Near-Zero Downtime Database Migration to Unicode - Procedures

ORACLE WHITE PAPER | JUNE 2015

Table of Contents

Introduction

A Near-Zero Downtime Database Migration to Unicode is discussed in two white papers:

- Near-Zero Downtime Database Migration to Unicode - Overview

  (The paper should be read before proceeding with this document).
- Near-Zero Downtime Database Migration to Unicode - Procedures (this paper).

  This white paper describes the setup procedures and detailed steps to perform a near-zero downtime database migration to Unicode for Oracle Database 11*g* and 12*c* using Oracle Database Migration Assistant for Unicode, Oracle GoldenGate, and Oracle Recovery Manger.

Audience

This paper is designed to be used by database administrators (DBAs) who are responsible for migrating databases to Unicode. It is assumed that the DBAs have basic working knowledge of Oracle Database, Oracle Database Migration Assistant for Unicode, Oracle GoldenGate, and Oracle Recovery Manager.

Assumption

We have used the following information for the purpose of illustration in this paper:

- Source database - WE8ISO8859P1 character set
- Target database - AL32UTF8 character set
- Platform - Linux x64
- Oracle Database and Oracle GoldenGate run on the same host

Software Versions

A list of minimum software requirements:

- Oracle Database version 11*g* or later
- Oracle Database Migration Assistant for Unicode (DMU) version 2.1 or later
- Oracle GoldenGate (OGG) version 12.1.2.1.0 or later

Near-Zero Downtime Unicode Migration Process

The diagram depicts the migration process using Oracle Database Migration Assistant for Unicode (DMU), Oracle GoldenGate (OGG) and Oracle Recovery Manager (RMAN) to implement a near-zero downtime database migration to Unicode.

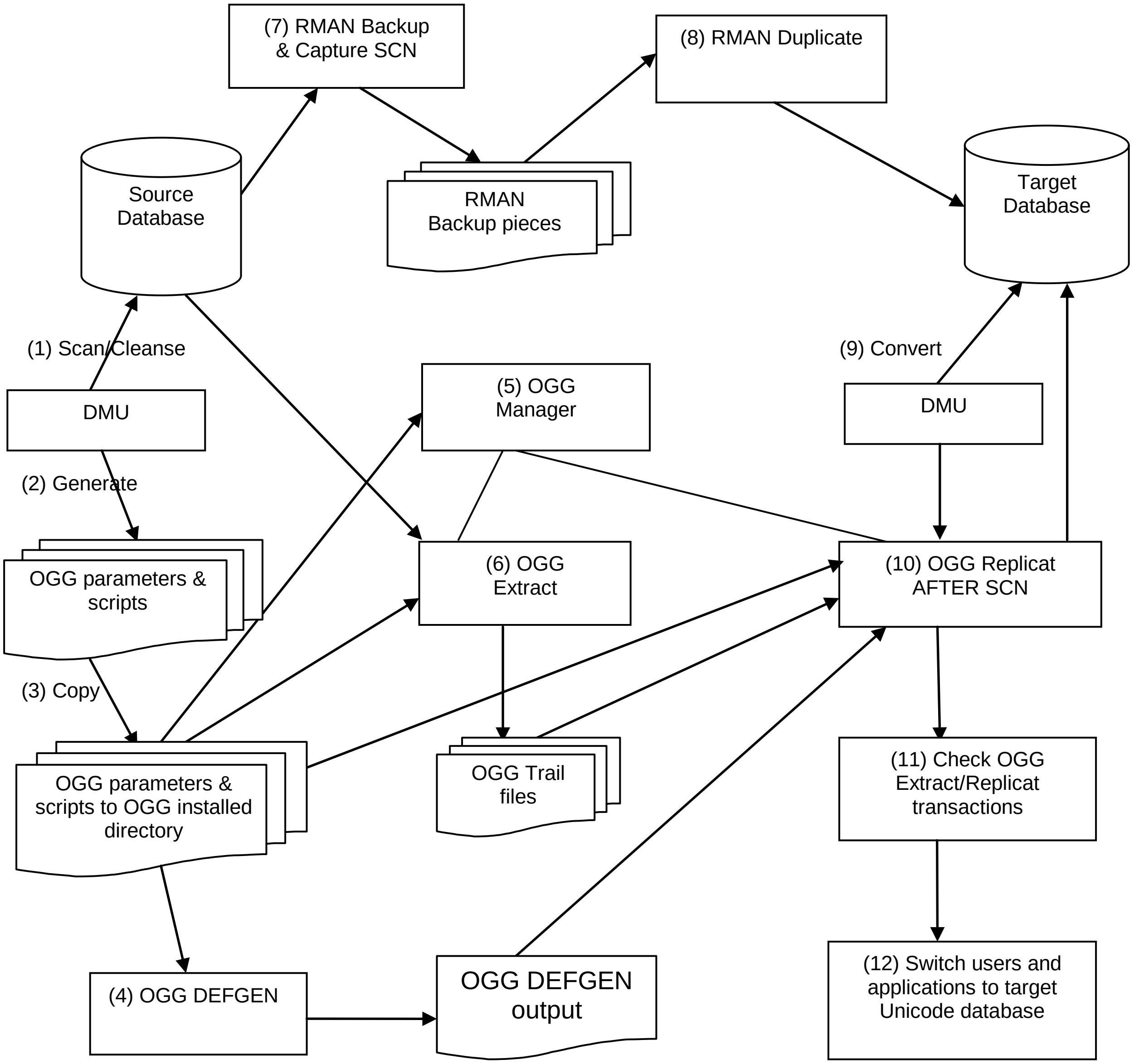

Fig. 1 – Near-Zero Downtime Unicode Migration Process

(1) Use DMU to scan and cleanse the source database until all the reported convertibility issues are resolved.

(2) Use DMU to generate OGG configuration files (OGG parameter files and Obey scripts). DMU can generate two types of configurations:

Local Replicat – OGG Extract and Replicat processes run on the same host.

Remote Replicat – OGG Extract and Replicat processes run on different hosts.

(3) Copy the DMU generated OGG configuration files that were generated in step 2 to the Oracle GoldenGate installed directory.

