IOUG :#861

“Tech refresh of existing system with ZERO downtime using RAC, ASM Technology”

We are sharing our experiences based on our observations at PayPal.

-- by Amit Das

PayPal Engineering Team

Introduction: about our team

- Sehmuz Bayhan – Our visionary director. Executed great changes in lightning speed.
- Saibabu Devabhaktuni – Our fearless leader at PayPal for at least 9 years.
  - http://sai-oracle.blogspot.com/
- Kyle Towle – Our fearless database architect at Paypal for at least 8 years.
- Dong Wang – Goldengate expert, speaker at multiple conferences, PayPal DBA for going on 7 years.
- John Kanagaraj – Author, Oracle ACE, frequent speaker at Oracle conferences
- Sarah Brydon – One of the very few Oracle Certified Masters.

Who Am I?

- 11 years in Oracle RAC Development team.
- Technical lead for world’s first Exadata production go-live (Apple), while at Oracle.
- Currently Engineering lead/architect for World largest Exadata OLTP system (PayPal).
- Frequent presenter inside/outside of Oracle.
- Love fishing.

PayPal’s Amazing Growth and Requirements

- Amazing Growth
  - Exponential growth in PayPal business year over year
- Business is growing rapidly
  - New users, features, transaction
  - New channels: POS, Mobile, etc
- Massive growth in database demand every year
  - Not uncommon to see database workloads grow 50-100% every year

One of the Largest OLTP database on Oracle

- Measured by Executions X Processes (concurrency)
- Fast paced VLDB OLTP environment on Oracle
  - 500+ database instances
  - OLTP databases commonly 10-130 TB
  - 5-14K concurrent processes
  - Executions ➔ 100K/sec,   11GB Redo/Minute
- Continuously growing
  - High growth of PayPal's business per year ➔ up to 2 X workload increase
  - Tier one databases built to support 300+K execs/sec

Agenda

H/W choice and validation
Pre-Work installation/configuration
Runtime Execution for ZERO downtime
Post-Work validation
Interconnect upgrade with ZERO downtime
If I were allowed to take 10 Minutes Downtime.

H/W choice and validation

- Build your cluster on lab first with new H/W
- Build your DB with exact same patch level as used in production.
- Use your best testing tool to test the DB and Oracle Clusterware; e.g.
  - RAT, SLAMD, Swingbench. Verify the test result and compare the AWR statistics.
- Find the break point for the new H/W in terms of user, load, CPU usage, etc…

Pre-Work step before software installation as “root” user

- (root) Edit /etc/host to add private IPs for all existing and new nodes.
- (root) Create the oracle user with proper permission and groups like your existing nodes.

Pre-Work step before software installation as “oracle” user

- (oracle) set ssh between existing nodes of the cluster and new nodes.
- (oracle) Verify the visibility of all ASM disks on new nodes.

Pre-Work step with cluvfy for new nodes qualification.

- (oracle)Run cluvfy:
  - cluvfy stage -pre nodeadd -n < new node1, new node2…> [-fixup [-fixupdir fixup_dir]] [-verbose]

Add node on existing GRID

- For this part we followed the DOC:
  - http://docs.oracle.com/cd/E11882_01/rac.112/e16795/adddelunix.htm#BEICADHD
- $ cd $GRID_HOME/oui/bin
- $ export IGNORE_PREADDNODE_CHECKS=Y

  (Sometime OUI will do some pre-addnode check and it may fail, if you are 100% sure that you can ignore the error with above setting)
- $ ./addNode.sh "CLUSTER_NEW_NODES={newnodes}" "CLUSTER_NEW_VIRTUAL_HOSTNAMES={newnodes-vip}"
- Follow the instruction for “root” user after running addNode.sh

Add node on existing ORACLE_HOME

- Followed exactly as per DOC:
  - http://docs.oracle.com/cd/E11882_01/rac.112/e16795/adddelunix.htm#BEICADHD
- $ cd $ORACLE_HOME/oui/bin
- $ ./addNode.sh -silent "CLUSTER_NEW_NODES={ new node1, new node2}"
- Follow the instruction for “root” user after running addNode.sh

