Oracle on IBM Power systems Best Practices

Operating system checks to perform to review Oracle Performance

Tutorial and Lab Exercises

Andrew Braid - andrew.braid@fr.ibm.com

IBM Oracle Center Montpellier

Majidkhan Remtoula – majidkhan.remtoula@fr.ibm.com

IBM Oracle Center Montpellier

Frederic DUBOIS - Fred.dubois@fr.ibm.com

Global Power Competitive Sales Support

Contents

i

Course Objectives

The objective of this hands-on Lab class is to share best practices for running Oracle database on AIX/Power Systems.

The document is divided into two parts :

1/ Oracle & AIX/Power Best Practices Review

2/ Performance Healthcheck using OraDetective Tool
(Asset created by IBM Advanced Technology Group, ISV on Power - Oracle, North America)

The 1st part can be run by **IBMers and Partners** in order to review all components that need to be checked prior starting Performance Tuning Analysis. These are only Database Infrastructure related.

The oraDetective Tool in the 2nd part is an IBM Internal Tool and only accessible by IBMers. The tool analyzes existing Oracle database workloads to quickly identify workload bottlenecks and tuning opportunities and provides analysis results in a customer deliverable report.

The document here will describe required data source and format to leverage OraDetective Tool and will provides an illustration applied to our current environment.

We encourage Partners to contact their local Power Client Technical Specialist and engage OraDetective Execution Request. Sample reports based on Lab Workload are provided as illustration.

It can be used :

- To prepare a Power System refresh opportunity
- as 1st action plan to identify and solve a performance problem customer may meet on their Power System Infrastructure with their Oracle Database.
- Before engaging a more in-depth Performance HealthCheck Analysis by IBM Lab Services Consultant.

Lab Completion Steps

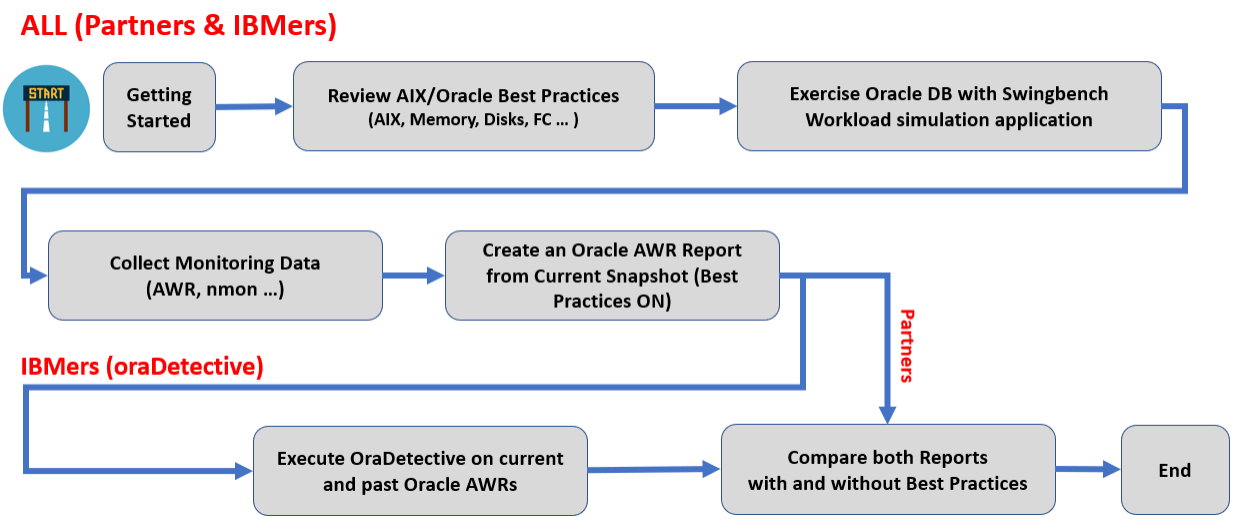
This course will walk through the following steps:

ALL

- Review of Oracle/AIX settings
- Simulation of a transactional (OLTP) workload within the pre-installed Oracle database via the charbench application
- Collection of monitoring data (Oracle AWR, nmon)
- Creation of an Oracle AWR report from the period when we executed the charbench workload
- Compare behaviours and performance impacts on the same workload when the Best Practices are ON or OFF.

IBMers Only

- Execution of OraDetective tool based on the historic Oracle AWR report and the current one you captured with best Practices ON.



Prerequisites

This Hands-On Lab recommends you to complete the following learning prerequisites in order to have a 1st theoretical overview about Oracle DB on AIX/Power Best Practices that you will review and apply in practice.

Mandatory learning prerequisites:

Power/AIX/Oracle best practices Part 1 CPU & Memory and Part 2 I/O & Networking sessions delivered by Ralf Schmidt-Dannert at 2021 TechU

- Oracle database on AIX best practices: Part 1 - Memory and CPU Link
- Oracle database on AIX best practices: Part 2 - I/O and more Link

Recommended learning:

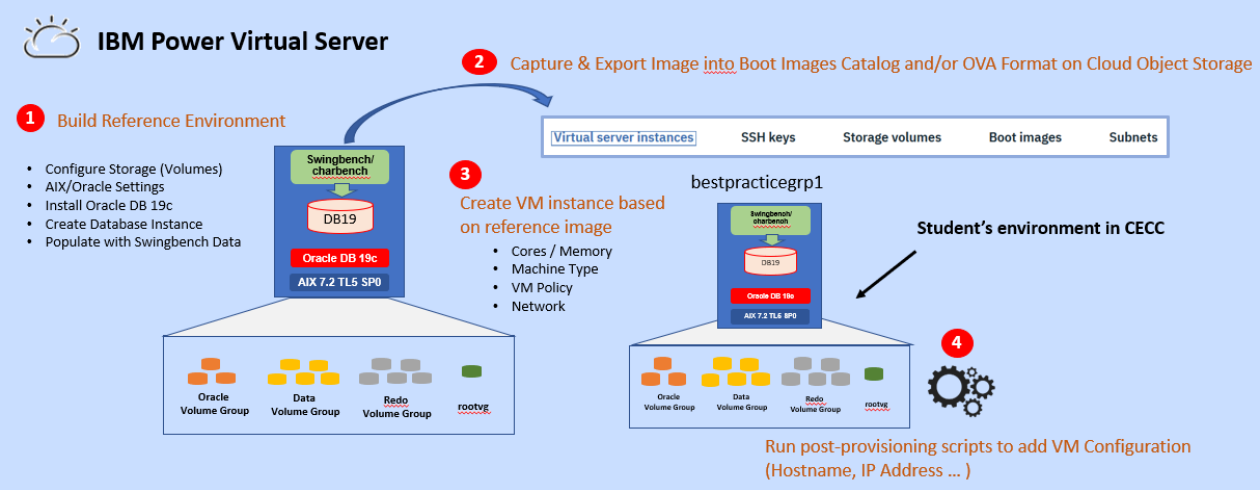
- **Managing the Stability and Performance of current Oracle Database versions running AIX on Power Systems including Power9** (**regularly updated**) ← bookmark this document
- **Oracle on POWER L1 Course** to understand why Power is the best server on which to run Oracle Database - from both a technology perspective and a cost perspective. (IBM Link and BP Link)
- **Oracle on POWER L2 Course** focuses on POWER Positioning against Oracle Exadata. (IBM Link and BP Link)

Hands-On Lab architecture

Architecture

This environment used for this Hands-On Lab is deployed in IBM Power Virtual Server. We have created an AIX 7.2 TL5 SP1 Image with all AIX & Oracle Best Practices applied and we then build VM in Power Virtual Server based on this image.

The same environment is deployed as detailed below for anybody running this Hands-On Lab.

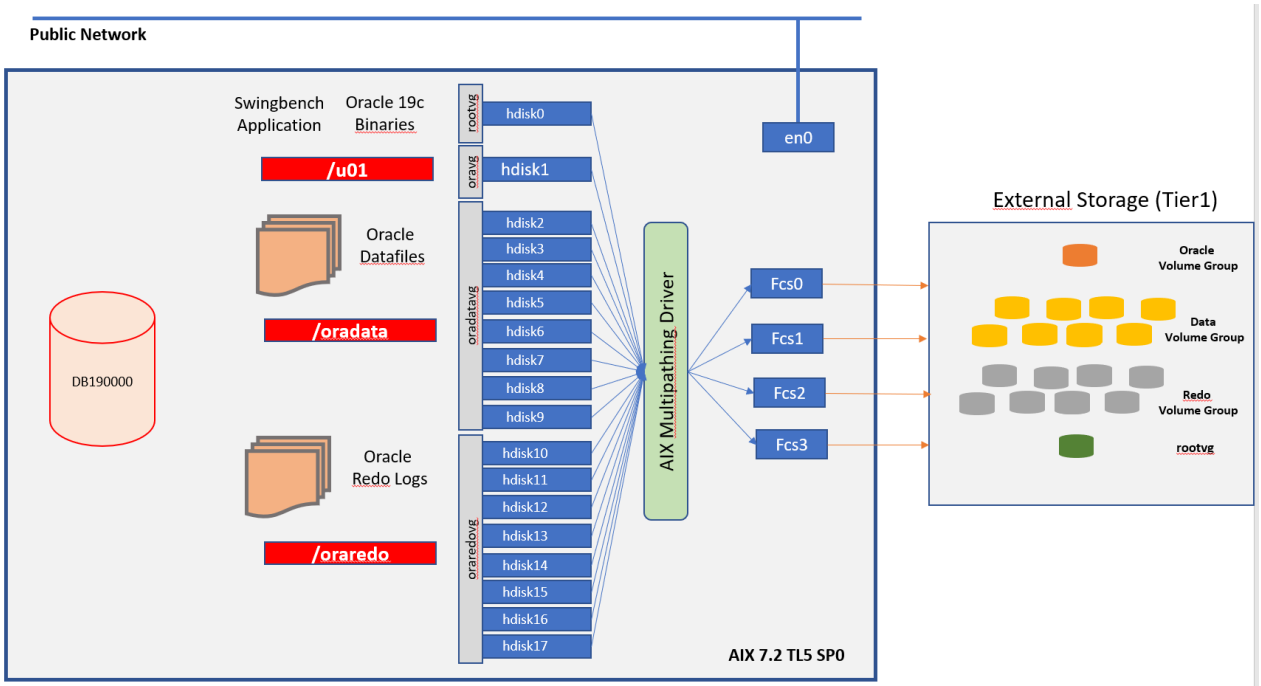


The following AIX/Oracle Best Practices apply to both environments whether you run Oracle DB on AIX/Power System on-premise or in Power Systems Virtual Server. However, as we do not have access to the underlying layers such as VIOS/HMC in Power Virtual Server, those settings will not be checked during the lab but guidance will be provided for Power Systems on-premise deployment.

Environment

Your Lab Environment is about an AIX 7.2 TL5 SP1 LPAR.
Oracle 19c version patched to the latest update is already installed. A database called "db190000" has been created and populated with 36GB of data in the Swingbench/Charbench application schemas.

We created a logical volume for database code installation, another one to host Oracle datafiles and a 3rd one for Oracle redo logs. The LPAR has 1 core assigned (Shared Capped) and 36GB of memory.



Getting Started

When the following symbol appears in this Hands-On Lab Document, Copy / Save values specific to your own environment into a Notepad or other text application. Those values will be used later for following exercise.

Before starting this Hands-On Lab, you initiate the creation of the Lab environment through Technology Zone Portal and have received a mail with a URL pointing to environment details.

Lab Details are under “Environment Summary” and “Account Credentials”.

Collect details and Save Values according to your own environment.

ENVIRONMENT SUMMARY

Below is a consolidated list with some key information about the externally provisioned environments for this project:

Type	Hostname	Alias	IP Address	External IP Address	External Hostname	Processor	Memory	OS	Reservation	Status
PowerVM Guest	d11310-pvs1	d11310-pvs1.pvs.cecc.ihost.com	192.168.140.230	158.176.147.230	e6.93.b09e.ip4.static.sl-reverse.com	POWER9	36864	AIX 7200-05-01-2038	CECC_11310_1	provisioned

- Hostname : **<given_hostname>**
- IP Address (Public IP) : **<given_public_ip>**

In our case and for the rest of the document, the Hostname given to the LPAR is **bestpracticesoracle-d11310-pvs1**

ACCOUNT CREDENTIALS & STORAGE

Root Account

By default direct root or primary administrative logins over SSH are disabled when possible. Unrestricted access to root is provided through sudo to all the users mentioned below.

Project Account

Every project has a generic project user account created locally on each provisioned environment when at all possible. The initial password for this account is as below although it can be changed after the fact on a case by case basis as desired. A SSH key pair is automatically generated for this account at the project level and the keys are provided below as well.

Note in IBM Cloud, Azure and AWS (except for Windows OS) only SSH keys can be used for remote access so if you have not saved a public SSH key for injection for your persistent user account you may need to use the project user private key below to access the enviroment. For Windows operating systems, you must use login/password authentication.

Note that the password for this account is also used on all the systems as the root or primary administrative account password.

Also note if you copy and paste the private key to use make sure a new line character is present at the end of your copy of it. Some SSH clients will complain the key is in an invalid format if the newline character at the end is missing.

Username	cecuser
Password	z!+L4b_G7dHKEGw

- User/password : **cecuser/<given_password>**
- Oracle user password defined in the LPAR is ***oracle1bm***

Connect to the environment

From your laptop, initiate an ssh connection to the AIX LPAR using putty.exe or other solutions.

You can download putty.exe file from the following website :

https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

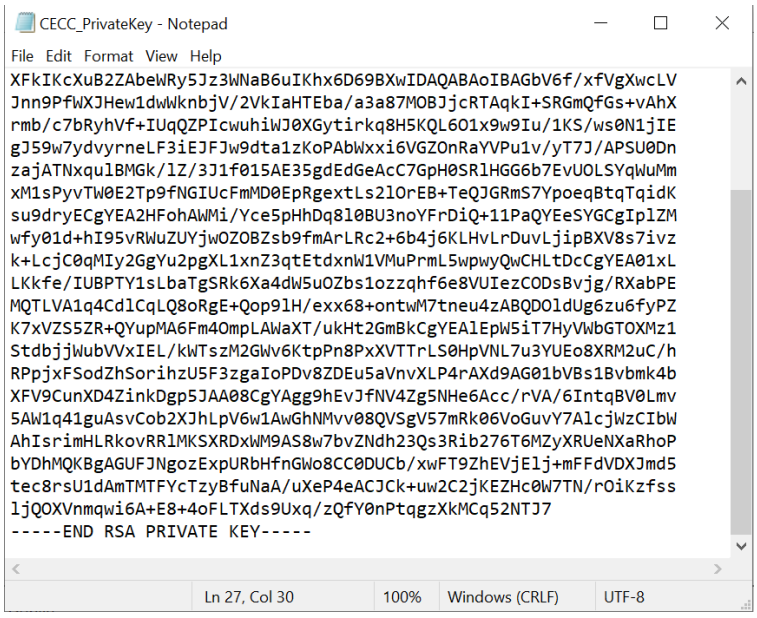
Save the Private Key provided in your CECC Environment Summary Page into a Notepad / Text File

Username	`cecuser`
Password	`z!+L4b_G7dHKEGw`

Private Key	`-----BEGIN RSA PRIVATE KEY-----`
	`MIIEowIBAAKCAQEAsrOBdVwe9rVAFQIPEDijXS25UO8+jN22DHKggAfqXFeAe6Ta`
	`9QxuthBewHwEDG8OM4aoqlQvDVbd9YR6cf7Z05cJnp7nnKS/g9JgjVQi8bdqdJNg`
	`HzVntt81LqFY0LBLyQUzxMjrcCeOEgUYY5Tf2t3T/dGRXPSYK6R0E76ZPRcNh18s`
	`wg0jvjoMG1cPyQXgNlZDhglU/0LlFfZEnnPs5akYtTDVZJXpPdeXGF229FJHj2j+`
	`ME8szNwrzf+CPNWhy0l5aK6R6jFq+JkwJBcQJjZzyWiFTxENQbpF2WDnObvl6kOE`
	`XFkIKcXuB2ZAbeWRy5Jz3WNaB6uIKhx6D69BXwIDAQABAoIBAGbV6f/xfVgXwcLV`
	`Jnn9PfWXJHew1dwWknbjV/2VkIaHTEba/a3a87MOBJjcRTAqkI+SRGmQfGs+vAhX`
	`rmb/c7bRyhVf+IUqQZPIcwuhiWJ0XGytirkq8H5KQL6O1x9w9Iu/1KS/ws0N1jIE`
	`gJ59w7ydvyrneLF3iEJFJw9dta1zKoPAbWxxi6VGZOnRaYVPu1v/yT7J/APSU0Dn`
	`zajATNxqulBMGk/lZ/3J1f015AE35gdEdGeAcC7GpH0SRlHGG6b7EvUOLSYqWuMm`
	`xM1sPyvTW0E2Tp9fNGIUcFmMD0EpRgextLs2lOrEB+TeQJGRmS7YpoeqBtqTqidK`
	`su9dryECgYEA2HFohAWMi/Yce5pHhDq8l0BU3noYFrDiQ+11PaQYEeSYGCgIplZM`
	`wfy01d+hI95vRWuZUYjwOZOBZsb9fmArLRc2+6b4j6KLHvLrDuvLjipBXV8s7ivz`
	`k+LcjC0qMIy2GgYu2pgXL1xnZ3qtEtdxnW1VMuPrmL5wpwyQwCHLtDcCgYEA01xL`
	`LKkfe/IUBPTY1sLbaTgSRk6Xa4dW5uOZbs1ozzqhf6e8VUIezCODsBvjg/RXabPE`
	`MQTLVA1q4CdlCqLQ8oRgE+Qop9lH/exx68+ontwM7tneu4zABQDOldUg6zu6fyPZ`
	`K7xVZS5ZR+QYupMA6Fm4OmpLAWaXT/ukHt2GmBkCgYEAlEpW5iT7HyVWbGTOXMz1`
	`StdbjjWubVVxIEL/kWTszM2GWv6KtpPn8PxXVTTrLS0HpVNL7u3YUEo8XRM2uC/h`
	`RPpjxFSodZhSorihzU5F3zgaIoPDv8ZDEu5aVnvXLP4rAXd9AG01bVBs1Bvbmk4b`
	`XFV9CunXD4ZinkDgp5JAA08CgYAgg9hEvJfNV4Zg5NHe6Acc/rVA/6IntqBV0Lmv`
	`5AW1q41guAsvCob2XJhLpV6w1AwGhNMvv08QVSgV57mRk06VoGuvY7AlcjWzCIbW`
	`AhIsrimHLRkovRRlMKSXRDxWM9AS8w7bvZNdh23Qs3Rib276T6MZyXRUeNXaRhoP`
	`bYDhMQKBgAGUFJNgozExpURbHfnGWo8CC0DUCb/xwFT9ZhEVjElj+mFFdVDXJmd5`
	`tec8rsU1dAmTMTFYcTzyBfuNaA/uXeP4eACJCk+uw2C2jKEZHc0W7TN/rOiKzfss`
	`ljQOXVnmqwi6A+E8+4oFLTXds9Uxq/zQfY0nPtqgzXkMCq52NTJ7`
	`-----END RSA PRIVATE KEY-----`

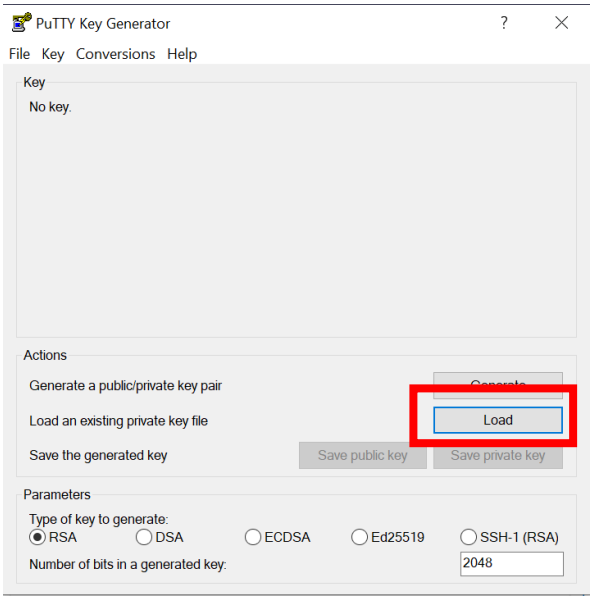
Public Key	`ssh-rsa`
	`AAAAB3NzaC1yc2EAAAADAQABAAABAQCys4F1XB72tUAVAg8QOKNdLblQ7z6M3bYMcqCAB+pc`
	`cecuser@cecc-11310`

Make sure a new line character is present at the end of your copy of it. Some SSH clients will complain the key is in an invalid format if the newline character at the end is missing.

