

Overcoming Performance Degradation on Primary DB when Standby DB Unavailable

Problem Description

When the Physical Standby Database in an Oracle 11gR1 Data Guarded RAC environment becomes unavailable, the performance of the Primary plummets due to excessive logfile switching. In this particular case, the Physical Standby Database had exhausted the free space in its Flash Recovery Area (FRA) causing the Foreign Archive Logs (FAL) to be rejected. The LNS (Log-write Network Server) process that manages the writing of redo to the Standby dies when it cannot successfully archive a redolog for a given Thread. When it restarts, it initiates a log file switch to “kick start” the archiver. This behaviour occurs at 1 minute intervals, resulting in the excessive logfile sync wait events as shown below in the Top 5 foreground events from the AWR report.

Top 5 Timed Foreground Events

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
log file switch completion	16,252	1,613	99	31.12	Configuration
log file sync	239,638	1,228	5	23.69	Commit
DB CPU		1,000		19.29	
log file switch (checkpoint incomplete)	9,440	939	99	18.11	Configuration
buffer busy waits	36,230	201	6	3.87	Concurrency

The Alert Log

The Alert log of the Oracle Instance shows the error details and the high frequency of logfile switches, also known to cause intermittent “Checkpoint not complete” warnings. Here we see the LNS process failing and restarting every minute when it can't send archive logs to the standby. This logfile switching supersedes the breaching of the archive lag target of 360 seconds and redolog file size of 1024MB.

```
ORA-00270: error creating archive log
LNS: Failed to archive log 5 thread 1 sequence 185789 (270)
Thu Oct 20 15:05:41 2011
Errors in file
/u01/app/oracle/diag/rdbms/mydb/MYDB1/trace/MYDB1_arc0_21634.trc:
ORA-00270: error creating archive log
FAL[server, ARC0]: FAL archive failed, see trace file.
Errors in file
/u01/app/oracle/diag/rdbms/mydb/MYDB1/trace/MYDB1_arc0_21634.trc:
ORA-16055: FAL request rejected
ARCH: FAL archive failed. Archiver continuing
ORACLE Instance MYDB1 - Archival Error. Archiver continuing.
Thu Oct 20 15:06:41 2011
LNS1 started with pid=66, OS id=23546
Thu Oct 20 15:06:44 2011
Thread 1 advanced to log sequence 185790 (LGWR switch)
Current log# 6 seq# 185790 mem# 0:
+DATA/mydb/onlinelog/group_6.324.713826365
```

```

Errors in file
/u01/app/oracle/diag/rdbms/mydb/MYDB1/trace/MYDB1_lns1_23546.trc:
ORA-00270: error creating archive log
LNS: Failed to archive log 6 thread 1 sequence 185790 (270)
Thu Oct 20 15:06:45 2011
Archived Log entry 1007376 added for thread 1 sequence 185789 ID
0x7bef4c81 dest 1:
Thu Oct 20 15:06:48 2011
Errors in file
/u01/app/oracle/diag/rdbms/mydb/MYDB1/trace/MYDB1_arc2_21638.trc:
ORA-00270: error creating archive log
FAL[server, ARC2]: FAL archive failed, see trace file.
Errors in file
/u01/app/oracle/diag/rdbms/mydb/MYDB1/trace/MYDB1_arc2_21638.trc:
ORA-16055: FAL request rejected
ARCH: FAL archive failed. Archiver continuing
ORACLE Instance MYDB1 - Archival Error. Archiver continuing.
Thu Oct 20 15:07:44 2011
LNS1 started with pid=69, OS id=25762
Thu Oct 20 15:07:47 2011
Thread 1 advanced to log sequence 185791 (LGWR switch)
  Current log# 1 seq# 185791 mem# 0:
+DATA/mydb/onlinelog/group_1.319.713826339

```

Solution

The obvious solution is to fix the FRA space issue and have LNS writing to the Standby logs with logfile switching occurring naturally; when either the archive lag target is reached (if set) or when the Primary Database's redolog is filled. However, it may be some time before issues on the Standby can be resolved.

As a quick win, the Data Guard archive destination on the Primary database can be temporarily deferred to resolve the performance degradation. Just follow these simple steps:

1. Logon (as sysdba) to the Primary database instance.
sqlplus / as sysdba

2. Confirm the database role.

```
SQL> select database_role from v$database;
```

```

DATABASE_ROLE
-----
PRIMARY

```

3. Show the database archive destination configurations.

```
SQL> show parameter archive_dest
```

NAME	TYPE	VALUE
log_archive_dest	string	
log_archive_dest_1	string	location="+FLASH", valid_for=(ALL_LOGFILES,ALL_ROLES)
log_archive_dest_10	string	
log_archive_dest_2	string	service="MYDBSBY", LGWR ASYN C NOAFFIRM delay=0 OPTIONAL co mpression=DISABLE max_failure= 0 max_connections=1 reopen=3

```
0 db_unique_name="MYDBSBY" net_
timeout=30 valid_for=(online_
logfile,primary_role)
```

4. In my case the archive destination for Data Guard is "log_archive_dest_2". Now set the "log_archive_dest_state_2" parameter to DEFER.

```
SQL> alter system set log_archive_dest_state_2=DEFER
scope=MEMORY sid='*';
```

System altered.

5. Check the status of the "log_archive_dest_state_2" parameter.

```
SQL> show parameter log_archive_dest_state_2
```

NAME	TYPE	VALUE
log_archive_dest_state_2	string	DEFER

We can see the dramatic positive effect deferring the destination has on the Primary Database performance by looking at the Instance AWR report and Alert log:

Top 5 Timed Foreground Events

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
DB CPU		70		87.55	
control file sequential read	15,234	13	1	15.80	System I/O
db file sequential read	4,898	4	1	4.61	User I/O
PX Deq: Slave Session Stats	2,385	1	1	1.57	Other
IPC send completion sync	2,954	1	0	1.21	Other

DB CPU has increased to 87.55 % of all DB time (1 hour period) with average waits of 1ms.

The Alert Log

```
Thu Oct 20 15:09:09 2011
ALTER SYSTEM SET log_archive_dest_state_2='DEFER' SCOPE=MEMORY
SID='*';
Thu Oct 20 15:14:34 2011
Thread 1 advanced to log sequence 185793 (LGWR switch)
  Current log# 3 seq# 185793 mem# 0:
+DATA/mydb/onlinelog/group_3.321.713826349
Thu Oct 20 15:14:34 2011
Archived Log entry 1007388 added for thread 1 sequence 185792 ID
0x7bef4c81 dest 1:
```

Once the Physical Standby database has been fixed, start the LNS process by re-enabling the "log_archive_dest_state_2" parameter as follows:

```
SQL> alter system set log_archive_dest_state_2=ENABLE
scope=BOTH sid='*';
```

Check the Alert log on the Standby to ensure Data Guard has resolved any logfile gaps and the Managed Recovery Process (MRP) has applied the missing logs.