

Reinstate Streams Schema Replication following Database Incomplete Recovery

Problem Description

After an incomplete recovery of the source database, where the database was opened using “ALTER DATABASE OPEN RESETLOGS”, your Oracle Streams downstream capture process will fail to the following errors, seen in the target database instance alert log:

```
ORA-01346: LogMiner processed redo beyond specified reset log scn
ORA-01280: Fatal LogMiner Error.
```

The following snippet is from the target database instance 1 alert log in a 2 node RAC environment running on Oracle Enterprise Linux:

```
$ tail -f alert_TGT1.log
```

```
<snip>
Mon Jan 18 13:00:33 2010
Archived Log entry 5763 added for thread 1 sequence 544 ID 0xba4d74f1 dest
1:
Mon Jan 18 13:21:29 2010
RFS[13]: Assigned to RFS process 9568
RFS[13]: Identified database type as 'streams capture': Client is ARCH pid
22022
RFS[13]: Identified database type as 'repository': Client is ARCH pid 22022
RFS[13]: Opened log for thread 2 sequence 1 dbid 2053587448 branch 708614321
Mon Jan 18 13:21:30 2010
Fatal Error: Stalled LogMiner processed scn 0.1973802967 beyond new branch
scn 0.1967682915
krvxrpt: Errors detected in process 127, role builder.
krvxmrs: Leaving by exception: 1346
Errors in file /u01/app/oracle/diag/rdbms/tgt/TGT1/trace/TGT1_ms01_5894.trc:
ORA-01346: LogMiner processed redo beyond specified reset log scn
LOGMINER: session#=36, builder MS01 pid=127 OS id=5894 sid=894 stopped
Errors in file /u01/app/oracle/diag/rdbms/tgt/TGT1/trace/TGT1_ms01_5894.trc:
Mon Jan 18 13:21:30 2010
LogMiner process death detected
Mon Jan 18 13:21:30 2010
LOGMINER: session#=37, preparer MS06 pid=132 OS id=5915 sid=887 stopped
RFS LogMiner: Registered logfile
[+FLASH/tgt/foreign_archive_log/src/tgt_2_1_708614321.dbf] to LogMiner
session id [36]
RFS LogMiner: Registered logfile
[+FLASH/tgt/foreign_archive_log/src/tgt_2_1_708614321.dbf] to LogMiner
session id [37]
Streams CAPTURE CP01 for SRC_SCHEMA_CAPTURE with pid=65, OS id=5708 stopped
Errors in file /u01/app/oracle/diag/rdbms/tgt/TGT1/trace/TGT1_cp01_5708.trc:
ORA-01280: Fatal LogMiner Error.
</snip>
```

Solution

The following procedure can be used to reinstate the downstream capture process:

1. Logon to target database as Streams Admin user and stop the downstream capture and apply processes.

```
sqlplus streams_admin/streams_admin

SQL> exec dbms_apply_adm.stop_apply(apply_name=>'SRC_SCHEMA_APPLY')

SQL> exec
dbms_capture_adm.stop_capture(capture_name=>'SRC_SCHEMA_CAPTURE')
```

2. Logon to source database as Streams Admin user. Generate and implicitly obtain the first SCN on source DB containing the data dictionary.

```
sqlplus streams_admin/streams_admin

SQL> SET SERVEROUTPUT ON
DECLARE
    scn NUMBER;
BEGIN
    DBMS_CAPTURE_ADM.BUILD(
        first_scn => scn);
    DBMS_OUTPUT.PUT_LINE('First SCN Value = ' || scn);
END;
/

First SCN Value = 1900359521

PL/SQL procedure successfully completed.
```

3. Logon to target database as Streams Admin user and drop capture process.

```
sqlplus streams_admin/streams_admin

SQL> exec dbms_capture_adm.drop_capture ('SRC_SCHEMA_CAPTURE');

PL/SQL procedure successfully completed.
```

