Converged Database Continuous Availability Workshop

Part II - Active Data Guard

Contents

Initial requirements

- SSH private key to Access the database server in the cloud. This private key is provided along with this manual.
- SSH client app, to login to the database servers
- Database servers public IP

Active Data Guard

Use Data Guard Broker

In this first part, we are going to review and use Data Guard Broker.

Data Guard Broker is the recommended tool to use in order to manage a Data Guard configuration.

First, gain access to the database nodes of your Data Guard configuration. You need to use the private key provided in this workshop, along with the public IP of your database servers.

Access to the primary database server as "**opc**" using ssh. Then connect as user "oracle" and run the following commands:

```
ssh -i privateKey opc@<public ip of node 1>

## Connect as "oracle" user and show some database instance parameters

sudo su - oracle

## Review the broker configuration:

sqlplus / as sysdba

show parameter dg_broker

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
dg_broker_config_file1               string      +DATA/ADG_FRA22S/dr1adg_fra22s
                                                 .dat
dg_broker_config_file2               string      +DATA/ADG_FRA22S/dr2adg_fra22s
                                                 .dat
dg_broker_start                      boolean     TRUE

## The broker backups its configuration in two files, for HA purposes.
## Review important parameters for Data Guard

show parameter log_archive_config

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
log_archive_config                   string      dg_config=(adg_fra22s,adg_fra34x)

## adg_fra22s and adg_fra34x are the unique names of the primary and standby databases.
Your own values will be different. Write them down, as they will be used extensively
during this workshop

show parameter log_archive_dest_2

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
log_archive_dest_2                   string      service="adg_fra34x", ASYNC NO
                                                 AFFIRM delay=0 optional compre
                                                 ssion=disable max_failure=0 re
                                                 open=300 db_unique_name="adg_f
                                                 ra269" net_timeout=30, valid_f
                                                 or=(online_logfile,all_roles)
```

```
## log_archive_dest_2 has been configured during the Data Guard creation, to enable
redolog propagation. Pay attention to "ASYNC NO AFFIRM", indicating that the configured
protection mode is "MAX PERFORMANCE"

SQL> show parameter standby_file

NAME                                 TYPE     VALUE
------------------------------------ -------- ------------------------------
standby_file_management              string   AUTO

## standby_file_management=AUTO ensures that any new datafile created on the primary
database will be automatically created on the standby database !!!
```

Connect to Data Guard Broker command line interface, and show the Data Guard configuration:

```
dgmgrl
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL>

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 10 seconds ago)

## As commented before, we can see the protection mode is configured to "MAX
PERFORMANCE".

--- Show database properties !!!

DGMGRL> show database verbose adg_fra22s

Database - adg_fra22s

  Role:               PRIMARY
  Intended State:     TRANSPORT-ON
  Instance(s):
    adg

  Properties:
    DGConnectIdentifier             = 'adg_fra22s'
    ObserverConnectIdentifier       = ''
    FastStartFailoverTarget         = ''
    PreferredObserverHosts          = ''
    LogShipping                     = 'ON'
```

```
    RedoRoutes                      = ''
    LogXptMode                      = 'ASYNC'
    DelayMins                       = '0'
    Binding                         = 'optional'
    MaxFailure                      = '0'
    ReopenSecs                      = '300'
    NetTimeout                      = '30'
    RedoCompression                 = 'DISABLE'
    PreferredApplyInstance          = ''
    ApplyInstanceTimeout            = '0'
    ApplyLagThreshold               = '30'
    TransportLagThreshold           = '30'
    TransportDisconnectedThreshold  = '30'
    ApplyParallel                   = 'AUTO'
    ApplyInstances                  = '0'
    StandbyFileManagement           = ''
    ArchiveLagTarget                = '0'
    LogArchiveMaxProcesses          = '0'
    LogArchiveMinSucceedDest        = '0'
    DataGuardSyncLatency            = '0'
    LogArchiveTrace                 = '0'
    LogArchiveFormat                = ''
    DbFileNameConvert               = ''
    LogFileNameConvert              = ''
    ArchiveLocation                 = ''
    AlternateLocation               = ''
    StandbyArchiveLocation          = ''
    StandbyAlternateLocation        = ''
    InconsistentProperties          = '(monitor)'
    InconsistentLogXptProps         = '(monitor)'
    LogXptStatus                    = '(monitor)'
    SendQEntries                    = '(monitor)'
    RecvQEntries                    = '(monitor)'
    HostName                        = 'adgdb-s01-2021-11-22-170552'
    StaticConnectIdentifier         = '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=adgdb-
s01-2021-11-22-
170552)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=adg_fra22s_DGMGRL.pub.racdblab.oraclevcn.
com)(INSTANCE_NAME=adg)(SERVER=DEDICATED)))'
    TopWaitEvents                   = '(monitor)'
    SidName                         = '(monitor)'

  Log file locations:
    Alert log               :
/u01/app/oracle/diag/rdbms/adg_fra22s/adg/trace/alert_adg.log
    Data Guard Broker log   : /u01/app/oracle/diag/rdbms/adg_fra22s/adg/trace/drcadg.log

Database Status:
SUCCESS
```

Use Data Guard Broker to change the protection mode to "MAX AVAILABILITY":

```
DGMGRL> EDIT DATABASE 'adg_fra22s' SET PROPERTY LogXptMode='SYNC';
Property "logxptmode" updated

DGMGRL> show database 'adg_fra22s' 'LogXptMode';
```

```
  LogXptMode = 'SYNC'

DGMGRL> show database 'adg_fra34x' 'LogXptMode';
  LogXptMode = 'ASYNC'
DGMGRL> EDIT DATABASE 'adg_fra34x' SET PROPERTY LogXptMode='SYNC';
Property "logxptmode" updated
DGMGRL> show database 'adg_fra34x' 'LogXptMode';
  LogXptMode = 'SYNC'

--- Now that log transport mode is configured to SYNC, we can apply "MAX AVAILABILITY"
mode:

DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MAXAVAILABILITY;
Succeeded.

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxAvailability
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 44 seconds ago)

## In "MAX AVAILABILITY" protection mode, the redolog propagation is synchronous. Should
something happen that prevents this synchronous propagation, and after a timeout, DG
Broker will automatically change the protection mode to "MAX PERFORMANCE" (ASYNC
propagation) after a timeout, until the problem is solved. DG Broker will check at
regular intervals whenever the issue has been solved, and change automatically the
protection mode back to "MAX AVAILABILITY" when possible.

## Configure the protection mode back to MAX PERFORMANCE

DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MAXPERFORMANCE;
Succeeded.
DGMGRL> EDIT DATABASE 'adg_fra34x' SET PROPERTY LogXptMode='ASYNC';
Property "logxptmode" updated
DGMGRL> EDIT DATABASE 'adg_fra22s' SET PROPERTY LogXptMode='ASYNC';
Property "logxptmode" updated
DGMGRL> show database 'adg_fra22s' 'LogXptMode';
  LogXptMode = 'ASYNC'
DGMGRL> show database 'adg_fra34x' 'LogXptMode';
  LogXptMode = 'ASYNC'

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database
```

```
Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 21 seconds ago)
```

