Implementing DDL Replication

by Ahmed Baraka

Introduction to Oracle Data Guard
In this lecture, we are going to talk about the basic concepts of Oracle Data Guard

Objectives

By the end of this lecture, you should be able to:

- Understand the requirements and restrictions on implementing DDL replication in Oracle GoldenGate
- Understand the DDL replication scopes
- Set up a DDL replication
- Control the behavior of a DDL replication
- Handling errors generated by DDL replication

About DDL Replication

- DDL synchronization is supported by GoldenGate 12c in Oracle and Teradata platforms
- By default, DDL is disabled in the Extract
- By default, DDL is enabled in the Replicat

DDL Replication Requirements

- Identical source and target data definitions: (`AssumeTargetDefs` is required for prior-12c)
- `Passthru` mode must be configured in Data Pump process
- `WildcardResolve` must be set to `DYNAMIC` (default)
- Oracle 10g requires to disable the Recyclebin

DDL Replication Restrictions

- `ddl_setup.sql` script fails if the tablespace is shared.
- Truncate table DDL is supported. Do not use `GetTruncates`
- `ALTER TABLE RENAME` fails if the old or new table name is longer than 18 characters
- DDL statements should not exceed 4 MB in length
- User exit logic cannot be triggered by DDL operations
- Some object names are excluded
- Some DDL operations are not supported

DDL Replication Restrictions (cont)

- For bi-directional configuration, it supports a configuration with only two databases
- Restrictions on DDL operations on user-defined types and LOB data:
  - Make sure all DMLs are applied before issuing the DDL

DDL Replication Setup

- DDL setup scripts included in Oracle GoldenGate.
- Setup scripts uses Trigger-based DDL Capture, needed when:
  - Database 11.2.0.3 or earlier
  - Processes running in Classic mode
- DDL Trigger must be enabled in the conditions above

```
marker_setup.sql
ddl_setup.sql
role_setup_.sql
ddl_enable.sql
```

DDL Replication Setup Procedure

1. Grant execute on `utl_file`
2. Dedicate a tablespace for the user
3. Specify the `GGSCHEMA` in the GLOBALS file
4. Run the setup scripts as SYSDBA (if needed)

DDL Scopes

- Mapped
- UnMapped
- OTHER

DDL Scopes: Mapped

- Mapped


```
# Extract
TABLE hr.emp;

# Replicat
MAP hr.emp, TARGET sales.employees;

# The command in the source:
ALTER TABLE hr.emp ADD notes VARCHAR2 (200);

# The command executed by the Replicat
ALTER TABLE sales.employees ADD notes VARCHAR2 (200);
```

DDL Scopes: UnMapped

- UnMapped
  - Does not have a Table or Map statement


```
CREATE TABLE report0134
(
 branch_id VARCHAR2(30),
 amount number(10)
);
```

DDL Scope: OTHER

- OTHER
  - Table or Map statements do not apply
  - DDL operations like: `CREATE USER, CREATE ROLE, ALTER TABLESPACE`

About the DDL Parameter

- Enables DDL support and filters the operations


```
DDL [
 {INCLUDE | EXCLUDE}
 [, MAPPED | UNMAPPED | OTHER | ALL]
 [, OPTYPE type]
 [, OBJTYPE 'type']
 [, OBJNAME name]
 [, INSTR 'string']
 [, INSTRCOMMENTS 'comment_string'] ] [...]
```

DDL Parameter Examples

```
DDL INCLUDE ALL
```


```
DDL INCLUDE UNMAPPED &
OPTYPE alter &
OBJTYPE 'table' &
OBJNAME HR.REPORT* &
INCLUDE MAPPED OBJNAME * &
EXCLUDE MAPPED OBJNAME TEMPORARY.TAB
```