Post Installation Check

- cluvfy stage -post nodeadd -n <new nodes> -verbose
- cluvfy comp admprv -o db_config -d $ORACLE_HOME -n <all nodes>
- (root) Disable CRS autostart while this Tech refresh
  - $GRID_HOME/bin/crsctl disable crs
- Inventory fix for GRID_HOME
  - ./runInstaller -updateNodeList ORACLE_HOME=$GRID_HOME "CLUSTER_NODES={All_nodes_list}" CRS=TRUE -silent
- Inventory fix for ORACLE_HOME
  - ./runInstaller -updateNodeList ORACLE_HOME=ORACLE_HOME "CLUSTER_NODES={All_nodes_list}"

Space Check and Create Thread and UNDO Tablespace for new instances

- Check the space for new redo thread.
- Check the space for new UNDO TS.
- Create threads for new instances.
- Create UNDO TS for new instances.

Runtime Execution for ZERO downtime

- Modify the DB resource in OCR to add new instances.
- Start one instance at a time on new nodes.
- Modify the service to start the services on new nodes.
- Stop the services on old nodes.
- Move the connections to new nodes.

Modify the DB resource in OCR to add new instances

- Registering the new instance with the existing DB resource in Oracle Cluster Registry (OCR) is mandatory.
  - ➢ Syntax: *srvctl add instance -d db_unique_name -i instance_name -n new_node_name*

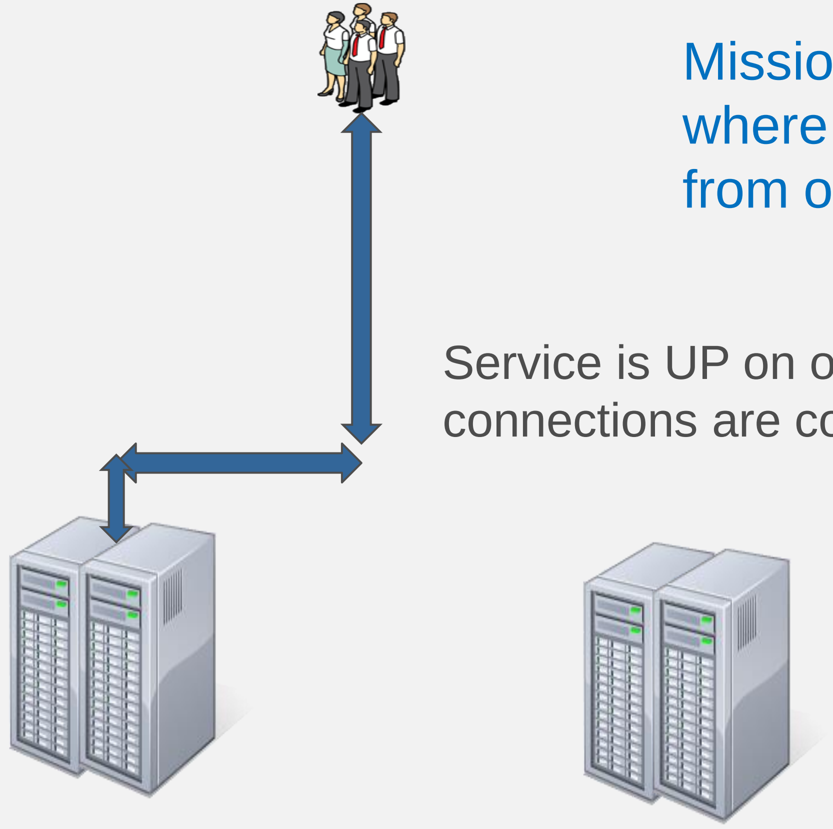
- Reason:
  - ➢ Without this step, you will not able to start your service on new nodes.
  - ➢ Without this step, you have to start the instance more than once to configure the system and OCR correctly.

Modify the service to start services on new nodes

- Modify the service from new node, then it will not stop the existing service on old node.
  - srvctl modify service -d <DB> -s SRV_PROD -n -i DB_old_1,….,DB_new_n
- Start the service on new node/nodes
  - crsctl start res ora.<DB>.srv_prod.svc –n <new_node>
- Stop the service from old node/nodes
  - crsctl stop res ora.<DB>.srv_prod.svc –n <old_node>

Service moving to new nodes – Pre.

