HOL9515
Delivering Continuous Database Services with Oracle Real Application Clusters (Oracle RAC) 12c

Mark Scardina
Director, Quality of Service Management,
mark.scardina@oracle.com

Troy Anthony
Cloud Evangelist, Oracle
troy.anthony@oracle.com
1 INTRODUCTION

1.1 LAB OBJECTIVE

This document details all actions that we will run during Oracle OpenWorld 2014 session Hands On Lab HOL9515 – Delivering Continuous Database Services with Oracle Real Application Clusters 12c.

Continuous database services is now possible with Oracle 12c technologies as will be demonstrated in this hands-on lab. Database 12c. If you are involved with database availability, performance or patching management of many Oracle databases, either using consolidated or dedicated resources or managing a database-as-a-service (DBaaS) environment, this hands-on lab is for you. There will be mini-tutorials explaining Rapid Home Provisioning, Application Continuity, and Quality-of-Service Management and how these technologies work together to offer you a database environment that helps you achieve higher availability more predictability through planned and unplanned downtime and demand surges to better meet your organization's policies and priorities.

This Hands-on-Lab is a one hour opportunity for you to work directly with a Multitenant Real Application Clusters Database and for you to perform the tasks necessary to configure and manage database services in order to have continuous availability during patching of a database home, unplanned instance failure, and demand surges.

There is far more content in these areas to cover in a brief, one hour session, so you are encouraged to use this lab as a starting point to explore these technologies and their associated best practices in the documentation, whitepapers, and collateral referenced in the “For Further Information” section.
1.2 ENVIRONMENT

During this lab, we will use a demo environment built on a single x86 laptop containing 3 virtual servers (Oracle VM VirtualBox virtual machines):

- Oracle RAC database node **Lightning**
- Oracle RAC database node **Thunder**
- Oracle Enterprise Manager Cloud Control server **EM12cREL4_DB12cPHH (EMCC)**

Lenovo
Oracle Linux Server 6.4
Memory 15.6 gb
Intel 4 Processor Core i5-3340M cpu 2.6ghz
1TB drive
GNOME 2.28.2
Kernel Linux 2.6.39-400.109.5 el6uek.x86_64

Host: ouser@hol-lap##
Users: ouser/Oracle1
        root/Oracle01
VMs:    Lightning OS Users: oracle/oracle
        Thunder OS Users: oracle/oracle
        EMCC OS User: oracle/welcome1
        EMCC: Admin User: sysman/welcome1
        grid/oracle;       Root User: root/oracle
        grid/oracle;       Root User: root/oracle
        Root User: oracle/welcome1
        QoS Admin: qosadmin/oracle12

Homes
        Grid Home: /u01/app/12.1.0.2
        DB Home: /u01/app/oracle/dbhome_1

1.3 THINGS YOU SHOULD NEVER DO
If you want to complete the lab, please do not:

- Close or shutdown any of the three virtual machines i.e accidently press “x” on any of three machines’ desktop’s upper right control buttons or select Close from the Machine Menu.

- Double click any of the icons like “Stop Enterprise Manager...” until instructed to do so. Otherwise, it may take quite a while to re-establish your environment so you can continue with the lab.

1.4 YOUR TOOLS

You will be using just a few tools during this lab:

- **Server Control** *(srvctl)* – a command line utility that can be run from a terminal window on Lightning and on Thunder as user “oracle”, password “oracle”. If you ever need to see the commands or command line options, just type: `srvctl -h`
- **Rapid Home Provisioning Control** *(rhpctl)* – a command line utility that can be run from a terminal window on Lightning and on Thunder as user “grid”, password “oracle”. If you ever need to see the commands or command line options, just type: `rhpctl --help`
- **Enterprise Manager Cloud Control** *(user=sysman, pw=welcome1)*
- **Oracle Quality of Service Management** from EMCC *(user=qosadmin, pw=oracle12)*

1.5 WHAT HAS ALREADY BEEN DONE FOR YOU

We have started:

- Each of the three virtual machines, Lightning, Thunder and EMCC.
- Oracle Real Application Clusters Multitenant database, “rhcdb” so that it is running on both nodes, Lightning and Thunder.
- Enterprise Manager Cloud Control (EMCC).
Log into your host with `ouser/Oracle1`. Then log into the Lightning and Thunder VMs as the Grid Infrastructure User with `grid/oracle`.