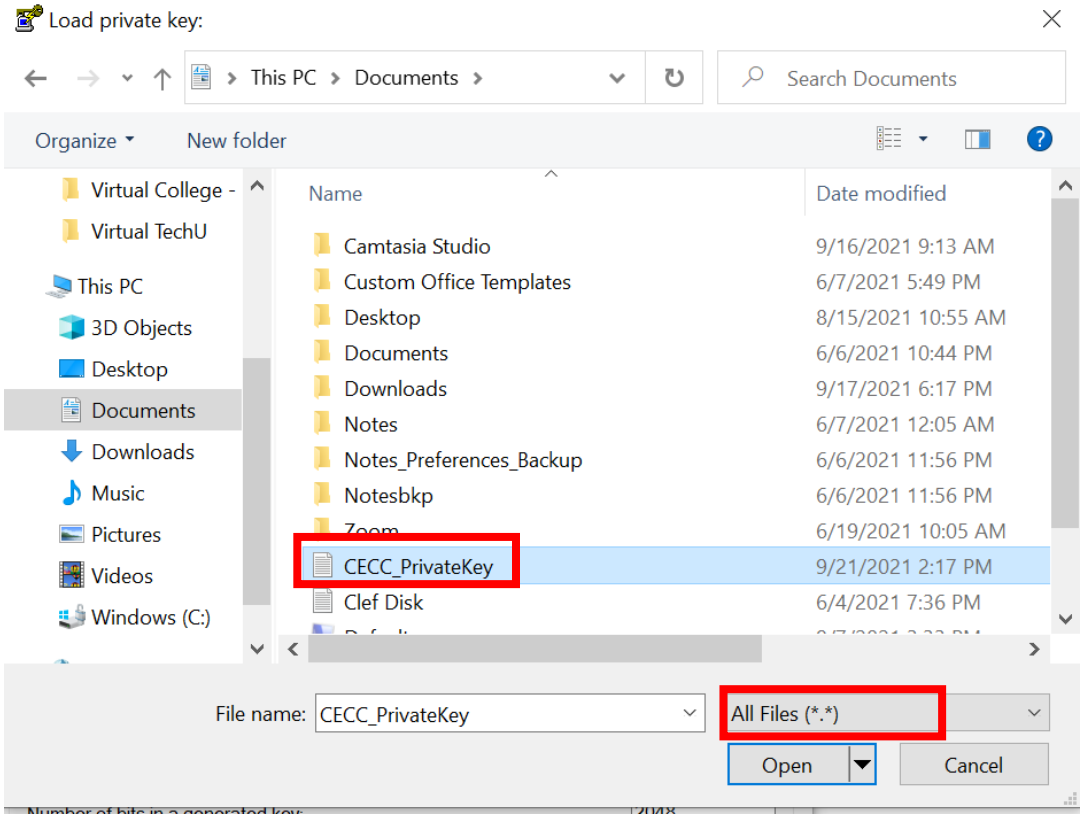


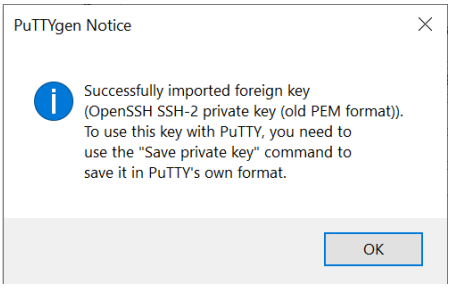
PuTTY stores keys in its own format in `.ppk` files. However, the tool can also convert keys to and from other formats.

Go to Windows **Start menu** → **All Programs** → **PuTTY** → **PuTTYgen**.



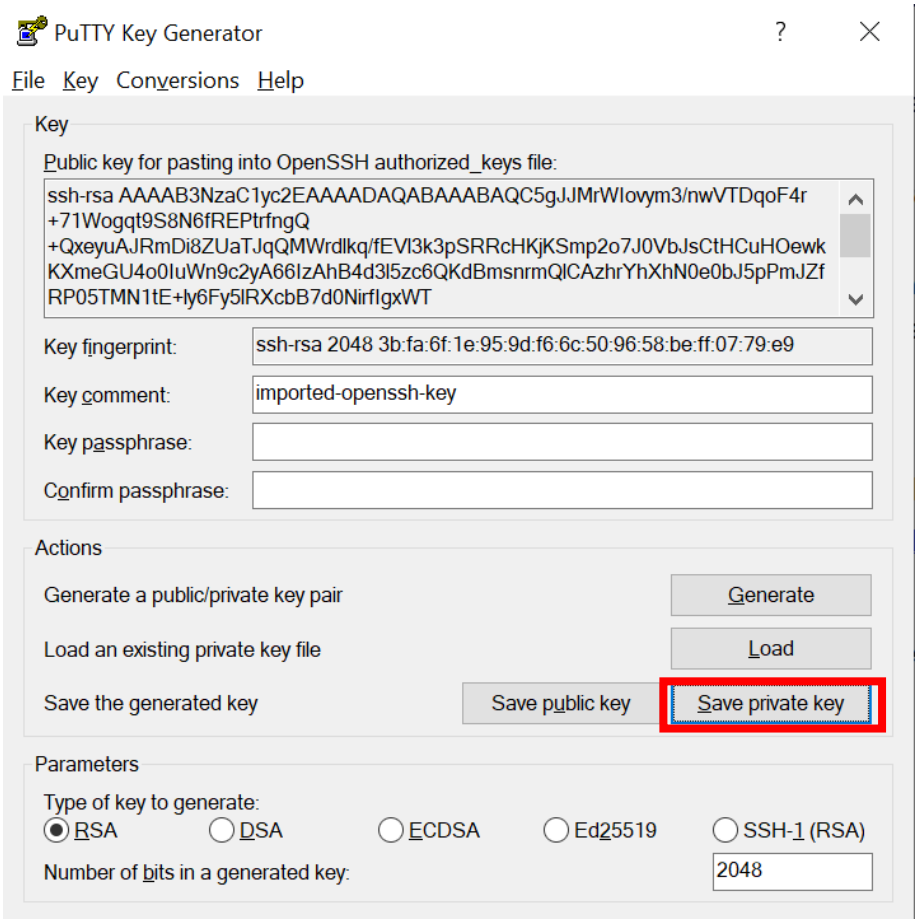
Click on “Load” Button and provide path of the text file you saved your private key.





Click on “Save Private Key” button and Provide Path to the private key into ppk putty format.

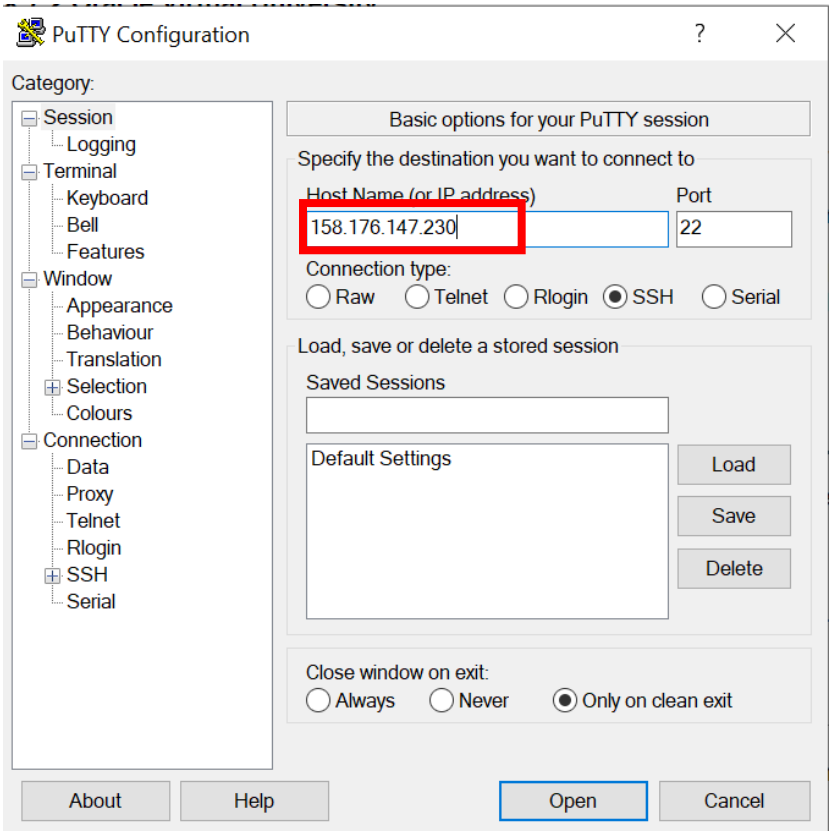
“C:\Documents\CECC_PrivateKey.ppk” in our example



Now you can proceed with Putty Executable and private Key Usage.

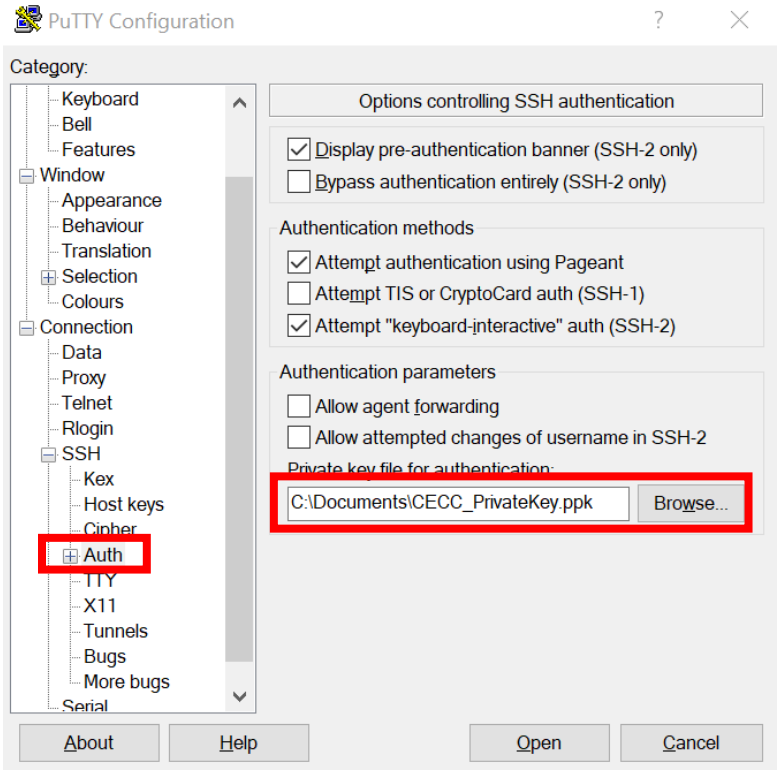
Launch Putty.exe Application.

Provide connections details to putty such as the external IP Address that has been assigned to your environment.

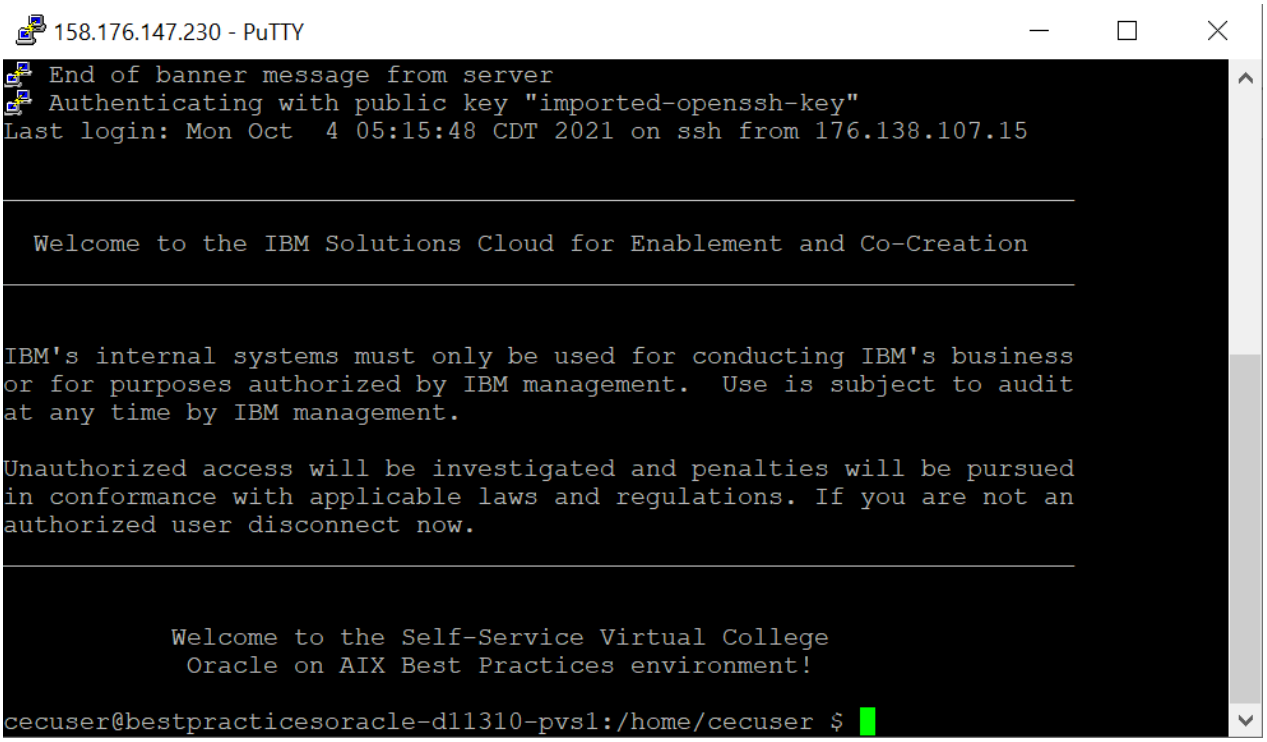


Navigate to *Connection* → *SSH* → *Auth* from the sidebar menu.

Click on the *Browse* button to look for your *PuTTY Private Key*.



Click on the *Open* button to start the connection.



You are now connected as cecuser and can start the Hands-On Lab 😉 !

You can perform sudo -i command to switch to root user.

Section 1 : Checking that the Best Practices have been applied

The 1st section of this Lab focuses on reviewing AIX & Oracle Best Practices that have been applied on your environment.

If you meet a different value for one of the following settings you are about to review on a PoC/Benchmark or customer environment, we would recommend to copy the corresponding parameter and clarify reasons leading to a different value.

This is a standard list of checks that we would perform on the Power Server of a client reporting performance issues.

1/ Check the firmware

The firmware should be as up-to-date as possible

To check the current FW level you will need to know the model of the Power Server and the Firmware level, both of these can be found with the prtconf command in AIX.

As root/cecuser, execute the following commands:

```
prtconf | grep "System Model"
prtconf | grep "Firmware" (or "lsmcode -c" command gives it directly)
```

```
cecuser@bestpracticesoracle-d11310-pvs1:/home/cecuser $ sudo -i
root@bestpracticesoracle-d11310-pvs1:/ # prtconf | grep "System Model"
System Model: IBM,9009-22A
root@bestpracticesoracle-d11310-pvs1:/ # prtconf | grep "Firmware"
Platform Firmware level: VL950_075
Firmware Version: IBM,FW950.11 (VL950_075)
root@bestpracticesoracle-d11310-pvs1:/ #
```

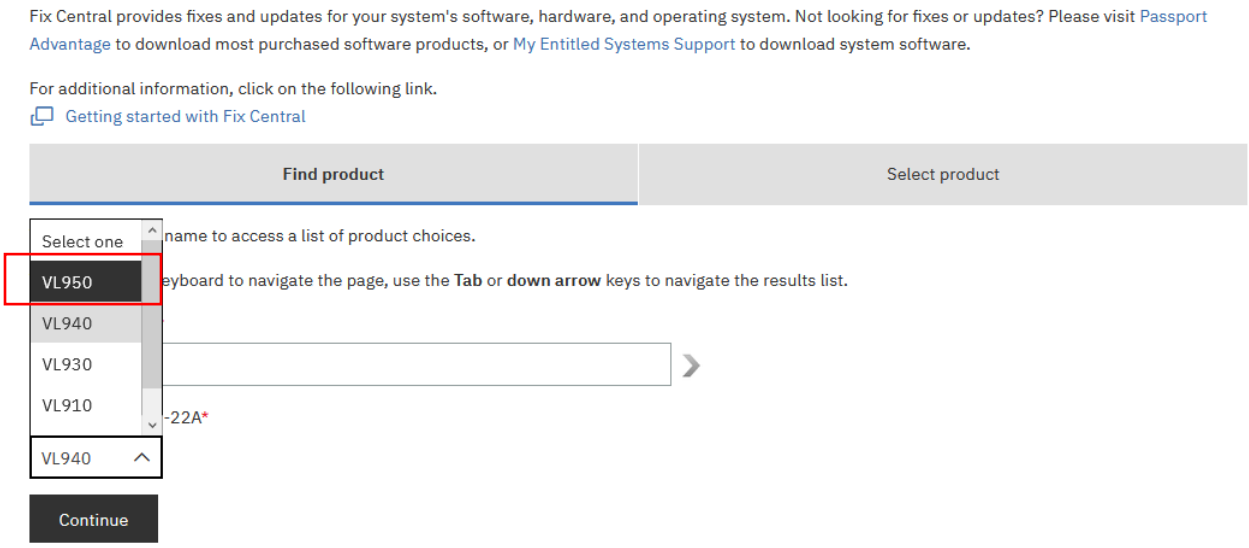
Note the Model and Firmware Version.

Check the latest firmware at the following url:

https://www-945.ibm.com/support/fixcentral

Scroll down to the Product selector box

Enter the system model. The web page will propose a link to fill the Product Selector code, click on the appropriate link. A further drop down will appear with the firmware version. Choose the firmware version noted above and click continue. Details of the latest versions of the firmware will then be displayed.



As you can see, our environment is running at the most recent Firmware version. If customer environment is not while you are reviewing its configuration, a recommendation would be to upgrade Firmware to VL950 Version (or newer if available).

You can also check if you are running at the latest level of your version.

You can see that there is a VL950_087 level available as we are running on this environment with VL950_075 level.

Once again, a Best Practice would be to update to that latest level if not possible to upgrade to most recent firmware version.

An alternative method to check Firmware updates can be done via the Fix Level Recommendation Tool (FLRT). The Fix Level Recommendation Tool (FLRT) provides cross-product compatibility information and fix recommendations for IBM products. https://www14.software.ibm.com/support/customercare/flrt/

Use FLRT to plan upgrades of key components or to verify the current health of a system. Enter your current levels of firmware and software to receive a recommendation. When planning upgrades, enter the levels of firmware or software you want to use, so you can verify levels and compatibility across products before you upgrade.

2/ AIX Level

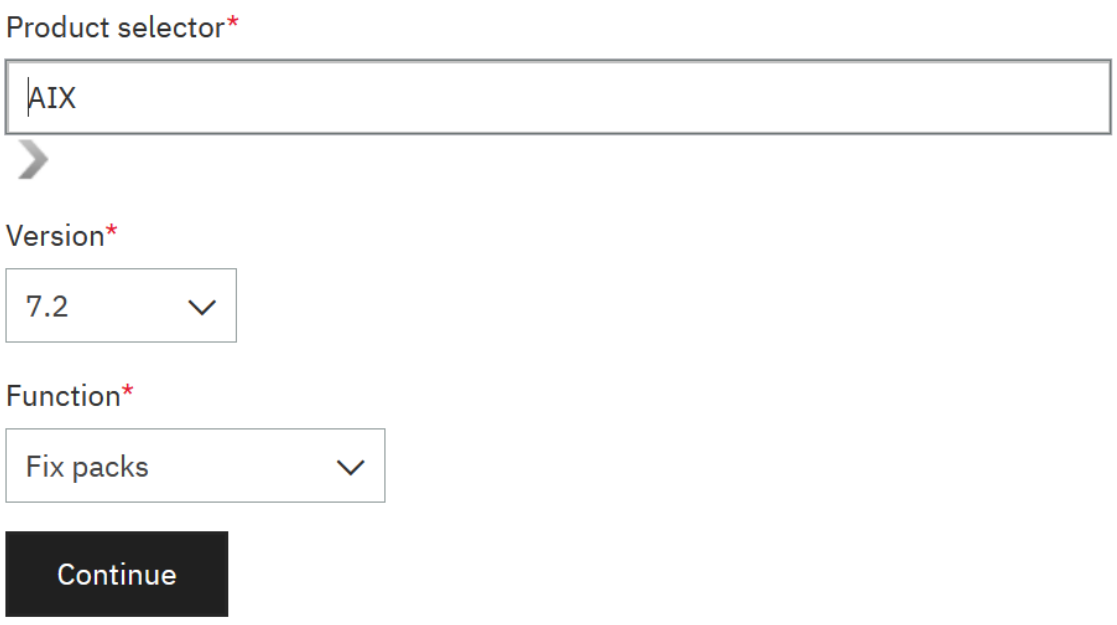
Check the level of the AIX operating system using the command:

oslevel -s

```
root@bestpracticesoracle-d11310-pvs1:/ # oslevel -s
7200-05-01-2038
root@bestpracticesoracle-d11310-pvs1:/ #
```

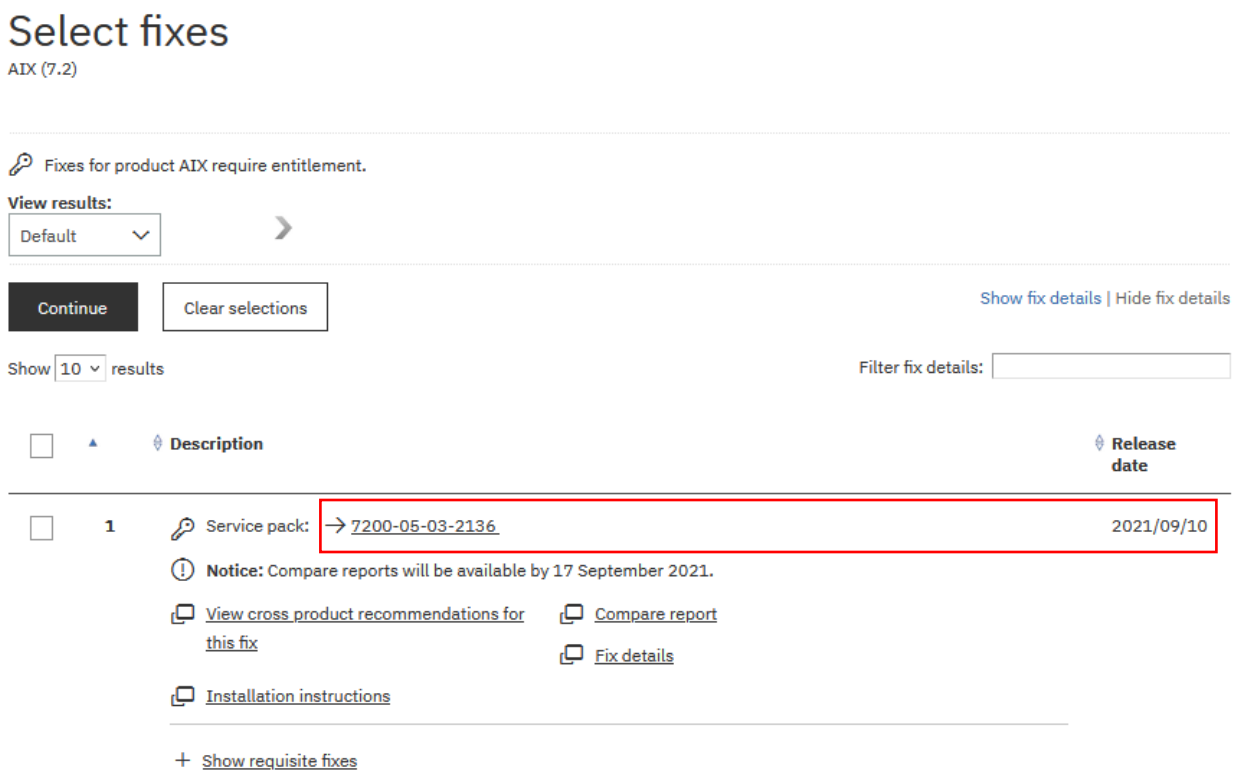
The latest AIX Technology Level (aka TL) or Service Pack (aka SP) can be verified and downloaded from https://www-945.ibm.com/support/fixcentral

Using the same search functionality that we used for the firmware we can find the latest level and check the support status of the installed version.