Mission-critical Databases,
where service is moving
from old node to new node



Service is UP on only old and existing/new connections are connecting on older side

Old Nodes

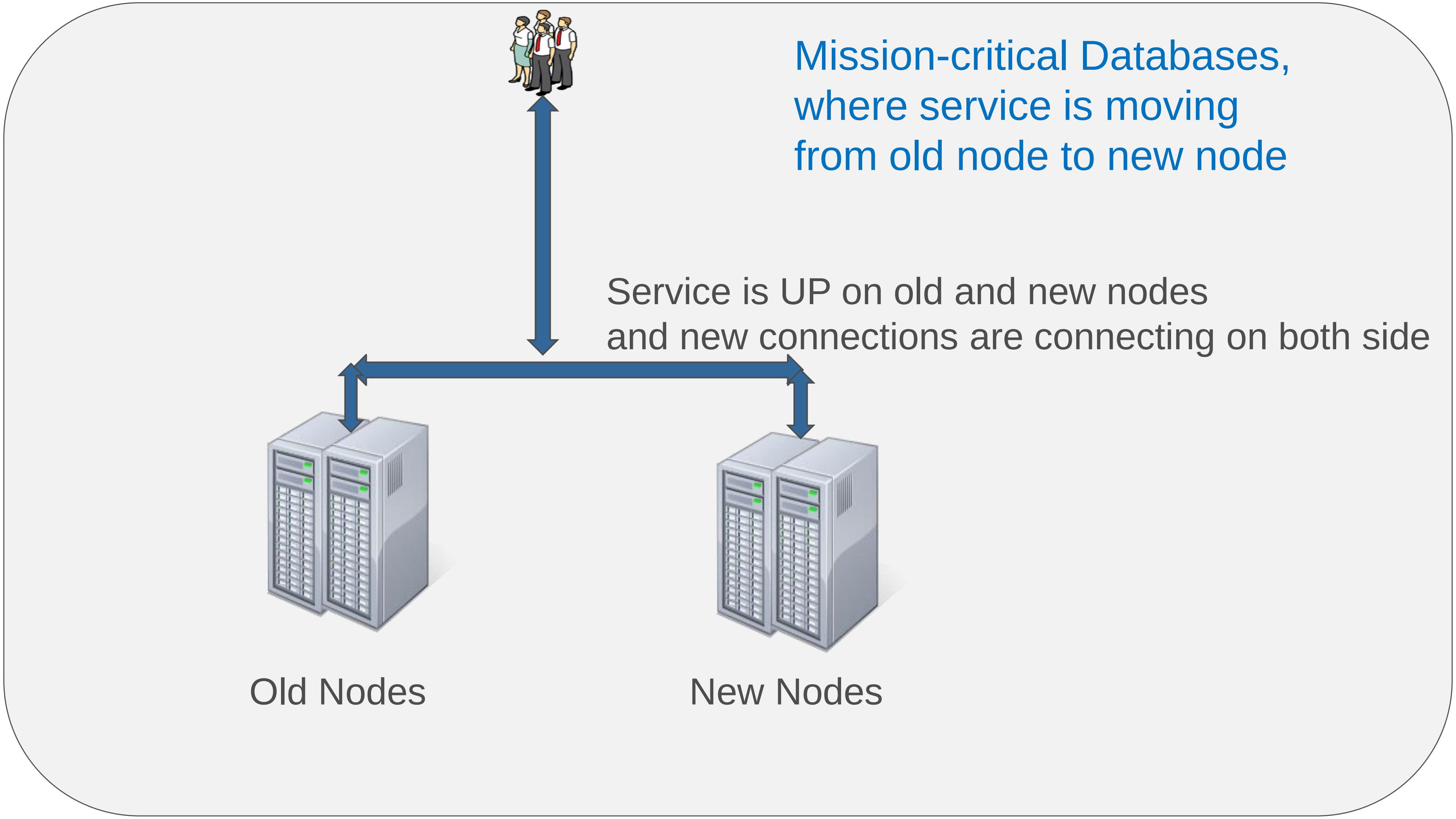
New Nodes

18

Sample TNS Entry to reconnect automatically.

```
@(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)
(HOST=<SCAN-NAME>)(PORT=<PortID>))
(CONNECT_DATA=
(SERVICE_NAME=<Service_Name>)
(FAILOVER_MODE = (TYPE=SESSION)
(METHOD=BASIC)(RETRIES=1000))))
```