Check to make sure your environment is running by navigating to the desktop of the Lightning node and double-clicking the **Check Cluster** icon. Confirm that you have two nodes running.

Confirm the database is running on both nodes by double-clicking the **CheckDB** icon.

Finally, once you log in with the credentials EMCC is running by double-clicking the **Log into Enterprise Manager Console** icon if its log in screen is not already available. Do not actually log in at this time. *If any of the above are not running please raise your hand for assistance.*

### 1.6 SUMMARY OF STEPS FOR HANDS-ON LAB

This lab will proceed through these steps:

- Exploration of your cluster and database home inventory.
- Configure resilient services using Application Continuity and start workloads.
- Patch your database Oracle Home rolling and online with Rapid Home Provisioning
- Create a default QoS Management policy to measure your workloads
- Create a new policy to monitor Service Levels.
- Demonstrate unplanned downtime by killing an instance and seeing how services remain continuous.
- If you have time - Surge the workload to observe dashboard violations and bottlenecks.

### 1.7 CONVENTIONS USED

In this lab, the following conventions will be used:

- Terminal window commands and responses are in **Courier** font.
• Commands you type, such as `srvctl status database -db rhpcdb` are also highlighted in **bold**.
• Items you should pay special attention to either on the command line responses or in screenshots are highlighted in **yellow**.
• Controls or icons you need to use will be in bold such as **CheckDB**.
• Menu navigation will be presented in bold with pipe separators such as **Favorites | Cluster Database (rhpcdb)**.
2 INTRODUCING RESILIENT SERVICES BY ENABLING APPLICATION CONTINUITY

It is complex for application development to mask outages of database sessions and as a result errors and timeouts are often exposed to the end users and applications. Application Continuity attempts to mask outages from end users and applications by recovering the database session following recoverable outages. Application Continuity performs this recovery beneath the application so that the outage appears to the application as a delayed execution. For the recovery to succeed, the data and messages restored to the client by Application Continuity must be exactly the same as those that the application has seen and potentially made decisions on.

Application Continuity is invoked for outages that are recoverable, typically related to underlying software, foreground, hardware, communications, network, or storage layers. Application Continuity is used to improve the user experience when handling both unplanned outages and planned outages.

2.1 MODIFY THE SALES AND SALES CART SERVICE PROPERTIES TO MAKE USE OF APPLICATION CONTINUITY

You will now modify the properties of each of the database services to be able to support Application Continuity. There are a number of new attributes that need to be set. A full description of these are included it the documentation, but for know the values selected will work for this lab.

Go to the Thunder terminal window, connect as ORACLE by entering

```
su - oracle  then password oracle
```

You need to use the oracle user as this is a role-separated environment where the grid user owns the GI Home and the oracle user owns the DB Home.
Modify the sales and shipping services as follows.

1. `[oracle@thunder:~]>$ srvctl modify service -db rhpcdb -failoverretry 30 -failoverdelay 10 -failovertypetransaction -commit_outcome TRUE -replay_init_time 1800 -retention 86400 -notification TRUE -rlbgoal SERVICE_TIME -clbgoal SHORT -service sales

2. `[oracle@thunder:~]>$ srvctl modify service -db rhpcdb -failoverretry 30 -failoverdelay 10 -failovertypetransaction -commit_outcome TRUE -replay_init_time 1800 -retention 86400 -notification TRUE -rlbgoal SERVICE_TIME -clbgoal SHORT -service shipping

This enables these services to now take advantage of Application Continuity such that if there is a disruption at the database tier, the transaction will complete without the application or mid-tier having to resubmit it.

Note: It is only necessary to set the attributes FAILOVERTYPE and COMMIT_OUTCOME to enable application continuity. The other attributes alter characteristics of the failover during a replay attempt, as well as provide runtime load balancing and connection load balancing. All of these characteristics should be considered when enabling database services.