Parameter/Script	**Source Directory**	**Destination Directory**
Manager parameter	<*DMU_DIR*>/dirprm/mgr.prm	<*OGG_HOME*>/dirprm/mgr.prm
Extract parameter	<*DMU_DIR*>/dirprm/ext.prm	<*OGG_HOME*>/dirprm/ext.prm
Replicat parameter	<*DMU_DIR*>/dirprm/rep.prm	<*OGG_HOME*>/dirprm/rep.prm
DEFGEN parameter	<*DMU_DIR*>/dirprm/defgen.prm	<*OGG_HOME*>/dirprm/defgen.prm
DEFGEN script	<*DMU_DIR*>/defgen.bat	<*OGG_HOME*>/defgen.bat
	<*DMU_DIR*>/defgen.sh	<*OGG_HOME*>/defgen.sh
Obey start script	<*DMU_DIR*>/start_extract	<*OGG_HOME*>/start_extract
	<*DMU_DIR*>/start_replicat	<*OGG_HOME*>/start_replicat
Obey stop script	<*DMU_DIR*>/stop_extract	<*OGG_HOME*>/stop_extract
	<*DMU_DIR*>/stop_replicat	<*OGG_HOME*>/stop_replicat

Fig. 2 - A Local Replicat Configuration

Note:

<*DMU_DIR*> is the directory on the DMU system where configuration files are generated.
<*OGG_HOME*> is the Oracle GoldenGate home directory on the system where Oracle GoldenGate software is installed.

Parameter/Script	**Source Directory**	**Destination Directory**
Manager parameter (Extract)	*<DMU_DIR>*/extract/dirprm/mgr.prm	*<OGG_HOME_EXT>*/dirprm/mgr.prm
Extract parameter	*<DMU_DIR>*/extract/dirprm/ext.prm	*<OGG_HOME_EXT>*/dirprm/ext.prm
DEFGEN parameter	*<DMU_DIR>*/extract/dirprm/defgen.prm	*<OGG_HOME_EXT>*/dirprm/defgen.prm
DEFGEN script	*<DMU_DIR>*/extract/defgen.bat	*<OGG_HOME_EXT>*/defgen.bat
	<DMU_DIR>/extract/defgen.sh	*<OGG_HOME_EXT>*/defgen.sh
Obey start Extract script	*<DMU_DIR>*/extract/start_extract	*<OGG_HOME_EXT>*/start_extract
Obey stop Extract script	*<DMU_DIR>*/extract/stop_extract	*<OGG_HOME_EXT>*/stop_extract
Manager parameter (Replicat)	*<DMU_DIR>*/replicat/dirprm/mgr.prm	*<OGG_HOME_REP>*/dirprm/mgr.prm
Replicat parameter	*<DMU_DIR>*/replicat/dirprm/rep.prm	*<OGG_HOME_REP>*/dirprm/rep.prm
Obey start Replicat script	*<DMU_DIR>*/replicat/start_replicat	*<OGG_HOME_REP>*/start_replicat
Obey stop Replicat script	*<DMU_DIR>*/replicat/stop_replicat	*<OGG_HOME_REP>*/stop_replicat

Fig. 3 - A Remote Replicat Configuration

Note:

<DMU_DIR> is the directory on the DMU system where configuration files are generated.
<OGG_HOME_EXT> is the Oracle GoldenGate home directory on the Extract host system.
<OGG_HOME_REP> is the Oracle GoldenGate home directory on the Replicat host system.

(4) Execute the defgen script present in the OGG installed directory to generate source table definitions which were scheduled for cleansing by DMU. The defgen script generates the output file “dmu_source.def” in the *<OGG_HOME>*/dirdef directory.

(5) Start the Oracle GoldenGate Manager process in the Oracle GoldenGate Command Interpreter. The Oracle GoldenGate Manger process is the parent process of Oracle GoldenGate for managing its Extract and Replicat processes and files.

```
GGSCI>start manager
```

(6) Start the Oracle GoldenGate Extract process in the Oracle GoldenGate Command Interpreter for capturing committed transactions from the source database into the Oracle GoldenGate trail files.

```
GGSCI>obey start_extract
```

(7) Use RMAN to back up the source database and capture the checkpoint SCN value to be used by the Oracle GoldenGate Replicat process in step 10.

(8) Use RMAN to create the target database by cloning the source database using the backup pieces in step 7.

(9) Use DMU to convert the target database character set to Unicode. You should be able to start the DMU conversion on the target database immediately, as all the convertibility issues on the source database have been cleansed in step 1 and the corresponding migration-related metadata information in the DMU repository has been cloned from the source database. Restart the target database after the DMU conversion is complete.

(10) Edit start_replicat Obey script by replacing the SCN value in it with the SCN value captured in step 7. Start the Oracle GoldenGate Replicat process in the Oracle GoldenGate Command Interface for applying the committed transactions from the Oracle GoldenGate trail files to the target database.

```
GGSCI>obey start_replicat
```

(11) Ensure that all the transactions have been extracted and replicated by Oracle GoldenGate:

» Check that there are no active transactions in the source and the target databases:

```
SQL>select count(*) from v$transaction;
COUNT(*)
--------
       0
```

» Query the current SCN of the source database:

```
SQL>select current_scn from v$database;
CURRENT_SCN
-----------
    1439706
```

» Query the status of the Extract process and verify that it has read past the last SCN value (for example, the SCN value of 1439706 mentioned in the above step:

```
GGSCI>send extract ext, status
Sending STATUS request to EXTRACT EXT ...
EXTRACT EXT (PID 6312)
  Current status: Recovery complete: Processing data
  Current read position:
  Redo thread #: 1
  Sequence #: 161
  RBA: 10483752
  Timestamp: 2015-01-09 13:56:14.000000
  SCN: 0.1439718
  Current write position:
  Sequence #: 0
  RBA: 1534
  Timestamp: 2015-01-09 13:56:13.756998
  Extract Trail: ./dirdat/dm
```

» Query the status of the Replicat process and verify that it shows At EOF status:

```
GGSCI>send replicat rep, status
Sending STATUS request to REPLICAT REP ...
  Current status: At EOF
  Sequence #: 0
  RBA: 1534
   0  records in current transaction
```

(12) Switch users and applications to the target Unicode database.

Generating Oracle GoldenGate Configuration Files

DMU generates parameter and script files according to the database version. See the table below:

Database version	OGG Capture Mode	OGG Transaction Data
11.1.0.7	Classic	Table Level (TRANDATA)
11.2.0.1	Classic	Table Level (TRANDATA)
11.2.0.2	Classic	Table Level (TRANDATA)
11.2.0.3	Classic	Schema Level (SCHEMATRANDATA)
11.2.0.4	Integrated	Schema Level (SCHEMATRANDATA)
12.1.0.1	Integrated	Schema Level (SCHEMATRANDATA)
12.1.0.2	Integrated	Schema Level (SCHEMATRANDATA)

Fig. 4 – Generation of Oracle GoldenGate Configuration Files

OGG Capture Modes:

- Classic capture - the Oracle GoldenGate Extract process captures data changes from the Oracle redo or archive log files on the source system.
- Integrated capture - the Oracle GoldenGate Extract process interacts directly with a database Logmining server to receive data changes in the form of logical change records (LCRs).