Now we will perform a switchover maneuver. Switchover switches the standby and primary roles between the databases. It can be run from either the primary or the standby server:

```
--- From the primary site, run the switchover

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 57 seconds ago)

-- Ensure switchover is possible: this is an optional step, as DG Broker will take care
of this even if you don't execute this command:

DGMGRL> validate database adg_fra34x

  Database Role:     Physical standby database
  Primary Database:  adg_fra22s

  Ready for Switchover:  Yes
  Ready for Failover:    Yes (Primary Running)

  Managed by Clusterware:
    adg_fra22s:  YES
    adg_fra34x:  YES

DGMGRL> switchover to adg_fra34x
Performing switchover NOW, please wait...
Operation requires a connection to database "adg_fra34x"
Connecting ...
Connected to "adg_fra34x"
Connected as SYSDBA.
New primary database "adg_fra34x" is opening...
Oracle Clusterware is restarting database "adg_fra22s" ...
Connected to "adg_fra22s"
Connected to "adg_fra22s"
Switchover succeeded, new primary is "adg_fra34x"

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
```

```
  Members:
  adg_fra34x - Primary database
    adg_fra22s - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 119 seconds ago)

DGMGRL> validate database adg_fra22s

  Database Role:     Physical standby database
  Primary Database:  adg_fra34x

  Ready for Switchover:  Yes
  Ready for Failover:    Yes (Primary Running)

  Managed by Clusterware:
    adg_fra34x:  YES
    adg_fra22s:  YES
```

In order to illustrate that switchover can be issued either from the primary or the standby site, stay on the new standby site and run a switchover command. This will do a switchback operation:

```
--- Illustrating that a switchover can be issued from either the primary or the standby

DGMGRL> switchover to adg_fra22s
Performing switchover NOW, please wait...
Operation requires a connection to database "adg_fra22s"
Connecting ...
Connected to "adg_fra22s"
Connected as SYSDBA.
New primary database "adg_fra22s" is opening...
Oracle Clusterware is restarting database "adg_fra34x" ...
Connected to "adg_fra34x"
Connected to "adg_fra34x"
Switchover succeeded, new primary is "adg_fra22s"

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 28 seconds ago)

DGMGRL> validate database adg_fra34x
```

```
  Database Role:     Physical standby database
  Primary Database:  adg_fra22s

  Ready for Switchover:  Yes
  Ready for Failover:    Yes (Primary Running)

  Managed by Clusterware:
    adg_fra22s:  YES
    adg_fra34x:  YES

## Show the transport and apply lags on the standby database: both should be near zero
when everything works fine.

DGMGRL> show database adg_fra34x

Database - adg_fra34x

  Role:               PHYSICAL STANDBY
  Intended State:     APPLY-ON
  Transport Lag:      0 seconds (computed 0 seconds ago)
  Apply Lag:          0 seconds (computed 0 seconds ago)
  Average Apply Rate: 5.00 KByte/s
  Real Time Query:    ON
  Instance(s):
    adg

Database Status:
SUCCESS
```

Now we will do a failover.
Failover is an emergency maneuver, you'll need to perform it when your primary database crashed, or more generally when a problem prevents the primary database to be accessed by the end users.
A failover maneuver MUST be issued from the standby site.
It's a destructive operation, meaning that the former primary will need to be re-instated. Re-instate can be performed from DG Broker, but for that "Flashback database" must be activated on both databases.

On the primary and the standby database, check that "flashback database" is ON:

```
-- ssh to the primary server as opc, and gain access to "oracle" user:

sudo su - oracle
sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Nov 23 11:08:51 2021
Version 19.12.0.0.0

Copyright (c) 1982, 2021, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c EE Extreme Perf Release 19.0.0.0.0 - Production
Version 19.12.0.0.0

SQL> select open_mode, database_role, flashback_on from v$database;
```

```
OPEN_MODE         DATABASE_ROLE    FLASHBACK_ON
------------------- ---------------- ------------------
READ WRITE        PRIMARY       YES


-- ssh to the standby server as opc, and gain access to "oracle" user:

sudo su - oracle

sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Nov 23 11:09:59 2021
Version 19.12.0.0.0

Copyright (c) 1982, 2021, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c EE Extreme Perf Release 19.0.0.0.0 - Production
Version 19.12.0.0.0

SQL> select open_mode, database_role, flashback_on from v$database;

OPEN_MODE         DATABASE_ROLE    FLASHBACK_ON
------------------- ---------------- ------------------
READ ONLY WITH APPLY PHYSICAL STANDBY YES
```

To simulate a crash on the primary database, we will take it down explicitely.

```
-- On the primary server, as user "oracle", abort the instance:

srvctl stop database -d $(srvctl config database) -o abort

-- Check the database status:

srvctl status database -d $(srvctl config database)

Instance adg is not running on node adgdb-s01-2021-11-22-170552
```

Connect to the standby server and gain access to the "oracle" user. Prepare to perform a failover.

```
sudo su - oracle

[oracle@adgsby ~]$ dgmgrl
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Tue Nov 23 11:12:25 2021
Version 19.12.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"
Connected to "adg_fra34x"
Connected as SYSDBA.
DGMGRL> show configuration
```

```
Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    Error: ORA-12514: TNS:listener does not currently know of service requested in
connect descriptor

    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
ERROR   (status updated 0 seconds ago)

## DG Broker cannot access to the primary database

DGMGRL> failover to adg_fra34x
Performing failover NOW, please wait...
Failover succeeded, new primary is "adg_fra34x"

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra34x - Primary database
    Warning: ORA-16857: member disconnected from redo source for longer than specified
threshold

    adg_fra22s - Physical standby database (disabled)
      ORA-16661: the standby database needs to be reinstated

Fast-Start Failover:  Disabled

Configuration Status:
WARNING   (status updated 58 seconds ago)

DGMGRL> validate database adg_fra22s
Error: ORA-16541: member is not enabled

-- The new standby needs to be reinstated.
```

Now we will startup the new standby database, in order to perform a reinstate. In a real crash situation, after a failover, the new standby needs to be restored from a backup.

```
--- On the former primary server, start the database

[oracle@adgdb-s01-2021-11-22-170552 ~]$ srvctl start database -d $(srvctl config
database)
[oracle@adgdb-s01-2021-11-22-170552 ~]$ srvctl status database -d $(srvctl config
database)
Instance adg is running on node adgdb-s01-2021-11-22-170552

--- Then we reinstate, from the new primary site ← ATTENTION!
```

```
[oracle@adgsbydb ~]$ dgmgrl
connect sys/"W3lc0m3#W3lc0m3#W"
DGMGRL> reinstate database adg_fra22s
Reinstating database "adg_fra22s", please wait...
Reinstatement of database "adg_fra22s" succeeded

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra34x - Primary database
    adg_fra22s - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 57 seconds ago)

DGMGRL> validate database adg_fra22s

  Database Role:     Physical standby database
  Primary Database:  adg_fra34x

  Ready for Switchover:  Yes
  Ready for Failover:    Yes (Primary Running)

  Managed by Clusterware:
    adg_fra34x:  YES
    adg_fra22s:  YES

DGMGRL> show database adg_fra22s

Database - adg_fra22s

  Role:               PHYSICAL STANDBY
  Intended State:     APPLY-ON
  Transport Lag:      0 seconds (computed 0 seconds ago)
  Apply Lag:          0 seconds (computed 0 seconds ago)
  Average Apply Rate: 21.00 KByte/s
  Real Time Query:    ON
  Instance(s):
    adg

Database Status:
SUCCESS
```

