

**MANOJ ENTERPRISES**  
Plot 40, Gasathri Nagar  
Ameerpet, Hyderabad

**LETTEST- 7-8-2014**

# **ORACLE 11G**

**DBA WITH RAC**

**BY**

**HANMANTH REDDY**

**MANOJ ENTERPRISES & XEROX**

**All soft ware institute materials, spiral-binding,**

**Printouts & stationery also available ..,**

**CONTACT: 9963111226**

**Add: Plot No.40, Gayatri Nagar, Behind HUDA, mithrivannam, HYD.**



## Applications

Custom Application  
Front-end programming language  
Back end database

Readymade Applications  
(1) SAP  
(2) oracle E-business suite  
(3) people soft  
*Oracle Apps*

MANOJ ENTERPRISES  
Plot: 40, Gasathri Nagar  
Ameerpet, Hyderabad

(4) JD Edwards  
(5) RAN

## RDBMS - products

ORACLE - Dary. El. Son (founder)

MySQL Server

DB2

Sybase

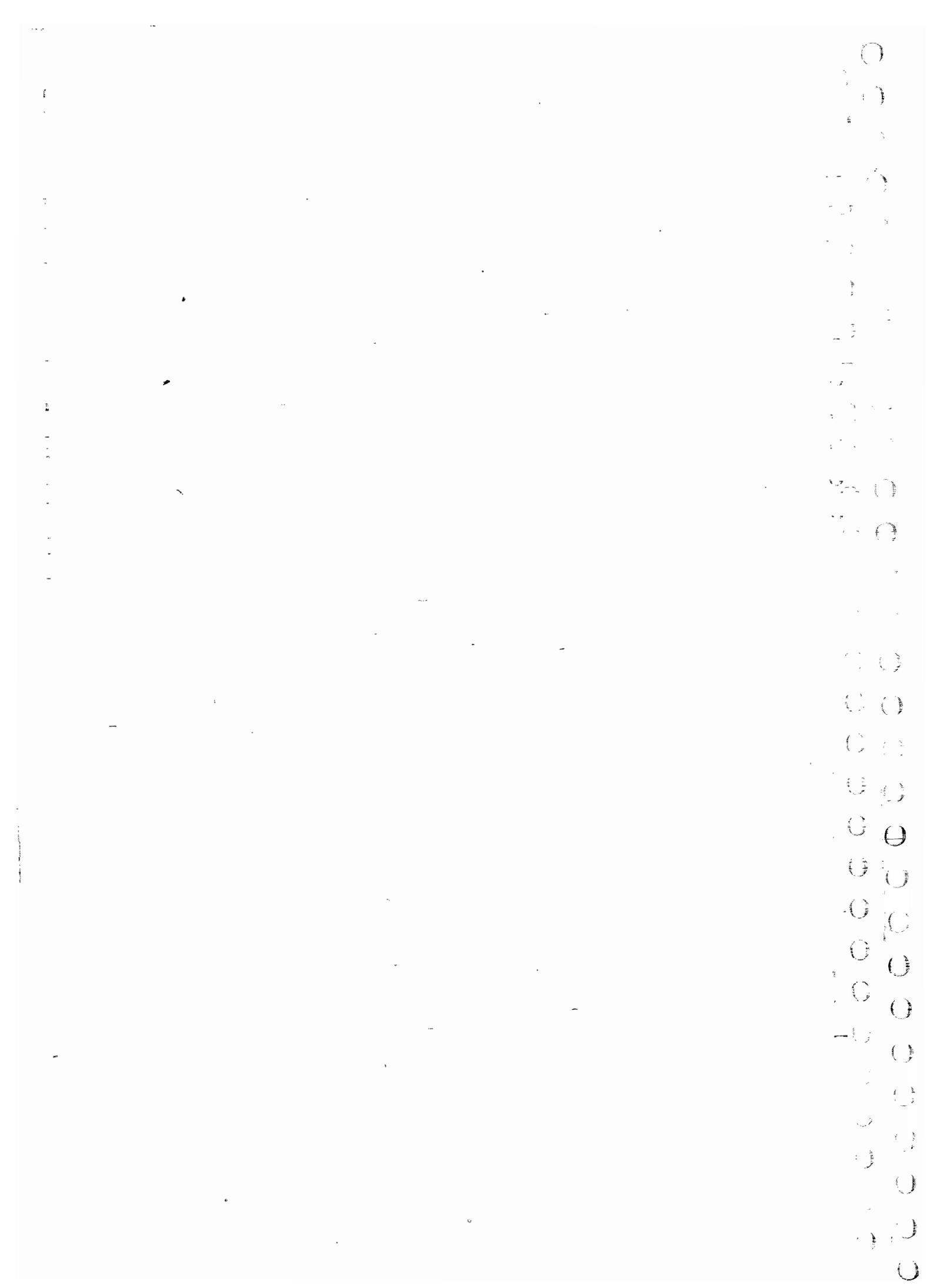
MySQL → Sun Micro Systems - 2009

post-gress

in-gress

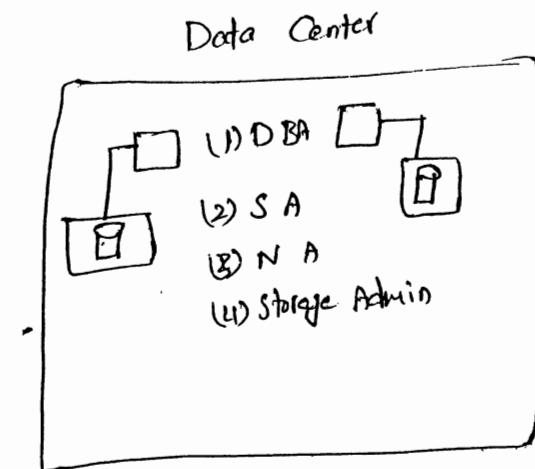
Tera data

MANOJ ENTERPRISES  
Plot: 40, Gasathri Nagar  
Ameerpet, Hyderabad



# ORACLE DBA

DBA - DL



Service provider

Vendor

ORACLE

IBM

HP

Microsoft

SAP

SWITCH

V/SP - Infysys + pinnacle.

VPN - virtual private network.

To connect the servers

putty -

SLA

Severity - I

Response time

5-10 min.

Resolution time

10-15 min

Monitoring } Mechanism.  
Alerting }

Severity - II

10 min

1 hr.

Severity - III

1 hr.