OGG Transaction Data:

- TRANDATA (table level) enables Oracle GoldenGate to acquire the transaction information that it needs from the transaction records.
- SCHEMATRANDATA (schema level) enables schema-level supplemental logging for a table.

> THERE IS A SIGNIFICANT PERFORMANCE BENEFIT OF USING SCHEMATRANDATA OVER TRANDATA IF SCHEMAS CONTAIN LARGE NUMBER OF TABLES.

Manager parameter (mgr.prm)

This parameter file is used by the OGG Manager process.

The parameter file contains these parameters:

- PORT - Establish the TCP/IP port number on which Manager listens for requests.
- USERIDALIAS - Provide login information for Manager when it needs to access the database.

Example: mgr.prm

```
PORT 7809
USERIDALIAS oggadm DOMAIN dmu
```

DEFGEN parameter (defgen.prm)

This parameter file is used by the OGG DEFGEN utility. It creates the definition file with data definition for the source tables. The data definition is needed when source and target tables have different definitions. This is true when DMU has performed the scheduled cleansing. DMU generates a list of tables in the definition file with the cleansing actions such as:

- Migrate to character semantics VARCHAR2(n BYTE) to VARCHAR2(n CHAR)
- Lengthen columns, e.g., VARCHAR2(10) to VARCHAR2(20)
- Migrate data type to a new data type, e.g., VARCHAR2 to CLOB
- Lengthen attributes of user-defined object types (ADT)

The parameter file contains these parameters:

- DEFSFILE – Identify the name of the file to which DEFGEN writes the definitions.
- USERIDALIAS - Specify database connection information.
- SOURCECATALOG - Specify a default container for all following TABLE or MAP statements.
- TABLE - Identify a table for which you want to capture a definition.

```
Example: defgen.prm
DEFSFILE ./dirdef/dmu_source.def
USERIDALIAS oggadm DOMAIN dmu
SOURCECATALOG PDB1
TABLE HR2.CONTACTS;
...
TABLE HR2.WAREHOUSES;
```

Extract parameter (ext.prm)

This parameter file is used by the OGG Extract process. DMU generates in the Extract parameter file:

- A list of DDL object names to be replicated.
- A list of tables with ADT columns which have been tagged with the “Assumed Character Set”. e.g.

  ```
  TABLE HR2.CONTACTS, COLCHARSET (WE8MSWIN1252, CONTACT_RECORD");
  ```

- A list of all tables with ADT columns if the “Assumed Database Character Set” is set. e.g.

  ```
  TABLE PM.PRINT_MEDIA, COLCHARSET (WE8ISO8859P15, "AD_HEADER");
  ```

  This provides the database level character set override for tables with ADT columns, similar to the “SOURCECHARSET OVERRIDE” parameter in the Replicat parameter file.

- A list of tables to be captured for replication. e.g.

  ```
  TABLE SH.*;
  ```

The parameter file contains these parameters:

- EXTRACT – Specify an Extract group name.
- setenv - Specify a value for an environment variable from within the GoldenGate Command Interface.
- USERIDALIAS - Specify database connection information when a credential store is in use.
- EXTTRAIL - Specify a trail to which extracted data is written on the local system.

- RMTHOST - Specify the target system and Manager port number.
- RMTTRAIL- Specify a trail to which extracted data is written on a remote system.
- SOURCECATALOG -Specify a default container for all following TABLE or MAP statements.
- DDL INCLUDE OBJNAME - Enable and filter the capture of DDL operations.
- TABLE - Specify tables for extraction and control column mapping and conversion.

Example: ext.prm

```
EXTRACT ext
setenv (ORACLE_HOME="/scratch/oracle/product/12.1.0/dbhome_1")
setenv (ORACLE_SID="cdb1")
USERIDALIAS oggadm DOMAIN dmu
EXTTRAIL ./dirdat/dm
SOURCECATALOG PDB1
DDL INCLUDE OBJNAME APEX_040200.* &
    INCLUDE OBJNAME APEX_PUBLIC_USER.* &
    ...
    INCLUDE OBJNAME SH.*
TABLE HR2.CONTACTS, COLCHARSET (WE8MSWIN1252, "CONTACT_RECORD");
...
TABLE PM.PRINT_MEDIA, COLCHARSET (WE8ISO8859P15, "AD_HEADER");
TABLE APEX_040200.*;
...
TABLE SH.*;
```

Replicat parameter (rep.prm)

This parameter file is used by the Replicat process. DMU generates in the Replicat parameter file:

- The database level override if the “Assumed Database Character Set” is set. e.g.

  ```
  SOURCECHARSET OVERRIDE WE8ISO8859P15;
  ```

- A relative path of the definition file created by the OGG DEFGEN utility e.g.

  ```
  SOURCEDEFS ./dirdef/dmu_source.def;
  ```

- A list of maps for tables with native data type columns which have been tagged with the “Assumed Character Set”. e.g.

  ```
  MAP HR2.CONTACTS, TARGET HR2.CONTACTS,
    COLCHARSET (WE8MSWIN1252, FIRST_NAME, LAST_NAME, RESUME);
  ```

- A list of maps for mapping data from source to target objects. e.g.

  ```
  MAP SH.*, TARGET SH.*;
  ```

The parameter file contains these parameters:

- REPLICAT- Specify a Replicat group name.
- setenv - Specify a value for an environment variable from within the GoldenGate Command Interface.
- USERIDALIAS - Specify database connection information when a credential store is in use.
- SOURCECHARSET OVERRIDE – Specify the source character set to use for converting data to the target character set.

- SOURCEDEFS - Specify a file that contains source data definitions created by the DEFGEN utility.
- SOURCECATALOG - Specify a default container for all following TABLE or MAP statements.
- ASSUMETARGETDEFS - Assume that source and target tables have the same column structure.
- MAP - Specify a relationship between one or more source and target tables and control column mapping and conversion.

`Example: rep.prm`

```
REPLICAT rep
setenv (ORACLE_HOME="/scratch/oracle/product/12.1.0/dbhome_1")
setenv (ORACLE_SID="cdb2")
USERIDALIAS oggadm2 DOMAIN dmu
SOURCECHARSET OVERRIDE WE8ISO8859P15;
SOURCEDEFS ./dirdef/dmu_source.def
SOURCECATALOG PDB1
MAP HR2.CONTACTS, TARGET HR2.CONTACTS,
    COLCHARSET (WE8MSWIN1252, FIRST_NAME, LAST_NAME, RESUME);
...
MAP HR2.WAREHOUSES, TARGET HR2.WAREHOUSES,
    COLCHARSET (WE8MSWIN1252, WAREHOUSE_NAME);
ASSUMETARGETDEFS
MAP APEX_040200.*, TARGET APEX_040200.*;
...
MAP SH.*, TARGET SH.*;
```

Start Extract obey script (start_extract)

This is an OGG obey script for starting the Extract process in the Oracle GoldenGate Command Interface.