Examine the service configuration for both services to understand which attributes have been modified. The following example shows the shipping service:
[oracle@lightning ~]$ srvctl config service -d rhpcdb -s shipping -v

Service name: shipping
Server pool: gold
Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: true
Global: false
Commit Outcome: true
Failover type: TRANSACTION
Failover method:
TAF failover retries: 30
TAF failover delay: 10
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: SERVICE_TIME
TAF policy specification: NONE
Edition:
Pluggable database name: SHIPPING_PDB
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 1800 seconds
Session State Consistency: DYNAMIC
GSM Flags: 0
Service is enabled
Service is individually enabled on nodes:
Service is individually disabled on nodes:
Patching large software deployments can be time consuming and error-prone. Automation can help, but does not sufficiently reduce the complexity of supporting large software deployments. The existing processes were designed to enable customized deployments, where each home could be tailored to a different level and configuration.

As customers are starting to standardize their environments, new approaches to patching and upgrades become possible. In fact, customers often buy additional software to enforce standardization in their current environment, to prevent level and configuration drift.

With Rapid Home Provisioning, customers can now create a gold image for each standardized software image they want to support, and easily provision and update homes using these gold images. In this way, they need only apply a new set of patches once, to the gold image, and then make this new binary level available throughout their enterprise for new, as well as existing deployments.

Firstly let us examine what has been created for you. Login in to Thunder as the GRID user and execute the following commands:

```bash
[grid@thunder ~]$ rhpctl query image
Image name: IMGDB12

[grid@thunder ~]$ rhpctl query workingcopy -image IMGDB12
Working copy name: wcdb12
Working copy name: wcdb12patch
[grid@thunder ~]$
```
What this tells us is that we have one gold image, named IMGDB12, and this gold image has two workingcopies, wc12db, and wc12dbpatch built from it. A gold image is representative of an ORACLE_HOME template. It is stored in an ASM diskgroup. Each workingcopy is a writeable Oracle Automatic Storage Management Cluster File System (Oracle ACFS) snapshot of the gold image and is a deployed ORACLE_HOME.

In this example wcdb12 is the 12.1.0.2 base release, wcdb12patch, while built from the same image has had a patch applied to it.

More information is available on the image. Run a query against the image:

```
/grid@thunder ~]$ rhpctl query image -image IMGDB12
Image name: IMGDB12
Owner: grid@stormcloud
Site: stormcloud
Access control: USER:grid@stormcloud
Access control: ROLE:OTHER
Access control: ROLE:GH_IMG_PUBLISH
Access control: ROLE:GH_IMG_ADMIN
Access control: ROLE:GH_IMG_VISIBILITY
Parent Image:
Image Type: ORACLEDBSOFTWARE
Image Version: 12.1.0.2.0
Software home path: /u01/app/rhpstorage/images/iIMGDB12894396/swhome
Image state: PUBLISHED
Image size: 5333MB
Image platform: Linux_AMD64
Interim patches installed:
Complete: TRUE
```

As you can see, this image is Oracle database software (ORACLEDBSOFTWARE), is version 12.1.0.2, owned by the grid user, and is for the Linux_AMD64 platform. The Software home
A path is created on the Oracle ACFS volume. Examining the `/etc/mtab` file will show an entry for this path and that it is mounted as `acfs`.

```
[grid@thunder ~]$ cat /etc/mtab
/dev/mapper/vg_thunder-lv_root / ext4 rw 0 0
... 
/u01/app/rhpstorage/images/iIMGDB12894396 acfs rw 0 0
```

You have been told previously that we have an existing database, `rhpcdb`, running in this cluster. In order to find the ORACLE_HOME this database is using, we can query each workingcopy.

```
[grid@thunder ~]$ rhpctl query workingcopy -workingcopy wcdb12
Working copy name: wcdb12
Image name: IMGDB12
Owner: oracle@stormcloud
Site: stormcloud
Access control: USER:oracle@stormcloud
Access control: USER:grid@stormcloud
Access control: ROLE:GH_WC_ADMIN
Software home path:
/u01/app/rhpstorage/images/iIMGDB12894396/.ACFS/snaps/wwcdb12/
swhome
Oracle base: /u01/app/dbbase
Storage type: RHP Managed
Configured databases: rhpcdb
Interim patches installed:
Complete: TRUE
```

The line showing “Configured databases” indicates whether the database is being served from this home or not. **Make a note of the workingcopy which has rhpcdb as its configured database.**
Query the workingcopy `wcdb12patch` to show that a patch, `123456788`, is applied

Ensure that you are logged into the Lightning GI User Desktop. Start the workloads against the “sales” and “shipping” services respectively by using the `Start_Sales` and `Start_Shipping` buttons on the desktop:
In order to monitor the sessions during the move operation, open up two terminal windows on Lightning and in one window log in as the oracle user with `su - oracle` and password `oracle`.

