

Integrating Oracle Streams and Physical Standby

John Garmany
garmanyj@proxitec.com

Who Am I

John Garmany
Senior Consultant

ORACLE | CERTIFIED
PROFESSIONAL



- West Point Graduate – GO ARMY!
- Masters Degree Information Systems
- Graduate Certificate in Software Engineering

Why Have a Duplicate Database

- Recovery
 - Corruption, failure or disaster
- Off Loading Work
 - Reporting
 - Large Queries
 - End of Month
- Availability (replication vs RAC)
- All of the Above!

Oracle Standby Database

- A copy of the primary database
 - Physical
 - Exact Copy
 - Database Mounted
 - REDO Apply
 - Logical – not addressed in this presentation.
 - Logical Copy – Tables read-only
 - Database Open
 - SQL Apply

Oracle Standby Database

- Physical Standby
 - Changes on primary passed to standby using redo or archivelogs.
 - Changes rolled into standby using redo apply.
 - FAL Client/FAL Server resolve gaps.
 - Entire Database

Redo Apply

- Applies REDO
 - Recovers changes into the database files.
 - Database must be in MOUNT mode.***
 - Results are datafiles that exactly match source. Can be used to recover the source!
 - Redo Apply is used on physical standby using
 - Archivelogs
 - Standby redo logs

Oracle Standby Database

- What happens on primary, happens on standby!
 - New datafiles, users, grants, etc
 - Replication Components
 - Database Links
 - NOT FLASHBACK!!!!!!!!!!!!
 - Flashback primary does not flashback standby.

Oracle Standby Database

- Supports Switchover and Failover.
 - Switchover
 - Coordinated switch of roles between standby and primary.
 - No dataloss.
 - Failover
 - Loss of Primary.
 - Likely some dataloss.
 - Primary must be rebuilt.

Physical Standby

- Exact Copy
 - Datafiles on Standby can be used to Recover the Primary
 - Standby in MOUNT mode.
 - Can be used by RMAN to backup Primary
 - Can be opened Read only but Apply stops.
 - Can be opened Read Write but must be Flashed back to resume standby.
 - Failover/Switchover

Physical Standby

- THE Disaster Recovery Option!
- Not suitable for Reporting.
 - Redo Apply stops when open.
 - Open only READ ONLY.
 - Flashback or rebuild required if opened READ WRITE.

Physical Standby 11g

- Open READ-ONLY
- Restart REDO Apply.
 - Read only queries while redo apply running.
 - Cannot open while redo apply is running on any instance.

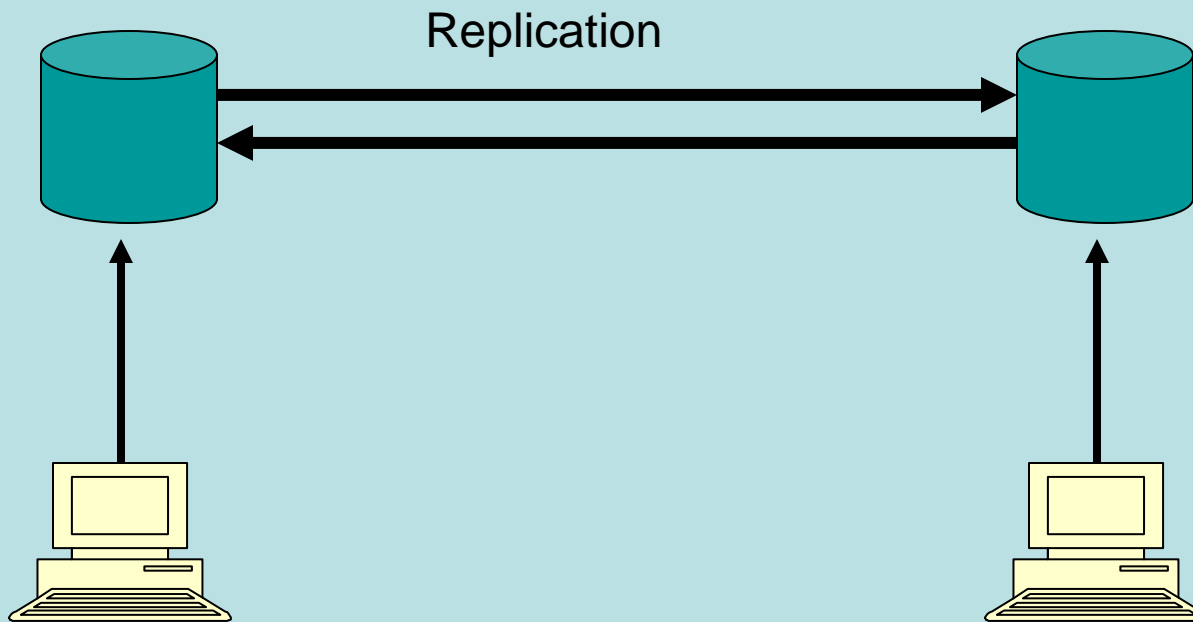
Oracle Streams

- Replication not Disaster Recovery
- Can be bi-directional, hub and spoke, etc
- All databases open.
- Update by Queued LRC and SQL Apply
- No switchover/failover

- Can and will fall behind.

