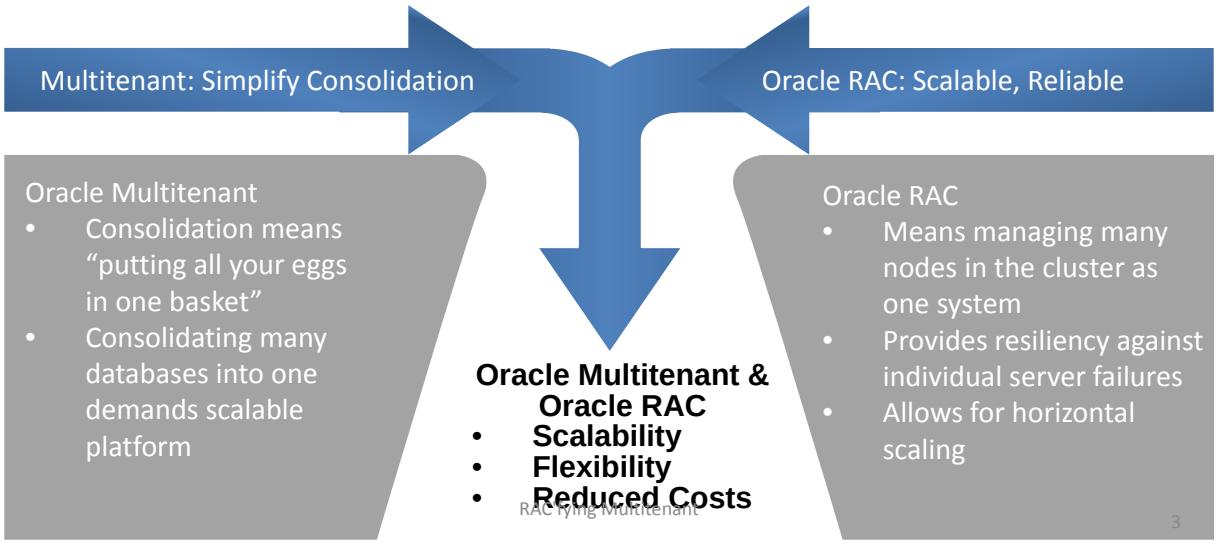
RACifying Multitenant

Arup Nanda
Principal Database Architect
Starwood Hotels
Deba Chatterjee
Principal Product Manager
Oracle Multitenant