The following site lists the available Technology Levels and their End of Support Date
https://www.ibm.com/support/pages/aix-support-lifecycle-information

Latest AIX 7.2 version is AIX 7.2 TL5 SP3



An update from TL5 SP0 to TL5 SP3 would be recommended.

It is strongly advised to **relink Oracle Home** binaries in case of AIX version and/or TL upgrade.

The relink is done via the oracle command “relink all”. If this has been performed it will create a log file in the $ORACLE_HOME/install directory. It has already been executed on this environment, you can either check the log or execute it once again.

Connect as the **Oracle user** and cd to the $ORACLE_HOME/install directory to see if there is a relink log

```
root@bestpracticesoracle-d11310-pvs1:/ # su - oracle
$ cd $ORACLE_HOME/install
$ ls relink*
relinkActions2021-02-24_09-20-49AM.log
```

Oracle keeps up to date certification information in the MyOracleSupport Note: **1307544.1**

You will require a MyOracleSupport account to access this information directly on the MOS website.

Certification Information for Oracl

The attached pdf contains the current version of the document as of September 26th, 2021.

3/ LPAR affinity

LPAR affinity was much more important in older versions of Power Systems but there is still be an influence on POWER9 if the memory and CPUs are too remote.

Check the affinity on your AIX LPAR with the following command as root user

lssrad -va

```
root@bestpracticesoracle-d11310-pvs1:/ # lssrad -va
REF1    SRAD        MEM      CPU
0
          0    35066.00      0-7
root@bestpracticesoracle-d11310-pvs1:/ #
```

The 'lssrad-va' command displays a summary of the way physical processors and memory is allocated for a given LPAR

- REF1: Hardware provided reference point identifying sets of resources that are near each other. e.g. socket in scale-out servers or node in scale-up servers.
- SRAD: A Scheduler Resource Affinity Domain, i.e. an individual group of processors that all reside on the same chip
- MEM: The amount of local memory (in Megabytes) allocated to the SRAD
- CPU: The logical CPUs within the SRAD, e.g. with SMT8 enabled, 0-7 would be for the first physical CPU, 8-15 would be for the second physical CPU, etc…

As this class is performed on the IBM Power Virtual Server, we cannot change the processor policy but for a client site you should consider setting processor policy to either Dedicated or Dedicated Donating for the most critical production LPARs. If any specific performance requirement on I/O and/or network is expected, consider also to dedicate the corresponding adapters.

With Shared processors configurations, set the EC and VP to be within a range of no more than 30 to 50% of each other to mitigate the overhead of folding activity.

You can check these values with the command: lparstat -i

```
root@bestpracticesoracle-d11310-pvs1:/ # lparstat -i
Node Name                                  : bestpracticesoracle-d11310-pvs1
Partition Name                             : bestpractices-09b8b9c8-00006eee
Partition Number                           : 12
Type                                       : Shared-SMT-8
Mode                                       : Capped
Entitled Capacity                          : 1.00
Partition Group-ID                         : 32780
Shared Pool ID                             : 0
Online Virtual CPUs                        : 1
Maximum Virtual CPUs                       : 8
Minimum Virtual CPUs                       : 1
Online Memory                              : 36864 MB
Maximum Memory                             : 294912 MB
Minimum Memory                             : 4608 MB
Variable Capacity Weight                   : 0
Minimum Capacity                           : 0.25
Maximum Capacity                           : 8.00
Capacity Increment                         : 0.01
Maximum Physical CPUs in system            : 20
```

```
Active Physical CPUs in system             : 20
Active CPUs in Pool                        : 20
Shared Physical CPUs in system             : 20
Maximum Capacity of Pool                   : 2000
Entitled Capacity of Pool                  : 1275
Unallocated Capacity                       : 0.00
Physical CPU Percentage                    : 100.00%
Unallocated Weight                         : 0
Memory Mode                                : Dedicated
Total I/O Memory Entitlement               : -
Variable Memory Capacity Weight            : -
Memory Pool ID                             : -
Physical Memory in the Pool                : -
Hypervisor Page Size                       : -
Unallocated Variable Memory Capacity Weight: -
Unallocated I/O Memory entitlement         : -
Memory Group ID of LPAR                    : -
Desired Virtual CPUs                       : 1
Desired Memory                             : 36864 MB
Desired Variable Capacity Weight           : 0
Desired Capacity                           : 1.00
Target Memory Expansion Factor             : -
Target Memory Expansion Size               : -
Power Saving Mode                          : Dynamic Power Savings (Favor
Performance)
Sub Processor Mode                         : -
root@bestpracticesoracle-d11310-pvs1:/ #
```

When deploying a virtual machine in Power Virtual Server, customers can choose between dedicated, capped shared or uncapped shared processors for their virtual CPUs (vCPU). The following list describes their differences in a few words:

- **Dedicated**: resources are allocated for a specific client (used for specific third-party considerations)
- **uncapped shared**: shared with other customers
- **capped shared**: shared, but resources are limited to those requested (used mainly for licensing)

In the context of Oracle software, the customer is under the Bring Your own license (BYOL) Model. Therefore, the option selected here will impact the Oracle Licenses Costs/requirements. The recommendation is to use the Shared Capped Option.

In this instance you can see that Desired Virtual CPUs and Entitled Capacity are both set to 1.

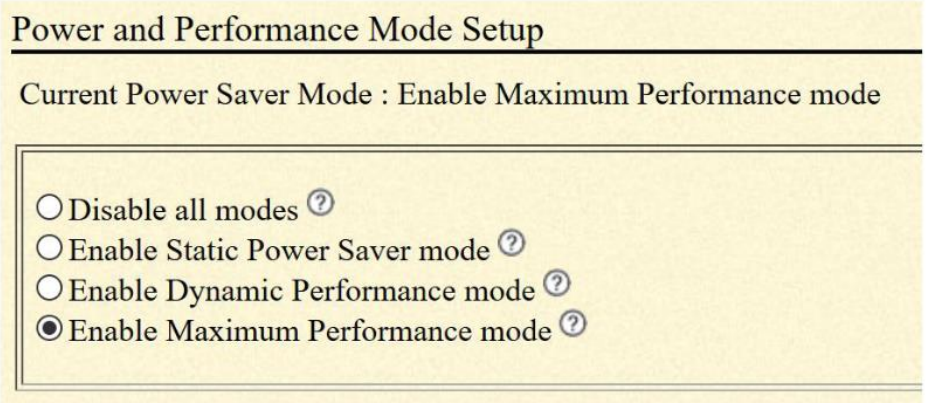
The processors aren't dedicated but there will be no folding activity.

In this cloud environment we don't have access to the HMC so we are unable to change the Power Saving Mode. On an on-premise Power Systems Server, we would recommend to switch to "Maximum Performance Mode".

When maximum performance is required check that Power Saving Mode is enabled and set to **"Maximum Performance Mode"**

This is done via the HMC (from Advanced System Management menu (aka ASM) or Command Line Interface (aka CLI))(" Maximum Performance Mode" is the default from S924 model to E980 except from S922 it is set to "Dynamic_Favor_Perf")

- ASM menu -> System Configuration -> Power Management -> Power and Performance Mode Setup, if not set check "Enable Maximum Performance mode" box



- CLI: **lspwrmgmt -m <managed system name> -r sys | cut -d, -f4,5**, if not set run **chpwrmgmt -m <managed system name> -r sys -o enable -t fixed_max_frequency**

4/ SMT

The Power Systems SMT capability enables concurrent execution of instruction streams, or threads, on the same core. With POWER9 processors, up to eight threads (SMT8) can concurrently run on a single core. The benefit of SMT8 on POWER9 is significantly higher than on the previous generation of POWER8 servers.

For this reason, it is recommended that customers moving to POWER9 evaluate changing from the default SMT4 setting of an AIX7.x logical partition on POWER8 by tuning it to SMT8 on POWER9 by default. This may not be applicable for all workloads. The nature of the application and characteristics of the workload being used in the LPAR may require a different SMT configuration if they do not take advantage of the multi-threading.

Leverage SMT-8 for performance improvement (whenever possible set Processor Implementation Mode to POWER9). AIX 7.1 only allows POWER8 mode and not POWER9. AIX 7.2 allows both settings.

Check the SMT value using the command:

lparstat | grep smt

Check Processor Implementation Mode with the AIX command

prtconf | grep "Processor Implementation Mode"

```
root@bestpracticesoracle-d11310-pvs1:/ #  lparstat | grep smt
System configuration: type=Shared mode=Capped smt=8 lcpu=8 mem=36864MB
psize=20 ent=1.00
root@bestpracticesoracle-d11310-pvs1:/ # prtconf | grep "Processor
Implementation Mode"
Processor Implementation Mode: POWER 9
root@bestpracticesoracle-d11310-pvs1:/ #
```

You can use **"smtclt -t #SMT_Value -w now"** to change the SMT configuration dynamically.

To make the change persistent, you need to run **"bosboot -a"**

https://www.ibm.com/support/pages/processor-compatibility-modes-ibm-power9-based-systems

5/ Spectre and Meltdown

Because this is a cloud environment, Spectre and Meltdown protection is set to the highest level but if the server is behind a firewall on a client site then the protection is already provided and activation at AIX level is no longer required.

Check the level of support with the command as root user:

lparstat -x

```
root@bestpracticesoracle-d11310-pvs1:/ # lparstat -x
LPAR Speculative Execution Mode              : 2
root@bestpracticesoracle-d11310-pvs1:/ #
```

There are three options.

- 0 = Speculative execution fully enabled
- 1 = Speculative execution controls to mitigate user-to-kernel side-channel attacks
- 2 = Speculative execution controls to mitigate user-to-kernel and user-to-user side-channel attacks

2 is the highest level of protection, 0 is the lowest despite the misleading name.

The “Speculative execution fully enabled” option is described as follows in the documentation:

> This optional mode is designed for systems where the hypervisor, operating system, and applications can be fully trusted. Enabling this option could expose the system to CVE-2017-5753, CVE-2017- 5715, and CVE-2017-5754. This includes any partitions that are migrated (using Live Partition Mobility) to this system. **This option has the least possible impact on the performance at the cost of possible exposure to both User accessible data as well as System data.**

We are unable to change this value during this workshop as it is done via the ASM interface from the HMC. The overhead of this option being enabled can be as high as 6% additional execution time.

In older versions of AIX lparstat -x is not available so you will need to check via the hmc.

To access the Advanced System Management menu from the HMC select the server and navigate to: Operations -> Launch Advanced System Management (ASM)

You will be asked for an admin login and password.

•ASM menu -> System Configuration -> Speculative Execution Control -> choose "Speculative execution fully enabled" radio button -> press "Save Settings" text button

Note that this change is only possible when the system is powered off.

6/ Online patching

- **Turn off Oracle Online Patching in your environment (if not required)**

Update your oracle AIX user .profile file with the following parameter:

MPROTECT_TXT=OFF

This parameter prevents the CPU from skyrocketing in case of memory page claims under certain circumstances

su - oracle
cat .profile
echo "export MPROTECT_TXT=OFF" >> .profile
cat .profile
. .profile

```
root@bestpracticesoracle-d11310-pvs1:/ # su - oracle
$ cat .profile
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.

# ADDED for Oracle
ORACLE_SID=db19000
export ORACLE_SID
ORACLE_HOME=/u01/app/product/19c
export ORACLE_HOME

PATH=$PATH:$ORACLE_HOME/bin:$ORACLE_HOME/OPatch

# ADDED for Swingbench
PATH=$PATH:/usr/java8_64/jre/bin

export PATH

if [ -s "$MAIL" ]           # This is at Shell startup.  In normal
then echo "$MAILMSG"        # operation, the Shell checks
fi                          # periodically.

set -o vi

export LDR_CNTRL=DATAPSIZE=64K@TEXTPSIZE=64K@STACKPSIZE=64K@SHMPSIZE=64K
oracle
VMM_CNTRL=vmm_fork_policy=COR
$ echo "export MPROTECT_TXT=OFF" >> .profile
$ cat .profile
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.

# ADDED for Oracle
ORACLE_SID=db19000
export ORACLE_SID
ORACLE_HOME=/u01/app/product/19c
export ORACLE_HOME

PATH=$PATH:$ORACLE_HOME/bin:$ORACLE_HOME/OPatch

# ADDED for Swingbench
PATH=$PATH:/usr/java8_64/jre/bin

export PATH

if [ -s "$MAIL" ]           # This is at Shell startup.  In normal
then echo "$MAILMSG"        # operation, the Shell checks
fi                          # periodically.

set -o vi
```

```
export LDR_CNTRL=DATAPSIZE=64K@TEXTPSIZE=64K@STACKPSIZE=64K@SHMPSIZE=64K
oracle
VMM_CNTRL=vmm_fork_policy=COR
export MPROTECT_TXT=OFF
$ . .profile
```

7/ Memory

In older versions of Oracle the default page size of 4K was used. Since 12c Oracle uses 64K pages by default.

Before 12c it was common to allocate large pages to improve memory performance and some clients continue to do so. Large Pages can still be considered for large Oracle memory configurations for example when the SGA Size exceeds 100GB. You need to be careful when configuring Large Pages because the pages are dedicated for Oracle and the number of pages needs to be sufficient for the total SGA memory space plus one page. If there are not enough pages available, the database instance will start using the 64k memory pages and you are likely to encounter paging activity and performance issues.

Today large pages are not always required as 64K pages will be sufficient for most workloads and increases flexibility as the pages don't need to be pre-allocated and dedicated for the size of the Oracle SGA Memory space.

This Lab environment is not using Large Pages

Commands to set up Large Pages for Oracle Database

```
@AIX level as root user
vmo -p -o v_pinshm=1
vmo -r -o lgpg_size=16777216 -o lgpg_regions=((at least SGA_MAX_SIZE instance parameter
in MB/16)+10)
chuser capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE <oracle user>

in <oracle user> .profile file set "export ORACLE_SGA_PGSZ=16M"
@Oracle Instance level
in the DB Instance initialization Parameter file (Spfile or init.ora) lock_sga=TRUE

For checking do
@Oracle Instance level
startup mount
@AIX level
svmon -nwU <oracle user> and check the output for
Large Pages configuration (beginning section) Inuse & Pin columns for L PageSize
shmat/mmap segments (at the end of the ouput) should be tagged as L
```

The svmon command allows us to check what pages are being used. If large pages have been allocated but are not being used this can cause memory issues as the allocated pages show as free memory but cannot be used by processes requiring 4k or 64k pages.

To check what pages are being used by the oracle user you can use the command:

svmon -nwU oracle|pg #use pg because there is a lot of output not shown in the extract below

```
$ svmon -nwU oracle |pg
================================================================================
=
User                                Inuse      Pin     Pgsp  Virtual
oracle                              20863     5428        0    20863

     PageSize                Inuse        Pin       Pgsp    Virtual
```

```
      s     4 KB                  351          4          0        351
      m    64 KB                 1282        339          0       1282
      L    16 MB                    0          0          0          0
      S    16 GB                    0          0          0          0

................................................................................
.
EXCLUSIVE segments                       Inuse      Pin     Pgsp  Virtual
                                          4303       68        0     4303

      PageSize                  Inuse        Pin       Pgsp    Virtual
      s     4 KB                  351          4          0        351
      m    64 KB                  247          4          0        247
      L    16 MB                    0          0          0          0
      S    16 GB                    0          0          0          0
```

You can see that there is limited memory allocated to oracle.

Connect as the oracle user using the command: su - oracle

From the home directory of the oracle user start the database processes using the script: start_db.sh

This will start the database and the listener.

If you cat the script you will see the following

```
$ cat start_db.sh
 sqlplus / as sysdba @startup.sql
lsnrctl start
```

The first command connects to sqlplus and runs the startup script.

The second command launches the listener process which allows remote connections and will be used later by the swingbench tool.

The startup script contains the commands:

```
$ cat startup.sql
startup
exit;
```

```
$ start_db.sh

LSNRCTL for IBM/AIX RISC System/6000: Version 19.0.0.0.0 - Production on 04-
OCT-2021 10:00:37

Copyright (c) 1991, 2020, Oracle.  All rights reserved.

Starting /u01/app/product/19c/bin/tnslsnr: please wait...

TNSLSNR for IBM/AIX RISC System/6000: Version 19.0.0.0.0 - Production
Log messages written to /u01/oracle/diag/tnslsnr/bestpracticesoracle-d11310-
pvs1/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=bestpracticesoracle-
d11310-pvs1)(PORT=1521)))

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------
```

```
Alias                     LISTENER
Version                   TNSLSNR for IBM/AIX RISC System/6000: Version
19.0.0.0.0 - Production
Start Date                04-OCT-2021 10:00:38
Uptime                    0 days 0 hr. 0 min. 1 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Log File         /u01/oracle/diag/tnslsnr/bestpracticesoracle-d11310-
pvs1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=bestpracticesoracle-d11310-
pvs1)(PORT=1521)))
The listener supports no services
The command completed successfully

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 10:00:39 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.

Connected to an idle instance.

ORACLE instance started.

Total System Global Area 2.1475E+10 bytes
Fixed Size                  12605904 bytes
Variable Size             2684354560 bytes
Database Buffers          1.8723E+10 bytes
Redo Buffers                54460416 bytes
Database mounted.
Database opened.
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.9.1.0.0
$
```

The listener will automatically scan for database instances to add. It takes around 1 minute. If you start the listener first the database registers immediately on startup. You can check the status by running the following command as the oracle user lsnrctl status.

Now run the svmon command again with the command:

svmon -nwU oracle|pg

```
$ svmon -nwU oracle |pg
===============================================================================
=
User                              Inuse      Pin     Pgsp  Virtual
oracle                           791110     9380        0   791110

     PageSize                Inuse        Pin       Pgsp    Virtual
     s    4 KB                9126          4          0       9126
     m   64 KB               48874        586          0      48874
     L   16 MB                   0          0          0          0
     S   16 GB                   0          0          0          0

..............................................................................
.
EXCLUSIVE segments                Inuse      Pin     Pgsp  Virtual
```

```
                                          295752      9380          0    295752

      PageSize                        Inuse          Pin         Pgsp     Virtual
      s    4 KB                        8904            4            0        8904
      m   64 KB                       17928          586            0       17928
      L   16 MB                           0            0            0           0
      S   16 GB                           0            0            0           0

.........................................................................
SHARED segments                              Inuse         Pin       Pgsp   Virtual
                                            495358           0          0    495358

      PageSize                        Inuse          Pin         Pgsp     Virtual
      s    4 KB                         222            0            0         222
      m   64 KB                       30946            0            0       30946
      L   16 MB                           0            0            0           0
      S   16 GB                           0            0            0           0
```

In the extract above you can see that the shared pages are all 64KB pages. If the client was using Large pages we would expect the inuse pages to appear in the 16 MB row.

