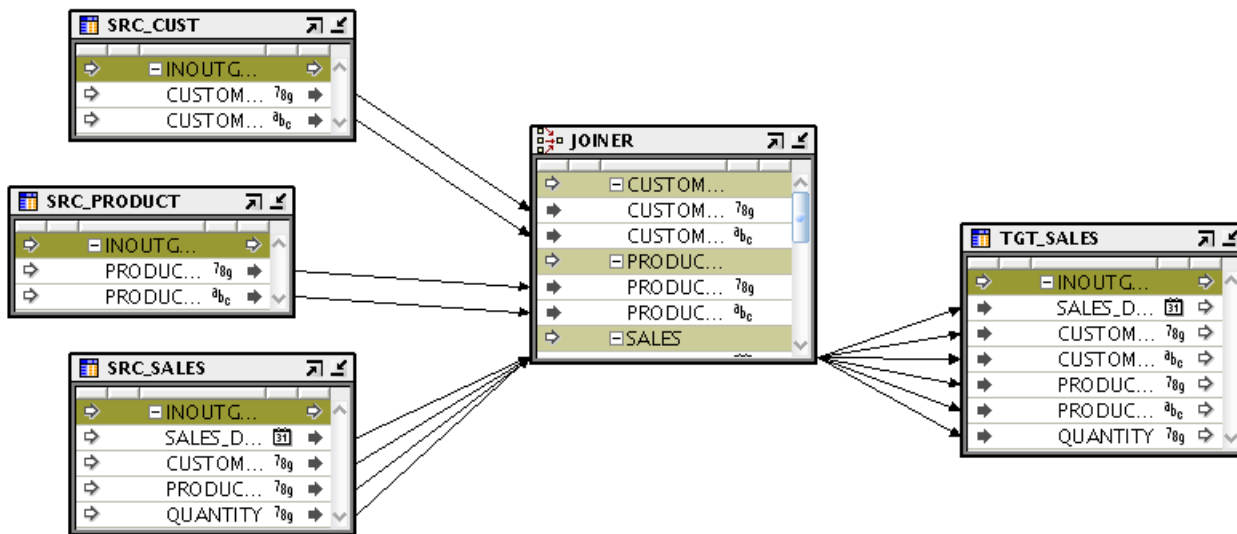


```
oracle : bash
File Edit View Scrollback Bookmarks Settings Help
[oracle@ODI12C ~]$ ls
app Desktop log Oracle software Videos
Config.txt Documents mdl Pictures start_migration.sh
datamodeler.log Downloads Music Public Templates
[oracle@ODI12C ~]$ cat start_migration.sh
cd $ORACLE_HOME/owb/bin/unix/
./migration.sh Welcome Welcome Welcome /home/oracle/Config.txt
cd
```

- Progress of the migration can be followed in the log files:
(in Linux: `watch -n 5 tail Migrate.log`)