Agenda

1. Introduction
2. Oracle Multitenant and RAC Basics
3. Why use database Services
4. Q & A

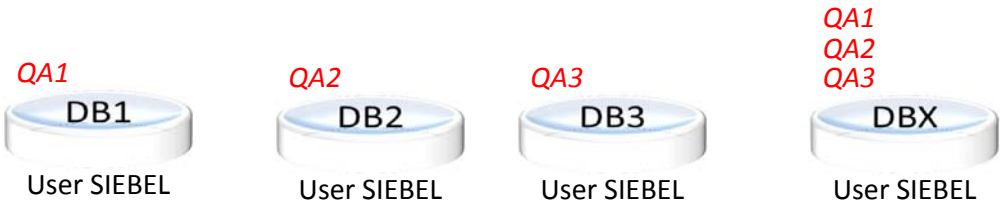
Oracle Multitenant and Oracle RAC



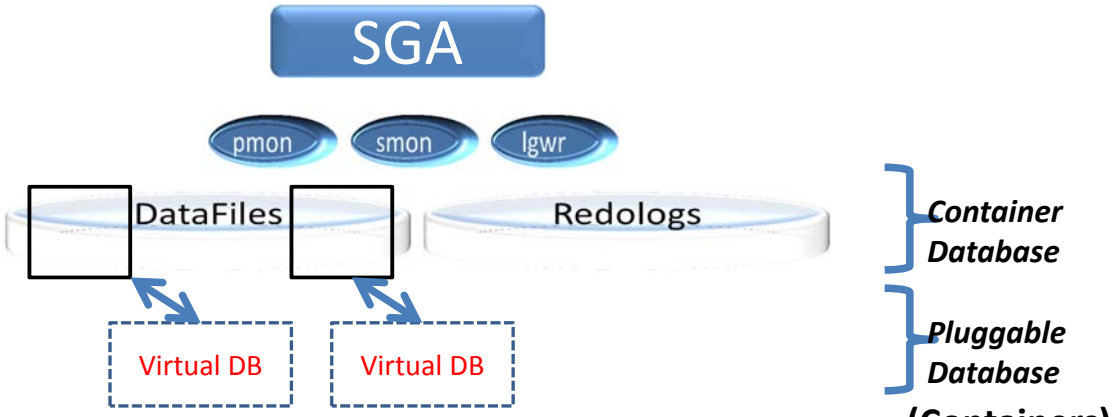
Agenda

1. Introduction
2. Oracle Multitenant and RAC Basics
3. Why use database Services
4. Q & A

Vexing Problem of Consolidation



Enter: Pluggable Database



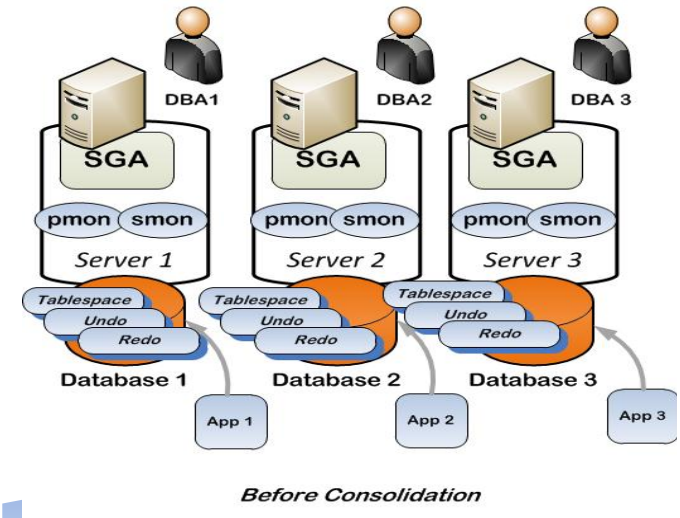
DBA_USERS

```
SELECT 1 AS CON_ID, NAME
FROM USER$@cdb
SELECT 2 AS CON_ID, NAME
FROM USER$@pdb1
SELECT 3 AS CON_ID, NAME
FROM USER$@pdb2
```