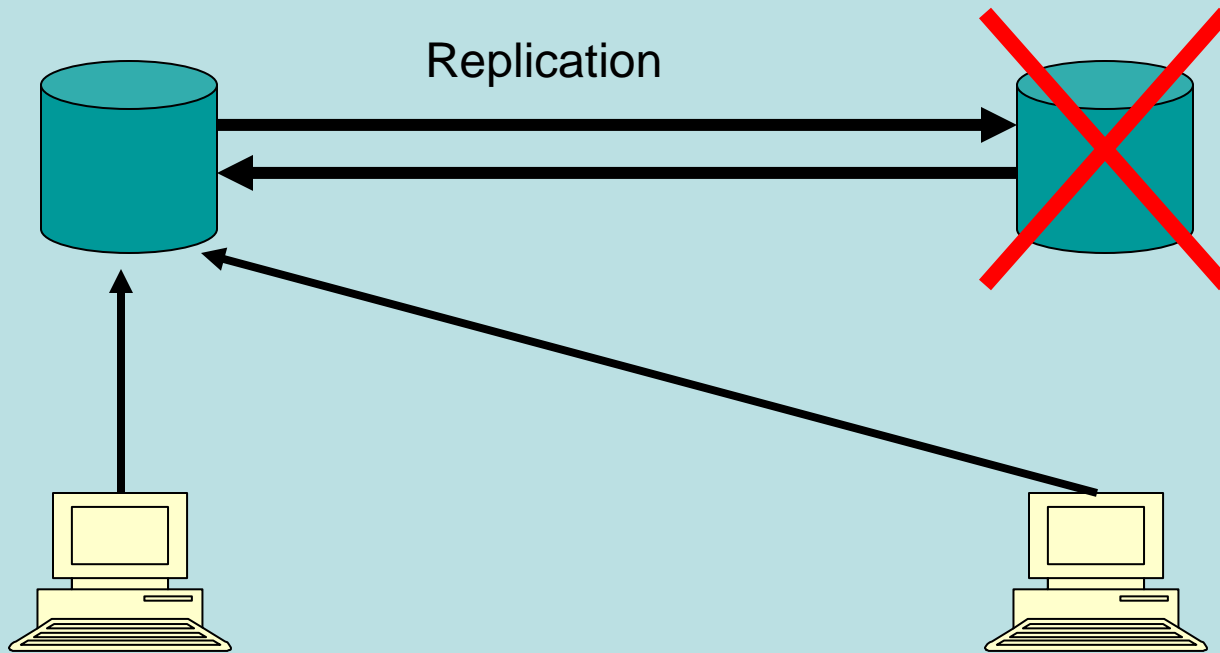
Oracle Streams

- Replication as high availability solution



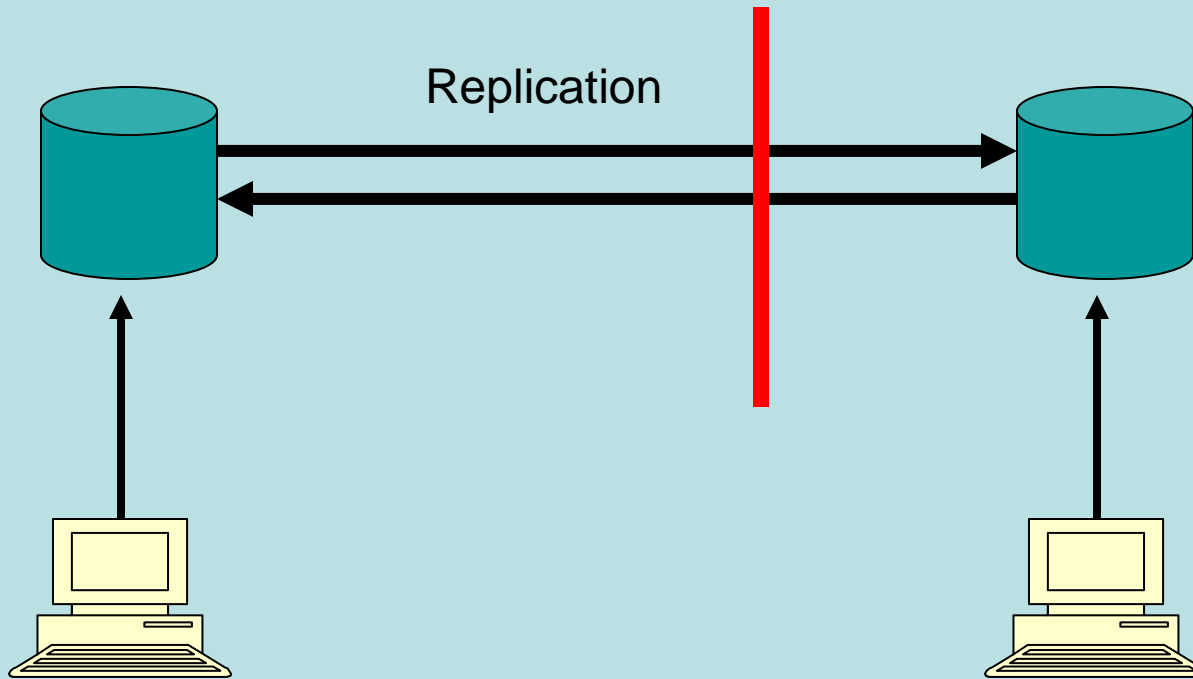
Oracle Streams

- Replication as high availability solution
 - Loss of a database.



Oracle Streams

- Replication as high availability solution
 - Loss of Network Connection