97748 memory pages of 64k are shown in the example above. **That totals around 6GB**.

Start nmon and use the **m option** to display the memory details.

You can use M option and check # of allocated 64KB pages prior and after starting the database instance. Verify that the number of free pages decreased as expected.

```
lqtopas_nmonqqU=Top-with-WLMqqqqqHost=bestpracticesorRefresh=2 secss1q10:07.05qqk
x Memory                            qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqx
x          Physical  PageSpace |        pages/sec  In      Out | FileSystemCache x
x% Used       21.1%      4.5%  | to Paging Space   0.0     0.0 | (numperm)  1.6% x
x% Free       78.9%     95.5%  | to File System    0.0     0.0 | Process   13.7% x
xMB Used    7764.7MB    22.9MB | Page Scans        0.0         | System     5.8% x
xMB Free   29099.3MB   489.1MB | Page Cycles       0.0         | Free      78.9% x
xTotal(MB) 36864.0MB   512.0MB | Page Steals       0.0         |          ------x
x                              | Page Faults     206.5         | Total    100.0% x
x------------------------------------------------------------- | numclient  1.6% x
xMin/Maxperm     1050MB(  3%)  31498MB( 90%) <--% of RAM       | maxclient 90.0% x
xMin/Maxfree     960    1088      Total Virtual    36.5GB      | User      10.4% x
xMin/Maxpgahead    2       8    Accessed Virtual     7.0GB 19.2%| Pinned    11.1% x
x                                                              | lruable pages
8959424.0                                                                     x
xqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
```

```
lqtopas_nmonqqc=CPUqqqqqqqqqqqqqqHost= bestpracticesorRefresh=2
secsqqq10:00.44qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
x Multiple-Page-Size (in Pages)
qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
qqqqqqqqx
xPage Sizes= 4   Page Size ->           4KB       64KB       16MB       16GB
x  numframes memory frames          1224240     972061          0          0
x  numfrb free list pages          170420     832154          0          0
x  numclient client frames           242002          0          0          0
x  numcompress compressed segments          0          0          0          0
x  numperm non-working segments    242002          0          0          0
x  numvpages accessed virtual pages  803519     140162          0          0
x  minfree min free list             960         60          0          0
x  maxfree max free list            1088         68          0          0
x  numpout page-outs                   0          0          0          0
x  numremote remote page-outs            0          0          0          0
```

```
x  numwseguse in use working seg        803519     139907          0          0
x  numpseguse in use persistent              0          0          0          0
x  xnumclseguse in use client             250301          0          0          0
x  numwsegpin pinned working            766014      33081          0          0
x  numpsegpin pinned persistent              0          0          0          0
x  numclsegpin pinned client seg          4184          0          0          0
x  numpgsp_pgs allocated PageSpace       10972        280          0          0
x  numralloc remote allocations              0          0          0          0
x  pfrsvdblks system reserv.blocks       65067      97206          0          0
x  pfavail pin available              454042     938980          0          0
x  pfpinavail pinnable@apps level       612564     946003          0          0
x  system_pgs SCBs marked V_SYS         187145      29638          0          0
x  nonsys_pgs SCBs not V_SYS            292851     110255          0          0
x--- Below are Rates per Second ---        4KB       64KB       16MB       16GB
x
x  numpermio non-w.s. pageouts               0          0          0          0
x     pgexct Page Faults                     0        850          0          0
x     pgrclm Page Reclaims                   0          0          0          0
x     pageins Paged in   -All                0          0          0          0
x     pageouts Paged out -All                0          0          0          0
x     pgspgins Paged in  -PageSpace          0          0          0          0
x     pgspgouts Paged out-PageSpace          0          0          0          0
x     numsios I/O Started                    0          0          0          0
x     numiodone I/O Completed                0          0          0          0
x     zerofills Zero filled                  0        570          0          0
x     exfills Exec() filled                  0          0          0          0
x     scans Scans by clock                   0          0          0          0
x     cycles Clock hand cycles               0          0          0          0
x     pgsteals Page Steals                   0          0          0          0
xqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqx
```

We will see that used memory is just under 10GB.

When Oracle starts, the default behavior is that it does not immediately allocate all of the memory it requires for the SGA and PGA memory zones.

The SGA or Shared Global Area is the main memory allocation for the internals of the oracle database instance. The PGA or Program Global Area is the area of memory allocated for the processes. This covers the data held by the processes in memory but doesn't cover the processes themselves meaning that with a large number of connections Oracle can take more memory than the PGA and SGA combined.

As a rule of thumb we try to not to exceed 60 to 70% as the limit of the size of the SGA+PGA compared to the memory allocation of the partition. This is to allow sufficient memory for the operating system.

If too much memory is allocated paging can occur and this causes significant performance degradation on an Oracle database.

The PGA and SGA are defined in the Oracle database initialization file. This can be found in the $ORACLE_HOME/dbs directory.

There are two naming formats, the legacy format is init<ORACLE_SID>.ora where ORACLE_SID is the name of the database instance. Our ORACLE_SID is db19000.

The other naming format is spfile<ORACLE_SID>.ora

Unlike the init.ora file the spfile is not a text file, you can cat the files but the special characters will probably cause a glitch in the putty display.

Create a pfile containing initialization parameter of the Database Instance. (Same method works for Oracle Automatic Storage Management (ASM))

As oracle user,

```
$ sqlplus '/as sysdba'
```

```
SQL*Plus: Release 19.0.0.0.0 - Production on Fri Dec 10 10:05:30 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2021, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.12.0.0.0

SQL> create pfile='/tmp/db19000.ora' from spfile;

File created.

SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.9.1.0.0
$
```

```
$ cat /tmp/db19000.ora
db19000.__data_transfer_cache_size=0
db19000.__db_cache_size=18589155328
db19000.__inmemory_ext_roarea=0
db19000.__inmemory_ext_rwarea=0
db19000.__java_pool_size=134217728
db19000.__large_pool_size=134217728
db19000.__oracle_base='/u01/oracle'#ORACLE_BASE set from environment
db19000.__pga_aggregate_target=5368709120
db19000.__sga_target=21474836480
db19000.__shared_io_pool_size=134217728
db19000.__shared_pool_size=2415919104
db19000.__streams_pool_size=0
db19000.__unified_pga_pool_s
ize=0
*.audit_file_dest='/u01/oracle/admin/db19000/adump'
*.audit_trail='db'
*.compatible='19.0.0'
*.control_files='/oradata/db19000/DB19000/controlfile/o1_mf_j2b2795w_.ctl','/u
01/oracle/fast_recovery_area/DB19000/controlfile/o1_mf_j2b2796o_.ctl'#Restore
Controlfile
*.db_block_size=8192
*.db_create_file_dest='/oradata/db19000/'
*.db_name='db19000'
*.db_recovery_file_dest='/u01/oracle/fast_recovery_area'
*.db_recovery_file_dest_size=8226m
*.diagnostic_dest='/u01/oracle'
*.dispatchers=
'(PROTOCOL=TCP) (SERVICE=db19000XDB)'
*.filesystemio_options='SETALL'
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=300
*.pga_aggregate_target=5120m
*.processes=2560
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=20480m
*.undo_tablespace='UNDOTBS1'
```

The * before each parameter indicates that it is for all instances of the database, this is only relevant when the file is shared between different instances of a RAC cluster.

You can see here that the pga_aggregate_target is 5GB, none of this will have been allocated because there are no Oracle processes running, this is allocated at runtime and released when no longer used.

The sga_target is set to 20GB, this is allocated once used unless specifically pre-allocated. This memory is not freed when unused so for most production database the target will be fully allocated.

As a general rule, the bigger the sga the more data is stored in memory and the less I/O a database will perform. The AWR report contains a section analysing the potential impact of changing this value.

Another parameter that is highlighted is the filesystemio_options=SETALL parameter. The default for this parameter is ASYNCH. If the parameter is not changed Oracle does not use CIO to access the database files (unless set at the file system level) and the file system cache can take a significant amount of memory which is not required as Oracle stores the same data internally. This setting is required when jfs2 file systems are used for storing the data. It is not relevant when ASM is used.

8/ Fibre Channel activity

Check the adapters attached to the partition using the command

lsdev | grep Fibre

```
$ lsdev  | grep Fibre
fcs0       Available C4-T1       Virtual Fibre Channel Client Adapter
fcs1       Available C6-T1       Virtual Fibre Channel Client Adapter
fcs2       Available C7-T1       Virtual Fibre Channel Client Adapter
fcs3       Available C8-T1       Virtual Fibre Channel Client Adapter
sfwcomm0   Available C4-T1-01-FF Fibre Channel Storage Framework Comm
sfwcomm1   Available C6-T1-01-FF Fibre Channel Storage Framework Comm
sfwcomm2   Available C7-T1-01-FF Fibre Channel Storage Framework Comm
sfwcomm3   Available C8-T1-01-FF Fibre Channel Storage Framework Comm
```

You can check if there are waits on a Fibre Channel adapter using the command fcstat, check the following section in the output of the command:

fcstat -D fcs0

```
$ fcstat -D fcs0
 FC SCSI Adapter Driver Information
  No DMA Resource Count: 0
  No Adapter Elements Count: 0
  No Command Resource Count: 0
```

If the “No Adapter Elements Count” is non-zero it means that the adapter is not able to handle the current activity. You can increase the value of num_cmd_elems to improve the throughput of the adapter.

To check the current value run the command: lsattr -El fcs0 -a num_cmd_elems

```
$ lsattr -El fcs0 -a num_cmd_elems
num_cmd_elems 1024 Maximum Number of COMMAND Elements True
```

you can check the range of settings allowed by the driver with the command lsattr –Rl

```
$ lsattr -Rl fcs0 -a num_cmd_elems
20...2048 (+1)
```

You could set a new value using chdev as below but this is not necessary for this workload.

```
chdev -l fcs0 -a num_cmd_elems=2048 -P
```

Increasing the value of num_cmd_elems should always be done in consultation with the system administrators particularly in a system where resources such as adapters and storage servers are shared. It is also recommended to consult IBM Support before changing this value.

The value of max_xfer_size can be gradually changed to a minimum of 0x200000 for an Oracle workload in certain circumstances. Increasing this value improves the performance for the transfer of large block sizes for workloads such as olap or rman backups.

This value needs to be kept inline with the value set at the level of the VIOS.

You can check the value with the command:

lsattr -El fcs0 -a max_xfer_size

```
$ lsattr -El fcs0 -a max_xfer_size
max_xfer_size 0x100000 Maximum Transfer Size True
```

User the lsattr command as root user to find the possible values.

```
root@bestpracticesoracle-d11310-pvs1:/ # lsattr -Rl fcs0 -a max_xfer_size
0x100000
0x200000
0x400000
0x800000
0x1000000
root@bestpracticesoracle-d11310-pvs1:/ #
```

Change the value with the following command (root user)

```
root@bestpracticesoracle-d11310-pvs1:/ #  chdev -l fcs0 -a
max_xfer_size=0x200000 -P
fcs0 changed
root@bestpracticesoracle-d11310-pvs1:/ #
```

9/ Hdisk attributes

On a busy system we can check to see if there is queue activity on a disk using nmon for live monitoring type D three times for the correct screen.

```
lqtopas_nmonqqn=NetworkqqqqqqqqqqHost=bestpracticesorRefresh=1 secsqqq10:37.46qqqqqqqqqqqqqqk
x Disk - Service times and Queues          qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqx
x Disk         Service      Read Service  Write Service       Wait      ServQ  WaitQ  ServQ   x
x Name      milli-seconds  milli-seconds  milli-seconds  milli-seconds  Size   Size   Full   x
xhdisk0         0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk16        0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk1         0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk15        0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk14        0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk13        0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk4         0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk11        0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk9         0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
xhdisk12        0.0           0.0           0.0           0.0         0.0    0.0    0.0    x
```

```
xhdisk5         0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk2         0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk3         0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk6         0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk10        0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk8         0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk17        0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xhdisk7         0.0             0.0             0.0             0.0          0.0     0.0     0.0    x
xTotals(MB/s)  Read=0.0    Write=0.0    Size(GB)=278    Free(GB)=50                                 x
xqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqx
```

If the value in the “wait milli-seconds” column is constantly above 0 this means that there is queueing either on the fibre channel or on the hdisk. If you see no issue on the fibre channel you may need to increase the queue_depth on the device or add more i/o queues by increasing the number of hdisk devices in the volume group.

Increasing the queue_depth should always be done in consultation with the system administrators particularly in a system where resources such as adapters and storage servers are shared. It is recommended that you also consult your Storage Vendor before making this change.

If you want to check the current **queue_depth** value for a hdisk device you can use the command:

```
root@bestpracticesoracle-d11310-pvs1:/ # lsattr -El hdisk1 -a queue_depth
queue_depth 20 Queue DEPTH True+
root@bestpracticesoracle-d11310-pvs1:/ #
```

To check the range of settings allowed by the driver you can run the command:

```
root@bestpracticesoracle-d11310-pvs1:/ # lsattr -Rl hdisk1 -a queue_depth
1...256 (+1)
root@bestpracticesoracle-d11310-pvs1:/ #
```

To set a new value a command such as the following with your own device and value:

```
chdev -l hdisk1 -a queue_depth=256 -P
```

Check Disk Attributes

```
root@bestpracticesoracle-d11310-pvs1:/ # lsattr -El hdisk9
PCM             PCM/friend/fcpother              Path Control Module              False
PR_key_value    none                             Persistant Reserve Key Value     True+
algorithm       fail_over                        Algorithm                        True+
clr_q           no                               Device CLEARS its Queue on error True
dist_err_pcnt   0                                Distributed Error Percentage     True
dist_tw_width   50                               Distributed Error Sample Time    True
hcheck_cmd      test_unit_rdy                    Health Check Command             True+
hcheck_interval 60                               Health Check Interval            True+
hcheck_mode     nonactive                        Health Check Mode                True+
location                                         Location Label                   True+
lun_id          0x9000000000000                  Logical Unit Number ID           False
lun_reset_spt   yes                              LUN Reset Supported              True
max_coalesce    0x40000                          Maximum Coalesce Size            True
max_retry_delay 60                               Maximum Quiesce Time             True
max_transfer    0x80000                          Maximum TRANSFER Size            True
min_rw_to       10                               Minimum value for rw_timeout     False
node_name       0x5005076810003179               FC Node Name                     False
pvid            00ca22508cd77f280000000000000000   Physical volume identifier     False
q_err           yes                              Use QERR bit                     True
q_type          simple                           Queuing TYPE                     True
queue_depth     20                               Queue DEPTH                      True+
reassign_to     120                              REASSIGN time out value          True
reserve_policy  single_path                      Reserve Policy                   True+
```

```
rw_max_time     0                               Maximum I/O completion time      True+
rw_timeout      30                              READ/WRITE time out value        True
scsi_id         0xa2d00                         SCSI ID                          False
start_timeout   60                              START unit time out value        True
timeout_policy  fail_path                       Timeout Policy                   True+
unique_id       332136005076810820 18BC8000000000039 7304 2145 03IBMfcp Unique device
identifier        False
ww_name         0x5005076810133179              FC World Wide Name               False
root@bestpracticesoracle-d11310-pvs1:/ #
```

The reserve_policy should be set to no_reserve and the algorithm should be set to shortest_queue for all hdisks that will be used to store Oracle database files (datafiles, redologs etc).

The following script can be used to position these parameters in your environment.
Connect as root and run the script /home/UpdateDiskAttributes.sh

```
#!/bin/sh
DISKLIST=`lspv | awk '{print $1}'`
for i in $DISKLIST
do
echo $i
VAL=`lspv |grep -w $i | awk '{print $3}'`
if [ $VAL = "oradatavg" ] || [ $VAL = "oraredovg" ]
then
   echo "Update disk $i"
chdev -l $i  -a reserve_policy=no_reserve -a algorithm=shortest_queue -P
fi
done
```

10/ Mount options, luns and file systems

The Oracle binaries need to be mounted in a file system with the mount options **'noatime,rw'** to turn off access-time update on i-node. The blocksize should be 4k.

For this workshop the Oracle binaries have been installed in the file system /u01

As the **root** user check Options column from **lsfs -q /<repository>** command:

lsfs -q /u01

```
root@bestpracticesoracle-d11310-pvs1:/ # lsfs -q /u01

Name            Nodename   Mount Pt               VFS   Size    Options
Auto Accounting

/dev/oralv      --         /u01                   jfs2  207618048 noatime,rw
yes  no

  (lv size: 207618048, fs size: 207618048, block size: 4096, sparse files:
yes, inline log: yes, inline log size: 396, EAformat: v1, Quota: no, DMAPI:
no, VIX: yes, EFS: no, ISNAPSHOT: no, MAXEXT: 0, MountGuard: no)

root@bestpracticesoracle-d11310-pvs1:/ #
```

For any Oracle Database files, set multiple IO queues by **configuring several LUNs** to improve IO flow. The recommendation for IBM flash storage is 8 luns or a multiple of 8 for datafiles, redologs or FRA (Flash Recovery Area) if used for online redos. This is true for databases storing data using JFS2 or ASM.

When hosting Oracle Database **on JFS2**, Physical Partition size (PPsize) should be set to 32 or 64MB to avoid contention which can occur with a larger ppsize.

The best practice is to create a Scalable Volume Group (aka VG) with Physical Partition (aka PP) Size set to 32 MB with a command such as : mkvg -S -y'<VG name>' -s '32' '-f' <space separated hdisk list>

Check the details of the oradatavg where our database files are stored with the command:

lsvg -L oradatavg

```
root@bestpracticesoracle-d11310-pvs1:/ # lsvg -L oradatavg
VOLUME GROUP:       oradatavg               VG IDENTIFIER:
00ca225000004b00000001778cbb1e74
VG STATE:           active                  PP SIZE:        32 megabyte(s)
VG PERMISSION:      read/write              TOTAL PPs:      2536 (81152 megabytes)
MAX LVs:            256                     FREE PPs:       36 (1152 megabytes)
LVs:                1                       USED PPs:       2500 (80000 megabytes)
OPEN LVs:           1                       QUORUM:         5 (Enabled)
TOTAL PVs:          8                       VG DESCRIPTORS: 8
STALE PVs:          0                       STALE PPs:      0
ACTIVE PVs:         8                       AUTO ON:        yes
MAX PPs per VG:     32768                   MAX PVs:        1024
LTG size (Dynamic): 512 kilobyte(s)         AUTO SYNC:      no
HOT SPARE:          no                      BB POLICY:      relocatable
MIRROR POOL STRICT: off
PV RESTRICTION:     none                    INFINITE RETRY: no
DISK BLOCK SIZE:    512                     CRITICAL VG:    no
FS SYNC OPTION:     no                      CRITICAL PVs:   no
ENCRYPTION:         no
root@bestpracticesoracle-d11310-pvs1:/ #
```

In this example we can see that the ppsize is 32mb and there are 8 physical volumes.

Logical Volumes have already been created for the purpose of this Lab.

Check the logical volumes created in the volume group oradatavg with the command:

lsvg -l oradatavg

```
root@bestpracticesoracle-d11310-pvs1:/ # lsvg -l oradatavg
oradatavg:
LV NAME             TYPE       LPs     PPs     PVs  LV STATE      MOUNT POINT
oradatalv           jfs2       2500    2500    8    open/syncd    /oradata
root@bestpracticesoracle-d11310-pvs1:/ #
```

When creating LV, Use "Maximum Range of Physical Volume" option to spread PPs on different hdisk in a Round Robin Fashion : # mklv -e x

For all of the logical volumes the INTER-POLICY should be set to "maximum"

Check this with the command: lslv oradatalv |grep INTER-POLICY

```
$ lslv oradatalv |grep INTER-POLICY
INTER-POLICY:       maximum                 RELOCATABLE:    yes
```

Redologs should be stored in a separate jfs2 file system with a blocksize of 512 bytes.

We have created a file system called /oraredo you can check the details with the command:

lsfs -q /oraredo

```
root@bestpracticesoracle-d11310-pvs1:/ # lsfs -q /oraredo
Name            Nodename   Mount Pt               VFS   Size    Options    Auto
Accounting
/dev/oraredolv  --         /oraredo               jfs2  65536000 noatime,rw yes  no
  (lv size: 65536000, fs size: 65536000, block size: 512, sparse files: yes, inline
log: yes, inline log size: 125, EAformat: v1, Quota: no, DMAPI: no, VIX: yes, EFS: no,
ISNAPSHOT: no, MAXEXT: 0, MountGuard: no)
root@bestpracticesoracle-d11310-pvs1:/ #
```

We will talk more about this later.