The start_extract obey script contains multiple OGG commands:

- Login using USERIDALIAS
- Add a list of TRANDATA (for database version 11.2.0.2 or older) or SCHEMATRANDATA (for database version 11.2.0.3 or later)
- Register an Integrated Extract (For database version 11.2.0.4 or later)
- Add an Extract
- Add an Extract trail
- Start the Extract process

`Example: start_extract`

```
DBLOGIN USERIDALIAS oggadm DOMAIN dmu
ADD SCHEMATRANDATA PDB1.APEX_040200
...
ADD SCHEMATRANDATA PDB1.SH
REGISTER EXTRACT ext DATABASE CONTAINER (PDB1)
ADD EXTRACT ext, INTEGRATED TRANLOG, BEGIN NOW
ADD EXTTRAIL ./dirdat/dm, EXTRACT ext
START EXTRACT ext
```

Start Replicat obey script (start_replicat)

This is an OGG obey script for starting the Replicat process in the Oracle GoldenGate Command Interface.

The start_replicat obey script contains multiple OGG commands:

- Login using USERIDALIAS
- Add a Replicat
- Start the Replicat process using AFTERCSN <*SCN value*> where the value of SCN was captured during RMAN backup.

```
Example: start_replicat
DBLOGIN USERIDALIAS oggadm2 DOMAIN dmu
ADD REPLICAT rep, EXTTRAIL ./dirdat/dm, NODBCHECKPOINT
START REPLICAT rep, AFTERCSN <SCN value>
```

Stop Extract obey script (stop_extract)

This is an OGG obey script for stopping the Extract process in the Oracle GoldenGate Command Interface.

The stop_extract obey script contains multiple OGG commands:

- Login using USERIDALIAS
- Stop the Extract process
- Delete the Extract process
- Delete the Extract trail
- Unregister the Integrated Extract (for database version 11.2.0.4 or later)
- Delete a list of TRANDATA (for database version 11.2.0.2 or older) or SCHEMATRANDATA (for database version 11.2.0.3 or later)

```
Example: stop_extract
DBLOGIN USERIDALIAS oggadm DOMAIN dmu
STOP EXTRACT ext
DELETE EXTRACT ext
DELETE EXTTRAIL ./dirdat/dm
UNREGISTER EXTRACT ext DATABASE
DELETE SCHEMATRANDATA PDB1.APEX_040200
...
DELETE SCHEMATRANDATA PDB1.SH
```

Stop Replicat obey script (stop_replicat)

This is an OGG obey script for stopping the Replicat process in the Oracle GoldenGate Command Interface.

The stop_replicat obey script contains multiple OGG commands:

- Login using USERIDALIAS
- Stop the Replicat process
- Delete the Replicat process

Example: stop replicat

```
DBLOGIN USERIDALIAS oggadm2 DOMAIN dmu
STOP REPLICAT rep
DELETE REPLICAT rep
```

defgen script files (defgen.bat, defgen.sh)

The convenient script files for running the OGG DEFGEN utility on UNIX and Windows platforms.

Example: defgen.sh

```
#!/bin/bash
echo ./defgen paramfile ./dirprm/defgen.prm reportfile ./dirrpt/defgen.rpt
./defgen paramfile ./dirprm/defgen.prm reportfile ./dirrpt/defgen.rpt
rc=$?
if [ $rc -eq 0 ]; then
   echo "defgen executed successfully."
else
   echo "defgen exited with error, check ggserr.log!"
fi
```

Example: defgen.bat

```
@ECHO OFF
REM File. defgen.bat
REM Description.
REM    A script file for creating a defgen file.
REM
ECHO .\defgen.exe paramfile .\dirprm\defgen.prm reportfile .\dirrpt\defgen.rpt
.\defgen.exe paramfile .\dirprm\defgen.prm reportfile .\dirrpt\defgen.rpt
IF ERRORLEVEL 1 GOTO GEN_ERROR
ECHO defgen executed successfully.
GOTO END
:GEN_ERROR
ECHO defgen exited with error, check ggserr.log!
:END
```

Setting up a Near-Zero Downtime Unicode Migration

Oracle Database 11*g* and 12*c* Non-Container Database (Non-CDB)

For the purpose of illustration in this section, we assume that the source database character set is WE8ISO8859P1 on a Linux-x64 platform with Oracle database and Oracle GoldenGate running on the same host.

In this configuration, it uses two instances:

Source Database: WE8ISO8859P1 <SOURCE_SID>
Target Database: WEISO8859P1 → AL32UTF8 <TARGET_SID>

Setup procedures for Oracle Database 11*g* and 12*c* Non-Container Database (Non-CDB):

- Create a Golden Gate user (e.g. OGG_OWNER) on the source database

```
$export ORACLE_SID=<SOURCE_SID>
$sqlplus / as sysdba
SQL>create user <OGG_OWNER> identified by <PASSWORD>
      default tablespace users temporary tablespace temp;
SQL>grant create session to <OGG_OWNER>;
SQL>grant connect,resource to <OGG_OWNER>;
SQL>grant select any dictionary, select any table to <OGG_OWNER>;
SQL>grant all privileges to <OGG_OWNER>;
SQL>grant create table to <OGG_OWNER>;
SQL>grant flashback any table to <OGG_OWNER>;
SQL>grant execute on dbms_flashback to <OGG_OWNER>;
SQL>grant execute on utl_file to <OGG_OWNER>;
SQL>alter database add supplemental log data (all) columns;
SQL>exec dbms_goldengate_auth.grant_admin_privilege ('<OGG_OWNER>');
```

- Create a PFILE from the SPFILE of the source database

```
$export ORACLE_SID=<SOURCE_SID>
$sqlplus / as sysdba
SQL>create pfile='?/dbs/init<SOURCE_SID>.ora' from spfile;
```

- Include the GoldenGate parameters in the source database PFILE e.g.

```
streams_pool_size=1280M              # minimum 1280M
enable_goldengate_replication=TRUE   # for DB 11.2.0.4 and 12.1.0.2
```

- Restart the source database using the PFILE and create the SPFILE

```
$export ORACLE_SID=<SOURCE_SID>
$sqlplus / as sysdba
SQL>shutdown immediate
SQL>startup pfile=?/dbs/init<SOURCE_SID>.ora
SQL>create spfile from pfile='?/dbs/init<SOURCE_SID>.ora';
```