DDL String Substitution

- `DDLSUBST` parameter substitutes strings in a DDL operation
- Can be used multiple times in a parameter file

```
DDLSUBST 'search_string' WITH 'replace_string'
[INCLUDE inclusion_clause | EXCLUDE exclusion_clause]
```

```
DDLSubst 'EMP' With 'EMPLOYEES'
Include All ObjType 'table' ObjName FIN.*
```

About the `DDLOptions` Parameter

```
DDLOptions
 [, AddTranData [ABEND | RetryOp <RetryDelay <seconds>
    MaxRetries <retries>]
 [, DefaultUserPassword <password> <encrypt_type>
 [EncryptKey {Default | <keyname>]}]
 [, GetApplOps | IgnoreApplOps]
 [, GetReplicates | IgnoreReplicates]
 [, MapDerived | NoMapDerived]
 [, MapSessionSchema] <src_schema> Target <trgt_schema>
 [, Password EncryptKey [Default | EncryptKey <keyname>]
 [, RemoveComments {Before | After}]
 [, ReplicatePassword | NoReplicatePassword]
 [, Report | NoReport] [, UpdateMetadata] [, UseOwnerForSession]
```

`DDLOptions` Parameter Options

Option	Description
`AddTranData`	Update supplemental logging for tables created/altered
`DefaultUserPassword`	Change the password of Create|Alter User statements
`GETAPPLOPS` | `IGNOREAPPLOPS`	Controls whether or not the Extract should capture the DDLs executed by the business applications
`GETREPLICATES` | `IGNOREREPLICATES`	Controls whether or not DDL operations produced by the Replicat should be captured
`MapSessionSchema`	Map a source session schema to a different session schema on the target

`DDLOptions` Parameter Options (cont)

Option	Description
`Password EncryptKey`	Instruct the Extract to encrypt all passwords in the source DDL before writing the DDL to the trail
`Report | NoReport`	Whether or not to write expanded DDL statistics on report file
`UpdateMetadata`	Instructs the Replicat to update its object metadata cache

Using `MapSessionSchema` Example

- Extract:


```
DDL INCLUDE OBJNAME SRC1.* INCLUDE OBJNAME SRC2.*
TABLE SRC1.*; TABLE SRC2.*;
```

- Replicate:

```
DDLOPTIONS MAPSESSIONSCHEMA SRC1 TARGET DST1
DDLOPTIONS MAPSESSIONSCHEMA SRC2 TARGET DST2
MAP SRC1.*, TARGET DST1.*;
MAP SRC2.*, TARGET DST2.*;
DDL INCLUDE OBJNAME DST1.* INCLUDE OBJNAME DST2.*
```

DDL Error Handling

- Use `DDLError` parameter to handle anticipated DDL errors

```
DDLError [RestartSkip <num_skips>]
         [SkipTriggerError <num_errors>]
```

```
DDLError
 {<error> | Default} {<response>}
 [RetryOp MaxRetries <n> [RetryDelay <delay>]]
 {Include <clause> | Exclude <clause>}
 [, IgnoreMissingTables | AbendOnMissingTables]
 [, RestartCollisions | NoRestartCollisions]
```

`DDLError` Examples

```
DDLERROR 1234 ABEND INCLUDE OBJNAME tab*
DDLERROR 5678 IGNORE
```

```
DDLERROR 1234 IGNORE RETRYOP MAXRETRIES 3
 RETRYDELAY 10 INCLUDE ALL OBJTYPE TABLE OBJNAME tab*
 EXCLUDE OBJNAME tab1*
```

Filtering DDL Replication

- Filtering with the `DDL` parameter
- Filtering with built-in filter rules
- Filtering with PL/SQL code

- More information in “Installing and Configuring Oracle GoldenGate for Oracle Database 12c”

Summary

In this lecture, you should learnt the following:

- Understand the requirements and restrictions on implementing DDL replication in Oracle GoldenGate
- Understand the DDL replication scopes
- Set up a DDL replication
- Control the behavior of a DDL replication
- Handling errors generated by DDL replication