Monitoring Oracle GoldenGate

Practice Overview

In this practice you will study the commands and tools that you use to monitor an Oracle GoldenGate configuration.

Assumption

This practice assumes that you have the two systems that you created in the previous practices, their databases, and their GoldenGate processes up and running.

Ahmed Baraka
Oracle Database Administrator

Monitoring Oracle GoldenGate

A. Check out the status of all processes

1. To have a look at the status of all the GoldenGate processes at a glance, issue the following command.

When GoldenGate is in operation, all the processes should have the "RUNNING" status. If the lag value keeps increasing, it is usually an indication of an incident and troubleshooting is needed.

Tip: to re-execute the last command, use the exclamation mark "`!`" in the command prompt.

```
info all
```

B. Monitor the Manager process

2. Issue the following command to display the status of the Manager process.

You can know from this output the status of the process, the port number the process is listening to, and the process ID.

```
info mgr
Manager is running (IP port ggsrv1.localdomain.7809, Process ID 4559).
```

3. Display the process in the OS shell, using the process ID obtained from the previous command output.

You can execute OS shell commands from the ggsci command prompt using the "`sh`" command.

Observer that the primary Extract and the Data Pump processes appear as sub-processes to the Manager process.

```
GGSCI (ggsrv1.localdomain) 38> sh ps -ef | grep 4559

oracle    4559  4555  0 01:25 ?        00:00:17 ./mgr PARAMFILE
/u01/app/oracle/product/ogg/dirprm/mgr.prm REPORTFILE
/u01/app/oracle/product/ogg/dirrpt/MGR.rpt PROCESSID MGR USESUBDIRS

oracle    4566  4559  0 01:26 ?        00:01:22 /u01/app/oracle/product/ogg/extract
PARAMFILE /u01/app/oracle/product/ogg/dirprm/esrv1.prm REPORTFILE
/u01/app/oracle/product/ogg/dirrpt/ESRV1.rpt PROCESSID ESRV1 USESUBDIRS

oracle    4567  4559  0 01:26 ?        00:01:05 /u01/app/oracle/product/ogg/extract
PARAMFILE /u01/app/oracle/product/ogg/dirprm/psrv1.prm REPORTFILE
/u01/app/oracle/product/ogg/dirrpt/PSRV1.rpt PROCESSID PSRV1 USESUBDIRS
...
```

C. Monitor the Extract process

4. Issue the `info` command against the Extract process to display the basic information about the process. A shortcut to the same command is "`info e*`"

You can know from the output the status of the process, the last time it was started, checkpoint lag, the sequence number, the RBA and the SCN.

```
info extract e*

EXTRACT    ESRV1     Last Started 2017-**-** 01:26   Status RUNNING
Checkpoint Lag       00:00:00 (updated 00:00:05 ago)
Process ID           4566
Log Read Checkpoint  Oracle Redo Logs
                     2017-**-** 09:36:51  Seqno 67, RBA 12241408
                     SCN 0.3090752 (3090752)
```

5. As SYSDBA, retrieve the sequence number of the redo log files in `db1`. Compare the output with the `Seqno` reported by the "`info`" command above.

```
-- in ggsrv1
sqlplus / as sysdba
SQL> SELECT SEQUENCE# FROM V$LOG ;
```

6. Retrieve the database SCN number and compare it with the SCN number reported by the "`info`" command.

```
SQL>  SELECT CURRENT_SCN FROM V$DATABASE;
```

7. Issue the `info` command again. Observe that the RBA is incrementing.

RBA never stops incrementing because the Extract keeps reading from the redo log regardless if the data in it is related to the replicated dataset or not. However, the rate at which RBA is incrementing rises as DML operations is being executed on the replicated tables.

```
info extract e*
```

8. Issue the `info` command with the detail option.

Observe the `Seqno` and the `RBA` of the "`Target Extract Trails`" section. They are different from those under the "`Log Read Checkpoint`" section.

```
info extract e*, detail
```