In the second window SSH to Thunder as the oracle user with `ssh oracle@thunder` and password `oracle`.

In each window open up a SQLPLUS session using the command lines:

```
[oracle@lightning ~]$ sqlplus sys/oracle@lightning/rhpodb as sysdba
SQL> COL SERVICE_NAME FORMAT A10
SQL> select service_name, count(*) as count from v$session
    where service_name in ('sales', 'shipping') group by service_name;
```
During the Move Database operation below, re-execute each query by entering / at the SQL prompt.

We are now going to perform a maintenance operation, moving the database to the patched/unpatched Oracle Home being offered by the respective workingcopy.

```
[oracle@thunder ~]$ sqlplus sys/oracle@thunder/rhpcdb as sysdba
SQL> COL SERVICE_NAME FORMAT A10
SQL> select service_name, count(*) as count from v$session
where service_name in ('sales','shipping') group by service_name;
```

Execute as the `grid` user the command to move the database to a new workingcopy on lightning as follows.

**Note the sourcewc workingcopy should be the one with the configured rhpcdb database.**
[grid@lightning ~]$ rhpctl move database -sourcewc wcdb12 -patchedwc wcdb12patch --dbname rhpcdb

lightning.oracle.com: Starting to move database from 
"/u01/app/rhpstorage/images/iIMGDB12894396/.ACFS/snaps/wwcdb12/swhome" to 
"/u01/app/rhpstorage/images/iIMGDB12894396/.ACFS/snaps/wwcdb12patch/swhome" on 
server cluster 

lightning.oracle.com: Transferring data to 2 nodes 
lightning.oracle.com: 10% complete 
lightning.oracle.com: 20% complete 
lightning.oracle.com: 30% complete 
lightning.oracle.com: 40% complete 
lightning.oracle.com: 50% complete 
lightning.oracle.com: 60% complete 
lightning.oracle.com: 70% complete 
lightning.oracle.com: 80% complete 
lightning.oracle.com: 90% complete 
lightning.oracle.com: 100% complete 

lightning.oracle.com: starting to move the following databases: "rhpcdb"

lightning.oracle.com: SQL Patching tool version 12.2.0.0.0 on Wed Sep 17 16:18:44 2014
Copyright (c) 2014, Oracle. All rights reserved.
Connecting to database...OK

Note: Datapatch will only apply or rollback SQL fixes for PDBs 
that are in an open state, no patches will be applied to closed PDBs. 
Please refer to Note: Datapatch: Database 12c Post Patch SQL Automation 
(Doc ID 1585822.1)

Determining current state...done
Adding patches to installation queue and performing prereq checks...done
Installation queue:
For the following PDBs: CDB$ROOT PDB$SEED SALES_PDB SHIPPING_PDB
Nothing to roll back
The following patches will be applied:
123456788 ( ORACLE DATABASE PATCHSET UPDATE : 12.0.0.0 (12345)
catcon: ALL catcon-related output will be written to 
/tmp/sqlpatch_catcon__catcon_26055.lst
catcon: See /tmp/sqlpatch_catcon_* .log files for output generated by scripts
catcon: See /tmp/sqlpatch_catcon_* .lst files for spool files, if any

Installing patches...
Installation complete. Total patches installed: 4
Validating logfiles...done

SQL Patching tool complete on Wed Sep 17 16:26:16 2014
lightning.oracle.com: Completed the 'move database' operation on server 
cluster
The move database operation will be performed by default in a rolling fashion, but the database instances will be shut down, one after the other, and have patching operations performed on them automatically. Application Continuity will have protected the SwingBench application and no errors should have been seen.

This process takes approximately 15 minutes, therefore in the interest of your time, once the SQL query fails on one node indicating the first instance shutdown you may proceed to the next section and return here one the move has been completed.