File Systems Creation Commands

for datafiles

crfs -v jfs2 -d'<LV name>' -m'<JFS2 mount point>' -A'yes' -p'rw' -a options='noatime' -a agblksize='4096' -a logname='INLINE' -a isnapshot='no'

for redolog and control files

crfs -v jfs2 -d'<LV name>' -m'<JFS2 mount point>' -A'yes' -p'rw' -a options='noatime' -a agblksize='512' -a logname='INLINE' -a isnapshot='no'

Performance issues with controlfiles are less common than issues with redologs. Placing the controlfiles in the same file system as the datafiles will be sufficient for most workloads but in the event of controlfile contention it is advisable to place them in their own file system with a 512 bytes blocksize. The I/O on the controlfiles does not warrant a separate volume group, the file system can be placed in the same volume group as the datafiles.

11/ Statistics

The oracle optimizer bases its optimization plans on statistics held within the database at several different levels. The statistics are essential for good performance.

Run the following scripts as the oracle user to gather statistics

su – oracle

The following output related statistics collected when Best Practices were not applied. This output will be used for comparison once script is executed now that best practices are on.

```
$ cat stats_output_2602_0927.out

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Feb 26 09:27:39 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0


'SYSTEMSTATS
------------
system stats


SNAME                          PNAME                               PVAL1 PVAL2
------------------------------ ------------------------------ ---------- --------------
------
```

```
SYSSTATS_INFO                  STATUS                                    COMPLETED
SYSSTATS_INFO                  DSTART                                    05-19-2019
03:43
SYSSTATS_INFO                  DSTOP                                     05-19-2019
03:43
SYSSTATS_INFO                  FLAGS                                            1
SYSSTATS_MAIN                  CPUSPEEDNW                             1516.97128
SYSSTATS_MAIN                  IOSEEKTIM                                      10
SYSSTATS_MAIN                  IOTFRSPEED                                   4096
SYSSTATS_MAIN                  SREADTIM
SYSSTATS_MAIN                  MREADTIM
SYSSTATS_MAIN                  CPUSPEED
SYSSTATS_MAIN                  MBRC
SYSSTATS_MAIN                  MAXTHR
SYSSTATS_MAIN                  SLAVETHR

13 rows selected.


'FIXEDOBJECTSTATS'
------------------
fixed object stats

ANALYZED_ON                     COUNT(*)
------------------------------ ----------
24/02/21                             1181
                                      154


'IOCALIB
--------
io calib

no rows selected

'SEGMENTSTATS'
--------------
segment stats


OWNER                ANALYZED_ON            COUNT(*)
-------------------- -------------------- ----------
APPQOSSYS            19/05/19                      5
AUDSYS               24/02/21                      2
AUDSYS               19/05/19                      1
CTXSYS               24/02/21                      7
CTXSYS               19/05/19                     46
DBSFWUSER            19/05/19                      3
DBSNMP                                             3
DBSNMP               24/02/21                      2
DBSNMP               19/05/19                     15
DVSYS                                              1
DVSYS                24/02/21                      3
DVSYS                19/05/19                     40
GSMADMIN_INTERNAL                                  2
GSMADMIN_INTERNAL    24/02/21                      2
GSMADMIN_INTERNAL    19/05/19                     36
LBACSYS              24/02/21                      2
LBACSYS              19/05/19                     20

OWNER                ANALYZED_ON            COUNT(*)
-------------------- -------------------- ----------
MDSYS                                             19
MDSYS                24/02/21                     51
MDSYS                19/05/19                     79
```

```
OJVMSYS              24/02/21                      2
OJVMSYS              19/05/19                      4
OLAPSYS              19/05/19                      2
ORDDATA                                           19
ORDDATA              19/05/19                     71
ORDSYS               19/05/19                      4
OUTLN                19/05/19                      3
SOE                  25/02/21                      1
SOE                  24/02/21                      2
SOE                  11/02/21                      8
SYSTEM                                             7
SYSTEM               24/02/21                      5
SYSTEM               19/05/19                    170
WMSYS                                              2

OWNER                ANALYZED_ON            COUNT(*)
-------------------- -------------------- ----------
WMSYS                24/02/21                      2
WMSYS                19/05/19                     34
XDB                                                1
XDB                  24/02/21                      5
XDB                  19/05/19                     29

39 rows selected.
```

There are four boxes highlighting the output showing the date of the collection of the system statistics, 154 fixed objects with no dates, no io calibration data and one of several schemas without statistics. The schema that is most interesting for us is SOE as this is where the test workload is installed. The others are standard schemas that were created at the time of the database creation.

There are four main types of statistics.

System statistics make the optimizer aware about the infrastructure performance. These statistics should be gathered whenever a significant change occurs on Oracle Database Server such as an operating system upgrade or change on the logical CPUs configuration of the partition or movement of the partition to a new server.

The system statistics can be gathered by connecting to sqlplus as the sysdba user and running the command:

sqlplus / as sysdba

at the sql prompt type:

```
exec DBMS_STATS.GATHER_SYSTEM_STATS;
```

```
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 10:51:25 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0

SQL> exec DBMS_STATS.GATHER_SYSTEM_STATS;

PL/SQL procedure successfully completed.
```

```
SQL>
```

The system also stores statistics for the I/O performance which should be updated whenever there is a significant change to the storage system (change of storage server or increase of SAN bandwith). Again this is done from sqlplus as sysdba.

```
set timi on
SET SERVEROUTPUT ON
DECLARE
 lat  INTEGER;
 iops INTEGER;
 mbps INTEGER;
BEGIN
 DBMS_RESOURCE_MANAGER.CALIBRATE_IO (8, 10, iops, mbps, lat);
 DBMS_OUTPUT.PUT_LINE ('max_iops = ' || iops);
 DBMS_OUTPUT.PUT_LINE ('latency  = ' || lat);
 DBMS_OUTPUT.PUT_LINE ('max_mbps = ' || mbps);
end;
/
```

```
SQL>   2    3    4    5    6    7    8    9   10   11
max_iops = 787
latency  = 9.825
max_mbps = 363

Note: The high I/O latencies from the calibration run indicate that the
calibration I/Os are being serviced mostly from disk. If your storage has a
cache, you may achieve better results by rerunning.  Rerunning will warm or
populate the storage cache.

max_iops = 787
latency  = 10
max_mbps = 363

PL/SQL procedure successfully completed.

Elapsed: 00:05:32.97
SQL> SQL>
```

In the above command the number 8 represents the number of physical disks (means #SSDs or #FCMs or #Spindles) on which LUNs rely and the number 10 the maximum tolerable latency in milliseconds which is always set to 10.

Fixed objects statistics need to represent the system during “typical” load. As such they should be gathered during peak processing times for each type of workload on the database (e.g. OLTP, batch, …). It is also recommended to gather fixed objects statistics whenever the following changes will occur:

1. SGA size
2. Major application changes
3. Database or application upgrade

They are collected through the gather_fixed_objects_stats procedure from the dbms_stats package via sqlplus.

```
exec DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
```

```
SQL> SQL> exec DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

PL/SQL procedure successfully completed.

Elapsed: 00:01:12.94
SQL>
```

Database segment statistics make the optimizer aware about the tables and indexes in the database and the structure of the data within them.

```
exec dbms_stats.gather_schema_stats('SOE',options=>'GATHER AUTO', estimate_percent =>
dbms_stats.auto_sample_size,degree=>DBMS_STATS.AUTO_DEGREE);
```

```
SQL> exec dbms_stats.gather_schema_stats('SOE',options=>'GATHER AUTO', estimate_percent
=> dbms_stats.auto_sample_size,degree=>DBMS_STATS.AUTO_DEGREE);

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.31
SQL> exit
```

The above command gathers statistics on all of the objects in the SOE schema of the database. This is the user or schema where we have installed the database objects required for the swingbench demo that we'll be using later.

Execute the system, fixed objects & database segment statistics in addition to IO Calibration.

Run the the statistics check script to see the changes (as oracle user)

/home/oracle/check_stats.sh

```
$ /home/oracle/check_stats.sh

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:03:34 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0


'SYSTEMSTATS
------------
system stats


SNAME                          PNAME                               PVAL1 PVAL2
------------------------------ ------------------------------ ---------- --------------
------
SYSSTATS_INFO                  STATUS                                     COMPLETED
SYSSTATS_INFO                  DSTART                                     10-04-2021
10:51
```

```
SYSSTATS_INFO                  DSTOP                                    10-04-2021
10:51
SYSSTATS_INFO                  FLAGS                                     1
SYSSTATS_MAIN                  CPUSPEEDNW                             1953
SYSSTATS_MAIN                  IOSEEKTIM                                10
SYSSTATS_MAIN                  IOTFRSPEED                             4096
SYSSTATS_MAIN                  SREADTIM
SYSSTATS_MAIN                  MREADTIM
SYSSTATS_MAIN                  CPUSPEED
SYSSTATS_MAIN                  MBRC
SYSSTATS_MAIN                  MAXTHR
SYSSTATS_MAIN                  SLAVETHR

13 rows selected.


'FIXEDOBJECTSTATS'
------------------
fixed object stats

ANALYZED_ON                      COUNT(*)
------------------------------ ----------
04/10/21                             1181
                                      154


'IOCALIB
--------
io calib

START_TIME           END_TIME               MAX_IOPS   MAX_MBPS  MAX_PMBPS    LATENCY
DISKS
-------------------- -------------------- ---------- ---------- ---------- ---------- -
---------
04-OCT-021 10:52:32  04-OCT-021 10:58:05         787        363        365      9.825
8


'SEGMENTSTATS'
--------------
segment stats


OWNER                ANALYZED_ON            COUNT(*)
-------------------- -------------------- ----------
APPQOSSYS            19/05/19                      5
AUDSYS                                             2
AUDSYS               24/02/21                      2
AUDSYS               19/05/19                      1
CTXSYS               24/02/21                      7
CTXSYS               19/05/19                     46
DBSFWUSER            19/05/19                      3
DBSNMP                                             3
DBSNMP               28/02/21                      3
DBSNMP               19/05/19                     14
DVSYS                                              1
DVSYS                24/02/21                      3
DVSYS                19/05/19                     40
GSMADMIN_INTERNAL                                  2
GSMADMIN_INTERNAL    24/02/21                      2
GSMADMIN_INTERNAL    19/05/19                     36
LBACSYS              24/02/21                      2

OWNER                ANALYZED_ON            COUNT(*)
-------------------- -------------------- ----------
LBACSYS              19/05/19                     20
```

```
MDSYS                                          19
MDSYS                 24/02/21                 51
MDSYS                 19/05/19                 79
OJVMSYS               24/02/21                  2
OJVMSYS               19/05/19                  4
OLAPSYS               19/05/19                  2
ORDDATA                                        19
ORDDATA               19/05/19                 71
ORDSYS                19/05/19                  4
OUTLN                 19/05/19                  3
SOE                   26/02/21                  1
SOE                   24/02/21                  2
SOE                   11/02/21                  8
SYSTEM                                         7
SYSTEM                24/02/21                  5
SYSTEM                19/05/19                170

OWNER                 ANALYZED_ON            COUNT(*)
--------------------- -------------------- ----------
WMSYS                                          2
WMSYS                 24/02/21                 2
WMSYS                 19/05/19                34
XDB                                            1
XDB                   24/02/21                 5
XDB                   19/05/19                29

40 rows selected.

Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.9.1.0.0
$
$
```

Only one of the SOE tables has a new date associated because the segment statistics are not considered stale. This means that the difference between the existing statistics and the contents of the table have not significantly changed since the last time the statistics were collected. Collecting statistics can take hours on larger databases, this is often done at night or during the weekend to avoid impacting production.

Section 2: Swingbench Workload Execution & Monitoring Data Collection

We reviewed Oracle & AIX/Power Best Practices in Section1. The objective of Section 2 is now to stress the Oracle Database and simulate an OLTP Workload and collect required monitoring data for performance analysis.

1/ Tools for Monitoring Data Collection

While we execute the workload, we will collect monitoring data at both AIX LPAR and Oracle Database level. We then correlate nmon data and Oracle Performance data in the Oracle Automatic Workload Repository (AWR) to efficiently determine bottleneck and next steps of action.

1. Collection time period for all collection metrics needs to be the same
2. Collection resolution needs to be selected so that time period before, and potentially after the event can be included, without making the amount of collected data unmanageable.

Nmon and AWR are the two key information sources for investigation during either a benchmark/PoC testing or to solve performance issues at our clients.

Oracle Automatic Workload Repository (AWR)

We recommend to take AWR snapshots either just before and just after the run or sometimes after the ramp up and before the cool down to catch the peak activity.

AWR reports give details of the delta between two snapshots. The longer the period between the two snapshots the more chance that a short peak of activity can be lost in the mass of the business as usual activity.

The Use of AWR views & functions requires Oracle Diagnostics Pack License.

By default snapshots of the relevant data are taken every hour and retained for 7 days. The default values for these settings can be altered using the procedure :

DBMS_WORKLOAD_REPOSITORY.modify_snapshot_settings

For Performance Analysis, we recommend to collect Oracle Automatic Workload Repository (AWR) reports every 15min or 30min depending on total duration of the workload execution to analyze.

The shorter is the better will be to focus on the performance analysis & its resolution

NMON – Nigel's Monitoring

NMON displays local system statistics in interactive mode and records system statistics in recording mode. https://www.ibm.com/docs/en/aix/7.2?topic=n-nmon-command

Once recording nmon file can then be integrated in different tools for graphical analysis : pGraph, NMON Visualizer, nmonchart, nmonanalyser, nmon consolidator …

http://nmon.sourceforge.net/pmwiki.php?n=Site.Nmon-Analyser

https://www.ibm.com/support/pages/pgraph-performance-data-graphing-tool

https://nmonvisualizer.github.io/nmonvisualizer/

Njmon for AIX & Linux extends nmon to collect more data and save it to JSON format or directly to InfluxDB.

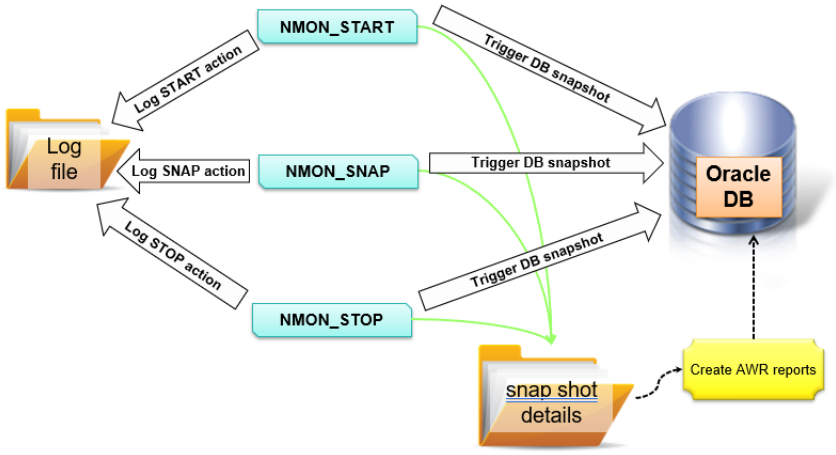
http://nmon.sourceforge.net/pmwiki.php?n=Site.Njmon

Here below are flags we recommend to use for database recording with nmon

- -f spreadsheet output format
- -s <seconds> between refreshing the screen
- -c <number> number of intervals to collect (*we typically use 999999)*
- -T spreadsheet includes top processes and UARG section
- -g <filename> User specified Disk Groups *(also specify –V)*
- -V Include disk Volume Group section
- -d Include Disk Service Time
- -w <number> Timestamp size (Tnnnn) *(we use a value of 5)*
- -P Include Paging Space section
- -M Include MEMPAGES section →detailed memory stats per pagesize
- -^ Include Fibre Channel (FC) sections
- -O Include Shared Ethernet Adpater (SEA) VIOS only sections
- -L Include LARGE page section *(If 16MB pages are used)*
- -A Include Async I/O Section
- -p Print process ID of nmon background process
- -Z <priority> set nice priority -20=important to 20=unimportant (negative only for root user)

For performance analysis use interval length of 10s for up to 2h of capture time.

In order to trigger AWR Snapshot and Nmon recording at the same time, you can leverage NMON "callback hooks".



Hooks are configured via shell environment variables:

- **NMON_START** - External command to run when nmon recording begins.
- **NMON_SNAP** - External command to run periodically while nmon executes.
- **NMON_END** - External command to run when nmon recording ends.
- NMON_START/SNAP/END environment variables are parsed **once** at startup; the first <space> terminates the command to be executed; any '&' is removed.

A single command will trigger both nmon and awr snapshots. Examples scripts are provided in Appendix on this Lab Document.

We first proceed with manual snapshots execution to let you see what are command lines/scripts executed and their outputs. Then you can leverage those nmon callback hooks if you want.

2/ Launch Swingbench Workload

We are now going to stress the Oracle database using swingbench OLTP Workload simulation tool and capture monitoring data using tools described previously.

Check if the Oracle Database Instance is up and running. Restart the Database if running by using following scripts : stop_db.sh and start_db.sh under /home/oracle as **oracle** user.

```
$ ps -ef |grep pmon
  oracle 16581110 25166256   0 02:53:09  pts/0  0:00 grep pmon
  oracle 26280376        1   0 02:52:47      -  0:00 ora_pmon_db19000  ← DBInstance up
$
$ stop_db.sh

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:06:07 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0

Database closed.
Database dismounted.
ORACLE instance shut down.
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.9.1.0.0

LSNRCTL for IBM/AIX RISC System/6000: Version 19.0.0.0.0 - Production on 04-OCT-2021
11:06:24

Copyright (c) 1991, 2020, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
The command completed successfully
$
$ start_db.sh

LSNRCTL for IBM/AIX RISC System/6000: Version 19.0.0.0.0 - Production on 04-OCT-2021
11:06:59

Copyright (c) 1991, 2020, Oracle.  All rights reserved.

Starting /u01/app/product/19c/bin/tnslsnr: please wait...

TNSLSNR for IBM/AIX RISC System/6000: Version 19.0.0.0.0 - Production
Log messages written to /u01/oracle/diag/tnslsnr/bestpracticesoracle-d11310-
pvs1/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=bestpracticesoracle-d11310-
pvs1)(PORT=1521)))

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------
```

```
Alias                     LISTENER
Version                   TNSLSNR for IBM/AIX RISC System/6000: Version 19.0.0.0.0 -
Production
Start Date                04-OCT-2021 11:06:59
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Log File         /u01/oracle/diag/tnslsnr/bestpracticesoracle-d11310-
pvs1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=bestpracticesoracle-d11310-
pvs1)(PORT=1521)))
The listener supports no services
The command completed successfully

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:06:59 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.

Connected to an idle instance.

ORACLE instance started.

Total System Global Area 2.1475E+10 bytes
Fixed Size                 12605904 bytes
Variable Size            2684354560 bytes
Database Buffers         1.8723E+10 bytes
Redo Buffers               54460416 bytes
Database mounted.
Database opened.
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.9.1.0.0
$
```

The test will run for around 5 minutes. You can monitor it from another window using nmon.

As oracle user, connect to the database and launch an AWR Snapshot. Capture date and time you start the workload & AWR Snapshot

Then immediately launch the swingbench scenario as the oracle user using the script:

/home/oracle/launch_test.sh

The Launch_test script also initiates nmon monitoring capture.

```
$ id
uid=54321(oracle) gid=54421(oinstall) groups=54322(dba),54323(oper)
$ date
Mon Oct  4 11:11:02 CDT 2021
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:11:08 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.



Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0
```

```
SQL> EXEC DBMS_WORKLOAD_REPOSITORY.create_snapshot;

PL/SQL procedure successfully completed.

SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.9.1.0.0
$ /home/oracle/launch_test.sh
16122134
Author  :        Dominic Giles
Version :        2.6.0.1137

Results will be written to scale10_100user.xml.

Time            Users   TPM     TPS

Saved results to scale10_100user.xml
11:17:05 AM     0       434     1
Completed Run.

$
```

The test will run for around 5 minutes. You can monitor it from another window using nmon.

Once finished, launch a 2nd AWR snapshot in order to create the interval that will be used to build the AWR report corresponding to the period we stressed the database and want to perform the analysis.

```
$ date
Mon Oct  4 11:18:08 CDT 2021
$ sqlplus '/as sysdba'

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:18:14 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0

SQL> EXEC DBMS_WORKLOAD_REPOSITORY.create_snapshot;

PL/SQL procedure successfully completed.

SQL> exit
```

Test is now completed.

NMON has been recorded and file (scale10_100user_<date>.nmon) is under location you initiate the launch_test.sh script (most probably from /home/oracle as oracle user.

Swingbench Workload Execution Summary is in the same folder.

As oracle User.

```
$ pwd
/home/oracle
$ ls *.nmon
scale10_100user_0410_1117.nmon
$ ls -al *.xml
-rw-r--r--    1 oracle   oinstall      15071 Oct 04 11:17 scale10_100user_0410_1117.xml
-rw-r--r--    1 oracle   oinstall      15414 Feb 26 2021  scale10_100user_2602_0422.xml
```

`scale10_100user_2602_0422.xml` corresponds to the Swingbench Workload that has been executed on this infrastructure where best practices were not correctly set and customer met performance issue.

Last step is now the generation of the Oracle AWR report correspond to the two snapshots we captured at the beginning and the end of the Swingbench Workload.

3/ Generate Oracle AWR Report for Database Activity Analysis or OraDetective Execution (2nd Part)

The AWR report is created via the script awrrpt.sql which is found in the directory $ORACLE_HOME/rdbms/admin

This directory is the home for a lot of the administration scripts used to manage an oracle database.