Perform another switchover to get back in the initial situation. This step is optional in "real life", but needed for this lab:

```
-- Perform this step from either the primary or the standby site

DGMGRL> switchover to adg_fra22s
Performing switchover NOW, please wait...
Operation requires a connection to database "adg_fra22s"
Connecting ...
```

```
Connected to "adg_fra22s"
Connected as SYSDBA.
New primary database "adg_fra22s" is opening...
Oracle Clusterware is restarting database "adg_fra34x" ...
Connected to "adg_fra34x"
Connected to "adg_fra34x"
Switchover succeeded, new primary is "adg_fra22s"

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 92 seconds ago)

DGMGRL> show database adg_fra34x

Database - adg_fra34x

  Role:               PHYSICAL STANDBY
  Intended State:     APPLY-ON
  Transport Lag:      0 seconds (computed 0 seconds ago)
  Apply Lag:          0 seconds (computed 0 seconds ago)
  Average Apply Rate: 43.00 KByte/s
  Real Time Query:    ON
  Instance(s):
    adg

Database Status:
SUCCESS
```

This concludes the Data Guard Broker section.

Active Data Guard and TAC

In the following section, we will demonstrate that TAC extends application protection beyond RAC, with a single instance database protected by an Active Data Guard.

Connect to the primary server and check the Data Guard configuration:

```
[oracle@adgdb-s01-2021-11-22-170552 ~]$ dgmgrl
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Tue Nov 23 11:26:40 2021
Version 19.12.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> show configuration
```

```
Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 51 seconds ago)
```

Install ACDemo application on the primary node.

Connect to the primary node and gain access to the "oracle" user:

```
sudo su - oracle
cd /home/oracle

wget https://objectstorage.us-ashburn-
1.oraclecloud.com/p/O8AOujhwl1dSTqhfH69f3nkV6TNZWU3KaIF4TZ-XuCaZ5w-
xHEQ14ViOVhUXQjPB/n/oradbclouducm/b/LiveLabTemp/o/ACDemo_19c.zip

--2021-11-23 11:27:42--  https://objectstorage.us-ashburn-
1.oraclecloud.com/p/O8AOujhwl1dSTqhfH69f3nkV6TNZWU3KaIF4TZ-XuCaZ5w-
xHEQ14ViOVhUXQjPB/n/oradbclouducm/b/LiveLabTemp/o/ACDemo_19c.zip
Resolving objectstorage.us-ashburn-1.oraclecloud.com (objectstorage.us-ashburn-
1.oraclecloud.com)... 134.70.24.1, 134.70.32.1, 134.70.28.1
Connecting to objectstorage.us-ashburn-1.oraclecloud.com (objectstorage.us-ashburn-
1.oraclecloud.com)|134.70.24.1|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 8573765 (8.2M) [application/x-zip-compressed]
Saving to: 'ACDemo_19c.zip'

100%[=================================================================================
=============>] 8,573,765   10.3MB/s   in 0.8s

2021-11-23 11:27:43 (10.3 MB/s) - 'ACDemo_19c.zip' saved [8573765/8573765]

[oracle@adgdb-s01-2021-11-22-170552 ~]$ ls -ltr
total 8376
-rw-r--r-- 1 oracle oinstall 8573765 Sep 10 04:21 ACDemo_19c.zip

## Unzip the application

[oracle@adgdb-s01-2021-11-22-170552 ~]$ unzip ACDemo_19c.zip
Archive:  ACDemo_19c.zip
   creating: acdemo/
  inflating: acdemo/ac_noreplay.sbs
  inflating: acdemo/ac_replay.sbs
  inflating: acdemo/ac_setup_sql.sbs
  inflating: acdemo/build.xml
   creating: acdemo/classes/
   creating: acdemo/classes/acdemo/
  inflating: acdemo/classes/acdemo/ACDemo.class
```

```
  inflating: acdemo/classes/acdemo/PrintACStatThread.class
  inflating: acdemo/classes/acdemo/PrintStatThread.class
  inflating: acdemo/classes/acdemo/Worker.class
  inflating: acdemo/kill_session.sbs
  inflating: acdemo/lbtest.sbs
   creating: acdemo/lib/
  inflating: acdemo/lib/acdemo.jar
  inflating: acdemo/lib/ojdbc8-19.10.0.0.jar
  inflating: acdemo/lib/ons-19.10.0.0.jar
  inflating: acdemo/lib/oraclepki-19.10.0.0.jar
  inflating: acdemo/lib/orai18n-19.10.0.0.jar
  inflating: acdemo/lib/orajsoda-1.1.4.jar
  inflating: acdemo/lib/osdt_cert-19.10.0.0.jar
  inflating: acdemo/lib/osdt_core-19.10.0.0.jar
  inflating: acdemo/lib/ucp-19.10.0.0.jar
 extracting: acdemo/MANIFEST.MF
  inflating: acdemo/my_setup.sql
  inflating: acdemo/README.txt
  inflating: acdemo/runlbtest
  inflating: acdemo/runnoreplay
  inflating: acdemo/runreplay
  inflating: acdemo/runtacreplay
   creating: acdemo/src/
   creating: acdemo/src/acdemo/
  inflating: acdemo/src/acdemo/ACDemo.java
  inflating: acdemo/src/acdemo/PrintACStatsThread.java
  inflating: acdemo/src/acdemo/PrintStatThread.java
  inflating: acdemo/src/acdemo/Worker.java
  inflating: acdemo/tac_replay.sbs
   creating: acdemo/win/
  inflating: acdemo/win/ac_noreplay.install
  inflating: acdemo/win/ac_noreplay.properties
  inflating: acdemo/win/ac_replay.install
  inflating: acdemo/win/create_hr.sql
  inflating: acdemo/win/hr_tab.sql
  inflating: acdemo/win/replace.bat
  inflating: acdemo/win/replace.vbs
  inflating: acdemo/win/SETUP_AC_TEST.bat
  inflating: README.txt
  inflating: SETUP_AC_TEST.sh
```

Create a new database service on the primary and the standby database.

The service will be associated to the primary role, to ensure that it is active only at the primary site.