4. Create capture process on target database. Use SCN obtained in step 2 for both start_scn and first_scn parameters of DBMS_CAPTURE_ADM.CREATE_CAPTURE procedure.

```
BEGIN
    DBMS_CAPTURE_ADM.CREATE_CAPTURE (
        queue_name      => 'STREAMS_ADMIN.STREAMS_SRC_Q',
        capture_name     => 'SRC_SCHEMA_CAPTURE',
        rule_set_name    => NULL,
        start_scn        => 1900359521,
        source_database  => '&src_db_name',
        use_database_link => TRUE,
        first_scn        => 1900359521,
        logfile_assignment => 'implicit');
END;
```

/

5. Add capture rules using `DBMS_STREAMS_ADM.ADD_SCHEMA_RULES` procedure, as in example below:

```
BEGIN
  DBMS_STREAMS_ADM.ADD_SCHEMA_RULES (
    schema_name      => 'SRC',
    streams_type     => 'CAPTURE',
    streams_name     => 'SRC_SCHEMA_CAPTURE',
    queue_name       => 'STREAMS_ADMIN.STREAMS_SRC_Q',
    include_dml      => TRUE,
    include_ddl      => FALSE,
    source_database  => '&src_db_link',
    and_condition    => ':dml.get_command_type() != ''DELETE''');
END;
/
```

6. Add capture performance settings using `DBMS_STREAMS_ADM.SET_PARAMETER` and `DBMS_STREAMS_ADM.ALTER_CAPTURE` procedures, as in example below:

```
BEGIN
  dbms_capture_adm.set_parameter( capture_name =>
'SRC_SCHEMA_CAPTURE',
                                parameter     =>
'_CHECKPOINT_FREQUENCY',
                                VALUE        => '1000');
  dbms_capture_adm.alter_capture( capture_name =>
'SRC_SCHEMA_CAPTURE',
                                checkpoint_retention_time => 7);
  dbms_capture_adm.set_parameter( capture_name =>
'SRC_SCHEMA_CAPTURE',
                                parameter     => 'PARALLELISM',
                                VALUE        => '2');
  dbms_capture_adm.set_parameter( capture_name =>
'SRC_SCHEMA_CAPTURE',
                                parameter     => '_SGA_SIZE',
                                VALUE        => '100');
END;
/
```

7. Start the combined capture and apply process on the target database.

```
SQL> exec dbms_apply_adm.start_apply(apply_name=>'SRC_SCHEMA_APPLY')
SQL> exec
dbms_capture_adm.start_capture(capture_name=>'SRC_SCHEMA_CAPTURE')
```

8. Log back onto source database and switch log files to initiate the downstream capture process.

```
sqlplus / as sysdba
```

```
SQL> alter system archive log current;
```

```
System altered.
```

9. Logon to target database and check capture process is “capturing changes”.

```
sqlplus streams_admin/streams_admin
```

```
SQL> select CAPTURE_NAME, STATE from v$streams_capture;
```

```
CAPTURE_NAME          STATE
-----
SRC_SCHEMA_CAPTURE    CAPTURING CHANGES
```

10. If you wish to re-enable downstream real-time mine, this can be done by executing the following procedure after Streams archivelog downstream mine has reinitialized. (downstream real-time mine also requires standby redo log files on the target database). Then re-execute step 8 to make the transition.

```
BEGIN
  dbms_capture_adm.set_parameter( capture_name =>
'SRC_SCHEMA_CAPTURE',
                                parameter      =>
'downstream_real_time_mine',
                                VALUE         => 'Y');
END;
/
```