- Alter the source database with archivelog option

  ```
  $export ORACLE_SID=<SOURCE_SID>
  $sqlplus / as sysdba
  SQL>shutdown immediate
  SQL>startup mount
  SQL>alter database archivelog;
  SQL>alter database open;
  ```

- Setup the Oracle GoldenGate environments

  ```
  $export ORACLE_HOME=<ORACLE_HOME_DIR>
  $export OGG_HOME=<OGG_HOME_DIR>
  $export PATH=$PATH:$ORACLE_HOME/bin:$OGG_HOME
  $export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$OGG_HOME
  $export TNS_ADMIN=$ORACLE_HOME/network/admin
  $export ORACLE_SID=<SOURCE_SID>
  ```

- Configure Oracle Golden Gate

  ```
  GGSCI>CREATE SUBDIRS
  ```

- Add the *<OGG_OWNER>* user to the OGG CREDENTIALSTORE for *<SOURCE_SID>*

  ```
  GGSCI>ADD CREDENTIALSTORE
  GGSCI>ALTER CREDENTIALSTORE ADD USER <OGG_OWNER>@<SOURCE_SID>,
       PASSWORD <PASSWORD>, ALIAS <OGG_SOURCE_ALIAS>, DOMAIN <OGG_DOMAIN>
  GGSCI>INFO CREDENTIALSTORE DOMAIN <OGG_DOMAIN>
  GGSCI>DBLOGIN USERIDALIAS <OGG_SOURCE_ALIAS> DOMAIN <OGG_DOMAIN>
  ```

- Enable GoldenGate DDL support for Extract on the source database (Replicat is enabled by default)

  Execute the following SQL scripts in $OGG_HOME directory to enable OGG DDL support:

  ```
  $sqlplus / as sysdba
  SQL>@marker_setup
  SQL>alter session set recyclebin=OFF;
  SQL>@ddl_setup
  SQL>@role_setup
  SQL>grant ggs_ggsuser_role to ogg_owner;
  SQL>@ddl_enable
  SQL>@ddl_pin OGG_OWNER
  ```

  Note: The steps above are only needed if the database version is 11.2.0.3 or older.

- Install the DMU SYS.DBMS_DUMA_INTERNAL package for the source database

```
$export ORACLE_SID=<SOURCE_SID>
$sqlplus / as sysdba
SQL>@?/rdbms/admin/prvtdumi.plb
```

- Perform migration tasks using DMU on the source database
  - Install DMU migration repository
  - Scan the entire database
  - Cleanse any reported convertibility issues
  - Repeat the scanning and cleansing operations until the DMU indicates the database is ready for conversion
  - Generate OGG configuration files using DMU

- Copy the DMU generated configuration files

  Copy the generated OGG parameter and script files from the DMU output directory to the OGG installed directory. See Fig. 2 and Fig. 3 above for local and remote Replicat configurations.

- Run the defgen script to generate the source table definitions

  Execute the defgen script in the $OGG_HOME directory. The OGG DEFGEN utility generates the output file "dmu_source.def" in the $OGG_HOME/dirdef directory.

  Modify the file permission of defgen.sh by running

```
chmod +x defgen.sh
```

  It is helpful to monitor ggserr.log for any warning and error.

```
$tail -f $OGG_HOME/ggserr.log
```

- Start the OGG Manger for managing Extract and Replicat processes and files

```
GGSCI>start manager
```

- Start the OGG Extract for capturing committed transactions from the source database

```
GGSCI>obey start_extract
```

- Use RMAN to backup and capture the checkpoint SCN value of the source database

```
$export ORACLE_SID=<SOURCE_SID>
$rman target /
RMAN>backup database plus archivelog;
RMAN>restore database preview summary;
```

» Clone the target database for Initial Load

» Copy the source parameter PFILE to the target PFILE

```
$cp $ORACLE_HOME/dbs/init<SOURCE_SID>.ora
     $ORACLE_HOME/dbs/init<TARGET_SID>.ora
```

» Modify the target PFILE

Replace the occurrences of <*SOURCE_SID*> with <*TARGET_SID*>*,* and add db_file_name_convert and log_file_name_convert to the PFILE e.g.

```
db_file_name_convert=(<ORACLE_BASE>/oradata/<SOURCE_SID>/,
                       <ORACLE_BASE>/oradata/<TARGET_SID>/)
log_file_name_convert=(<ORACLE_BASE>/oradata/<SOURCE_SID>/,
                       <ORACLE_BASE>/oradata/<TARGET_SID>/)
```

» Create the directories referenced in the target PFILE

```
$mkdir -p <ORACLE_BASE>/admin/<TARGET_SID>/adump
$mkdir <ORACLE_BASE>/oradata/<TARGET_SID>
$mkdir <ORACLE_BASE>/fast_recovery_area/<TARGET_SID>
```

» Create the target database using RMAN

```
$export ORACLE_SID=<TARGET_SID>
$sqlplus /as sysdba
SQL>startup force nomount pfile=?/dbs/init<TARGET_SID>.ora
$rman target sys/<PASSWORD>@<SOURCE_SID> auxiliary /
RMAN>duplicate target database to <TARGET_SID>
        pfile='?/dbs/init<TARGET_SID>.ora';
```

» Comment out the parameters in the target PFILE

```
#db_file_name_convert=(<ORACLE_BASE>/oradata/<SOURCE_SID>/,
                       <ORACLE_BASE>/oradata/<TARGET_SID>/)
#log_file_name_convert=(<ORACLE_BASE>/oradata/<SOURCE_SID>/,
                        <ORACLE_BASE>/oradata/<TARGET_SID>/)
```

» Restart the target database using the target PFILE

```
$export ORACLE_SID=<TARGET_SID>
$sqlplus /as sysdba
SQL>shutdown immediate
SQL>startup pfile=?/dbs/init<TARGET_SID>.ora
```

» Create a password file for the target database

```
$cd $ORACLE_HOME/dbs
$orapwd file=orapw<TARGET_SID> password=<PASSWORD> entries=<NUM_ENTRIES>
```

» Modify listener.ora and tnsnames.ora to include *<TARGET_SID>* and restart the listener

```
$lsnrctl reload
```

» Install the DMU SYS.DBMS_DUMA_INTERNAL package for the target database

```
$export ORACLE_SID=<TARGET_SID>
$sqlplus / as sysdba
SQL>@?/rdbms/admin/prvtdumi.plb
```

» Use DMU to convert the target database character set to Unicode. Make sure that there is no process connected to the target database. You may need to scan OGG schema on the target database.