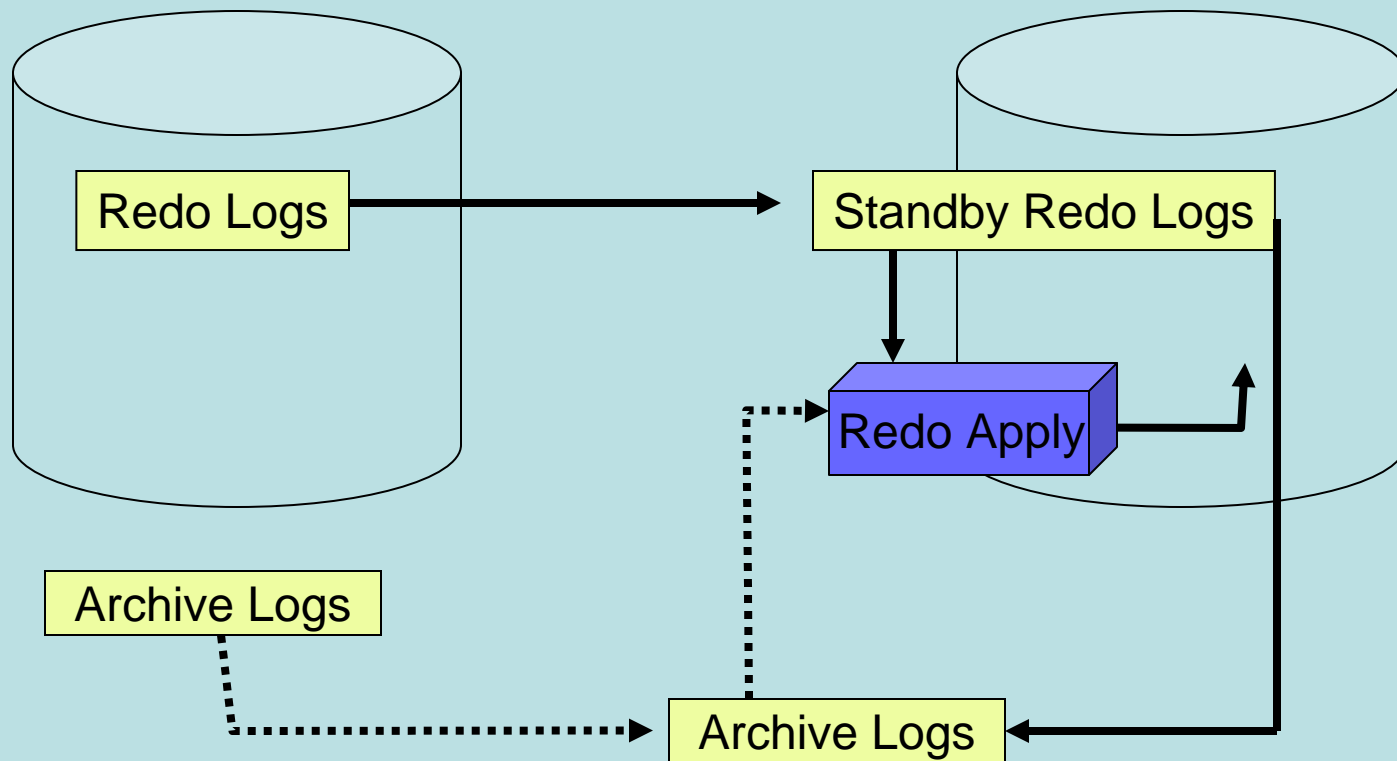
How Things Work

Physical Standby

Physical Standby

- Redo shipped to Standby Redo Logs.
 - Standby logs are used to recover redo into the standby. Not used in primary role.
 - On log switch, local archive logs are created from the standby logs.
- Archive Logs
 - Shipped if needed
 - Large gaps
 - No standby redo logs.