The following commands must be run on both the primary and standby database:

```
## On the primary server

[oracle@adgdb-s01-2021-11-22-170552 ~]$ srvctl add service -d $(srvctl config database)
-s svc_tac -pdb PDB1 -role primary -replay_init_time 1800 -failoverretry 30 -
failoverdelay 3 -commit_outcome TRUE -failovertype AUTO -failover_restore AUTO

[oracle@adgdb-s01-2021-11-22-170552 ~]$
[oracle@adgdb-s01-2021-11-22-170552 ~]$ srvctl start service -d $(srvctl config
database) -s svc_tac
```

```
[oracle@adgdb-s01-2021-11-22-170552 ~]$ srvctl status service -d $(srvctl config
database) -s svc_tac

Service svc_tac is running on instance(s) adg

[oracle@adgdb-s01-2021-11-22-170552 ~]$ srvctl config service -d $(srvctl config
database) -s svc_tac
Service name: svc_tac
Server pool:
Cardinality: 1
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: true
Failover type: AUTO
Failover method:
Failover retries: 30
Failover delay: 3
Failover restore: AUTO
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: PDB1
Hub service:
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 1000 seconds
Drain timeout:
Stop option:
Session State Consistency: AUTO
GSM Flags: 0
Service is enabled
Preferred instances: adg
Available instances:
CSS critical: no
Service uses Java: false

## On the standby server
## On the standby database, we don't start the service
## It will be started automatically when the database assumes the primary role

[oracle@adgsby ~]$ srvctl add service -d $(srvctl config database) -s svc_tac -pdb PDB1
-role primary -replay_init_time 1800 -failoverretry 30 -failoverdelay 3 -commit_outcome
TRUE -failovertype AUTO -failover_restore AUTO
[oracle@adgsby ~]$ srvctl status service -d $(srvctl config database) -s svc_tac
Service svc_tac is not running.
```

On the primary database, create HR schema and populate some tables.

Connect to the primary server and gain access to "oracle" user:

```
## Connect to the PDB as system
```


```
sudo su - oracle

# We can get an example of the service fully qualified name from the existing entries of
tnsnames.ora file. We can get the server private IP from the same file as well, if
needed.

cat $ORACLE_HOME/network/admin/tnsnames.ora

# In the following command, we use the fully qualified service name.

sqlplus system/"W3lc0m3#W3lc0m3#W"@10.0.0.107:1521/svc_tac.pub.adgdblab.oraclevcn.com

   set heading off
   set feedback off
   drop user hr cascade;
   create user hr identified by "W3lc0m3#W3lc0m3#W" default tablespace USERS temporary
tablespace temp;
   grant connect, create session, resource to hr;
   alter user hr quota unlimited on USERS;

   create table HR.emp4AC(
    empno number(4) not null,
    ename varchar2(10),
    job char(9),
    mgr number(4),
    hiredate date,
    sal number(7,2),
    comm number(7,2),
    deptno number(2),
    constraint emp_primary_key primary key (empno));

   insert into hr.emp4AC values(7839,'KING','PRESIDENT',NULL,'17-NOV-81',50000,NULL,10);
   insert into hr.emp4AC values(7698,'BLAKE','MANAGER',NULL,'17-NOV-81',8000,NULL,10);
   insert into hr.emp4AC values(7782,'CLARK','MANAGER',NULL,'17-NOV-81',8000,NULL,10);
   insert into hr.emp4AC values(7566,'JONES','MANAGER',NULL,'17-NOV-81',8000,NULL,10);
   insert into hr.emp4AC values(7654,'MARTIN','SALESMAN',NULL,'17-NOV-81',7000,NULL,10);
   insert into hr.emp4AC values(7499,'ALLEN','MANAGER',NULL,'17-NOV-81',9000,NULL,10);
   insert into hr.emp4AC values(7844,'TURNER','CLERK',NULL,'17-NOV-81',5000,NULL,10);
   insert into hr.emp4AC values(7900,'JAMES','MANAGER',NULL,'17-NOV-81',9000,NULL,10);
   insert into hr.emp4AC values(7521,'WARD','PRGRMMER',NULL,'17-NOV-81',9000,NULL,10);
   insert into hr.emp4AC values(7902,'FORD','SALESMAN',NULL,'17-NOV-81',7000,NULL,10);
   insert into hr.emp4AC values(7369,'SMITH','PRGRMMER',NULL,'17-NOV-81',8000,NULL,10);
   insert into hr.emp4AC values(7788,'SCOTT','CLERK',NULL,'17-NOV-81',6000,NULL,10);
   insert into hr.emp4AC values(7876,'ADAMS','PRGRMMER',NULL,'17-NOV-81',7000,NULL,10);
   insert into hr.emp4AC values(7934,'MILLER','SALESMAN',NULL,'17-NOV-81',9000,NULL,10);
   commit;
```

Before running the application, configure a "tac_replay.properties" file.
Edit the file and change the connect string according to your own values. You can use the hostnames already prepared in the /etc/hosts file.
The connect string is formatted for ADG failover.

```
[oracle@adgdb-s01-2021-11-22-170552 acdemo] cd /home/oracle/acdemo
```


```
[oracle@adgdb-s01-2021-11-22-170552 acdemo]$ cp  /home/oracle/acdemo/tac_replay.sbs
/home/oracle/acdemo/tac_replay.properties

[oracle@adgdb-s01-2021-11-22-170552 acdemo]$ vi
/home/oracle/acdemo/tac_replay.properties



# Stub file to create tac_replay.properties
# Use replay datasource
datasource=oracle.jdbc.replay.OracleDataSourceImpl

# Set verbose mode
VERBOSE=FALSE

# database JDBC URL

url=jdbc:oracle:thin:@(DESCRIPTION=(CONNECT_TIMEOUT=90)(RETRY_COUNT=50)(RETRY_DELAY=3)(T
RANSPORT_CONNECT_TIMEOUT=3)(ADDRESS_LIST = (FAILOVER = ON) (LOAD_BALANCE = OFF)(ADDRESS
= (PROTOCOL = TCP)(HOST = adgdb-scan.pub.adgdblab.oraclevcn.com)(PORT = 1521))(ADDRESS =
(PROTOCOL = TCP)(HOST = adgsbydb.pub.adgdblab.oraclevcn.com)(PORT = 1521)))(CONNECT_DATA
=(SERVICE_NAME = svc_tac.pub.adgdblab.oraclevcn.com)))


# database username and password:
username=hr
password=W3lc0m3#W3lc0m3#W

# Enable FAN
fastConnectionFailover=TRUE

#Disable connection tests
validateConnectionOnBorrow=TRUE

# number of connections in the UCP''s pool:
ucp_pool_size=20

#Connection Wait Timeout for busy pool
connectionWaitTimeout=5

# number of active threads (this simulates concurrent load):
number_of_threads=10

# think time is how much time the threads will sleep before looping:
thread_think_time=50
```

Run the application.

```
cd /home/oracle/acdemo
chmod +x ./runtacreplay

[oracle@adgdb-s01-2021-11-22-170552 acdemo]$ ./runtacreplay
#########################################################
# of Threads             : 10
 UCP pool size            : 20
FCF Enabled:  true
VCoB Enabled: true
```

```
ONS Configuration:  null
Enable Intensive Wload:  false
Thread think time        : 50 ms
#########################################################

Starting the pool now... (please wait)
Pool is started in 12385ms
10 borrowed, 0 pending, 2ms getConnection wait, TotalBorrowed 278, avg response time
from db 66ms
5 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 810, avg response time from
db 38ms
4 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 1371, avg response time
from db 33ms
[...]
```