The AWR output is written to the local directory so you can run this from /home/oracle

Connect as the Oracle user.

```
$ sqlplus / as sysdba @?/rdbms/admin/awrrpt.sql

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:22:47 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0


Specify the Report Type
~~~~~~~~~~~~~~~~~~~~~~~
AWR reports can be generated in the following formats.  Please enter the
name of the format at the prompt.  Default value is 'html'.

'html'          HTML format (default)
'text'          Text format
'active-html'   Includes Performance Hub active report

Enter value for report_type: html
old   1: select 'Type Specified: ',lower(nvl('&&report_type','html')) report_type from
dual
new   1: select 'Type Specified: ',lower(nvl('html','html')) report_type from dual

Type Specified:  html

old   1: select '&&report_type' report_type_def from dual
new   1: select 'html' report_type_def from dual



old   1: select '&&view_loc' view_loc_def from dual
new   1: select 'AWR_PDB' view_loc_def from dual



Current Instance
~~~~~~~~~~~~~~~~
DB Id          DB Name        Inst Num       Instance       Container Name
-------------- -------------- -------------- -------------- --------------
  363528841     DB19000                     1 db19000        db19000


```

```
Instances in this Workload Repository schema
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  DB Id      Inst Num   DB Name      Instance     Host
------------ ---------- ---------    ----------   ------
  363528841      1      DB19000      db19000      bestpractice
  363528841      1      DB19000      db19000      bestpractice
* 363528841      1      DB19000      db19000      bestpractice

Using 363528841 for database Id
Using          1 for instance number


Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Entering the number of days (n) will result in the most recent
(n) days of snapshots being listed.  Pressing <return> without
specifying a number lists all completed snapshots.


Enter value for num_days:

Listing all Completed Snapshots
Instance     DB Name      Snap Id       Snap Started    Snap Level
------------ ------------ ---------- ------------------ ----------

db19000      DB19000            169  04 Oct 2021 03:12    1
                                170  04 Oct 2021 03:53    1
                                171  04 Oct 2021 04:00    1
                                172  04 Oct 2021 04:20    1
                                173  04 Oct 2021 04:25    1
                                174  04 Oct 2021 05:00    1
                                175  04 Oct 2021 10:11    1
                                176  04 Oct 2021 11:00    1
                                177  04 Oct 2021 11:11    1
                                178  04 Oct 2021 11:18    1


Specify the Begin and End Snapshot Ids
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter value for begin_snap: 177
Begin Snapshot Id specified: 177

Enter value for end_snap: 178
End   Snapshot Id specified: 178



Specify the Report Name
~~~~~~~~~~~~~~~~~~~~~~~
The default report file name is awrrpt_1_177_178.html.  To use this name,
press <return> to continue, otherwise enter an alternative.

Enter value for report_name: AWRBestPracticesON_COLD.html

Using the report name AWRBestPracticesON_COLD.html
<p />
<p />
End of Report
</body></html>
Report written to AWRBestPracticesON_COLD.html
- - - -
- - - -
```

```
SQL>
SQL> exit
```

The Oracle AWR corresponding to this 1st Cold Swingbench Execution is now available and get by used by Oracle DBA or be injected into OraDetective Tool.

```
$ pwd
/home/oracle
$ ls -al *.html
-rw-r--r--    1 oracle   oinstall     707062 Oct 04 04:27 AWRBestPracticesOFF.html
-rw-r--r--    1 oracle   oinstall     906182 Oct 04 11:24 AWRBestPracticesON_COLD.html
$
```

`AWRBestPracticesOFF.html` corresponds to the initial run of the swingbench application and same workload that has been executed on this environment with some of the best practices OFF.

4/ Re-Execute Swingbench Workload with HOT Database Cache

As you have just started the database instance prior executing the swingbench workload, the data is not stored in the database instance's memory as was the case for the previous run we recorded with some the Best Practices OFF.

For this reason, it will be necessary to run the scenario twice. The first run with a cold database will have poor performance compared to the second with the memory of the instance populated. This second run should be used for comparison in the 2nd Part of this Hands-On Lab related to oraDetective report.

We will replay same steps as previously i.e launch of Swingbench Application in addition of Monitoring Data Collection. You'll use this time the nmon callback hooks previously introduced in order to trigger at the same time both nmon and Oracle AWR snapshots execution.

Let's finalize configuration of nmon callback hooks. Sample scripts triggered at start and stop of nmon recording (.ie myNmonStartAWR.ksh & myNmonStopAWR.ksh) are in Appendix for reference.

Feel Free to adapt them according to your environment and any other actions you'd like initiate at the same time as nmon recording.

As Oracle User, Update Oracle User Profile with the following variable and re-load Oracle's profile file.

export NMON_START=/usr/local/bin/myNmonStartAWR.ksh
export NMON_STOP=/usr/local/bin/myNmonStopAWR.ksh
export NMON_TIMESTAMP=1

echo "export NMON_START=/usr/local/bin/myNmonStartAWR.ksh" >> /home/oracle/.profile
echo "export NMON_STOP=/usr/local/bin/myNmonStopAWR.ksh" >> /home/oracle/.profile
echo "export NMON_TIMESTAMP=1" >> /home/oracle/.profile
. /home/oracle/.profile

```
$ echo "export NMON_START=/usr/local/bin/myNmonStartAWR.ksh" >> /home/oracle/.profile
$ echo "export NMON_STOP=/usr/local/bin/myNmonStopAWR.ksh" >> /home/oracle/.profile
$ echo "export NMON_TIMESTAMP=1" >> /home/oracle/.profile
$ . /home/oracle/.profile
$
$ cat /home/oracle/.profile
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.
```

```
# ADDED for Oracle
ORACLE_SID=db19000
export ORACLE_SID
ORACLE_HOME=/u01/app/product/19c
export ORACLE_HOME

PATH=$PATH:$ORACLE_HOME/bin:$ORACLE_HOME/OPatch

# ADDED for Swingbench
PATH=$PATH:/usr/java8_64/jre/bin

export PATH

if [ -s "$MAIL" ]           # This is at Shell startup.  In normal
then echo "$MAILMSG"        # operation, the Shell checks
fi                          # periodically.

set -o vi

export LDR_CNTRL=DATAPSIZE=64K@TEXTPSIZE=64K@STACKPSIZE=64K@SHMPSIZE=64K oracle
VMM_CNTRL=vmm_fork_policy=COR
export MPROTECT_TXT=OFF
export NMON_START=/usr/local/bin/myNmonStartAWR.ksh
export NMON_STOP=/usr/local/bin/myNmonStopAWR.ksh
export NMON_TIMESTAMP=1
$
$ env |grep NMON
NMON_START=/usr/local/bin/myNmonStartAWR.ksh
NMON_TIMESTAMP=1
NMON_STOP=/usr/local/bin/myNmonStopAWR.ksh
$
```

As Oracle user under /home/oracle, Execute the launch_test2.sh script that combines monitoring data collection and swingbench workload execution.

```
$ cd /home/oracle
$ launch_test2.sh
17826144
Author  :        Dominic Giles
Version :        2.6.0.1137

Results will be written to scale10_100user.xml.

Time            Users   TPM     TPS

Saved results to scale10_100user.xml
11:38:25 AM     0       8687    91
Completed Run.
$
```

Once run completed, xml file summarizing swingbench OLTP workload details and completed transactions is created, corresponding nmon file and Oracle AWR snapshots have been triggered into the database.

You can compare the performance of the different swingbench tests using the command:

grep TotalCompletedTransactions *.xml

```
$ pwd
/home/oracle
$ grep TotalCompletedTransactions *.xml
scale10_100user_0410_1117.xml:
<TotalCompletedTransactions>2717</TotalCompletedTransactions>
```

```
scale10_100user_0410_1133.xml:
<TotalCompletedTransactions>42785</TotalCompletedTransactions>
scale10_100user_2602_0422.xml:
<TotalCompletedTransactions>41515</TotalCompletedTransactions>
$
```

The nmon callback hooks keeps tracks of start / stop / interval execution in the following logs files : /tmp/dbsnap.txt & /tmp/snapDet.txt

Check the corresponding files have been created.

As oracle user

```
$ pwd
/home/oracle
$ ls -al *.xml
-rw-r--r--    1 oracle   oinstall      15071 Oct 04 11:17 scale10_100user_0410_1117.xml
-rw-r--r--    1 oracle   oinstall      15401 Oct 04 11:38 scale10_100user_0410_1133.xml
-rw-r--r--    1 oracle   oinstall      15414 Feb 26 2021  scale10_100user_2602_0422.xml
$
$ cat /tmp/dbsnap.txt
NMONstart - DB snap at: 11:33:21,04,10,2021
NMONstop - DB snap at: 11:38:25,04,10,2021
$
$ cat /tmp/snapDet.txt
AWRsnapshot 11:33:21,04,10,2021 179
AWRsnapshot 11:38:25,04,10,2021 180
$
```

Dbsnap.txt keeps trace of nmon start/stop/interval callback hooks and allows you to see time you initiate recording of nmon monitoring and when it ends.

snapDet.txt keeps trace of time of Oracle AWR snapshot execution. It also provides associated AWR Snapshot ID that will be then used to create the corresponding Oracle AWR Report.

The Oracle AWR related to the 2nd Execution of Swingbench Workload with Hot Database Cache will leverage Snapshot IDs 179 & 180.

5/ Generate Oracle AWR Report

As you previously generated Oracle AWR Report following the 1st Swingbench Execution and Cold Database Cache, you'll repeat the same procedure here.

As Oracle User, Execute following command

```
$ sqlplus / as sysdba @?/rdbms/admin/awrrpt.sql

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 4 11:46:01 2021
Version 19.9.1.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.9.1.0.0


Specify the Report Type
~~~~~~~~~~~~~~~~~~~~~~~
AWR reports can be generated in the following formats.  Please enter the
```

```
name of the format at the prompt.  Default value is 'html'.

'html'          HTML format (default)
'text'          Text format
'active-html'   Includes Performance Hub active report

Enter value for report_type: html
old   1: select 'Type Specified: ',lower(nvl('&&report_type','html')) report_type from
dual
new   1: select 'Type Specified: ',lower(nvl('html','html')) report_type from dual

Type Specified:  html

old   1: select '&&report_type' report_type_def from dual
new   1: select 'html' report_type_def from dual



old   1: select '&&view_loc' view_loc_def from dual
new   1: select 'AWR_PDB' view_loc_def from dual



Current Instance
~~~~~~~~~~~~~~~~
DB Id          DB Name        Inst Num       Instance       Container Name
-------------- -------------- -------------- -------------- --------------
  363528841     DB19000                     1 db19000        db19000









Instances in this Workload Repository schema
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  DB Id      Inst Num   DB Name      Instance     Host
------------ ---------- ---------    ----------   ------
  363528841      1      DB19000      db19000      bestpractice
  363528841      1      DB19000      db19000      bestpractice
* 363528841      1      DB19000      db19000      bestpractice

Using 363528841 for database Id
Using          1 for instance number


Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Entering the number of days (n) will result in the most recent
(n) days of snapshots being listed.  Pressing <return> without
specifying a number lists all completed snapshots.


Enter value for num_days:

Listing all Completed Snapshots
Instance     DB Name      Snap Id       Snap Started    Snap Level
------------ ------------ ---------- ------------------ ----------

db19000      DB19000            169  04 Oct 2021 03:12    1
                                170  04 Oct 2021 03:53    1
                                171  04 Oct 2021 04:00    1
                                172  04 Oct 2021 04:20    1
                                173  04 Oct 2021 04:25    1
                                174  04 Oct 2021 05:00    1
                                175  04 Oct 2021 10:11    1
```

```
                                 176  04 Oct 2021 11:00    1
                                 177  04 Oct 2021 11:11    1
                                 178  04 Oct 2021 11:18    1
                                 179  04 Oct 2021 11:33    1
                                 180  04 Oct 2021 11:38    1


Specify the Begin and End Snapshot Ids
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter value for begin_snap: 179
Begin Snapshot Id specified: 179

Enter value for end_snap: 180
End   Snapshot Id specified: 180



Specify the Report Name
~~~~~~~~~~~~~~~~~~~~~~~
The default report file name is awrrpt_1_179_180.html.  To use this name,
press <return> to continue, otherwise enter an alternative.

Enter value for report_name: AWRBestPracticesON_HOT.html

Using the report name AWRBestPracticesON_HOT.html
<p />
<p />
End of Report
</body></html>
Report written to AWRBestPracticesON_HOT.html
SQL>
```

Check you have now the 3 Oracle AWRs listed under /home/oracle

```
$ pwd
/home/oracle
$ ls *.html
AWRBestPracticesOFF.html      AWRBestPracticesON_COLD.html  AWRBestPracticesON_HOT.html
$
```

Section 3: Database Performance Analysis & OraDetective

This section is reserved for IBMers who can run oraDetective Tool and proceed with Performance Analysis leveraging Monitoring Data Collection that has been captured during Swingbench Workload Execution.

Partners : Although you can NOT access and execute oraDetective Tool, you'll find in this section the generated reports directly built from oraDetective. We strongly recommend to contact your local IBM Power Client Technical Specialist to request execution of OraDetective for performance report analysis

It can be used as a 1st Level of engagement before going deeper in the analysis and involving L2 Oracle/AIX Support SMEs.

OraDetective has been developed by the IBM Advanced Technology Group, ISV on Power (Oracle), Ralf Schmid-Dannert and his team. The tool is access controlled and that access permissions may need to be requested if not already granted. For any questions, comments related to the tool, You can send an email to the development team. (dannert@us.ibm.com)

The Tool is often updated, we recommend to bookmark the following URL and download the latest version of the tool here : https://w3.ibm.com/w3publisher/advanced-technical-group-oracle/oradetective

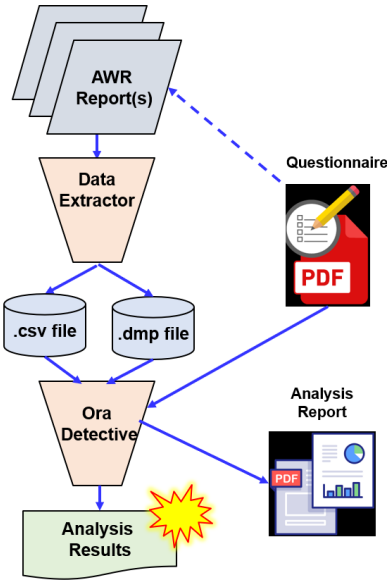
The objective of oraDetective is to analyze existing Oracle database workloads and quickly identify workload bottlenecks and tuning opportunities to provides analysis results in a customer deliverable report.

The Oracle Wait Events are analyzed via an automated process. Oracle Wait Time is an indication of physical system (CPU, I/O, Memory) resources being used by the workload and/or potential contention bottlenecks on those critical hardware resources .

Here is the process flow to use OraDetective Tool.

OraDetective process flow

1. Oracle AWR report(s) for an existing Oracle database environment are obtained from the customer
2. AWR report(s) are processed by the "Data Extractor" tool to extract desired data fields:
   - Most metrics are written to ROW.csv file
   - Detailed File I/O statistics, if available, written to .dmp file
3. The resulting ROW.csv and .dmp files are imported into OraDetective
4. Analysis results can be interactively reviewed directly within OraDetective
5. A customer deliverable report, extended with input from the questionnaire, can be generated from within the tool



1/ Install Oracle Detective on your laptop. OraDetective requires **Excel on Windows only.**

If you can not run OraDetective on your Laptop, you will not be able to complete the following part of the Lab which is creating Performance Report based on the Oracle AWR you captured. We placed under /home/oracle/oraDetectiveReports both OraDetective reports you would have got to perform comparison between one run where best practices were not applied and the same run with best practices on. You can jump directly to the next section 4.6 for a read through the report.

You can not run DataExtractor and can not process ROW.csv files on MAC OS, BUT but you can open an existing OraDetective spreadsheet with already processed data.

Download latest oraDetective version from w3publisher Oracle ATG page

Unzip OraDetective_<version>.zip on your laptop at the location you want. Adapt ORADETECTIVE_HOME variable with your Installation Path

ORADETECTIVE_HOME = "C:\Users\FREDERICDubois\Downloads"

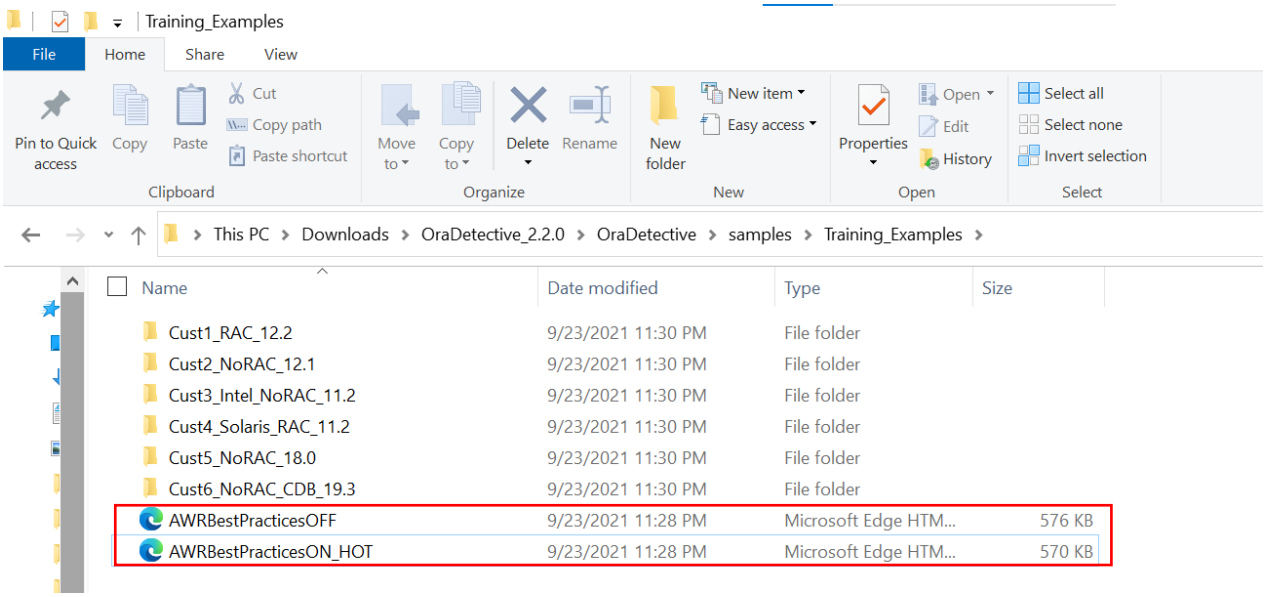
2/ Transfer the AWR report from your AIX LPAR to your Laptop to perform OraDetective using FTP or SCP method.

We want to compare Database behaviour with and without best practices. AWRs that will be used as data source are **AWRBestPracticesOFF.html** and **AWRBestPracticesON_HOT.html**

3/ Copy the file to the following directory:

$ORADECTIVE_HOME\OraDetective_2.2.0\OraDetective\samples\Training_Examples

Review in the %ORADETECTIVE_HOME%\OraDetective_2.2.0\OraDetective\samples\Training_Examples\ that the AWRs are present.



4/ go to your OraDetective home in the cmd window and run the data extractor tool command line

cd %ORADETECTIVE_HOME%\OraDetective_2.2.0\OraDetective

The DataExtractor application is used to extract data from specific files using the rules specified in a rules spreadsheet. In the context of Performance Detective for Oracle databases (OraDetective) those specifc files are Oracle AWR files in html format. OraDetective distribution package includes several different rules files. The rules file used will need to match the Oracle database version the Oracle AWR files were created on.

It is recommended to use a separate destination directory per customer as a practical best practice. Without filtering on AWR Name, DataExtractor will process any AWR reports in HTML recursively from the inputs directory. Isolating AWRs set per customer under a dedicated directory will avoid such issues.

You can run DataExtractor twice specifying each time, the Oracle AWR you consider to build OraDetective Report. You could also run DataExtractor to combine both AWRs (captured during to different intervals) to compare behaviour changes.

The Following instructions will generate one oraDetective Report per AWR Interval capture. 1st Report (with Best Practices OFF) may correspond to the situation you got looking at the customer environment and thus looking for root cause of poor performance. Then you can build same report (based on Oracle AWR with Best Practices ON to identify if any Infrastructure Tuning iteration are required. Otherwise you can then build a report combining both situations i.e with and without Best Practices.