9. List the trail files generated by the Extract process.

```
GGSCI> sh ls ./dirdat/e*
```

10. Issue the `info` command with the "`detail`" option against the Data Pump process.

```
info extract p*, detail
```

11. View the primary Extract process report.

```
View Report esrv1
```

12. Display statistics about the DML operations captured by the Extract.

Tip: “extract” option in the command can be omitted.

```
stats extract esrv1
```

13. Display the lag information of the Extract.

```
lag extract esrv1
```

14. Display the runtime parameter values of the Extract process.

Send command is used to get a vast list of information from any GoldenGate process. Refer to the documentation for the information that you can get by using this command.

```
send esrv1 GetParamInfo
```

Ahmed Baraka
Oracle Database Administrator

D. Monitor the Replicat

15. Issue the `info` command against the Replicate process

Observe the RBA of the `Log Read Checkpoint`. It does not get incremented unless some data has been added to the remote trail files. In your scenario, it will definitely get incremented after at most one minute when the job updates the SAMPLE table.

```
info replicat r*
```

16. Issue the `info` command with the `detail` option.

The output will include the message "`Last Committed Transaction CSN value: (requires database login)`"

```
info r*, detail
```

17. Login to the database and issue the `info` command again. Observe the difference in the output.

```
dblogin userid ogg, PASSWORD oracle
info replicat r*, detail
```

18. List the trail files that the Replicat process reads from.

```
sh ls ./dirdat/r*
```

19. Display the runtime parameter values of the Replicat process.

```
send rsrv2 GetParamInfo
```

E. Configure the Report and Discard file rollover parameters

A process abends when its report or discard files reach to their maximum size. To avoid such a scenario, configure the `REPORTROLLOVER` and the `DISCARDROLLOVER` parameters.

20. Add the following parameters to both the Extract and Replicat parameter files. For the parameter to take effect, restart the processes.

```
REPORTROLLOVER AT 03:00 ON sunday
DISCARDROLLOVER AT 03:00 ON sunday
```

F. Set up Automatic Heartbeat Tables functionality

In this practice you will configure the Automatic Heartbeat Tables functionality.

Note: this functionality is available starting from Oracle GoldenGate release 12.2. With the versions earlier than 12.2, GoldenGate administrators should develop a manual solution to achieve a similar functionality. One procedure is available in Oracle Support site as described in the Doc ID 1299679.1.

21. Login to the source database and add the heartbeat functionality.

```
dblogin userid ogg, password oracle
ADD HEARTBEATTABLE
```

22. Perform the same action for the target database.

```
dblogin userid ogg, PASSWORD oracle
ADD HEARTBEATTABLE
```

23. Restart the processes

```
stop *
start *
```

24. Verify the heartbeat functionality components are there

```
info heartbeattable
```

25. Login to the database via SQL*Plus and list the heartbeat table functionality components.

```
sqlplus / as sysdba
SELECT TABLE_NAME FROM DBA_TABLES WHERE OWNER= 'OGG' AND TABLE_NAME LIKE 'GG%';
SELECT JOB_NAME, REPEAT_INTERVAL FROM DBA_SCHEDULER_JOBS WHERE OWNER= 'OGG';
```

26. Test the new configuration by querying the GG_LAG view in the target database.

```
col LOCAL_DB format a7
col RMT_DB format a6
col INCOMING_PATH format a30
select REMOTE_DATABASE RMT_DB, LOCAL_DATABASE LOCAL_DB, INCOMING_PATH, INCOMING_LAG
from OGG.GG_LAG;
```

Summary

In this practice you used the basic monitoring tools in Oracle GoldenGate to monitor its operation.

You also configured the automatic heartbeat functionality which helps you to measure the end-to-end lag value.

Note: Oracle GoldenGate has more sophisticated products for monitoring an Oracle GoldenGate configuration. Using those products is beyond the scope of this course.

Ahmed Baraka
Oracle Database Administrator