In another terminal, connect to the primary or the standby node and perform a switchover with Data Guard Broker:

```
[oracle@adgsby ~]$ dgmgrl
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Tue Nov 23 11:46:35 2021
Version 19.12.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"
Connected to "adg_fra34x"
Connected as SYSDBA.
DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 22 seconds ago)

DGMGRL> switchover to adg_fra34x
Performing switchover NOW, please wait...
New primary database "adg_fra34x" is opening...
Oracle Clusterware is restarting database "adg_fra22s" ...
Connected to "adg_fra22s"
Connected to "adg_fra22s"
Switchover succeeded, new primary is "adg_fra34x"
DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra34x - Primary database
```

```
    adg_fra22s - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 124 seconds ago)

## Observe the application output

2 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 4560, avg response time
from db 28ms
9 borrowed, 1 pending, 0ms getConnection wait, TotalBorrowed 5193, avg response time
from db 25ms
3 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 5779, avg response time
from db 30ms
1 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 6388, avg response time
from db 27ms
4 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 7007, avg response time
from db 26ms
10 borrowed, 0 pending, 0ms getConnection wait, TotalBorrowed 7635, avg response time
from db 25ms
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718, avg response time
from db 43ms
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
0 borrowed, 10 pending, 0ms getConnection wait, TotalBorrowed 7718
4 borrowed, 0 pending, 95ms getConnection wait, TotalBorrowed 8034, avg response time
from db 28ms
3 borrowed, 0 pending, 88ms getConnection wait, TotalBorrowed 8631, avg response time
from db 29ms
1 borrowed, 0 pending, 82ms getConnection wait, TotalBorrowed 9290, avg response time
from db 21ms
4 borrowed, 0 pending, 77ms getConnection wait, TotalBorrowed 9856, avg response time
from db 33ms
5 borrowed, 0 pending, 73ms getConnection wait, TotalBorrowed 10486, avg response time
from db 25ms
1 borrowed, 0 pending, 68ms getConnection wait, TotalBorrowed 11135, avg response time
from db 22ms
0 borrowed, 0 pending, 64ms getConnection wait, TotalBorrowed 11803, avg response time
from db 20ms
5 borrowed, 4 pending, 61ms getConnection wait, TotalBorrowed 12443, avg response time
from db 24ms
1 borrowed, 0 pending, 58ms getConnection wait, TotalBorrowed 13120, avg response time
from db 19ms
0 borrowed, 0 pending, 55ms getConnection wait, TotalBorrowed 13787, avg response time
from db 20ms
```

```
5 borrowed, 1 pending, 52ms getConnection wait, TotalBorrowed 14470, avg response time
from db 19ms
4 borrowed, 0 pending, 50ms getConnection wait, TotalBorrowed 15141, avg response time
from db 20ms
6 borrowed, 0 pending, 48ms getConnection wait, TotalBorrowed 15819, avg response time
from db 19ms
```

The app froze during the switchover, but no error was reported (in red).

Stop the application typing CTRL-C, several times if needed.

Check the service on the new primary:

```
[oracle@adgsby ~]$ srvctl status service -d $(srvctl config database) -s svc_tac
Service svc_tac is running on instance(s) adg
```

Create a Fast Start Failover configuration

In the following steps, we will perform some installation and configuration steps, in order to enable Fast Start Failover (FSFO).
Three observer servers have been provisioned in your environment, with their corresponding public and private IPs.

Write down their public and private IP, as you will need them to ssh those servers. For example:

```
dgobserver1: 130.61.238.113/10.0.0.184   - ssh -i privateKey opc@130.61.238.113
dgobserver2: 130.61.212.66/10.0.0.76   - ssh -i privateKey opc@130.61.212.66
dgobserver3: 129.159.254.197/10.0.0.8 - ssh -i privateKey opc@129.159.254.197
```

On each server, an "oracle" OS user has been created, and the Oracle Client Software has been previously installed.
Connect to any observer host and check:

```
ssh -i privateKey opc@<public IP>

[opc@dgobserver1 ~]$ sudo su - oracle
Last login: Wed Jul  7 09:42:03 GMT 2021 on pts/2
[oracle@dgobserver1 ~]$ which dgmgrl
/u01/app/oracle/client/bin/dgmgrl
```

Modify /u01/app/oracle/client/network/admin/tnsnames.ora with your own values (copy the values from the Primary DB tnsnames.ora):

```
[oracle@dgobserver1 ~]$ cat /u01/app/oracle/client/network/admin/tnsnames.ora
DBSDU_TSE=(DESCRIPTION=(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(ADDRE
SS=(PROTOCOL=TCP)(HOST=<PRIMARY DB HOST IP>)
(PORT=1521))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_NAME=DBSDU_TSE.sub06221433571.skyne
t.oraclevcn.com)(UR=A)))
DBSDU_FRA2BW=(DESCRIPTION=(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(AD
DRESS=(PROTOCOL=TCP)(HOST=<STANDBY DB HOST IP>)
(PORT=1521))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_NAME=DBSDU_fra2bw.sub06221433571.sk
ynet.oraclevcn.com)(UR=A)))
```

Substitute the green parts by your values:

- DBSDU_TSE: the unique name of your primary database
- DBSDU_FRA2BW: the unique name of your standby database
- "database host 1": private IP of the primary database server
- "database host 2": private IP of the standby database server
- DBSDU_TSE.sub06221433571.skynet.oraclevcn.com: the fully qualified name of the primary CDB service (fully qualified DB unique name).
- DBSDU_fra2bw.sub06221433571.skynet.oraclevcn.com: the fully qualified name of the standby CDB service (fully qualified DB unique name).

For example:

```
adg_fra22s=(DESCRIPTION=(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(ADDR
ESS=(PROTOCOL=TCP)(HOST=10.0.0.180)(PORT=1521))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_
NAME=adg_fra22s.pub.adgdblab.oraclevcn.com)(UR=A)))

adg_fra34x=(DESCRIPTION=(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(ADDR
ESS=(PROTOCOL=TCP)(HOST=10.0.0.109)(PORT=1521))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_
NAME=adg_fra34x.pub.adgdblab.oraclevcn.com)(UR=A)))
```