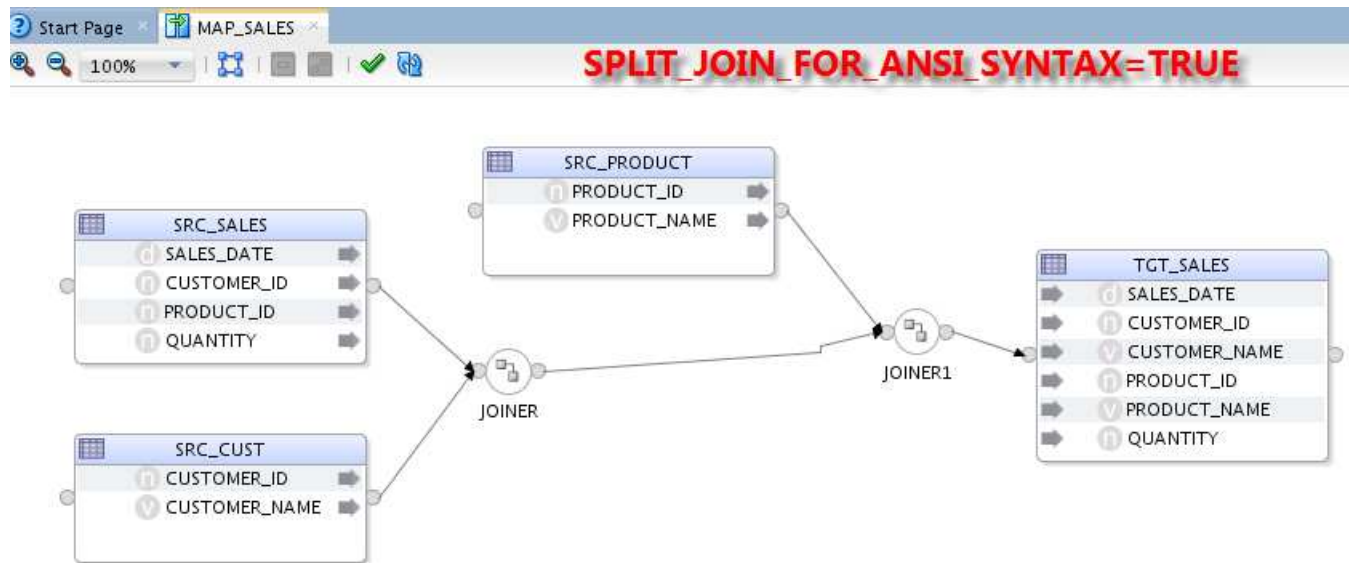
Impact of Configuration Setting on Migration

OWB:

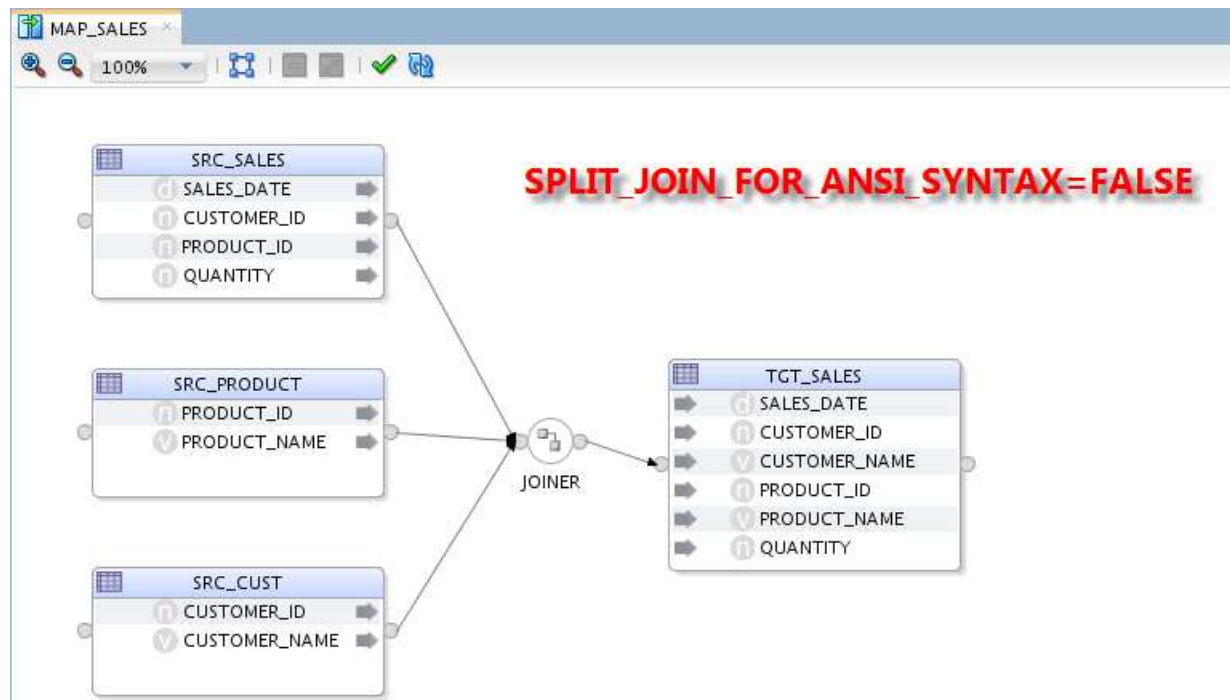


Impact of Configuration Setting on Migration

ODI:



Impact of Configuration Setting on Migration



Migration Utility – Limitations

- No GUI available ☹️
- 1 Workspace at a time
- Several consequences if an incremental approach is taken:
 - All datastores will be duplicated
 - Multiple Projects and Models will be created in ODI
- Not all operators from OWB are migrated to ODI:
 - Dimensions/Cubes/ Custom pl/sql
 - Processflows
- Not all OWB constructions can be migrated to ODI (yet):
 - Mappings without a dataflow (e.g. Mappings to execute a Procedure)
 - Using a single operator as input for multiple ingroups in a joiner
 - Delete as target operation is set to «insert append»
 - Mappings with Row-based operations (Cursor Loops)

Migration Utility – Known Issues in ODI!

- See open issues list in latest patch set

e.g (Extract)

★ **Patch 17836908: ORACLE DATA INTEGRATOR 12C (12.1.2.0.1) BUNDLE PATCH**

ANALYTIC SQL EXPRESSION "THEN" USED IN FILTER GENERATES INVALID CODE
After you migrate an OWB mapping that contains a filter with analytic SQL expression to ODI and execute it, incorrect code is generated. The analytic SQL expression is wrongly copied into the filter clause. [17735977]

MAPPINGS CONTAINING SPLITTER OPERATER GENERATES INVALID CODE
OWB mappings that contain Splitter operator are not migrated correctly to ODI. Executing such a mapping in ODI generates invalid code. [17671367]

ISSUES WITH MIGRATION OF SINGLE TABLE MAPPINGS
The following issues are seen when you migrate single table mappings from OWB to ODI:

- A table used as a target in OWB is positioned in the source group in ODI.
- The pre-mapping process is not migrated.
- The input parameter is migrated as a variable. However, it is not used in the mapping.

[17669872]

COMMENTS IN JOINER JOIN CONDITION NOT MIGRATED
When you migrate an OWB mapping that contains a Joiner Join operator, the Joiner Join operator is migrated as a Join component in ODI. However, any comments in the Joiner Join condition are not migrated. [17620844]

FILE ENCLOSURES NOT MIGRATED
OWB flat file property "Enclosures", which should be migrated as "Text Delimiter" of file datastore in ODI, is not migrated. [17231688]

Migration - Planning

- It's a bit complicated to plan for a migration
- Influencing factors:
 - Number of repositories, workspaces and projects
 - Number of objects within a project
 - Used features in OWB
- Not all problems are identified:
 - Mapping is migrated correctly: no errors in log file
 - When validating in ODI: several new errors might occur (e.g. with pre/post Mapping operators)
- Some issues are known and can be anticipated
 - Check in OWB Repositories if those issues exist (e.g. row-based mappings)
 - Estimate time needed
 - to change this in OWB
 - to change this in ODI

Migration – Plan of action

- Start with a dry-run, potentially in a separate environment (VM or DB)
- Determine common failures in the log file
- Common issues should if possible be corrected in OWB:
 - Manually
 - Tcl
- Execute a migration
 - Identify the Mappings that are not migrated due to errors
 - Identify the Mappings that have been migrated but had warnings
- Afterwards
 - Create the non-migrated Mappings manually (if possible, generate them!)
 - Validate and test the migrated Mappings
 - Fix potential erroneous Mappings

Conclusion

- Migration utility is **easy to use**
- Unfortunately only command line interface
- **Non-complex** mappings are **migrated** in most cases **correctly**
- **Complex Mappings** should be **checked** more carefully, even if the migration utility didn't report a problem
- **Corrections** in OWB (manually or via Tcl) can **improve** the **quality** and **completeness** of the migration
- Several **open issues exist** in ODI: till these are solved, a work-around must be created
- It will be **necessary to manually migrate some mappings** due to limitations in the utility

More Information

Trivadis TechnoCircle Oracle Data Integrator 12c

- 15.05 Zürich
- 22.05 Stuttgart
- 03.06 Frankfurt
- 05.06 Basel
- 16.06 Lausanne
- 17.06 Bern
- 03.07 Düsseldorf / Hamburg
- 07.07 München

Trivadis Stand