» Restart the target database and create the target SPFILE

```
$export ORACLE_SID=<TARGET_SID>
$sqlplus / as sysdba
SQL>shutdown immediate
SQL>startup pfile=?/dbs/init<TARGET_SID>.ora
SQL>create spfile from pfile='?/dbs/init<TARGET_SID>.ora';
```

» Add the *<OGG_OWNER>* user to OGG CREDENTIALSTORE for *<TARGET_SID>*

```
GGSCI>ALTER CREDENTIALSTORE ADD USER <OGG_OWNER>@<TARGET_SID>,
     PASSWORD <PASSWORD>, ALIAS <OGG_TARGET_ALIAS>, DOMAIN <OGG_DOMAIN>
GGSCI>INFO CREDENTIALSTORE DOMAIN <OGG_DOMAIN>
GGSCI>DBLOGIN USERIDALIAS <OGG_TARGET_ALIAS> DOMAIN <OGG_DOMAIN>
```

» Modify the *<SCN value>* of the start_replicat Obey script

START REPLICAT rep, AFTERCSN *<SCN value>*
Replace *<SCN value>* with the checkpoint SCN value captured during the RMAN backup step.

» Start the OGG Replicat for applying the committed transactions to the target database

```
GGSCI>obey start_replicat
```

» Check transactions are extracted and replicated correctly

For more information, refer to step 11 above “Ensure that all transactions have been extracted and replicated by Oracle GoldenGate”.

» To stop the OGG replication, execute the following:

```
GGSCI>obey stop_extract
GGSCI>obey stop_replicat
GGSCI>stop manager
```

Oracle Database 12*c* Container Database (CDB)

For the purpose of illustration in this section, we assume that the source database character set is WE8ISO8859P1 on a Linux-x64 platform with Oracle database and Oracle GoldenGate running on the same host.

In this configuration, it uses three instances:

Source Database:	WE8ISO8859P1	<CDB_SORCE_SID>
Cloned Database:	WEISO8859P1	<CDB_DUP_SID>
Target Database:	AL32UTF8	<CDB_TARGET_SID>

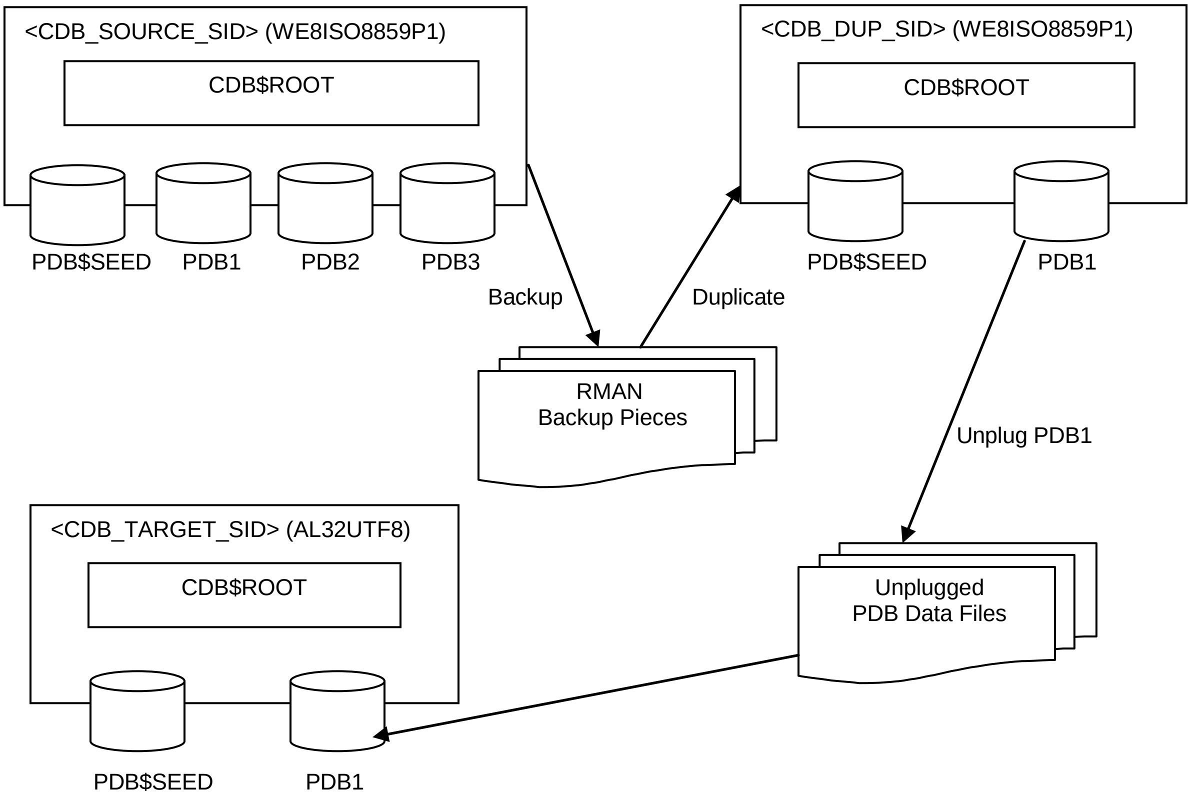

Fig. 5 Backup, Duplicate, Unplug and Plug-in Pluggable Database for Container Database (CDB)

Example: The diagram above shows steps to backup, duplicate, unplug, and plug-in a pluggable database for Container Database (CDB).

» <CDB_SOURCE_SID>
  Use RMAN to backup CDB$ROOT, PDB$SEED and PDB1 to backup pieces.

» <CDB_DUP_SID>
  Duplicate CDB$ROOT, PDB$SEED and PDB1 from the backup pieces, and unplug PDB1 to PDB data files.

» <CDB_TARGET_SID>
  Plug-in PDB1 using the unplugged PDB data files.

Setup procedures for Oracle Database 12*c* Container Database (CDB):

- Create or identify a target Unicode Container database <CDB_TARGET_SID> that will host the converted pluggable database (PDB).