Physical Standby



How Things Work

Oracle Streams

Oracle Streams

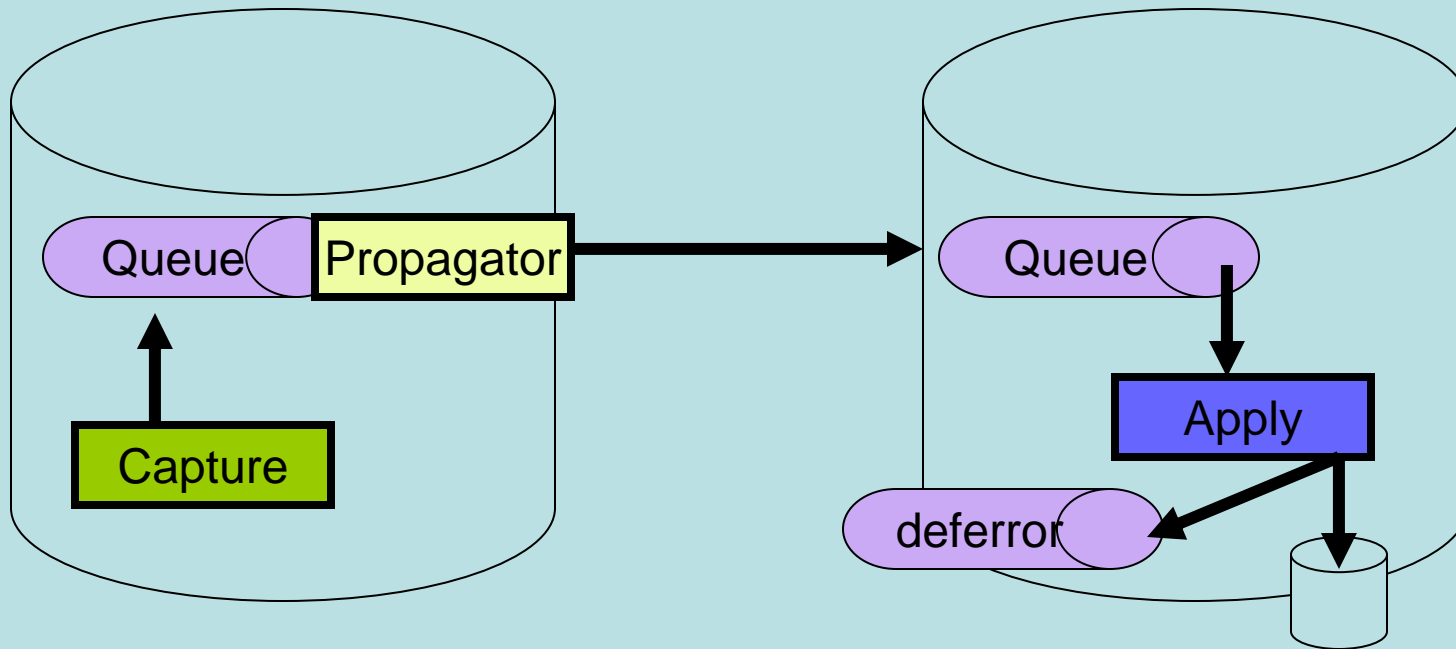
- Streams is replication
 - May replicate:
 - Database
 - Schema
 - Tables
 - Any combination of the above.
 - Multiple Streams on one database.

Oracle Streams

- Redo Mined on the Source database
- LCRs are queued on the Source.
- Propagator Sends LCR to Destination
- LCRs queued on the Destination
- Apply dequeues and applies changes.

- Archive logs are not involved at all.

Oracle Streams



Bi-directional is same going both directions.
Hub and spoke continues change propagation.

Oracle Streams

- Capture (Log Miner) operates on Source.
 - Can impact primary database!
 - Does not have to Capture all changes (Rules)
- SQL Apply with all its limitations
- User defined Apply (PL/SQL)
- Apply Error Handling
 - Conflict resolution
 - Used defined error handling (PL/SQL)

Oracle Streams RULES

Applied at each step

(capture, propagate, apply)

– Determine actions performed.

- Capture
 - Queue or skip
- Propagate
 - Send or skip
- Apply
 - Apply, skip or pass to PL/SQL Procedure.