Test the connections: connect to all the three observer hosts, and check the connections to the databases are working:

```
--- From the observer nodes, test the connections with dgmgrl

ssh -i privateKey opc@<Observer1 public IP>

[opc@dgobserver1 ~]$ sudo su - oracle
Last login: Tue Nov 23 16:10:11 GMT 2021 on pts/0

[oracle@dgobserver1 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 16:14:15 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL>
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra34x as sysdba
Connected to "adg_fra34x"
Connected as SYSDBA.
DGMGRL>exit

## If these checks are OK, then take note of the tnsnames.ora entries, as we will have
to update it on observer2 and observer3 hosts

-- Repeat the same tests on Observer2 and Observer3 hosts

[opc@dgobserver2 ~]$ sudo su - oracle
Last login: Tue Nov 23 16:11:17 GMT 2021 on pts/0

[opc@dgobserver2 ~]$ vi /u01/app/oracle/client/network/admin/tnsnames.ora

#paste the working entries from observer 1

[oracle@dgobserver2 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 16:32:48 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra34x as sysdba
Connected to "adg_fra34x"
```

```
Connected as SYSDBA.
DGMGRL>exit



[opc@dgobserver3 ~]$ vi /u01/app/oracle/client/network/admin/tnsnames.ora

#paste the working entries from observer 1

[opc@dgobserver3 ~]$ sudo su - oracle
Last login: Tue Nov 23 16:12:12 GMT 2021 on pts/0
[oracle@dgobserver3 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 16:31:43 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra34x as sysdba
Connected to "adg_fra34x"
Connected as SYSDBA.
DGMGRL>

--- If these tests are successful, we can proceed !!!
--- If not, we need to fix the errors before proceeding
```

We are ready to enable Fast Start Failover on the Data Guard configuration. From any observer, enable FSFO:

```
[oracle@dgobserver1 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 16:14:15 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - Physical standby database

Fast-Start Failover:  Disabled

Configuration Status:
SUCCESS   (status updated 39 seconds ago)
```

```
-- FSFO is disabled, let's enable it:

DGMGRL> ENABLE FAST_START FAILOVER
Enabled in Potential Data Loss Mode.

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    Warning: ORA-16819: fast-start failover observer not started

    adg_fra34x - (*) Physical standby database

Fast-Start Failover: Enabled in Potential Data Loss Mode

Configuration Status:
WARNING   (status updated 34 seconds ago)
```

FSFO is now reported to be enabled in Potential Data Loss mode: this is because our Data Guard protection mode is "MAXIMUM PERFORMANCE" (ASYNC). As redo transport is asynchronous, data might potentially be lost in the case the primary database crashes.

An additional warning is returned: " Warning: ORA-16819: fast-start failover observer not started ": this is because no observer has been started so far, so FSFO is enabled without any observer running. In the following steps, we will start the observers so that they can monitor the Data Guard configuration.

First, configure the Observer connect string property in the Data Guard configuration:

```
--- In the following commands, use your own database unique names !!!

DGMGRL> show database adg_fra22s 'DGConnectIdentifier'
  DGConnectIdentifier = 'adg_fra22s'

DGMGRL> show database adg_fra34x 'DGConnectIdentifier'
  DGConnectIdentifier = 'adg_fra34x'

DGMGRL> edit database adg_fra22s set property 'ObserverConnectIdentifier'='adg_fra22s';
Property "ObserverConnectIdentifier" updated

DGMGRL> edit database adg_fra34x set property 'ObserverConnectIdentifier'='adg_fra34x';
Property "ObserverConnectIdentifier" updated
```

Before starting the observers in background, we need to configure a wallet, so that each observer can connect to both the primary and the standby database.

Configure Wallet from the OBSERVER1 host:

```
 -- Clean-up the /u01/app/oracle/wallet/ directory:

[oracle@dgobserver1 wallet]$ rm /u01/app/oracle/wallet/*
```

```
[oracle@dgobserver1 wallet]$ cd /u01/app/oracle/wallet
[oracle@dgobserver1 wallet]$ ls -ltr
total 0

-- Create a new wallet:

[oracle@dgobserver1 wallet]$ mkstore -wrl /u01/app/oracle/wallet/ -create
Oracle Secret Store Tool Release 21.0.0.0.0 - Production
Version 21.0.0.0.0
Copyright (c) 2004, 2020, Oracle and/or its affiliates. All rights reserved.

Enter password: W3lc0m3#W3lc0m3#W
Enter password again: W3lc0m3#W3lc0m3#W

[oracle@dgobserver1 wallet]$ ls -ltr
total 8
-rw-------. 1 oracle oinstall   0 Nov 23 16:55 ewallet.p12.lck
-rw-------. 1 oracle oinstall 149 Nov 23 16:55 ewallet.p12
-rw-------. 1 oracle oinstall   0 Nov 23 16:55 cwallet.sso.lck
-rw-------. 1 oracle oinstall 194 Nov 23 16:55 cwallet.sso

--- A wallet has been created, protected by "W3lc0m3#W3lc0m3#W" password.
-- Now add a credential for SYS user to that wallet, for both the primary and the
standby database.
-- Use your own database unique names (in green) in the following commands !!!


[oracle@dgobserver1 wallet]$ mkstore -wrl /u01/app/oracle/wallet/ -createCredential
'adg_fra22s' sys W3lc0m3#W3lc0m3#W
Oracle Secret Store Tool Release 21.0.0.0.0 - Production
Version 21.0.0.0.0
Copyright (c) 2004, 2020, Oracle and/or its affiliates. All rights reserved.

Enter wallet password: W3lc0m3#W3lc0m3#W


[oracle@dgobserver1 wallet]$ mkstore -wrl /u01/app/oracle/wallet/ -createCredential
'adg_fra34x' sys W3lc0m3#W3lc0m3#W
Oracle Secret Store Tool Release 21.0.0.0.0 - Production
Version 21.0.0.0.0
Copyright (c) 2004, 2020, Oracle and/or its affiliates. All rights reserved.

Enter wallet password: W3lc0m3#W3lc0m3#W

-- Now check the sqlnet.ora file:

[oracle@dgobserver1] cat /u01/app/oracle/client/network/admin/sqlnet.ora
NAMES.DIRECTORY_PATH= (TNSNAMES, ONAMES, HOSTNAME,EZCONNECT)
WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=/u01/app/oracle/wallet/)))
SQLNET.WALLET_OVERRIDE=TRUE

-- Check that the "DIRECTORY" property points to the directory where you've just
generated the new wallet.
```

Repeat the "wallet configuration" steps on observer2 and observer3 servers. Once the wallet is created on the three observer hosts, we can proceed to start the observers in background. SSH to observer1 machine, connect to dgmgrl and start the observer:

```
[oracle@dgobserver1 wallet]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 17:23:46 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.

## Show the observers currently running on the configuration:

DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

No observers.

## No observer currently running is the expected behavior !!!

DGMGRL> start observer dgobs1 in background file is
'/u01/app/oracle/client/network/admin/fsfo.dat' logfile is
'/u01/app/oracle/client/network/admin/observer_dgobs1.log' connect identifier is
'adg_fra22s';

Connected to "adg_fra22s"
Submitted command "START OBSERVER" using connect identifier "adg_fra22s"
DGMGRL> DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 17:24:09 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
Connected to "adg_fra22s"
Connected as SYSDBA.
Succeeded in opening the observer file "/u01/app/oracle/client/network/admin/fsfo.dat".
[W000 2021-11-23T17:24:12.314+00:00] Observer could not validate the contents of the
observer file.
[W000 2021-11-23T17:24:12.476+00:00] FSFO target standby is adg_fra34x
Observer 'dgobs1' started
The observer log file is '/u01/app/oracle/client/network/admin/observer_dgobs1.log'.

#press enter to return to the prompt

## Show the running observers again !!!

DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

  Primary:            adg_fra22s
  Active Target:      adg_fra34x
```

```
Observer "dgobs1"(21.1.0.0.0) - Master

  Host Name:                    dgobserver1
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:
```