PDB1
CON_ID=2

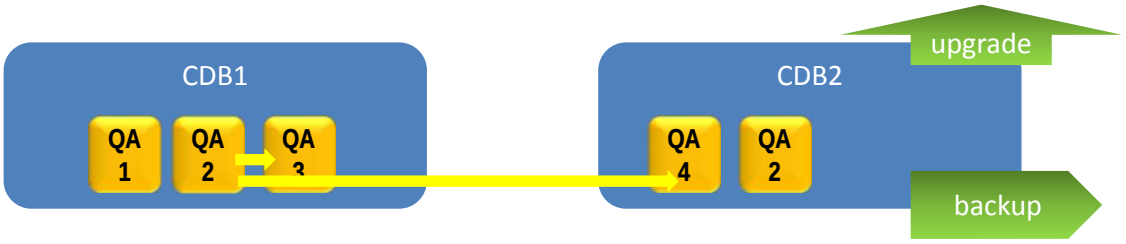
PDB2
CON_ID=3

PDB3
CON_ID=4



Before Consolidation

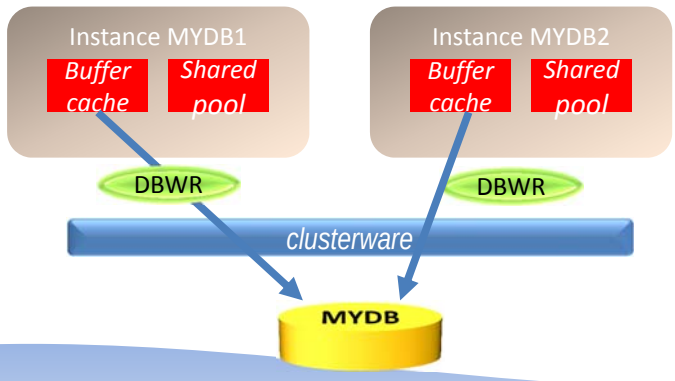
Why Multitenant



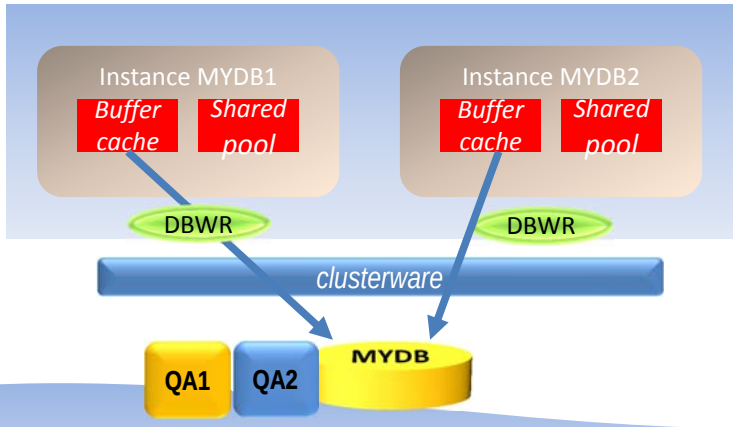
Cloning on same host
Cloning on a different host
Unplug and Plug
Upgrade/backup/ as a whole

Granular control when needed

Oracle RAC



RAC + Multitenant



Individualized Instances of PDBs

PDBs can be opened on selected instances

```
select name, inst_id, open_mode
from gv$pdbs;

NAME     IN OPEN_MODE
-------- -- ----------
PDB$SEED  1 READ ONLY
PDB$SEED  4 READ ONLY
PLUG1     1 READ WRITE
PLUG1     4 MOUNTED
PLUG2     1 READ WRITE
PLUG2     4 MOUNTED
SARPRD    1 READ WRITE
SARPRD    4 READ WRITE
```

Note: PLUG1 is opened on one instance and just mounted on the other.

Save State

- Set PDB SAVE STATE (12.1.0.2)
  - Service pulls up the PDB by default
  - Open mode must be consistent across the cluster

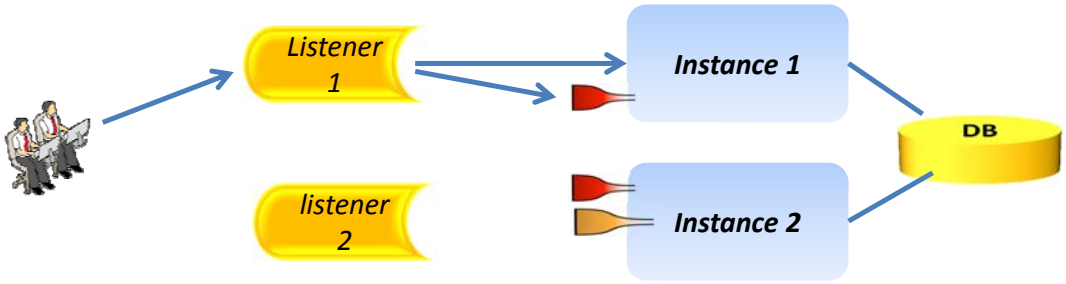
```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 TEST                           READ ONLY  NO
SQL>

SQL> alter pluggable database test save state;

Pluggable database altered.

SQL> alter pluggable database test open read write force;

Pluggable database altered.
```

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 TEST                           READ WRITE NO
SQL>

SQL> shutdown
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup
ORACLE instance started.