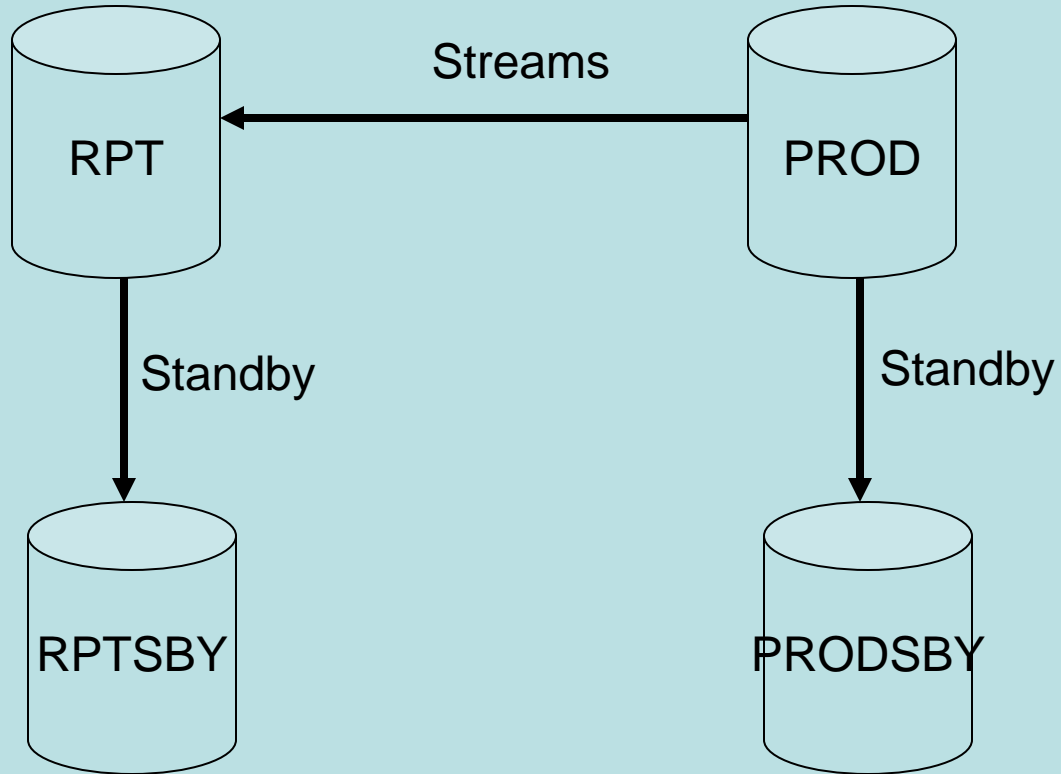
Important Oracle Streams Parts

- Capture
 - First_SCN
 - Where the Capture process starts reading.
 - Moved forward during capture checkpoint purges.
 - Start_SCN
 - Where the Capture process starts capturing changes.
 - Tracked by the Capture process.
 - LCR
 - Logical Change Record.