NOTE: You may see a failure of the datapatch script that is executed at the end of the move operation. This is a known timing issue due to these highly loaded laptops and is fixed in the first 12.1.0.2 PSU. You may ignore the error.

Confirm that the database has indeed moved to the patched workingcopy:
4 RUNTIME MONITORING USING QOS MANAGEMENT

4.1 LOG INTO EM CLOUD CONTROL

From the EMCC node launch your browser using the **Login to Enterprise Manager Console** shortcut on the desktop. Once you see the EMCC login screen enter **sysman** as User Name and **welcome1** as password. This may take a few minutes to load the summary screen as this is the first startup.

```
[grid@thunder ~]$ rhpctl query workingcopy -workingcopy wcdb12patch
Working copy name: wcdb12patch
Image name: IMGDB12
Owner: oracle@stormcloud
Site: stormcloud
Access control: USER:oracle@stormcloud
Access control: USER:grid@stormcloud
Access control: ROLE:GH_WC_ADMIN
Software home path: /u01/app/rhpstorage/images/iIMGDB12894396/.ACFS/snaps/wwcdb12patch
Oracle base: /u01/app/dbbase
Storage type: RHP Managed
**Configured databases:** rhpcdb
**Interim patches installed:** 123456788
**Complete:** TRUE
```
User Name: sysman  
Password: welcome1

The next screen will be the summary for the entire system. It will show the targets up and unknown. This is normal as we are not running the EM Agent on the EMCC node in order to conserve resources.

4.2 ENTERPRISE MANAGER CLOUD CONTROL – ENABLE QUALITY OF SERVICE MANAGEMENT

QoS Management needs to be enabled on database to be monitored or managed. **In the interest of time, this has already been done for you but the steps are as follows:**

1. From the Availability Menu on the rhpcdb database target screen select **Enable/Disable QoS Management.**
2. Specify the Cluster and Database credentials, which would be grid/oracle and sys/oracle with role SYSDBA and click Login.
3. Enter the APPQOSSYS password twice and click OK.

This permits the QoS Management server to access the database using a secure database wallet to gather metrics and manage its resource plan.

The database is now enabled for QoS Management.

4.3 CREATE A DEFAULT POLICY SET TO MONITOR WORKLOADS

The next steps are to create a policy set to specify the databases and workloads to be monitored. The QoS Management pages are hosted on the Cluster Target.

1. Go to the Cluster Target by selecting Favorites I stormcloud (Cluster) I stormcloud.
2. Go to the QoS Management Policy Set wizard by selecting Administration I Quality of Service Management I Create Policy Set.
3. Log into the QoS Management server with the following credentials:

   **User:** qosadmin  
   **Password:** oracle12

4. On the “Create Policy Set: General” page, make sure that all the server pool boxes are checked under “Manage.” This specifies the pools that will be monitored or managed. Click **Next**.

5. This is the Performance Classes page that allows you to specify the actual workloads to monitor or manage. Note that all the services have been automatically discovered and are listed as Performance Classes. Click **Next**.
6. The next page is the Create Policy Set: Classifier Ordering Performance Class Order page that allows specifying the filter order in case the definitions overlap. Note that all of the “Classifiers’” session parameters are mutually exclusive hence order is unnecessary. Click **Next**.

7. This is the Create Policy Set: Performance Policies page and you will see that a DefaultPolicy is already created. We will add custom policies later. Click **Next**.

8. This is the Set Policy page which specifies the initial policy from the set to activate. The one policy to activate at this point is the DefaultPolicy. Click **Next**.

9. The final page presents a review of the complete policy set before submission. Click **Submit Policy Set** and you will be taken to the QoS Management Dashboard.
The Dashboard shows the work running in each service or Performance Class.

10. In the upper right of the dashboard, change the refresh cycle to **60 seconds**.

11. It takes a couple of minutes for the data to be collected for the “Resource Use vs. Wait Time” display.

12. Hover your mouse over any of the blue bars and you will see the amount of response time that is consumed by the use of resources for that Performance Class (workload).
13. Hover your mouse over the gray bar to see the amount of response time that is consumed by waiting for resources to become available. The two bars add together to represent total response time as the middle tier or application server would see it.