There is now one observer running (**dgobs1**) with the master role.
Repeat the steps on dgobserver2 and dgobserver3:

```
   -- On dgobserver2 and dgobserver3, generate the wallet and the credentials exactly the
same way as on dgobserver1
-- Then start the observer in background on both machines:

-- dgobserver2

[oracle@dgobserver2 wallet]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 17:23:46 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.

start observer dgobs2 in background file is
'/u01/app/oracle/client/network/admin/fsfo.dat' logfile is
'/u01/app/oracle/client/network/admin/observer_dgobs2.log' connect identifier is
'adg_fra22s';

-- dgobserver3

[oracle@dgobserver3 wallet]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Tue Nov 23 17:23:46 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.

start observer dgobs3 in background file is
'/u01/app/oracle/client/network/admin/fsfo.dat' logfile is
'/u01/app/oracle/client/network/admin/observer_dgobs3.log' connect identifier is
'adg_fra22s';

-- Now from any observer, show the observers !!!

DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x
```

```
  Fast-Start Failover:     ENABLED

  Primary:            adg_fra22s
  Active Target:      adg_fra34x

Observer "dgobs1"(21.1.0.0.0) - Master

  Host Name:                    dgobserver1
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

Observer "dgobs2"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver2
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

Observer "dgobs3"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver3
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

-- We ended-up with 3 observers, 1 master and 2 backups
-- We can easily change the master role to another observer

DGMGRL> set masterobserver to dgobs2
Succeeded.

DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

  Primary:            adg_fra22s
  Active Target:      adg_fra34x

Observer "dgobs2"(21.1.0.0.0) - Master

  Host Name:                    dgobserver2
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

Observer "dgobs1"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver1
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          1 second ago
```

```
  Log File:
  State File:

Observer "dgobs3"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver3
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          1 second ago
  Log File:
  State File:
```

This concludes the "observers configuration" section. In the next section we will see observers and FSFO in action.

Resilience testing with FSFO and observers

The configuration we've just setup aims at leveraging Disaster Recovery in an automatic way.
Should an error occur on the primary database, FSFO and observers would ensure an automatic failover of the database layer. Once the former primary is fixed, FSFO will take the re-instate in charge automatically.
But let's start testing the observers resilience: we start with the three observers configuration as configured in the previous section. Connect to **dgobserver3**, and run the following:

```
[oracle@dgobserver3 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Wed Nov 24 10:29:00 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

  Primary:            adg_fra22s
  Active Target:      adg_fra34x

Observer "dgobs2"(21.1.0.0.0) - Master

  Host Name:                    dgobserver2
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

Observer "dgobs1"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver1
  Last Ping to Primary:         0 seconds ago
```

```
  Last Ping to Target:          1 second ago
  Log File:
  State File:

Observer "dgobs3"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver3
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          1 second ago
  Log File:
  State File:

## As we left it before, dgobs2 is the master, dgobs1 and dgobs3 are backup
-- The first test is about observer resilience
-- Kill the master observer, in this case dgobs2

[oracle@dgobserver2 wallet]$ ps -ef | grep dgmgrl
oracle   10385     1 31 10:14 ?        00:05:37 /u01/app/oracle/client/bin/dgmgrl
START OBSERVER dgobs2 FILE IS '/u01/app/oracle/client/network/admin/fsfo.dat' LOGFILE IS
'/u01/app/oracle/client/network/admin/observer_dgobs2.log'
oracle   15103  2217  0 10:32 pts/0    00:00:00 grep --color=auto dgmgrl

[oracle@dgobserver2 wallet]$ kill -9 10385

-- From observer 3 machine, observe the configuration:

[oracle@dgobserver3 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Wed Nov 24 10:33:29 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

  Primary:            adg_fra22s
  Active Target:      adg_fra34x

Observer "dgobs1"(21.1.0.0.0) - Master

  Host Name:                    dgobserver1
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

Observer "dgobs2"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver2
  Last Ping to Primary:         59 seconds ago
  Last Ping to Target:          28 seconds ago
```

```
  Log File:
  State File:

Observer "dgobs3"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver3
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

DGMGRL>
```

From the above output, we can observer that:

- dgobs1 became master
- dgobs2 is not pinging neither the primary nor the standby database, and is lagging behind. It's because it has been killed, and needs to be restarted manually.
- dgobs3 is backup

Let's restart dgobs2 and observe:

```
[oracle@dgobserver2 wallet]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Wed Nov 24 10:36:48 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL>
DGMGRL>
DGMGRL> start observer dgobs2 in background file is
'/u01/app/oracle/client/network/admin/fsfo.dat' logfile is
'/u01/app/oracle/client/network/admin/observer_dgobs2.log' connect identifier is
'adg_fra22s';
Connected to "adg_fra22s"
Submitted command "START OBSERVER" using connect identifier "adg_fra22s"
DGMGRL> DGMGRL for Linux: Release 21.0.0.0.0 - Production on Wed Nov 24 10:37:12 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
Connected to "adg_fra22s"
Connected as SYSDBA.
Succeeded in opening the observer file "/u01/app/oracle/client/network/admin/fsfo.dat".
Observer 'dgobs2' started
The observer log file is '/u01/app/oracle/client/network/admin/observer_dgobs2.log'.
```

Re-check the observers from any observer machine:

```
[oracle@dgobserver3 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Wed Nov 24 10:38:02 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra22s as sysdba
Connected to "adg_fra22s"
Connected as SYSDBA.
DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

  Primary:            adg_fra22s
  Active Target:      adg_fra34x

Observer "dgobs1"(21.1.0.0.0) - Master

  Host Name:                    dgobserver1
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          0 seconds ago
  Log File:
  State File:

Observer "dgobs2"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver2
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          2 seconds ago
  Log File:
  State File:

Observer "dgobs3"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver3
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          0 seconds ago
  Log File:
  State File:

## Let that dgmgrl session opened and proceed !!!
```

Now dgobs2 is not lagging anymore, but is still backup.

Now let's test FSFO: we are going to abort the primary instance and observe what occurs.
Connect to the primary database server and abort the instance:

```
[opc@adgdb-s01-2021-11-22-170552 ~]$ sudo su - oracle
Last login: Wed Nov 24 10:39:31 UTC 2021
[oracle@adgdb-s01-2021-11-22-170552 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 24 10:40:06 2021
Version 19.12.0.0.0
```

```
Copyright (c) 1982, 2021, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c EE Extreme Perf Release 19.0.0.0.0 - Production
Version 19.12.0.0.0

SQL> select open_mode, database_role from v$database;

OPEN_MODE         DATABASE_ROLE
-------------------- ----------------
READ WRITE        PRIMARY


SQL> exit
-- Abort the instance

srvctl stop database -d $(srvctl config database) -o abort

srvctl status database -d $(srvctl config database)
Instance adg is not running on node adgdb-s01-2021-11-22-170552

## Run a "show observer" command on the previous dgmgrl session on dgobserver3 machine:

DGMGRL> show observer
ORA-03113: end-of-file on communication channel
Process ID: 18645
Session ID: 209 Serial number: 28128

Configuration details cannot be determined by DGMGRL
```

This is because the dgmgrl session was connected to the primary database, that has just been aborted.