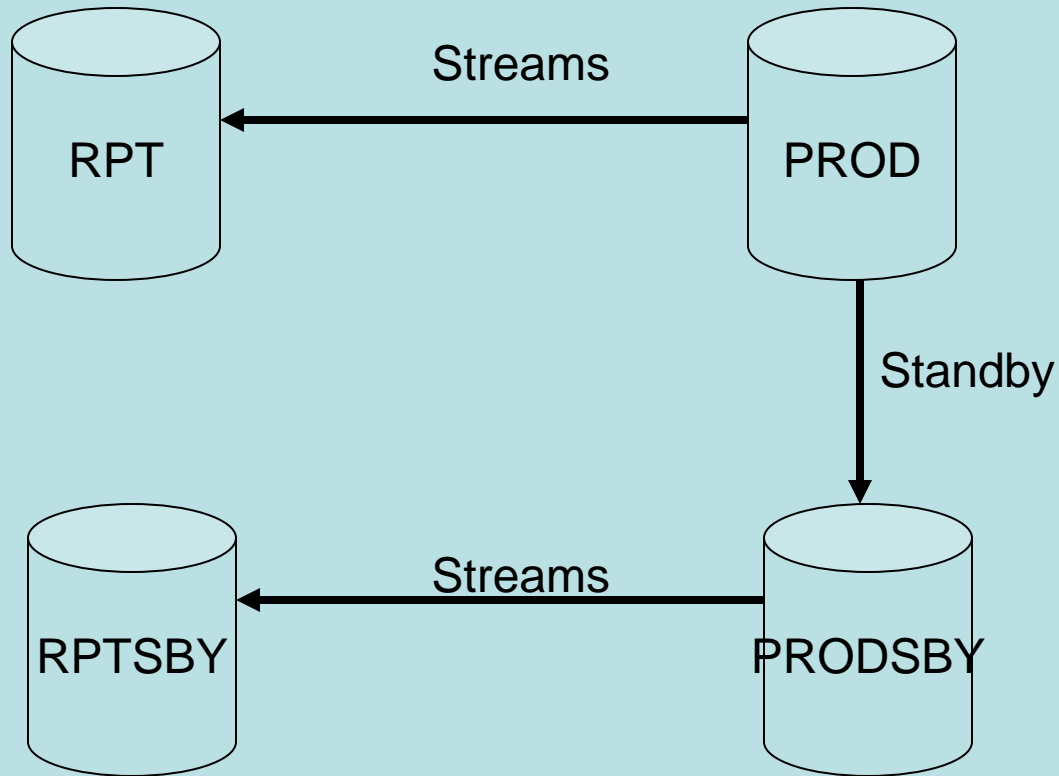
Important Oracle Streams Parts

- Apply
 - `oldest_message_number`
The last source scn applied to the destination.