5 CREATING A NEW POLICY & IMPLEMENTING IT

5.1 CREATE A NEW "MONITOR" POLICY

To demonstrate the policy-based capacity flexibility, create a new "Monitor" policy.

1. On the QoS Management Dashboard click Current Active Policy: DefaultPolicy to go to the Policy Editor Wizard page.
2. On the Policy Editor “General” page, make sure that you check “Manage” for all the server pools as this will cause them to be tracked and managed by QoS Management.
3. Click Next 3 times until you get to the Performance Policies Page.
4. Click Copy Policy and this will take you to a new page where you can enter policy name and details.
5. Enter **Monitor Policy** as the Policy Name and an optional description such as "**This is the monitor policy.**"

6. Enter **Objective Values** that are approximately 2x that shown for sales_pc and shipping_pc. You can leave Default_pc as 0 since it is for unclassified work. Set the sales_pc Rank to **Highest** and the shipping_pc rank to **High**.

7. Click **OK** to move to Set Policy page and then click select the **Monitor Policy** and click **Set Policy** and then **Next**.

8. To save this policy, click **Submit Policy Set** on the Policy Set Review Page. You will be taken back to the QoS Management Dashboard.

9. Confirm all the workloads are still running on the Dashboard and that the Monitor Policy is your Current Active Policy. Again, set your refresh rate to **60 seconds**.

10. Hover your mouse over the green bar to see the headroom available before the performance objective is exceeded.

You now have completed putting QoS Management into Monitor Mode and enabling EMCC’s metric notifications to send warning and critical alerts for violations.
6 UNPLANNED OUTAGES

Application Continuity will protect against planned and unplanned outages. In Section 5 you should have completed a planned maintenance task, moving the database to a patched version of the Oracle Home, and not seen any application errors.

This section will show that unplanned outages are also protected by Application Continuity, and enabled by database services and Fast Application Notification (FAN).

As the grid user, identify the PMON background process for the database:

```
$ ps -ef | grep pmon
grid      3309  2422  0 03:00 pts/0  00:00:00 grep pmon
grid      5573     1  0 Sep15 ?  00:00:01 asm_pmon_+ASM1
grid      6680     1  0 Sep15 ?  00:00:01 mdb_pmon_-MGMTDB
oracle   31681     1  0 02:36 ?  00:00:00 ora_pmon_rhpcdb_1
```

As the root user, kill the PMON background process for the database instance ora_pmon_rhpcdb_1

```
$ su -
Password:
[root@lightning ~]# kill -9 31681
[root@lightning ~]#
```

The database instance will die, but the services sales and shipping are still offered on the database instance running on the other node. New connections will be sent to the surviving instance, and the in-flight work will be protected by application continuity – so no errors should be seen as confirmed by the workload terminal windows.

Check the response times and demand by returning to the QoS Management Dashboard and clicking on the Details link in the upper section.
While you will see an increase in response time, demand will be unaffected and the application will not see any errors.

7 IF YOU HAVE TIME: CREATING A WORKLOAD SURGE

The final phase of preserving continuous database services is to respond to demand surges. The **Performance Satisfaction Metric** in QoS Management is a metric collected by EMCC and is available to assign a threshold for Warning and Critical alerts. These alerts are based upon the number of continuous seconds of violation per performance class. For this final section, you will create a demand surge that will cause a violation to the performance objectives that you previously set in your Monitor Policy.

1. From Lightning, start second client workloads for Sales and Shipping by double-clicking their icons and wait to ensure all four workloads are reporting transactions.
2. Switch to the EMCC QoS Dashboard and note whether you have workloads exceeding their objectives where the gray bar portion now displays as red with the performance objective see as a single blue line.
3. Scroll down to the **Resource Wait Times Breakdown** to identify the bottleneck resource, which is the column whose value is largest.
4. Click on the Performance Class in the first column of one that is exceeding its objective in order to view the increase in demand and increase in response time.

**Congratulations, you have just completed the Hands On Lab!**
This following are relevant links to the technologies contained in this Hands-On Lab for further information and reference.

- [Rapid Home Provisioning Documentation](#)
- [Rapid Home Provisioning on oracle.com](#)
- [Application Continuity Documentation](#)
- [Application Continuity on OTN](#)
- [QoS Management Documentation](#)
- [QoS Management on OTN](#)