Total System Global Area 4.0668E+10 bytes
Fixed Size                  7653480 bytes
Variable Size            6442452888 bytes
Database Buffers         3.4091E+10 bytes
Redo Buffers              126562304 bytes
Database mounted.
Database opened.
SQL>

SQL> select inst_id, name, open_mode from gv$pdbs;

   INST_ID NAME                           OPEN_MODE
---------- ------------------------------ ----------
         1 PDB$SEED                       READ ONLY
         1 TEST                           READ ONLY
```


Agenda

1. Introduction
2. Oracle Multitenant and RAC Basics
3. Why use database Services
4. Q & A

Services in RAC

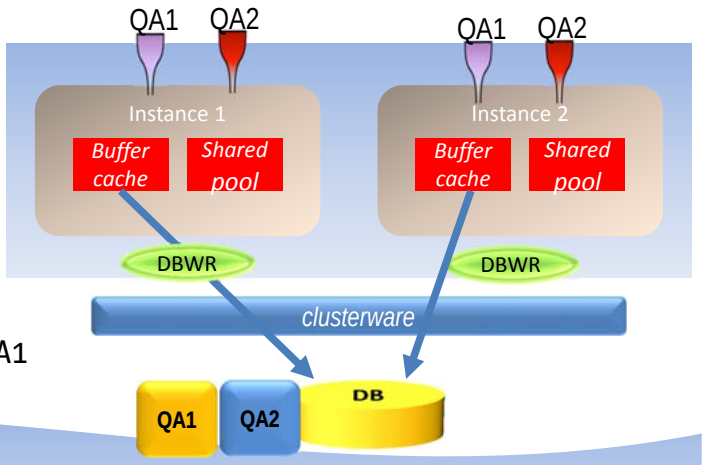



```
sqlplus <user>/<pwd>@SCAN/serv1
```

GV$SERVICEMETRIC
Service based Resource Manager

Services in Multitenant

- Default service of CDB is started
- Default service for each PDB is started



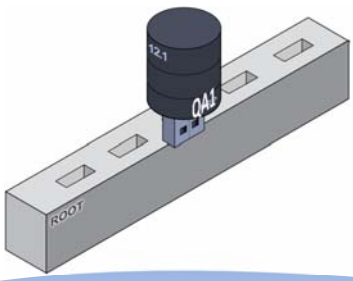