The Setup



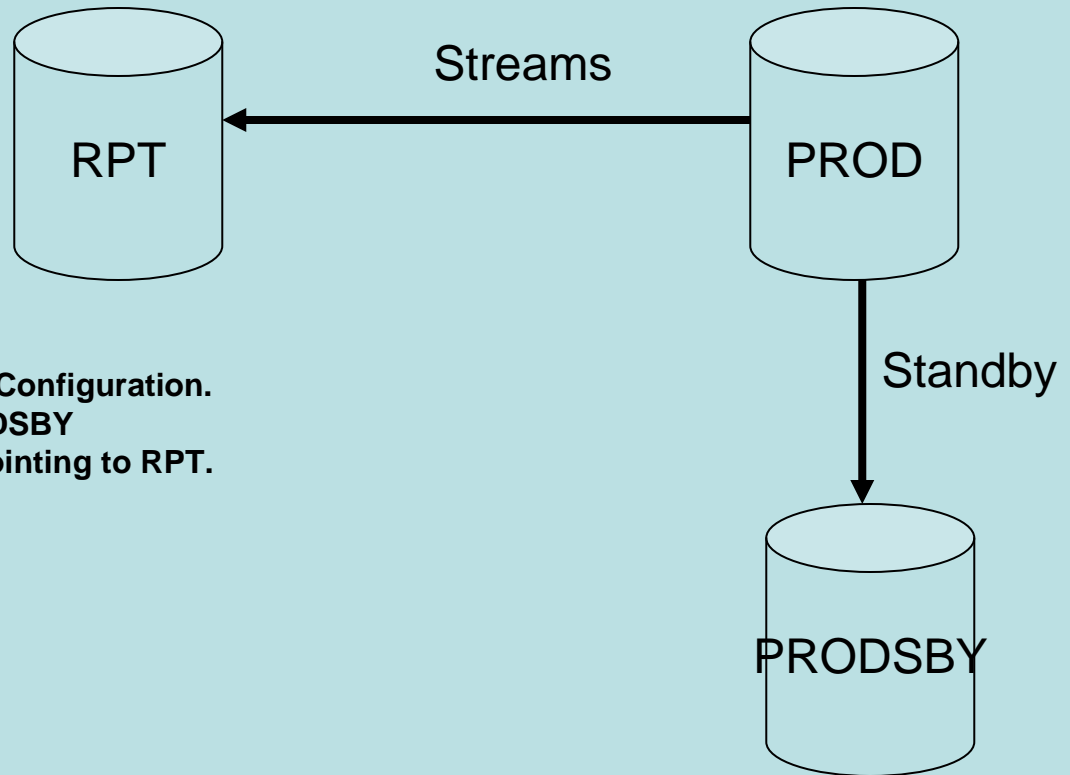
The Setup - Wrong



PRODSBY is mounted, not open. No Active Streams

The Easy Part

Switchover PROD

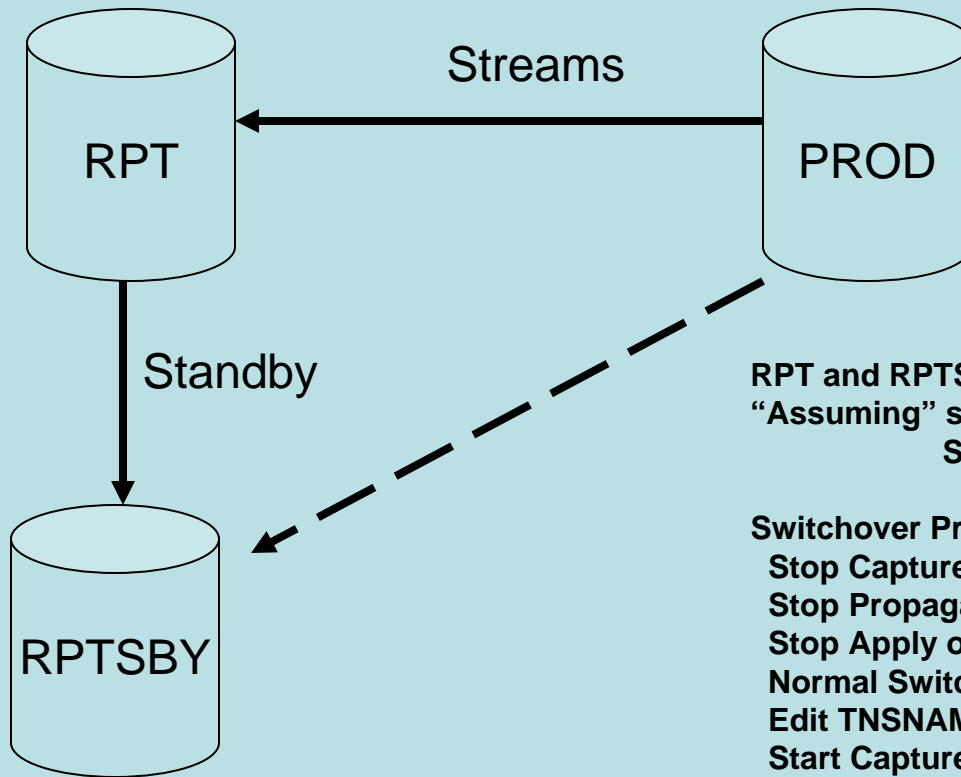


PROD and PRODSBY have same Streams Configuration.
“Assuming” same TNSNAMES.ORA, PRODSBY
Streams DBLINK is already pointing to RPT.

Switchover Procedure:
Stop Capture on PROD.
Stop Propagation on PROD.
Normal Switchover.
Start Capture on PRODSBY.
Start Propagation on PRODSBY.

The Easy Part II

Switchover RPT



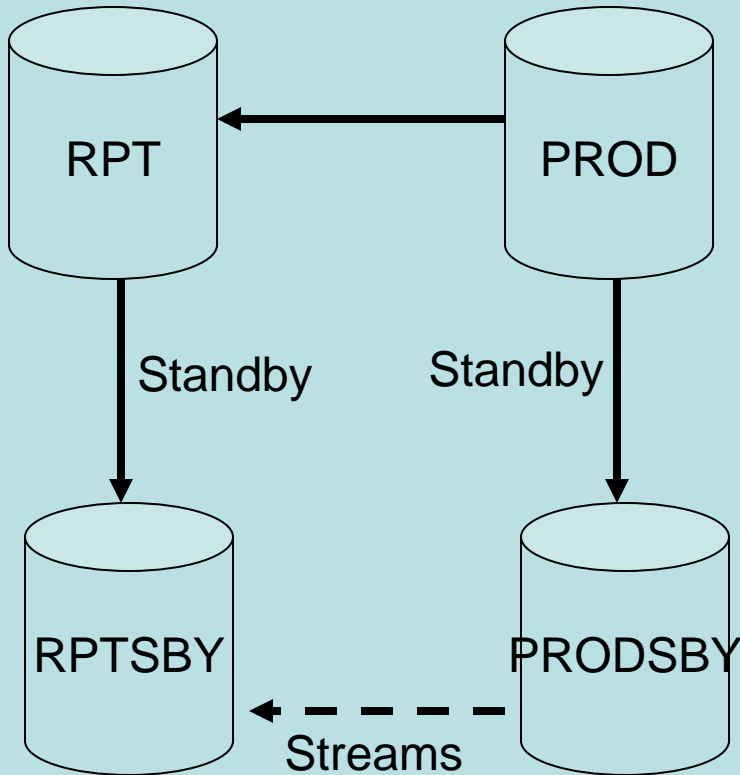
RPT and RPTSBY have same Streams Configuration.
"Assuming" same TNSNAMES.ORA, PROD
Streams DBLINK is pointing to RPT, not RPTSBY.

Switchover Procedure:

- Stop Capture on PROD.
- Stop Propagation on PROD.
- Stop Apply on RPT
- Normal Switchover.
- Edit TNSNAMES.ORA on PROD to point to RPTSBY.
- Start Capture on PROD.
- Start Propagation on PROD.
- Start Apply on RPTSBY.

The Easy Part III

Switchover Both



“Assuming” same TNSNAMES.ORA on each database.

Switchover Procedure:

Stop Capture on PROD.

Stop Propagation on PROD.

Stop Apply on RPT

Normal Switchover of both standbys.

Edit TNSNAMES.ORA on PRODSBY to point to RPTSBY.

Edit TNSNAMES.ORA on RPTSBY to point to PRODSBY.