Wait 1 minute and reconnect dgmgrl to the standby database to proceed:

```
[oracle@dgobserver3 ~]$ dgmgrl
DGMGRL for Linux: Release 21.0.0.0.0 - Production on Wed Nov 24 10:51:10 2021
Version 21.1.0.0.0

Copyright (c) 1982, 2020, Oracle and/or its affiliates.  All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/"W3lc0m3#W3lc0m3#W"@adg_fra34x as sysdba
Connected to "ADG_FRA34X"
Connected as SYSDBA.
DGMGRL> show observer

Configuration - adg_fra22s_adg_fra34x

  Fast-Start Failover:     ENABLED

  Primary:            adg_fra34x
  Active Target:      adg_fra22s

Observer "dgobs1"(21.1.0.0.0) - Master

  Host Name:                    dgobserver1
```

```
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          594 seconds ago
  Log File:
  State File:

Observer "dgobs2"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver2
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          594 seconds ago
  Log File:
  State File:

Observer "dgobs3"(21.1.0.0.0) - Backup

  Host Name:                    dgobserver3
  Last Ping to Primary:         0 seconds ago
  Last Ping to Target:          594 seconds ago
  Log File:
  State File:
```

Last ping to primary is 0 seconds ago, whereas last ping to standby is lagging behind. It seems that FSFO performed a failover. Let's check on that. From dgobserver3, run the following dgmgrl commands:

```
  DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra34x - Primary database
    Warning: ORA-16824: multiple warnings, including fast-start failover-related
warnings, detected for the database

    adg_fra22s - (*) Physical standby database (disabled)
      ORA-16661: the standby database needs to be reinstated

Fast-Start Failover: Enabled in Potential Data Loss Mode

Configuration Status:
WARNING   (status updated 13 seconds ago)

## Here we observer that a failover was performed automatically by FSFO
## Check the two databases:

DGMGRL> show database adg_fra22s

Database - adg_fra22s

  Role:               PHYSICAL STANDBY
  Intended State:     APPLY-ON
  Transport Lag:      (unknown)
  Apply Lag:          (unknown)
  Average Apply Rate: (unknown)
  Real Time Query:    OFF
  Instance(s):
```

```
    adg

Database Status:
DISABLED - ORA-16661: the standby database needs to be reinstated

## A reinstate is needed for the former primary, as it has been aborted.

DGMGRL> show database adg_fra34x

Database - adg_fra34x

  Role:               PRIMARY
  Intended State:     TRANSPORT-ON
  Instance(s):
    adg

  Database Warning(s):
    ORA-16829: fast-start failover configuration is lagging
    ORA-16869: fast-start failover target not initialized

Database Status:
WARNING
```

The new primary is OK, but the configuration is lagging because the new standby needs to be re-instated. At this stage, we are back in business with the new primary, but not protected against a new failure. We would need to restore/recover the standby from backup, and let FSFO do the re-instate automatically.

Now startup the former primary and observe the automatic FSFO re-instate:

```
## Connect to the former primary host and start the database
## In a real case, we would restore from a backup

srvctl start database -d $(srvctl config database)
srvctl status database -d $(srvctl config database)
Instance adg is running on node adgdb-s01-2021-11-22-170552
```

Go back to the dgobs3 session and observe:

```
DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra34x - Primary database
    Warning: ORA-16824: multiple warnings, including fast-start failover-related
warnings, detected for the database

    adg_fra22s - (*) Physical standby database
      Warning: ORA-16657: reinstatement of database in progress

Fast-Start Failover: Enabled in Potential Data Loss Mode

Configuration Status:
```

```
WARNING   (status updated 102 seconds ago)
```

After some minutes, the situation is back to normal:

```
DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra34x - Primary database
    adg_fra22s - (*) Physical standby database

Fast-Start Failover: Enabled in Potential Data Loss Mode

Configuration Status:
SUCCESS   (status updated 46 seconds ago)


DGMGRL> show database adg_fra34x

Database - adg_fra34x

  Role:                PRIMARY
  Intended State:      TRANSPORT-ON
  Instance(s):
    adg

Database Status:
SUCCESS

DGMGRL> show database adg_fra22s

Database - adg_fra22s

  Role:                PHYSICAL STANDBY
  Intended State:      APPLY-ON
  Transport Lag:       0 seconds (computed 1 second ago)
  Apply Lag:           0 seconds (computed 1 second ago)
  Average Apply Rate:  17.00 KByte/s
  Real Time Query:     ON
  Instance(s):
    adg

Database Status:
SUCCESS


DGMGRL> validate database adg_fra22s

  Database Role:     Physical standby database
  Primary Database:  adg_fra34x

  Ready for Switchover:  Yes
  Ready for Failover:    Yes (Primary Running)

  Managed by Clusterware:
    adg_fra34x:  YES
```

```
    adg_fra22s:  YES

=> OK
```

Optionally, we would perform a switchback to go back to the initial configuration:

```
--- From any observer, run the switchover command !!!

DGMGRL> switchover to adg_fra22s
2021-11-24T11:01:36.994+00:00
Performing switchover NOW, please wait...

2021-11-24T11:01:41.317+00:00
Operation requires a connection to database "adg_fra22s"
Connecting ...
Connected to "adg_fra22s"
Connected as SYSDBA.

2021-11-24T11:01:41.823+00:00
Continuing with the switchover...

2021-11-24T11:02:28.676+00:00
New primary database "adg_fra22s" is opening...

2021-11-24T11:02:28.676+00:00
Oracle Clusterware is restarting database "adg_fra34x" ...
Connected to "adg_fra34x"
Connected to "adg_fra34x"
2021-11-24T11:03:44.657+00:00
Switchover succeeded, new primary is "adg_fra22s"

2021-11-24T11:03:44.659+00:00
Switchover processing complete, broker ready.

--- After a while, we are back to the initial configuration

DGMGRL> show configuration

Configuration - adg_fra22s_adg_fra34x

  Protection Mode: MaxPerformance
  Members:
  adg_fra22s - Primary database
    adg_fra34x - (*) Physical standby database

Fast-Start Failover: Enabled in Potential Data Loss Mode

Configuration Status:
SUCCESS   (status updated 18 seconds ago)


DGMGRL> validate database adg_fra34x

  Database Role:     Physical standby database
  Primary Database:  adg_fra22s

  Ready for Switchover:  Yes
  Ready for Failover:    Yes (Primary Running)
```

```
Managed by Clusterware:
  adg_fra22s:  YES
  adg_fra34x:  YES
```

This concludes the ADG workshop.