- Create a Golden Gate common user (e.g. C##OGG_OWNER) on the source database

```
$export ORACLE_SID=<CDB_SOURCE_SID>
$sqlplus / as sysdba
SQL>create user <OGG_OWNER> identified by <PASSWORD>
       default tablespace users temporary tablespace temp CONTAINER=ALL;
SQL>grant create session to ogg_owner CONTAINER=ALL;
SQL>grant connect,resource to <OGG_OWNER> CONTAINER=ALL;
SQL>grant select any dictionary, select any table to <OGG_OWNER> ONTAINER=ALL;
SQL>grant all privileges to <OGG_OWNER> CONTAINER=ALL;
SQL>grant create table to <OGG_OWNER> CONTAINER=ALL;
SQL>grant flashback any table to <OGG_OWNER> CONTAINER=ALL;
SQL>grant execute on dbms_flashback to <OGG_OWNER> CONTAINER=ALL;
SQL>grant execute on utl_file to <OGG_OWNER> CONTAINER=ALL;
SQL>alter database add supplemental log data (all) columns;
SQL>exec dbms_goldengate_auth.grant_admin_privilege
       ('<OGG_OWNER>', container=>'ALL');
```

- Create a PFILE from the SPFILE of the source database

```
$export ORACLE_SID=<CDB_SOURCE_SID>
$sqlplus / as sysdba
SQL>create pfile='?/dbs/init<CDB_SOURCE_SID>.ora' from spfile;
```

- Include the GoldenGate parameters in the source database PFILE e.g.

```
*.streams_pool_size=1280M             # minimum 1280M
*.enable_goldengate_replication=TRUE  # for DB 11.2.0.4 and 12.1.0.2
```

- Restart the source database using the PFILE and create the SPFILE

```
$export ORACLE_SID=<CDB_SOURCE_SID>
$sqlplus / as sysdba
SQL>shutdown immediate
SQL>startup pfile=?/dbs/init<CDB_SOURCE_SID>.ora
SQL>create spfile from pfile='?/dbs/init<CDB_SOURCE_SID>.ora';
```

- Alter the source database with archivelog option

```
$export ORACLE_SID=<CDB_SOURCE_SID>
$sqlplus / as sysdba
SQL>shutdown immediate
SQL>startup mount
SQL>alter database archivelog;
SQL>alter database open;
```

- Setup the Oracle GoldenGate environments

```
$export ORACLE_HOME=<ORACLE_HOME_DIR>
$export OGG_HOME=<OGG_HOME_DIR>
$export PATH=$PATH:$ORACLE_HOME/bin:$OGG_HOME
$export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$OGG_HOME
$export TNS_ADMIN=$ORACLE_HOME/network/admin
$export ORACLE_SID=<CDB_SOURCE_SID>
```

- Configure Oracle Golden Gate

```
GGSCI>CREATE SUBDIRS
```

- Add the *<OGG_OWNER>* common user to the OGG CREDENTIALSTORE for *<CDB_SOURCE_SID>*

```
GGSCI>ADD CREDENTIALSTORE
GGSCI>ALTER CREDENTIALSTORE ADD USER <OGG_OWNER>@<CDB_SOURCE_SID>,
     PASSWORD <PASSWORD>, ALIAS <OGG_SOURCE_ALIAS>, DOMAIN <OGG_DOMAIN>
GGSCI>INFO CREDENTIALSTORE DOMAIN <OGG_DOMAIN>
GGSCI>DBLOGIN USERIDALIAS <OGG_SOURCE_ALIAS> DOMAIN <OGG_DOMAIN>
```

- Install the DMU SYS.DBMS_DUMA_INTERNAL package for the source database

```
$export ORACLE_SID=<CDB_SOURCE_SID>
$sqlplus sys/<PASSWORD>@<PDB_SOURCE_SID> as sysdba
SQL>@?/rdbms/admin/prvtdumi.plb
```

- Perform tasks using DMU on the source database
  - Install DMU migration repository
  - Scan the entire database
  - Cleanse any reported convertibility issues
  - Repeat the scanning and cleansing operations until the DMU indicates the database is ready for conversion
  - Generate OGG configuration files using DMU

- Copy the DMU generated configuration files

Copy the generated OGG parameter and script files from the DMU output directory to the OGG installed directory. See Fig. 2 and Fig. 3 above for local and remote Replicat configurations.

- Run the defgen script to generate the source table definitions

Execute the defgen script in the $OGG_HOME directory. The OGG DEFGEN utility generates the output file “dmu_source.def” in the $OGG_HOME/dirdef directory.

Modify the file permission of defgen.sh by running

```
chmod +x defgen.sh
```

It is helpful to monitor ggserr.log for any warning and error.

```
$tail -f $OGG_HOME/ggserr.log
```

» Start the OGG Manger for managing Extract and Replicat processes and files

```
GGSCI>start manager
```

» Start the OGG Extract for capturing committed transactions from the source database

```
GGSCI>obey start_extract
```

» Use RMAN to backup and capture the checkpoint SCN value of the source database

```
$export ORACLE_SID=<CDB_SOURCE_SID>
$rman target /
RMAN>
run {
      backup database root;
      backup database 'PDB$SEED';
      backup pluggable database <PDB_SOURCE_SID> plus archivelog;
}
RMAN>restore database preview summary;
```

» Clone the target database for Initial Load

- Copy the source parameter PFILE to the cloned PFILE

```
$cp $ORACLE_HOME/dbs/init<CDB_SOURCE_SID>.ora
    $ORACLE_HOME/dbs/init<CDB_DUP_SID>.ora
```

- Modify the PFILE for the cloned database

  Replace the occurrences of *<CDB_SOURCE_SID>* with *<CDB_DUP_SID>*, and add db_file_name_convert and log_file_name_convert to the PFILE e.g.

```
db_file_name_convert=(<ORACLE_BASE>/oradata/<CDB_SOURCE_SID>/,
                     <ORACLE_BASE>/oradata/<CDB_DUP_SID>/)
log_file_name_convert=(<ORACLE_BASE>/oradata/<CDB_SOURCE_SID>/,
                     <ORACLE_BASE>/oradata/<CDB_DUP_SID>/)
```

- Create the directories referenced in the cloned PFILE

```
$mkdir -p <ORACLE_BASE>/admin/<CDB_DUP_SID>/adump
$mkdir <ORACLE_BASE>/oradata/<CDB_DUP_SID>
$mkdir <ORACLE_BASE>/oradata/<CDB_DUP_SID>/pdbseed
$mkdir <ORACLE_BASE>/oradata/<CDB_DUP_SID>/<PDB_DUP_SID>
$mkdir <ORACLE_BASE>/fast_recovery_area/<CDB_DUP_SID>
```

- Create the cloned database using RMAN

```
$export ORACLE_SID=<CDB_DUP_SID>
```

```
$sqlplus /as sysdba
SQL>startup force nomount pfile=?/dbs/init<CDB_DUP_SID>.ora
$rman target sys/<PASSWORD>@<CDB_SOURCE_SID> auxiliary /
RMAN>duplicate target database to <CDB_DUP_SID> pluggable database pdb1
        pfile='?/dbs/init<CDB_DUP_SID>.ora';
```

