

OWB2ODI Migration Check - ein Feldtest



BASEL BERN BRUGG LAUSANNE ZUERICH DUESSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MUNICH STUTTGART VIENNA

■ AGENDA

1. Einführung
2. OWB2ODI Migration Check
3. Feldtest
4. Ergebnisse
5. Fazit

■ Vorstellung

Martin de Gooijer

- Über 20 Jahre Oracle Erfahrung
 - 11 Jahre Oracle Niederlande
 - Seit 2005 Trivadis (Schweiz)
- OWB Erfahrung: seit 2001
- ODI Erfahrung: seit 2012
- Referent OWB und ODI Kurse
- Teilnehmer ODI 12c und OWB Migration Utility Beta Programm

Einführung

■ Einführung – Warum migrieren?

- Desupport des Oracle Warehouse Builders ab Oracle DB 12.2
- Viele Jahre sind in die Entwicklung von OWB Mappings investiert worden
- Alles neu entwickeln (mit anderen Werkzeugen) ist teuer und kostet viel Zeit

■ Einführung – Migration Utility

- Command-Line, kein schöne Windows-Umgebung!
- Es findet ein Repository-to-Repository Migration statt
 - OWB und ODI Repository müssen beide verfügbar und erreichbar sein
- Patchen einer bestehende OWB Installation
 - OWB 11.2.0.3 oder 11.2.0.4
- Direktes Migrieren einer MDL-Datei ist nicht möglich
- Es gibt einige Einschränkungen
- Trotzdem können viele Mappings direkt migriert werden
- Migration funktioniert sehr schnell und ohne manuelle Eingriffe

■ Einführung – Migration Planung

- Es ist nicht einfach eine Migration zu planen
- Einflussfaktoren sind:
 - Anzahl Repositories, Workspaces und Projekte
 - Anzahl und Komplexität der Objekte innerhalb die Projekte
 - Verwendete OWB-Features
- Nicht alle Probleme werden erkannt:
 - Mapping ist erfolgreich migriert worden: keine Fehler in der Logdatei
 - Beim Validieren in ODI: verschiedene Fehler können auftreten (z.B. mit Pre/Post-Mapping Operators)
- Bestimmte Fehler können im Voraus schon beachtet werden

OWB2ODI Migration Check

■ OWB2ODI Migration Check

- Der Trivadis Migration Check dient als fundierte Grundlage, um ein Migrationsprojekt einer konkreten OWB Lösung einschätzen zu können
- Mit Hilfe von Tools und speziellen Skripten werden mögliche Migrationsprobleme analysiert und daraus resultierende Migrationsaufwände definiert
- Der Migrations-Check bringt Sicherheit in die Planung eines Migrationsvorhabens
- Möglich für OWB Repositories < 11.2.0.3!

■ OWB2ODI Migration Check - Vorgehen

- Workshop zur Voranalyse der Bestandslösung und zur zukünftigen Ausrichtung
- Test Run Migration Tool
- Auswertung und Aufbereitung der Ergebnisse aus dem Test-Run
- Problemanalyse des OWB Repository mittels OMB+ Skripte
- Aufbereitung der Ergebnisse und Präsentation

Feldtest

■ Feldtest

Aufgabe

- Durchführen einen Migration Check für OWB und eine Aufwandschätzung für eine eventuelle Migration erstellen
- Mögliche Varianten aufzeichnen

■ Feldtest - Vorgehen

Workshop

- Abklären wie die OWB Repository aufgesetzt und verwendet worden ist
 - OWB Version
 - OWB Lizenzen
 - Anzahl Workspaces und Projekte
- Struktur von OWB Projekten/Module analysieren

■ Feldtest - Execution

- Erstellen eines OWB Exports und Importieren in Analyse Umgebung
- OWB Mappings identifizieren welche nicht verwendet werden (deployable = false) oder bereits im OWB nicht kompilierbar sind
- OWB Repository analysieren:
 - Check auf Objekte welche nicht migriert werden können
 - Check auf Objekte welche manchmal Problemen machen
 - Complex mappings
 - Row-based mappings
 - Unbound Tables und View Operatoren

■ Feldtest - Execution

Ausführen ODI Migration, inklusiv Analyse Logdateien

- Fast Check
- Dry Run
- Run

Ergebnisse

■ Feldtest - Ergebnisse

Bereinigen von Repository

```
Started at 10:46:48
Search parameters
=====
Project : TEST

DWH
MAP_TEST_1 Deployable: false
Number of Mappings in Module      : 78
Number of Mappings not deployable : 1
-----

KMT_SOURCE
Number of Mappings in Module      : 0
Number of Mappings not deployable : 0
-----

STAGING
Number of Mappings in Module      : 0
Number of Mappings not deployable : 0
-----

STAR
BCK_MAP_1 Deployable: false
BCK_FCT_MAP1 Deployable: false
OLD_FCT_REPORT Deployable: false
OLD_FCT_RESULT Deployable: false
OLD_MAP_FCT_REPORT Deployable: false
Number of Mappings in Module      : 49
Number of Mappings not deployable : 5
-----

Total number of Mappings in project : 207
```

```
#Started at 10:46:48
OMBCC '/TEST/DWH'
OMBDROP MAPPING
'MAP_TEST_1'
OMBCC '/TEST/STAR'
OMBDROP MAPPING 'BCK_MAP_1'
OMBDROP MAPPING
'BCK_FCT_MAP1'
OMBDROP MAPPING
'OLD_FCT_REPORT'
OMBDROP MAPPING
'OLD_FCT_RESULT'
OMBDROP MAPPING
'OLD_MAP_FCT_REPORT'
OMBCOMMIT
```