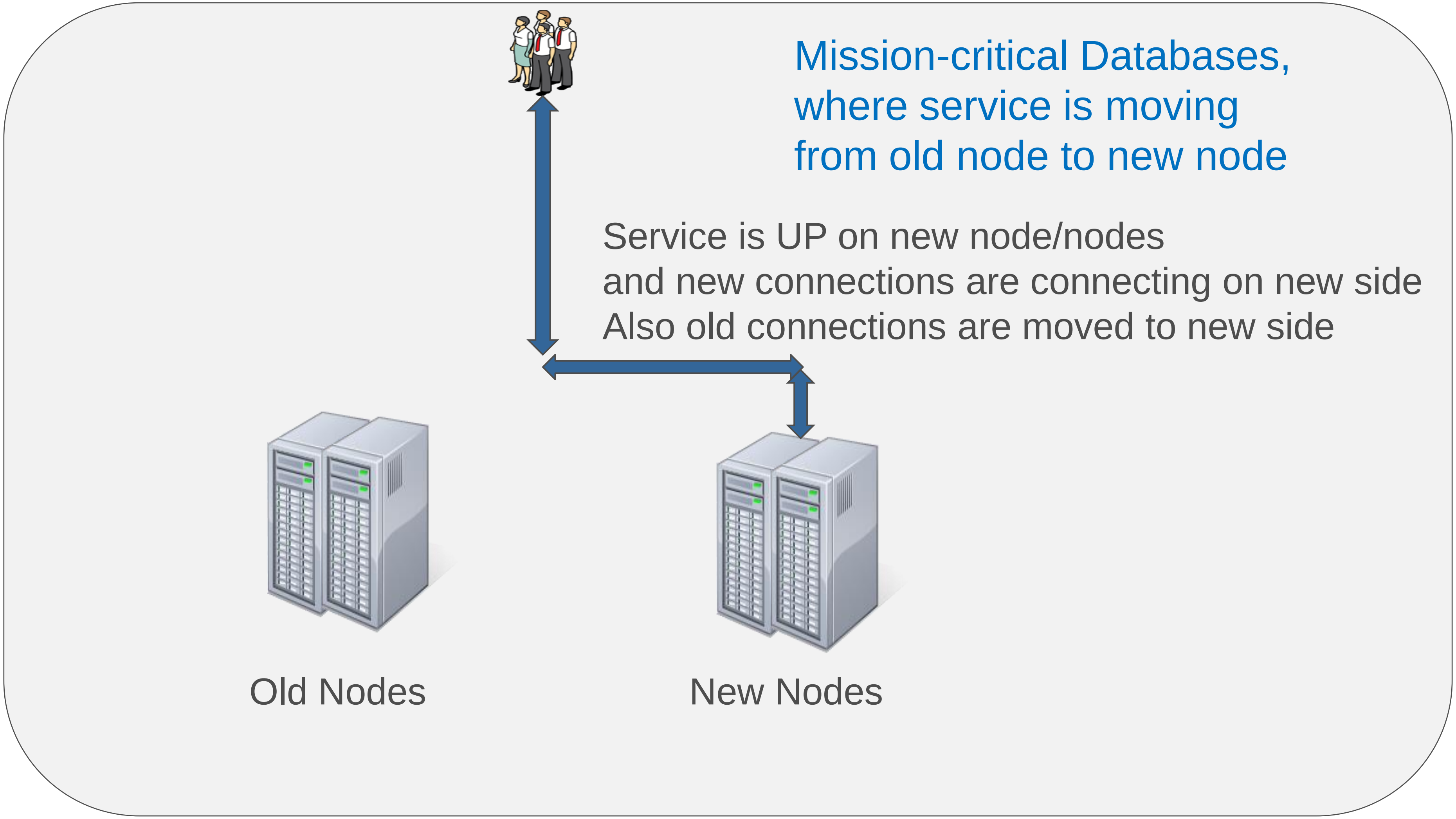
Service moving to new nodes



Mission-critical Databases, where service is moving from old node to new node

Service is UP on old and new nodes
and new connections are connecting on both side