- Comment out the parameters in the cloned PFILE

```
#db_file_name_convert=(<ORACLE_BASE>/oradata/<CDB_SOURCE_SID>/,
                      <ORACLE_BASE>/oradata/<CDB_DUP_SID>/)
#log_file_name_convert=(<ORACLE_BASE>/oradata/<CDB_SOURCE_SID>/,
                       <ORACLE_BASE>/oradata/<CDB_DUP_SID>/)
```

- Restart the cloned database using the cloned PFILE

```
$export ORACLE_SID=<CDB_DUP_SID>
$sqlplus /as sysdba
SQL>shutdown immediate
SQL>startup pfile=?/dbs/init<CDB_DUP_SID>.ora
```

- Unplug the cloned *<PDB_DUP_SID>* from the *<CDB_DUP_SID>*

```
$export ORACLE_SID=<CDB_DUP_SID>
$sqlplus /as sysdba
SQL>alter session set container = cdb$root;
SQL>alter pluggable database <PDB_DUP_SID> CLOSE IMMEDIATE;
SQL>alter pluggable database <PDB_DUP_SID> unplug into '<PDB_FILE_XML>';
SQL>drop pluggable database <PDB_DUP_SID> keep datafiles;
```

- Plug-in the *<PDB_DUP_SID>* into the *<CDB_TARGET_SID>* using the unplugged data

```
$export ORACLE_SID=<CDB_TARGET_SID>
$sqlplus / as sysdba
SQL>create pluggable database <PDB_TARGET_SID> using
       '<PDB_FILE_XML>' nocopy tempfile reuse;
SQL>alter pluggable database <PDB_TARGET_SID> open;
```

- Create a password file for the target database

```
$cd $ORACLE_HOME/dbs
$orapwd file=orapw<CDB_TARGET_SID> password=<PASSWORD> entries=<NUM_ENTRIES>
```

- Modify listener.ora and tnsnames.ora to include *<CDB_TARGET_SID>* and *<PDB_TARGET_SID>* and restart the listener

```
$lsnrctl reload
```

- Install the DMU SYS.DBMS_DUMA_INTERNAL package for the target database

```
$export ORACLE_SID=<CDB_TARGET_SID>
```

```
$sqlplus sys/<PASSWORD>@<PDB_TARGET_SID> as sysdba
SQL>@?/rdbms/admin/prvtdumi.plb
```

» Use DMU to convert the target database character set to Unicode. Make sure that there is no process connected to the target database. You may need to scan OGG schema on the target database.

» Include the GoldenGate parameters in the target database PFILE e.g.

```
*.streams_pool_size=1280M             # minimum 1280M
*.enable_goldengate_replication=TRUE  # for DB 11.2.0.4 and 12.1.0.2
```

» Restart the target database and create the target SPFILE

```
$export ORACLE_SID=<CDB_TARGET_SID>
$sqlplus / as sysdba
SQL>shutdown immediate
SQL>startup pfile=?/dbs/init<CDB_TARGET_SID>.ora
SQL>create spfile from pfile='?/dbs/init<CDB_TARGET_SID>.ora';
```

» Create a Golden Gate common user (e.g. C##OGG_OWNER) on the target database

```
$export ORACLE_SID=<CDB_TARGET_SID>
$sqlplus / as sysdba
/*
  Need to delete <OGG_OWNER> user in <PDB_TARGET_SID> which is inherited from
  <PDB_DUP_SID>.
*/
SQL>alter session set container = <PDB_TARGET_SID>;
SQL>drop user <OGG_OWNER> cascade;
SQL>alter session set container = cdb$root;
SQL>create user <OGG_OWNER> identified by <PASSWORD>
       default tablespace users temporary tablespace temp CONTAINER=ALL;
SQL>grant create session to ogg_owner CONTAINER=ALL;
SQL>grant connect,resource to <OGG_OWNER> CONTAINER=ALL;
SQL>grant select any dictionary, select any table to <OGG_OWNER> ONTAINER=ALL;
SQL>grant all privileges to <OGG_OWNER> CONTAINER=ALL;
SQL>grant create table to <OGG_OWNER> CONTAINER=ALL;
SQL>grant flashback any table to <OGG_OWNER> CONTAINER=ALL;
SQL>grant execute on dbms_flashback to <OGG_OWNER> CONTAINER=ALL;
SQL>grant execute on utl_file to <OGG_OWNER> CONTAINER=ALL;
SQL>alter database add supplemental log data (all) columns;
SQL>exec dbms_goldengate_auth.grant_admin_privilege
     ('<OGG_OWNER>', container=>'ALL');
```

» Add the *<OGG_OWNER>* common user to the OGG CREDENTIALSTORE for *<PDB_TARGET_SID>*

```
GGSCI>ALTER CREDENTIALSTORE ADD USER <OGG_OWNER>@<PDB_TARGET_SID>,
     PASSWORD <PASSWORD>, ALIAS <OGG_TARGET_ALIAS>, DOMAIN <OGG_DOMAIN>
GGSCI>INFO CREDENTIALSTORE DOMAIN <OGG_DOMAIN>
GGSCI>DBLOGIN USERIDALIAS <OGG_TARGET_ALIAS> DOMAIN <OGG_DOMAIN>
```

- Modify the <*SCN value*> of the start_replicat Obey script

  START REPLICAT rep, AFTERCSN <*SCN value*>

  Replace <*SCN value*> with the checkpoint SCN value captured during the RMAN backup step.

- Start the OGG Replicat for applying the committed transactions to the target database

  ```
  GGSCI>obey start_replicat
  ```

- Check transactions are extracted and replicated correctly

  For more information, refer to step 11 above “Ensure that all transactions have been extracted and replicated by Oracle GoldenGate”.

- To stop OGG Manager, Extract and Replicat

  ```
  GGSCI>obey stop_extract
  GGSCI>obey stop_replicat
  GGSCI>stop manager
  ```

Conclusion

In this paper we have presented the setup procedures and detailed steps to perform the near-zero downtime database migration to Unicode for Oracle Database 11*g* and 12*c* using Oracle Database Migration Assistant for Unicode, Oracle GoldenGate, and Oracle Recovery Manager.

The information presented in this document can be adapted as needed for your specific environments.

For further information, see the documents below:

- Near-Zero Downtime Database Migration to Unicode – Overview
- Oracle Database Migration Assistant for Unicode Documentation
- Oracle GoldenGate Documentation

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

blogs.oracle.com/oracle

facebook.com/oracle

twitter.com/oracle

oracle.com

Hardware and Software, Engineered to Work Together

Near-Zero Downtime Database Migration to Unicode - Procedures
June 2015

Oracle is committed to developing practices and products that help protect the environment