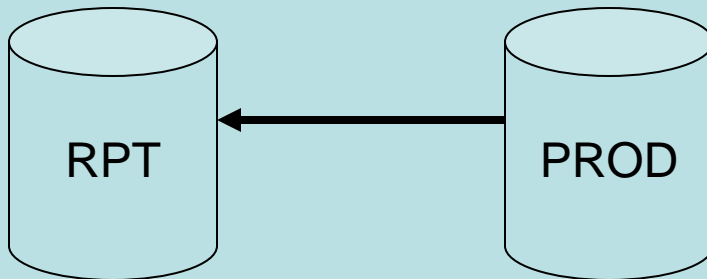
Start Capture on PRODSBY.

Start Propagation on PRODSBY.

Start Apply on RPTSBY.

The Hard Part - Failover

- Failover assumes some data loss.
- Primary is gone.
- Three Scenarios:
 - Failover of PROD leaves PRODSBY ahead of RPT.
 - Failover of PROD leaves PRODSBY behind RPT
 - Failover of RPT leaves RPTSBY behind PROD.



The Hard Part - Failover

- Is PROD ahead or behind RPT?

On PROD

```
SQL> select STANDBY_BECAME_PRIMARY_SCN from v$databases;
```

Now determine the apply point for the RPT database.

```
SQL> select hwm_message_number  
          From v$streams_apply_coordinator  
          Where apply_name = 'APPLY_GLV';
```

```
SQL> select oldest_message_number from dba_apply_progress;
```

If apply reports a higher number than STANDBY_BECAME_PRIMARY_SCN then RPT has transactions that are no longer on PROD.

The Hard Part – Failover

PROD is ahead of RPT

This is the easy one. Recapture the missing changes.

Execute the failover.

Determine if PROD is ahead of RPT.

Stop the Capture process and drop the Propagation.

```
Begin
  Dbms_propagation_adm.drop_propagation ('PROP_PROD');
  Dbms_capture_adm.stop_capture('CAPTURE_PROD');
End;
/
```

On the destination find the last applied message.

```
select oldest_message_number from dba_apply_progress;
```

The Hard Part – Failover PROD is ahead of RPT

Continued

Reset the Capture Start_SCN

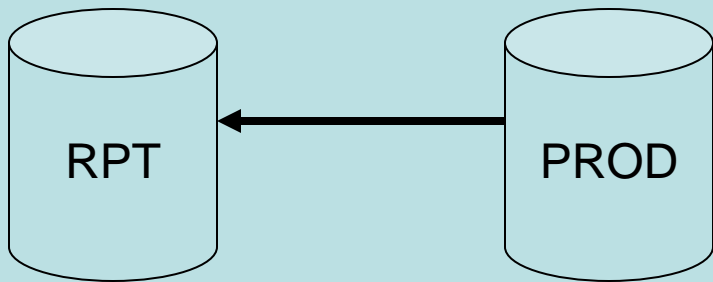
```
Begin
  Dbms_capture_adm.alter_capture(
    Capture_name=>'CAPTURE_PROD',
    Start_scn => <number for apply>);
End;
/
```

Recreate the Propagation. Restart Capture

```
Begin
  Dbms_propagation_adm.create_propagation(
    Propagation_name => 'PROP_PROD',
    Source_queue      => 'REP_CAPTURE_QUEUE',
    Destination_queue => 'REP_DEST_QUEUE',
    Destination_dblink => 'RPT.ORACLE.LOCAL',
    Rule_set_name     => 'RULESET$197');
End;
/
Begin
  Dbms_cpature_adm.start_capture('CAPTURE_PROD');
End;
/
```

The REALY Hard Part – Failover PRODSBY is behind RPT

- If PRODSBY is behind RPT, there are transactions applied to RPT that are lost on PROD.
 - Use a NULL TAG and apply the transactions to PROD.
 - Lose the transactions and resync the replication.



The REALY Hard Part – Failover PROD is behind RPT

- To replace the transactions in prod.
 - Set a null tag.
 - Manually add the transactions.
 - Restart replication.
 - Rebuild the Log Miner Dictionary retrieving the first_scn.
 - Reset the start_scn to the first_scn.
 - Recaptured transactions will not be reapplied at the dest.
 - Changes made under null tags will not be captured.
 - May need to drop and recreate the Capture.

The REALY Hard Part – Failover PROD is behind RPT

- Lose the transactions and resync the replication.
 - Reinstantiate the replication.
 - Flashback the RPT database.
 - Remember the scns of PRT and PROD are different.
 - Flashback small steps till the apply scn is behind the capture scn.
 - Once the RPT database is behind the PROD database, follow the earlier procedure.
 - Flashback must already be on to flashback the database.
 - If you flashback RPT then RPTSBY must be rebuilt.

Conclusion

- Streams and Standby work well together.
- Switchover operations are no data loss and easy to coordinate.
- Failover operations are more difficult and require some preparation (flashback on).

Contact Information

garmanyj@proxitec.com

www.proxitec.com