DataExtractor.exe cmd=extract "debug=no" "sourcepath=.\samples\Training_Examples" "outputfile=.\samples\Training_Examples\AWRBestPracticesOFF.xlsx" "rulesfile=.\config\OraDetective_Rules_18.6_to_19.X.xlsx" "maxThreads=4" "filter=AWRBestPracticesOFF.html"

```
C:\Users\081691706\Downloads\OraDetective_2.2.0\OraDetective>
C:\Users\081691706\Downloads\OraDetective_2.2.0\OraDetective>DataExtractor.exe cmd=extract "debug=no" "sourcepath=.\samp
les\Training_Examples" "outputfile=.\samples\Training_Examples\AWRBestPracticesOFF.xlsx" "rulesfile=.\config\OraDetectiv
e_Rules_18.6_to_19.11.xlsx" "maxThreads=4" "filter=AWRBestPracticesOFF.html"
Started Data Extractor at 23-Sep-2021 23:38:33.
Code Version: 1.4.0
Command: extract

Total files to process are 1.

Rules File: .\config\OraDetective_Rules_18.6_to_19.11.xlsx
Rules file version: OraDetective^2.1.2^18.6^19.11


Max threads is set to 4.
Queuing the files to process.
Total Files: 1 To process: 0 In Progress: 0 Complete: 1


Extracted data from 1 files.
Successfully written data to spreadsheet .\samples\Training_Examples\AWRBestPracticesOFF.xlsx.
Successfully written data to row CSV: .\samples\Training_Examples\AWRBestPracticesOFF_ROW.csv

Done executing Data Extractor at 23-Sep-2021 23:38:59.
Elapsed Time:    26 seconds

C:\Users\081691706\Downloads\OraDetective_2.2.0\OraDetective>
```

Repeat this step and run DataExtractor to process AWRBestPracticesON_HOT.html AWR Report.

Update Filter & output_file arguments.

→ "filter=AWRBestPracticesON_HOT.html"

➔ "outputfile=.\samples\Training_Examples\AWRBestPracticesON_HOT.xlsx"

```
C:\Users\081691706\Downloads\OraDetective_2.2.0\OraDetective>DataExtractor.exe cmd=extract "debug=no" "sourcepath=.\samp
les\Training_Examples" "outputfile=.\samples\Training_Examples\AWRBestPracticesON_HOT.xlsx" "rulesfile=.\config\OraDetec
tive_Rules_18.6_to_19.11.xlsx" "maxThreads=4" "filter=AWRBestPracticesON_HOT.html"
Started Data Extractor at 23-Sep-2021 23:47:01.
Code Version: 1.4.0
Command: extract

Total files to process are 1.

Rules File: .\config\OraDetective_Rules_18.6_to_19.11.xlsx
Rules file version: OraDetective^2.1.2^18.6^19.11


Max threads is set to 4.
Queuing the files to process.
Total Files: 1 To process: 0 In Progress: 0 Complete: 1


Extracted data from 1 files.
Successfully written data to spreadsheet .\samples\Training_Examples\AWRBestPracticesON_HOT.xlsx.
Successfully written data to row CSV: .\samples\Training_Examples\AWRBestPracticesON_HOT_ROW.csv

Done executing Data Extractor at 23-Sep-2021 23:47:27.
Elapsed Time:    26 seconds

C:\Users\081691706\Downloads\OraDetective_2.2.0\OraDetective>
```

Repeat this step and run DataExtractor to process both AWRBestPractices*.html AWR Report in order to combine them within the same oraDetective Report.

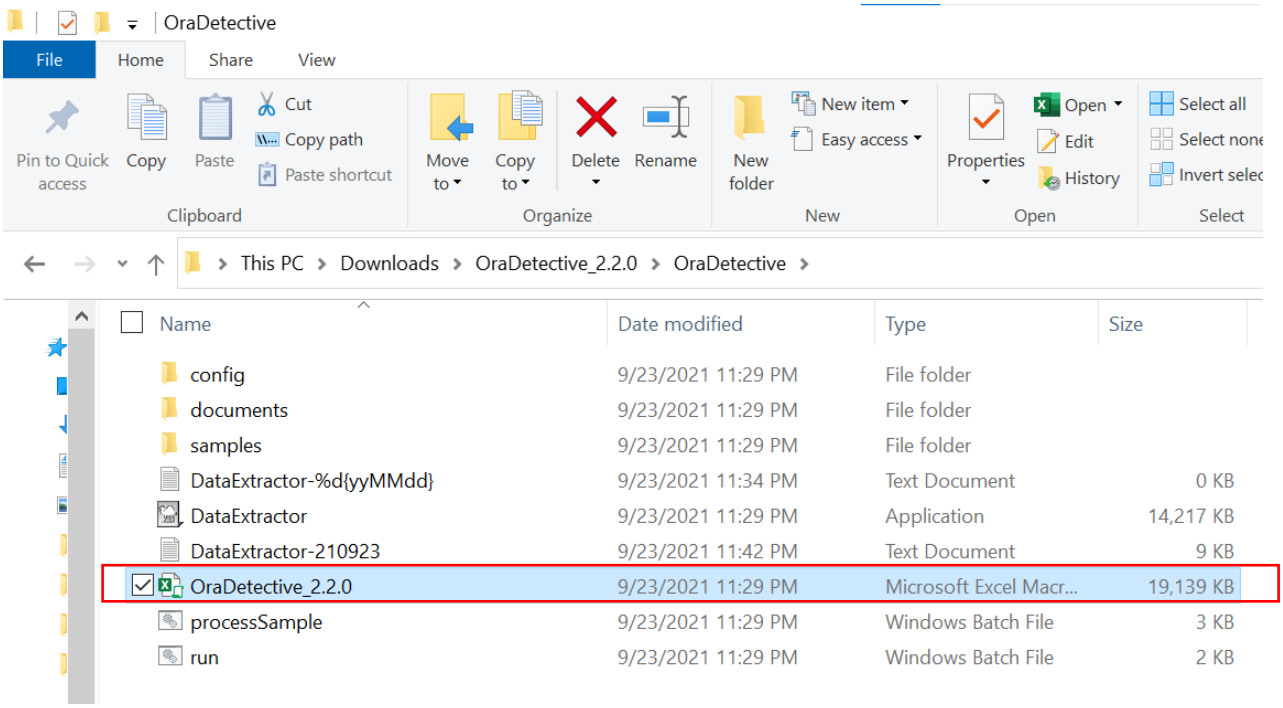
Update Filter & output_file arguments.

→ "filter=*.html"

➔ "outputfile=.\samples\Training_Examples\AWRBestPractices.xlsx"

```
C:\Users\081691706\Downloads\OraDetective_2.2.1\OraDetective>DataExtractor.exe cmd=extract "debug=no" "sourcepath=.\samp
les\Training_Examples" "outputfile=.\samples\Training_Examples\AWRBestPractices.xlsx" "rulesfile=.\config\OraDetective_R
ules_18.6_to_19.X.xlsx" "maxThreads=4" "filter=*.html"
Started Data Extractor at 13-Dec-2021 12:15:11.
Code Version: 1.4.1
Command: extract

Total files to process are 2.

Rules File: .\config\OraDetective_Rules_18.6_to_19.X.xlsx
Rules file version: OraDetective^2.1.3^18.6^19.11


Max threads is set to 4.

Queuing the files to process.
Total Files: 2 To process: 0 In Progress: 0 Complete: 2


Extracted data from 2 files.
Successfully written data to spreadsheet .\samples\Training_Examples\AWRBestPractices.xlsx.
Successfully written data to row CSV: .\samples\Training_Examples\AWRBestPractices_ROW.csv

Done executing Data Extractor at 13-Dec-2021 12:15:53.
Elapsed Time:    42 seconds

C:\Users\081691706\Downloads\OraDetective_2.2.1\OraDetective>^S
```

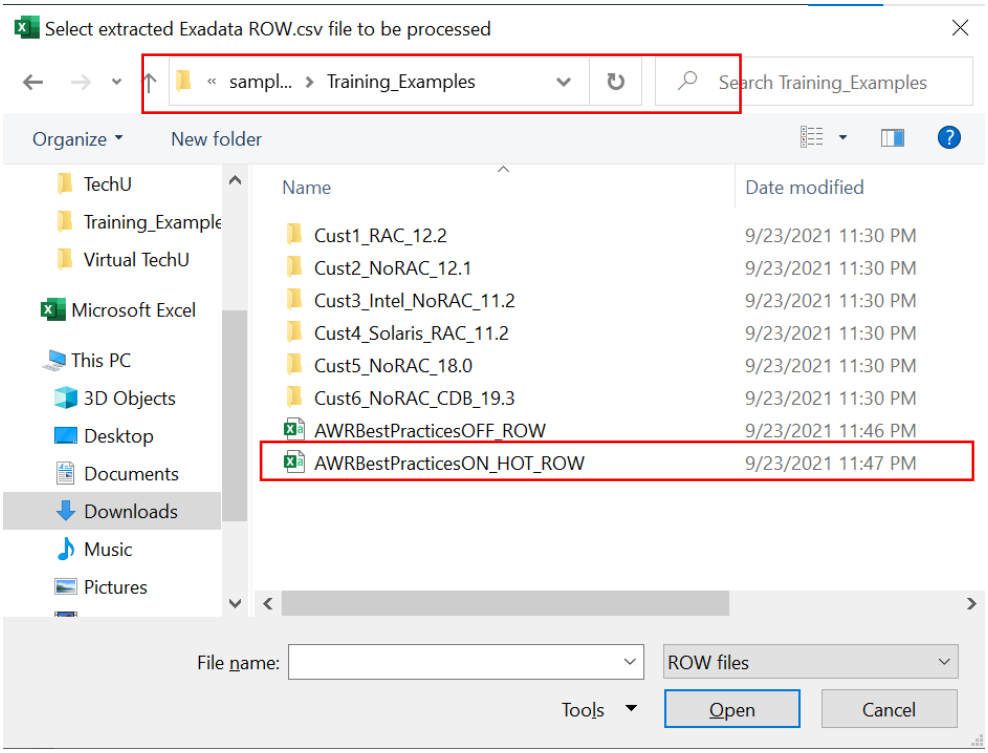
5/ open the OraDetective_v2.2.0 excel file.



If this is the first time you have used the spreadsheet you may need to enable editing and enable content before being able to use the tool.

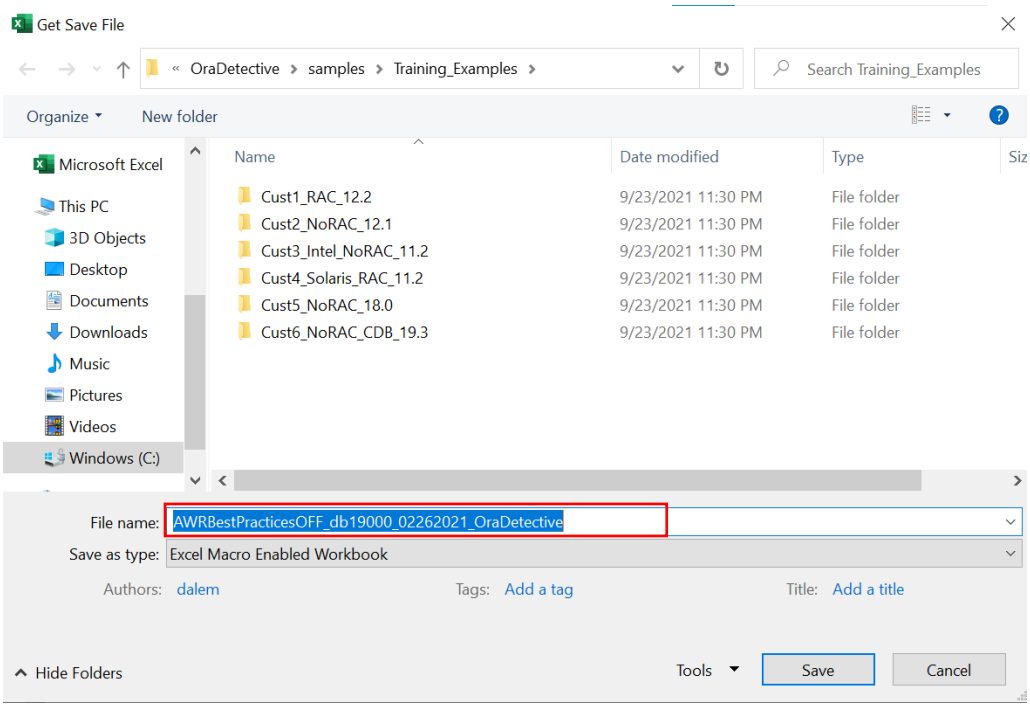
Click on the "**Reset All**" push button at the top of the "Main" worksheet to clear out any existing data in the tool.

Click on the "**Import Data**" push button at the top of the "Main" worksheet to import contents of the Data Extractor "*ROW.csv" output. You should find the **AWRBestPracticesOFF_ROW.csv** in the samples > Training Examples directory.

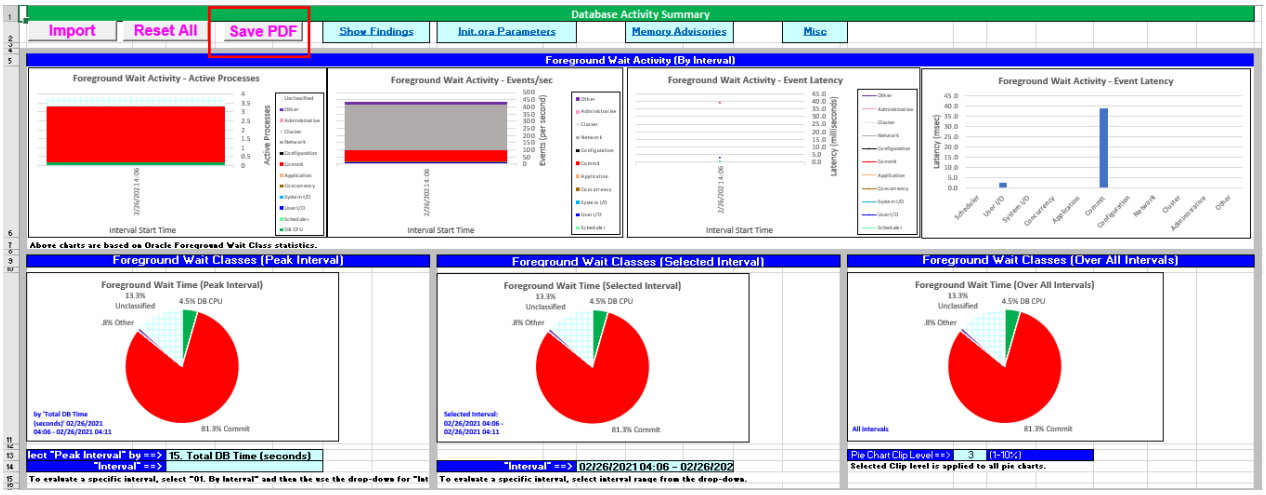


At this point, you may interactively review the results within the tool.

Once data imported, you are asked to save the spreadsheet under a new name, click Yes.



The screenshot below is an illustration of the OraDetective Tool.



If you want to generate a customer deliverable pdf report file, click on the "**Save PDF**" push button at the top of the "Main" worksheet.

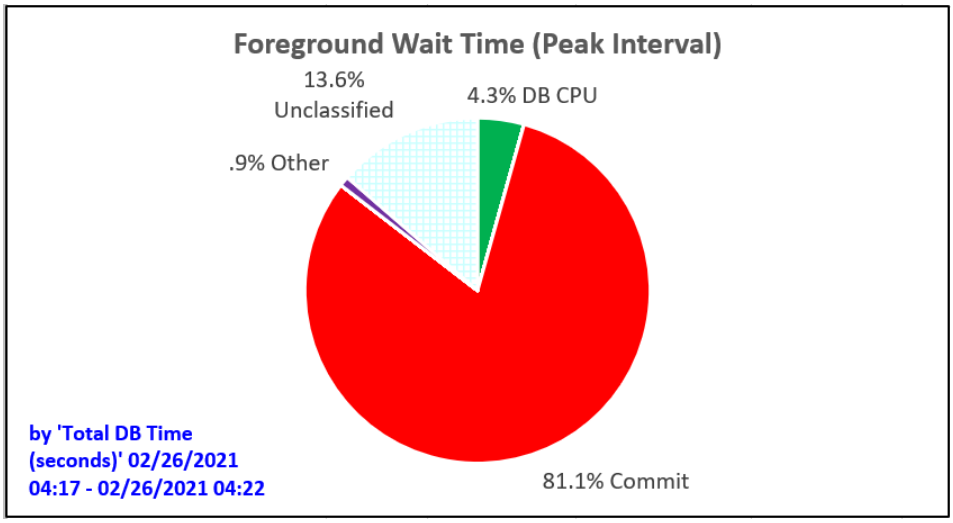
You will be prompted to add Customer details to add to the report, for this demonstration leave them blank and continue.

6/ Review the OraDetective report

For Partners or IBMers that may not be able to execute OraDetective, You can find a copy of the OraDetective with Best Practices OFF already generated in pdf under **/home/oracle/oraDetectiveReports directory.**

This section is about reviewing client ready report generated by the OraDetective Tool.

We see the following in the Top findings section of the Excel and the pdf.



The top foreground events should be DB CPU and I/O related. Anything else means that the database instance is spending time doing something other than processing data. Commit activity is essential for a database instance that is updating data but it shouldn't be more significant than the I/O itself.

Msg Id	Analysis Result
WC-07	'Commit' is the number 1 ranked Foreground Wait Class and accounted for 81.1% of the total wait time. High Commit activity may be normal behavior for this workload, but it may also an indication of issues such as excessive indexes, or poorly written SQL.
Commit-01	The 'log file sync' wait time for the peak interval (02/26/2021 04:17 - 02/26/2021 04:22) is 81.1% of total DB time. This occurs at the end of a transaction when the DBWR process must write redo log data to disk.Segregating the redo log file onto separate disk spindles, moving the redo logs to Flash storage, or increasing the log_buffer size above 10 megabytes (It is automatically set in 11g and beyond) may help. If I/O is slow (> 15 ms), there may be an I/O bandwidth issue. If the vmstat runqueue column is greater than cpu_count, then the instance is CPU-bound and this can manifest itself in high wait time. Tune SQL to reduce CPU overhead, add processors, or 'nice' the dispatching priority of the LGWR process. Poorly-written applications may issue COMMITs too frequently, causing high LGWR activity and log file sync waits. Reduce the frequency of COMMIT statements. Check the server for RAM swapping, and add RAM if the instance processes are getting paged-out.
CPU-02	On average, 80.1% of total system CPU usage is not accounted for as Oracle wait time. This could indicate significant non-DB related activity on this system, or unaccounted for Oracle activity.
WC-12	13.6% of the total DB time is not accounted for in one of the Foreground Wait Class categories. If there is a large amount of 'Non-DB' CPU time, much of this time may actually be DB CPU that was not captured by AWR.

The main thing that we see in the report is that there is excessive commit activity and that the Log File Sync wait event is very high.

The WC-07 advice highlights the 81.3% of commit activity but clearly the 'log file sync' event is 81.1% also so this is the more important point of the two as all of the commit activity is log file sync.

The recommendation from Oracle for the Log File Sync wait event is to improve the I/O throughput for the redo logs.

We mentioned earlier that the redo logs should be placed in a filesystem with 512 byte blocks but the Oracle DBCA tool automatically created the redologs database in the /oradata filesystem with a 4K blocksize. This means that when the test was run the database was experiencing demoted I/O. It had to

read the 4K block to update it with 512 bytes of data and then write it again causing 1 write I/O to become 1 read , 1 update and 1 write.

To resolve the issue the redologs were moved to /oraredo. This is a simple process that can be done online without stopping the database. The move has already been performed but the script is here for information:

```
-- 1. create the redolog
ALTER DATABASE ADD LOGFILE GROUP 1
('/oraredo/db19000/DB19000/onlinelog/redo01.log' ) size 1024M reuse;
ALTER DATABASE ADD LOGFILE GROUP 2
('/oraredo/db19000/DB19000/onlinelog/redo02.log' ) size 1024M reuse;
ALTER DATABASE ADD LOGFILE GROUP 3
('/oraredo/db19000/DB19000/onlinelog/redo03.log' ) size 1024M reuse;

-- 2. Switch to make current, one of the newly created logfiles.
alter system switch logfile;
alter system switch logfile;
alter system switch logfile;

-- 3. make all non current redologs inactive
alter system checkpoint;

-- 4. Drop the old redolog files
alter database drop logfile group 4;
alter database drop logfile group 5;
alter database drop logfile group 6;
```

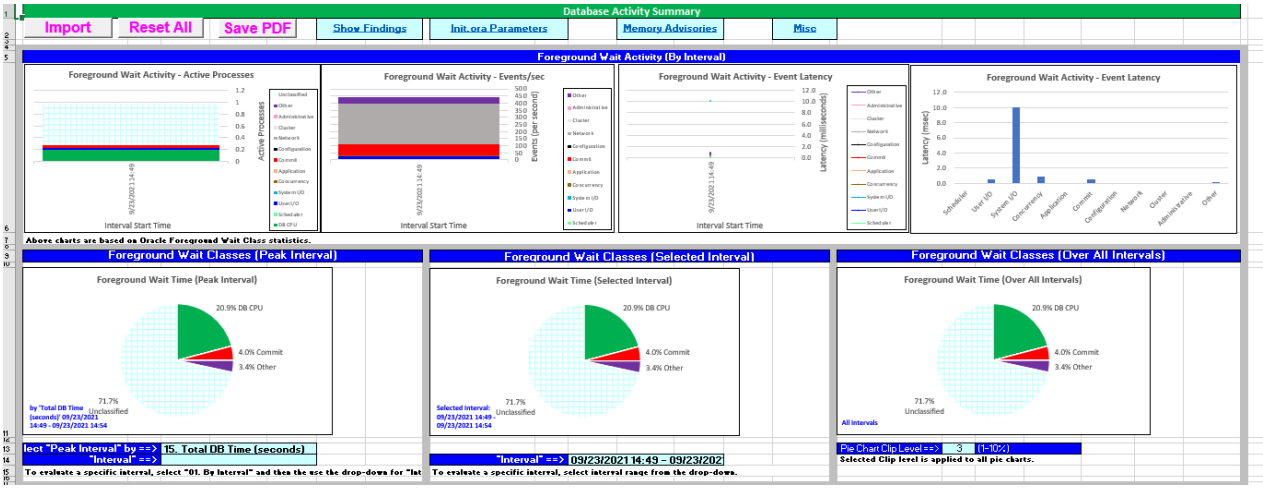
Now that you have completed your review of the Best Practices we are ready to test the scenario again with the redologs now on the oraredo filesystem with it's correctly configured 512 byte block size.