Disconnecting from old nodes



Mission-critical Databases, where service is moving from old node to new node

Service is UP on new node/nodes
and new connections are connecting on new side
Also old connections are moved to new side

Health Check of the Apps and DBs

- Monitor the health of DB after 100% application move.
  - Query gv$session
  - Active session count
  - Lock/latch contention
- Monitor the health of H/W and networks.
  - Active session count at any point
  - Network load
  - CPU run queue count
  - Memory usage
  - I/O service time
- Monitor the Apps health.
  - PD/Apps team to validate their matrix

Remove old DB and nodes

- Modify the service to remove the old instance name from service, and run the syntax from old node.
  - srvctl modify service -d <DB> -s SRV_PROD -n -i DB_new_nodes_only

- Stop one instance one at a time from old nodes.
  - Srvctl stop instance –i <old_instance> -d <DBNAME>
  - To reduce the impact for RAC reconfiguration, do one old instance at a time.
- Remove the instance from OCR.
  - Srvctl remove instance –i <old_instance> -d <DBNAME>
- STOP CRS from all old nodes as a root.
  - $GRID_HOME/bin/crsctl stop crs

Continued…

- Follow the DOC:
  - http://docs.oracle.com/cd/E11882_01/rac.112/e16794/adddelclusterware.htm#BEIFDCAF
- Delete the node from CRS
  - crsctl delete node -n oldnode
- Remove binary from old nodes are optional.
- Fix the inventory for GRID_HOME in all new nodes.
  - $GRID_HOME/bin/runInstaller -updateNodeList ORACLE_HOME=$GRID_HOME "CLUSTER_NODES={all new nodes}" CRS=TRUE -silent
- Fix the inventory for ORACLE_HOME in all new nodes.
  - $ORACLE_HOME/oui/bin/runInstaller -updateNodeList ORACLE_HOME=$ORACLE_HOME "CLUSTER_NODES={all new nodes}" -silent".

Interconnect Upgrade from 1GigE to 10GigE

- While doing this operation, I would highly recommend using 1 node in the cluster and STOP CRS from rest of the nodes in the cluster.

Successfully done on AIX with ZERO downtime

- Add Backup 10Gbit adapter

  /usr/lib/methods/ethchan_config -a -b ent7 ent12

- Fail ethchannel from 1Gbit primary to 10Gbit Backup

  /usr/lib/methods/ethchan_config -f ent7

- Remove Primary 1Gbit interface

  /usr/lib/methods/ethchan_config -d ent7 ent3

- Add Primary 10Gbit adapter

  /usr/lib/methods/ethchan_config -a ent7 ent11

- Fail Etherchannel from backup 10Gbit to Primary 10Gbit

  /usr/lib/methods/ethchan_config -f ent7

If I were allowed to take 10 minutes downtime

My options are:

- Dataguard switchover.
  - Needs extra storage.
  - And the switchover
- Use Oracle Clusterware and ASM technology
  - My next slides will share the detail.

If I were allowed to take 10 minutes downtime

Pre-steps

- Build a separate new cluster.
- Make sure the new cluster can see the disks of existing cluster.
  - Select * from v$asm_diskgroup;
  - Select path from v$asm_disk;
- Copy all the init.ora from existing cluster to new cluster.
  - Need to change local and remote listener parameter.
- Create all the resource entry on new cluster
  - srvctl add database …..
  - srvctl add service …..
- Push tnsnames entry to clients

If I were allowed to take 10 minutes downtime

Run time execution

- Stop DB in existing cluster.
- Stop CRS on existing cluster.
- Mount the DiskGroup of existing cluster to new cluster.
- Start the DB on new cluster.
- Start the service on new cluster.

If I were allowed to take 10 minutes downtime

Post execution

- You can move the OCR and VD to old FRA group and move spfile of ASM from new FRA to old FRA.
- Drop the new FRA DG and use the old FRA DG.
- You can use old SCAN IP to new cluster, by modifying the SCAN resource, then you do not need to push tns entry.

Q/A?