```
sqlplus <user>/<pwd>@SCAN/QA1
```

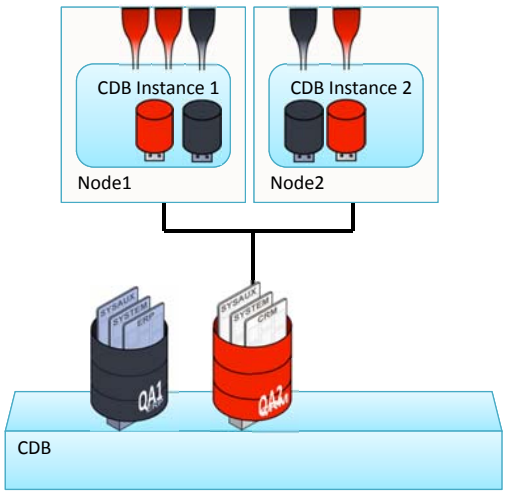
Do not Use Default Service

- PDBs provide location independence (plug/unplug/clone) but exist within a CDB
- Moving PDBs between CDB may need changing names of PDBs
- Use a unique service for an app
- If the PDB moves to another location the app does not need to change
- This requires uniqueness of service names

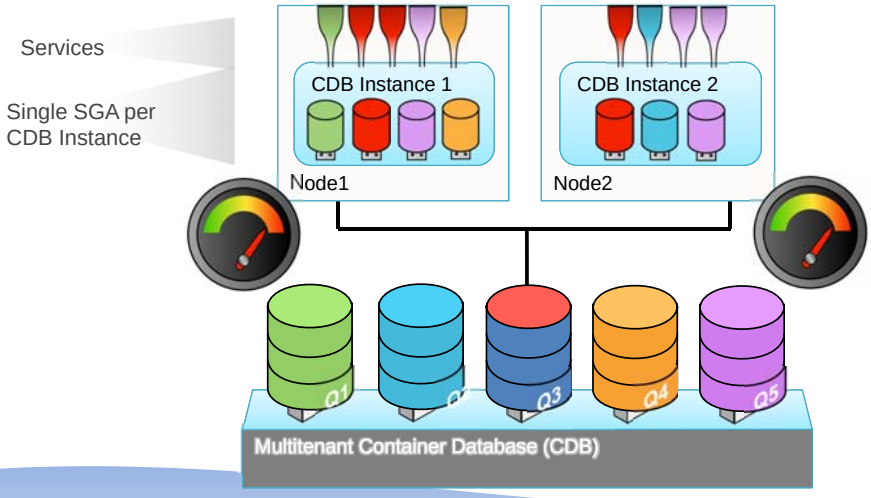
Converting to RAC

The simplest way of converting a SI PDB to RAC: unplug/plug

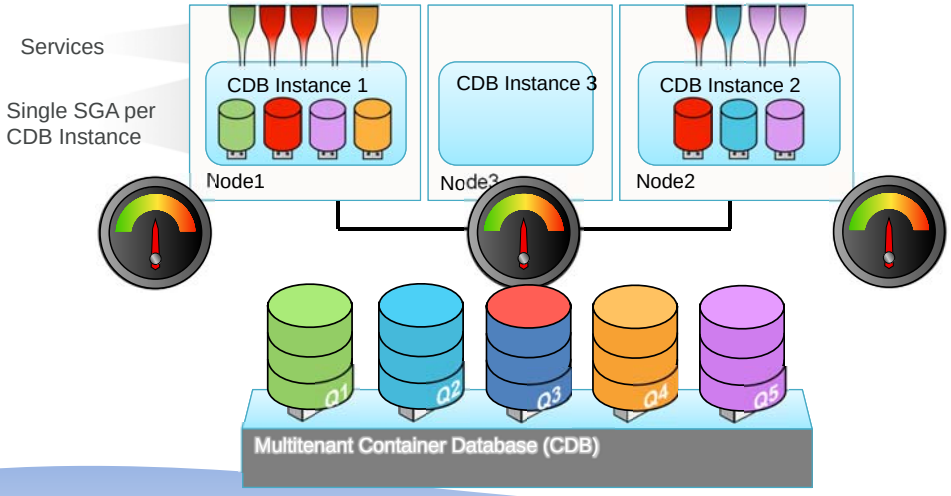




Changing Workloads



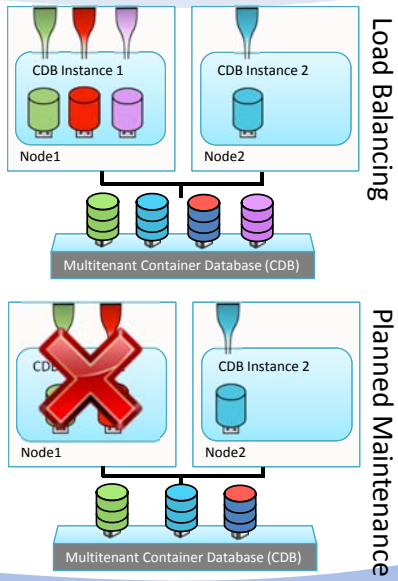
Changing Workloads



PDB Relocation

```
SQL> alter pluggable database pdb1 close
immediate relocate to 'cdb2';
```

- PDB Relocate
  - migrate some tenants off an overloaded node
  - migrate all tenants off a node that needs to be taken down for maintenance
    - without downtime for migrated tenants
    - minimal impact on QoS of migrated (and other) tenants



Takeaways

- During 12c upgrade, create a CDB, even if with just one container
- If you are migrating from another database, simply unplug and plug into the multitenant RAC database
- Later, as you consolidate, just create several containers instead of separate databases
- Create unique service names for all applications, even if they will share the same CDB, e.g. SALESPRD1 and SALESDEV1
- Make sure you understand the differences between CDB and PDB in V$ views

Agenda

1. Introduction
2. Oracle Multitenant and RAC Basics
3. Why use database Services
4. Q & A

Thank You!

Blog: arup.blogspot.com
Tweeter: @ArupNanda
Facebook.com/ArupKNanda