<snip>

```
$ tail -f alert_TGT1.log
```

```
Mon Jan 18 14:31:27 2010
Streams CAPTURE CP01 for SRC_SCHEMA_CAPTURE started with pid=62, OS id=21009
Streams CAPTURE CP01 for SRC_SCHEMA_CAPTURE with pid=62, OS id=21009 is in
combined capture and apply mode.
Streams downstream capture SRC_SCHEMA_CAPTURE uses
downstream_real_time_mine: TRUE
Starting persistent Logminer Session with sid = 38 for Streams Capture
SRC_SCHEMA_CAPTURE
LOGMINER: Parameters summary for session# = 38
LOGMINER: Number of processes = 4, Transaction Chunk Size = 1
LOGMINER: Memory Size = 100M, Checkpoint interval = 1000M
LOGMINER: SpillScn 0, ResetLogScn 1967682916
LOGMINER: krxpsr summary for session# = 38
LOGMINER: StartScn: 1981015050 (0x0000.7613e40a)
LOGMINER: EndScn: 0
LOGMINER: HighConsumedScn: 1981015050 (0x0000.7613e40a)
LOGMINER: session_flag 0x1
LOGMINER: LowCkptScn: 0 (0x0000.00000000)
LOGMINER: HighCkptScn: 0 (0x0000.00000000)
LOGMINER: SkipScn: 1980975666 (0x0000.76134a32)
</snip>
```

..

<snip>

```
RFS[17]: Selected log 20 for thread 1 sequence 11 dbid 2053587448 branch
708614321
LOGMINER: End mining logfile for session 38 thread 1 sequence 10,
+FLASH/tgt/foreign_archivelog/src/tgt_1_10_708614321.dbf
LOGMINER: Begin mining logfile for session 38 thread 1 sequence 11,
+FLASH/tgt/onlinelog/group_20.6324.704635273 ← Note transition to mine standby redo
logs (Realtime mine) for thread 1 sequence 11
LOGMINER: End mining logfile for session 39 thread 1 sequence 10,
+FLASH/tgt/foreign_archivelog/src/tgt_1_10_708614321.dbf
LOGMINER: Begin mining logfile for session 39 thread 1 sequence 11,
+FLASH/tgt/onlinelog/group_20.6324.704635273
LOGMINER: End mining logfile for session 38 thread 2 sequence 9,
+FLASH/tgt/foreign_archivelog/src/tgt_2_9_708614321.dbf
LOGMINER: End mining logfile for session 39 thread 2 sequence 9,
+FLASH/tgt/foreign_archivelog/src/tgt_2_9_708614321.dbf
LOGMINER: Begin mining logfile for session 38 thread 2 sequence 10,
+FLASH/tgt/onlinelog/group_26.6330.704635299
```

```
LOGMINER: Begin mining logfile for session 39 thread 2 sequence 10,  
+FLASH/tgt/online_log/group_26.6330.704635299  
</snip>
```

Additional Steps

Should the capture process not advance and appears stuck in one of the following states:

```
select * from v$streams_capture shows :
```

- INITIALIZING / DICTIONARY INITIALIZATION (the state alternates between these states), or
- WAITING FOR DICTIONARY REDO: FIRST SCN <SCN> , or
- WAITING FOR REDO: LAST SCN MINED <SCN> , or
- WAITING FOR DICTIONARY REDO: FILE <filename>

Then perform these additional steps:

11. Repeat step 1 and 2 above
12. Make a note of the System Change Number returned from step 2 and alter the downstream capture process on the target database to change the start SCN to this number.

```
SQL> exec  
dbms_capture_adm.alter_capture(capture_name=>'SRC_SCHEMA_CAPTURE', sta  
rt_scn=><SCN>)
```

13. Start the apply and capture processes on the target database.

```
SQL> exec dbms_apply_adm.start_apply(apply_name=>'SRC_SCHEMA_APPLY')  
SQL> exec  
dbms_capture_adm.start_capture(capture_name=>'SRC_SCHEMA_CAPTURE')
```

14. Check the capture process is “capturing changes”.

```
SQL> select CAPTURE_NAME, STATE from v$streams_capture;
```

| CAPTURE_NAME | STATE |
|--------------------|-------------------|
| ----- | ----- |
| SRC_SCHEMA_CAPTURE | CAPTURING CHANGES |