It is good practice to fix one problem at a time. If we fix several issues at once we are unsure which change was the most beneficial and equally if there are further problems we are unsure which change is the cause. Associating two changes can mean that one positive change is falsely labelled as the cause of an issue.

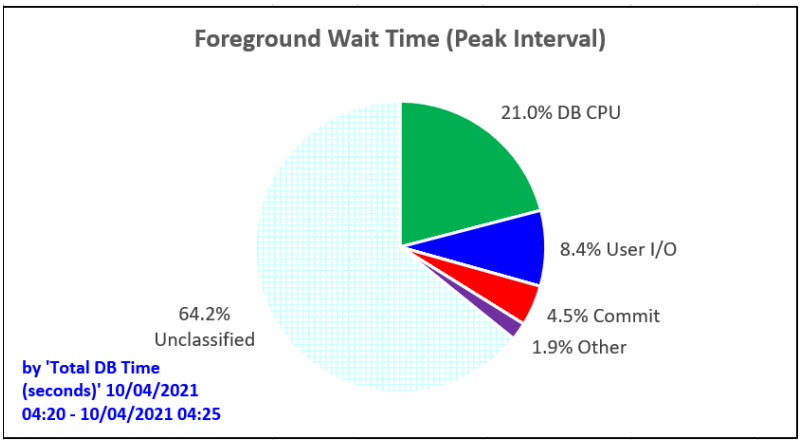
Repeat oraDetective to generate customer deliverable report using Oracle AWR corresponding to the period YOU exercise the database with Best Practices ON and HOT Database cache.

oraDetective2.2.0.xls → Import → AWRBestPracticesON_HOT_ROW.csv → Save as New Name → Save As PDF

The screenshot below is an illustration of the OraDetective Tool using Oracle AWR with Best Practices ON and HO DB Cache.



This is now the following Top findings section of the Excel and the pdf. You should have similar output.



The Commit activity has strongly decreased versus 1st without Best Practices On.

It represents now 4.5% of DB CPU versus 81% initially.

Msg Id	Analysis Result
WC-12	64.2% of the total DB time is not accounted for in one of the Foreground Wait Class categories. If there is a large amount of 'Non-DB' CPU time, much of this time may actually be DB CPU that was not captured by AWR.
CPU-02	On average, 78.6% of total system CPU usage is not accounted for as Oracle wait time. This could indicate significant non-DB related activity on this system, or unaccounted for Oracle activity.
Commit-01	The 'log file sync' wait time for the peak interval (10/04/2021 04:20 - 10/04/2021 04:25) is 4.5% of total DB time. This occurs at the end of a transaction when the DBWR process must write redo log data to disk.Segregating the redo log file onto separate disk spindles, moving the redo logs to Flash storage, or increasing the log_buffer size above 10 megabytes (It is automatically set in 11g and beyond) may help. If I/O is slow (> 15 ms), there may be an I/O bandwidth issue. If the vmstat runqueue column is greater than cpu_count, then the instance is CPU-bound and this can manifest itself in high wait time. Tune SQL to reduce CPU overhead, add processors, or 'nice' the dispatching priority of the LGWR process. Poorly-written applications may issue COMMITs too frequently, causing high LGWR activity and log file sync waits. Reduce the frequency of COMMIT statements. Check the server for RAM swapping, and add RAM if the instance processes are getting paged-out.
IO-13	The 'DB File Sequential Read' wait time for the peak interval (10/04/2021 04:20 - 10/04/2021 04:25) is 4.2% of total DB time. This is the amount of time related to single-block read requests. These are typically associated with transactional OLTP or batch workloads that perform index lookups by key. Buffer cache tuning to reduce physical read I/Os, or placing the most active data files on Flash storage may help alleviate this issue.
IO-11	The 'DB File Scattered Read' wait time for the peak interval (10/04/2021 04:20 - 10/04/2021 04:25) is 3.2% of total DB time. This is the amount of time related to multi-block read requests. These are typically associated with table or index scans. Tuning the 'hog' SQL statements (e.g. adding a more selecting index or running full table stats) or improving the I/O subsystem throughput bandwidth might help alleviate this issue.

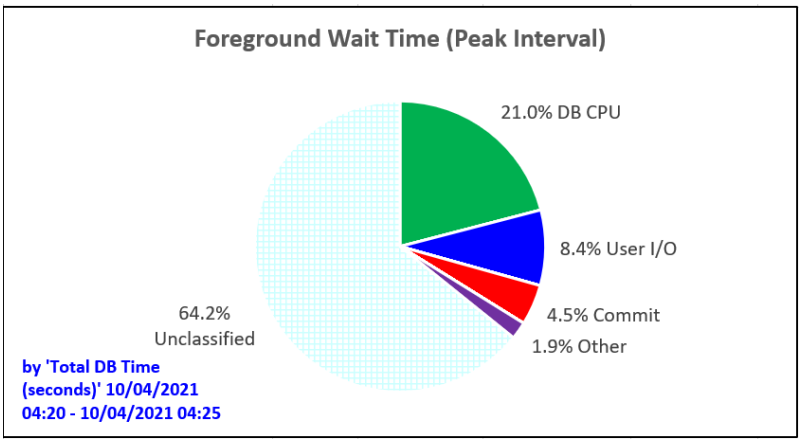
Section 5: Comparison.

We now have Oracle AWRs and OraDetective Reports corresponding to two runs : One pre-recorded with Best Practices OFF (Oracle Redo Logs placement on uncorrect LVM/Filesystems) and one with Best Practices ON (Oracle Redo Logs Placement changed).

We will now proceed in this section in reviewing both OraDetective Reports.

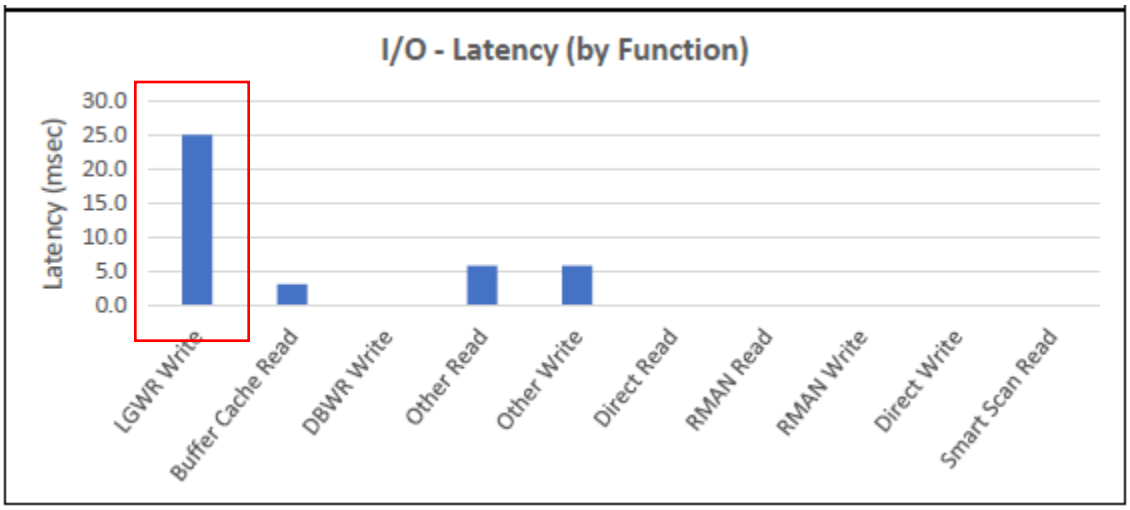
You can find both reports on your environment under /home/oracle/oraDetectiveReports directory. It contains both reports i.e the one with Best Practices OFF and one we captured while developing this Lab with Best Practices ON.

You can see in the Top 10 Events that DB CPU and User I/O have increased. Commit and Log File Sync related events have significantly decreased.

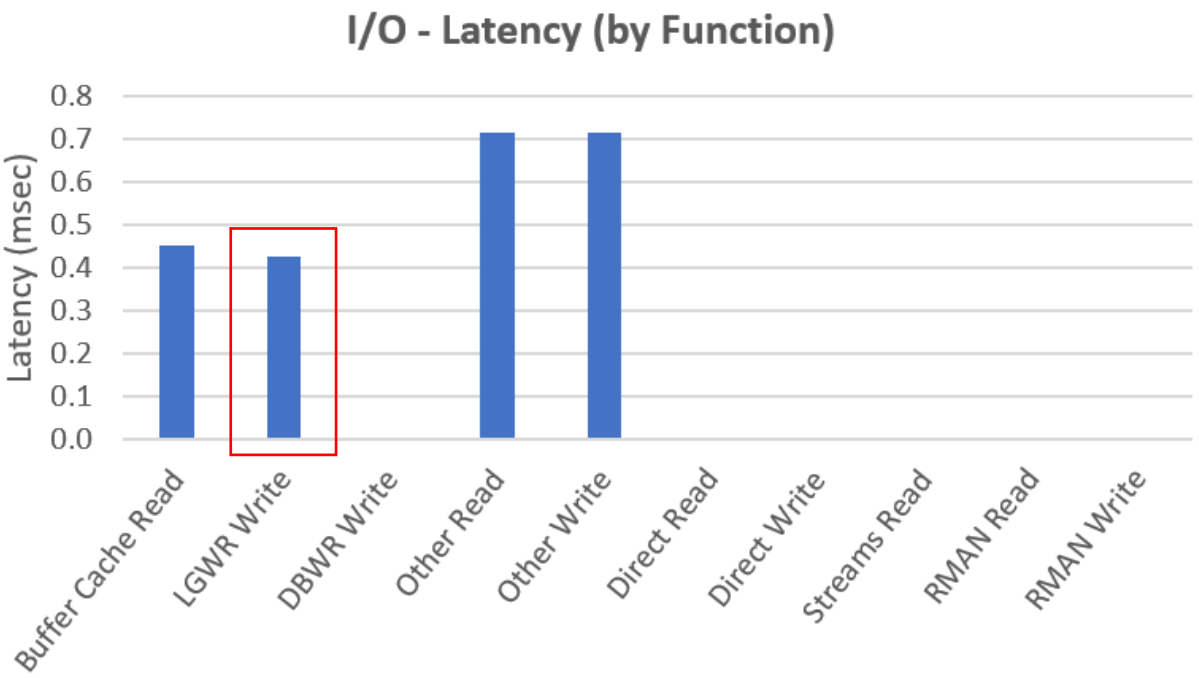


As mentioned before the target for an Oracle database is to have the foreground wait events dominated by User I/O and DB CPU. Clearly here we have gone from a situation where over 80% was long file sync to almost 80% of the DB related activity is User I/O and DB CPU.

For the first run before the move of the redologs we saw the following log writer latency:



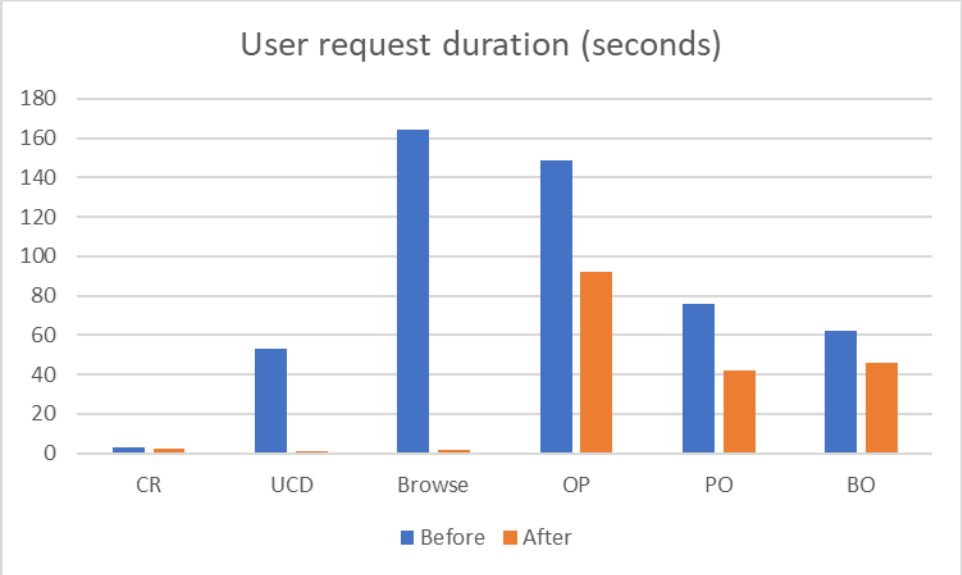
After the application of the change this becomes:



On the second run the LGWR write latency is at less than 1ms. A very clear improvement.

The number of transactions hasn't increased massively but the behavior has clearly changed.

The optimization has improved the Completed transactions from completed transaction 41500 to 42700 but the AverageResponse time should have improved and this is what an end user would be interested in.



	CR	UCD	Browse	OP	PO	BO
Before	**2.8**	**53**	**164**	**149**	**76**	**62**
After	**2.3**	**1.1**	**1.8**	**92**	**42**	**46**

We significantly reduced Oracle Log Writer Latency by 20x. Although the CPU utilization between both runs is roughly the same ~10%, the platform is now optimized to handle an increase in workload.

Without this change other performance issues are not visible.

The oraDetective is now identifying issues with the I/O reads and the real workload of the client and will allow us to further optimize the workload by increasing the DB Buffer Cache for example which is done by increasing the SGA_TARGET. To perform that change we may need to increase the memory on the lpar.

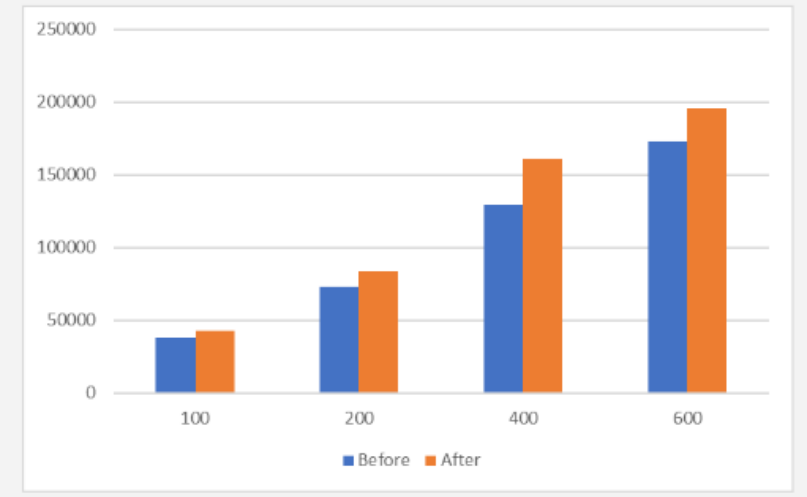
Top Findings (Best Practices)

indicate the most important areas to focus on to conform to Oracle best practices. Implementing best practices may improve

Msg Id	Analysis Result
Misc-10	There were 360.5 'table scans (short tables)/sec'. A short table is generally defined as being less than 4 data blocks or 2% of the db block buffer cache size (whichever is greater) in size. Performance may be improved by identifying those short tables with the highest table scan activity and placing them in the KEEP pool.
Init-01	The only db buffer cache area defined is for the default '8k' buffer pool. Depending on the workload, there may be benefits to using non-default db block sizes for some objects. For example, using a smaller block size for tables having a lot of random reads for a unique row (index access), or using a large block size for indexes that are frequently involved in index range scans.

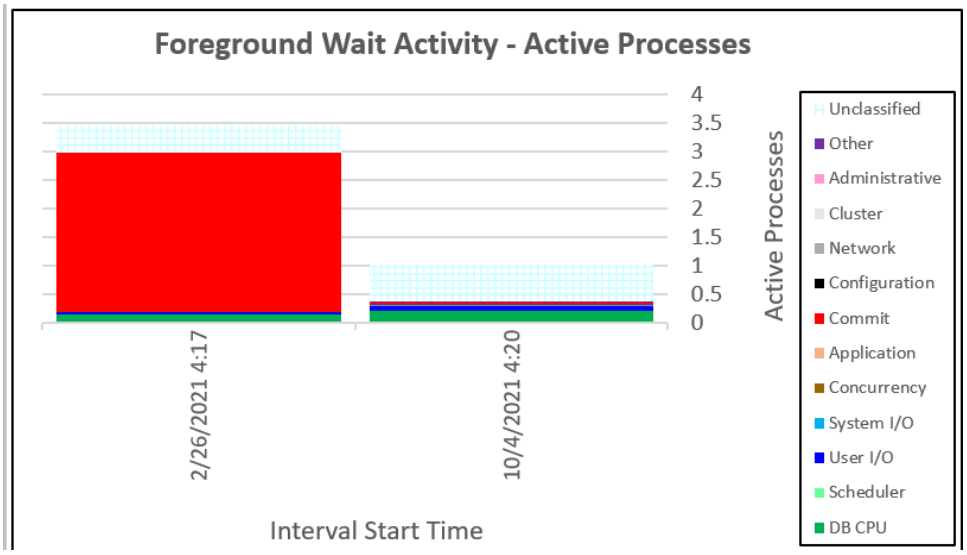
As mentioned earlier tuning a database is an iterative process and on large production systems this keeps DBAs busy full time but the essential thing is to have the best practices in place to ensure that the bottleneck is not the IBM Power Server or the AIX operating system.

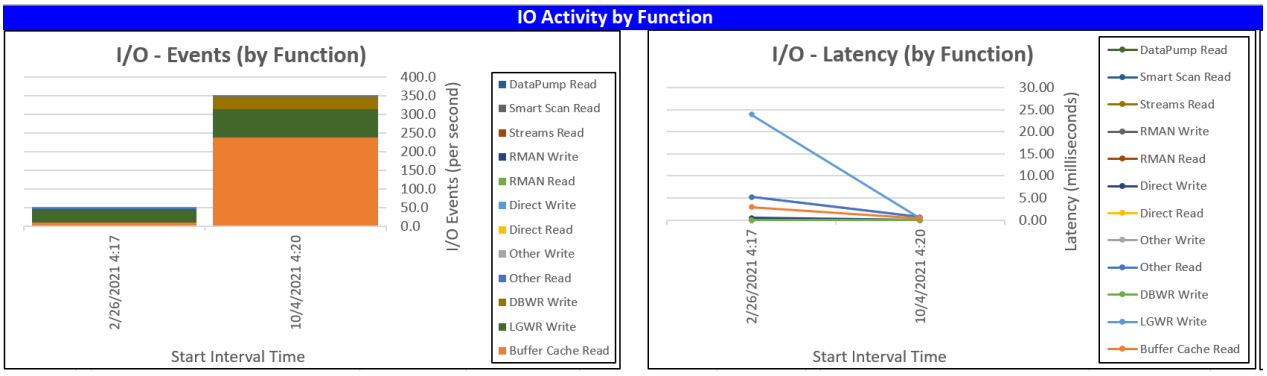
For the lab we ran the test with100 users but the benefit of the best practices is clear as we increase the number of users.



Users	Before	After
100	37912	42894
200	73076	84008
400	129301	161237
600	173429	195906

Here are oraDetective Report output of combined Oracle AWRs which illustrates difference in Foreground activity and IO Events by Function





Summary

This documents showed you steps needed to:

- Review Oracle best practices on IBM Power Systems / AIX (from architecture and performance perspective)
- Collect monitoring data -> change parameters -> collect data again to see the impact
- Perform OraDetective for performance analysis
- Please share feedbacks about this hands-on lab and/or if you have any questions related to Oracle on IBM POWER Systems and competition related topics.

Appendix

Here below is sample of nmon callback hook script used to execute Oracle AWR Snapshot when starting and stopping nmon recording.

```
#!/bin/ksh
# Set environment variables below by sourcing the file “oracleCfg”
# ORACLE_SID, DBID, TNS_ADMIN, oraUser, oraPW, snapLevel
. ~/oracleCfg
# Set output location
outdir=/tmp

# Write activity log - $1 contains time,date timestamp
echo "NMONstart - DB snap at: $1" >> $outdir/dbsnap.txt

# Trigger AWR snapshot and record date/time and snapshot number
result=`sqlplus -S ${oraUser}/${oraPW}<< EOF
set head off
set term off
set echo off
set trim on
set trimspool on
set feedback off
set pagesize 0
exec sys.dbms_workload_repository.create_snapshot(${snapLevel});
select max(snap_id) from dba_hist_snapshot where DBID=${DBID};
EOF`
echo AWRsnapshot $1 $result | tr -s \ >> $outdir/snapDet.txt 2>&1
```