■ Feldtest - Ergebnisse

```
Total Projects Migrated: 1
*****
*****
PROJECT: TEST
```

Object Types	Migrated	
Rejected	Skipped	
SEQUENCE:	12	0
0		
TABLE:	294	0
0		
MAPPING_MODULE:	4	0
0		
VIEW:	1	0
0		
MODULE:	4	0
0		
MAPPING:	201	0
0		
MODULE_FOR_LOGICALSCHEMA:	4	0
0		

```
In Report: [WARN][Migration][MU-5022]Unable to
find corresponding integration type in ODI
according to the loading type DELETE
```

```
Function GETDATAQUALITY (IN VARCHAR2, IN VARCHAR2)
RETURN NUMBER will not be migratedEWH
ERROR: Target Load Table Operator
EWH_ACCOUNTINFO_TARGET does not exist in the mapping
WARNING: Row Based Operation. Mapping:
MAP_EWH_ACCOUNTINFO Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
MAP_EWH_ACCOUNT_HAS_STAB Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping: MAP_EWH_ACTIVITY
Operation: SET_BASED_FAIL_OVER_TO_ROW_BASED
ERROR: Target Load Table Operator EWH_ADDRESS_TARGET
does not exist in the mapping
WARNING: Row Based Operation. Mapping:
MAP_EWH_ADDRESS Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
MAP_EWH_ADDRESS_TYPE Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
MAP_EWH_APPOINTMENT Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
MAP_EWH_APPOINTMENT_HAS_LEAD Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
MAP_EWH_APPOINTMENT_TYPE Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping: MAP_EWH_AZG
Operation: SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
MAP_EWH_CAMPAIGN Operation:
SET_BASED_FAIL_OVER_TO_ROW_BASED
WARNING: Row Based Operation. Mapping:
```

■ Feldtest - Ergebnisse

Es gab verschiedene Warnings und Errors:

Typ	Umschreibung	Anzahl
Error	Target Load Order falsch (kein ETL Option!)	39
Error	Target Load Operator Delete	4
Warning	Row-Based Mappings	199*
Warning	Komplexe Mappings	6
Warning	Pre/Post Mapping Operators	2

■ Feldtest - Ergebnisse

Allgemeine Aufwände:

Typ	Umschreibung	Aufwand
Allgemein	Migration	1 Tag
Betrieb	Scheduling (Jobpläne werden nicht migriert)	1-5 Tage
Betrieb	Definieren neu Deploymentkonzept	2 Tage
	Training	3 Tage/Mitarbeiter
	Testen	10 Tage

■ Feldtest - Ergebnisse

Bei Warnings eher Testaufwand - bei Issues/Errors gilt der genannte Aufwand

Typ	Umschreibung	Aufwand
Error	Target Load Order falsch (kein ETL Option!)	39 : 5 Min/Map
Error	Target Load Operator Delete	4: 0.5 Tag/Map
Warning	Row-Based Mappings	199 Maps
Warning	Komplexe Mappings	6: 2-3 Tage/Map
Warning	Pre/Post Mapping Operators	2: 0-2 Tage

■ Feldtest - Ergebnisse

Aufwände ohne Migration (manuelle Neuimplementierung):

Typ	Umschreibung	Aufwand
Allgemein	Siehe vorher (exklusiv migration)	
Mappings	Einfache Mappings: 4 Stunden/Mapping	195: 97.5 Tage
C	Complexe Mappings: 2 Tage/Mapping	6: 12 Tage

Fazit

■ Fazit

- Migration ist sehr gut möglich
- Aufwand wird bestimmt durch:
 - Anzahl Repositories/Workspaces/Projects
 - Verwendete OWB Operatoren
 - Anzahl Mappings
- Es gibt sowieso Änderungen:
 - Deployment/Executionkonzepte
 - Training

■ Fazit

- Eine Migration nach ODI ist günstiger als eine manuelle Migration
- Die OWB2ODI Migration Check von Trivadis listet innerhalb kurzer Zeit die zu erwartenden Aufwände für eine Migration auf
- Somit werden die existierenden Unsicherheiten beseitigt und gleichzeitig eine bessere Entscheidungsgrundlage geschaffen

Questions and answers ...



BASEL BERN BRUGG LAUSANNE ZUERICH DUESSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MUNICH STUTTGART VIENNA