2-4 hr.

Nagios, Big brother (Oracle), OEM (grid), Site scope. } Monitoring tools.

shell scripts submitted as cron jobs) } Alert's tools

OEM - oracle enterprise Manager

### Tickets

#### Ticketing tools

- (1) BMC Remedy
- (2) clarify. ... etc.

Q. Can u work on Ticketing tool?

A) Yes, if my company has developed some internal tool.

#### Alert's with respect to instance availability

Listener

CPU usage

Memory usage

File system Space.

[www.metalink.oracle.com](http://www.metalink.oracle.com)

(or)

[www.support.oracle.com](http://www.support.oracle.com)

[CSI - Customer support identify number]

Q. In order to login to the above portals what we require?

A) CSI number.

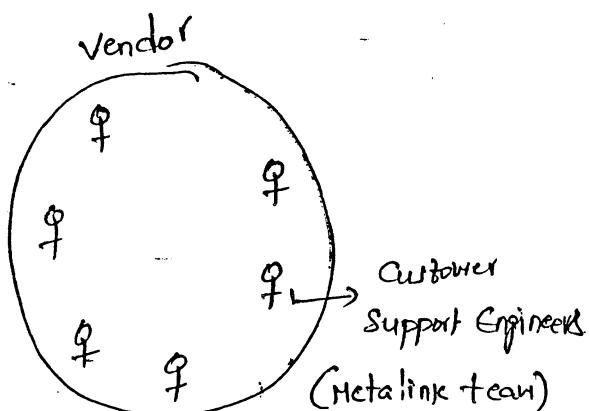
Service request

Knowledge Base

Patches

SR

Certify



## oracle Database

- ✓ (1) oracle enterprise edition - 80 - 90 %
- ✓ (2) oracle standard edition - 10 - 20 %
- ✗ (3) oracle standard edition one
- ✗ (4) oracle express edition

(1) www.google.com.

(2) www.support.oracle.com (or) www.metalink.oracle.com.

(3) www.otn.oracle.com (oracle technology network) ] download products.

(4) www.edelivery.oracle.com

(5) www.oracle.com

	<u>db</u>	<u>RAC</u>	<u>Max. No. of nodes/servers</u>
EE	47,500 \$	80000 23500	100
SE	17500 \$	[Free]	4

Features not available in S.E  
Data guard

partitioning

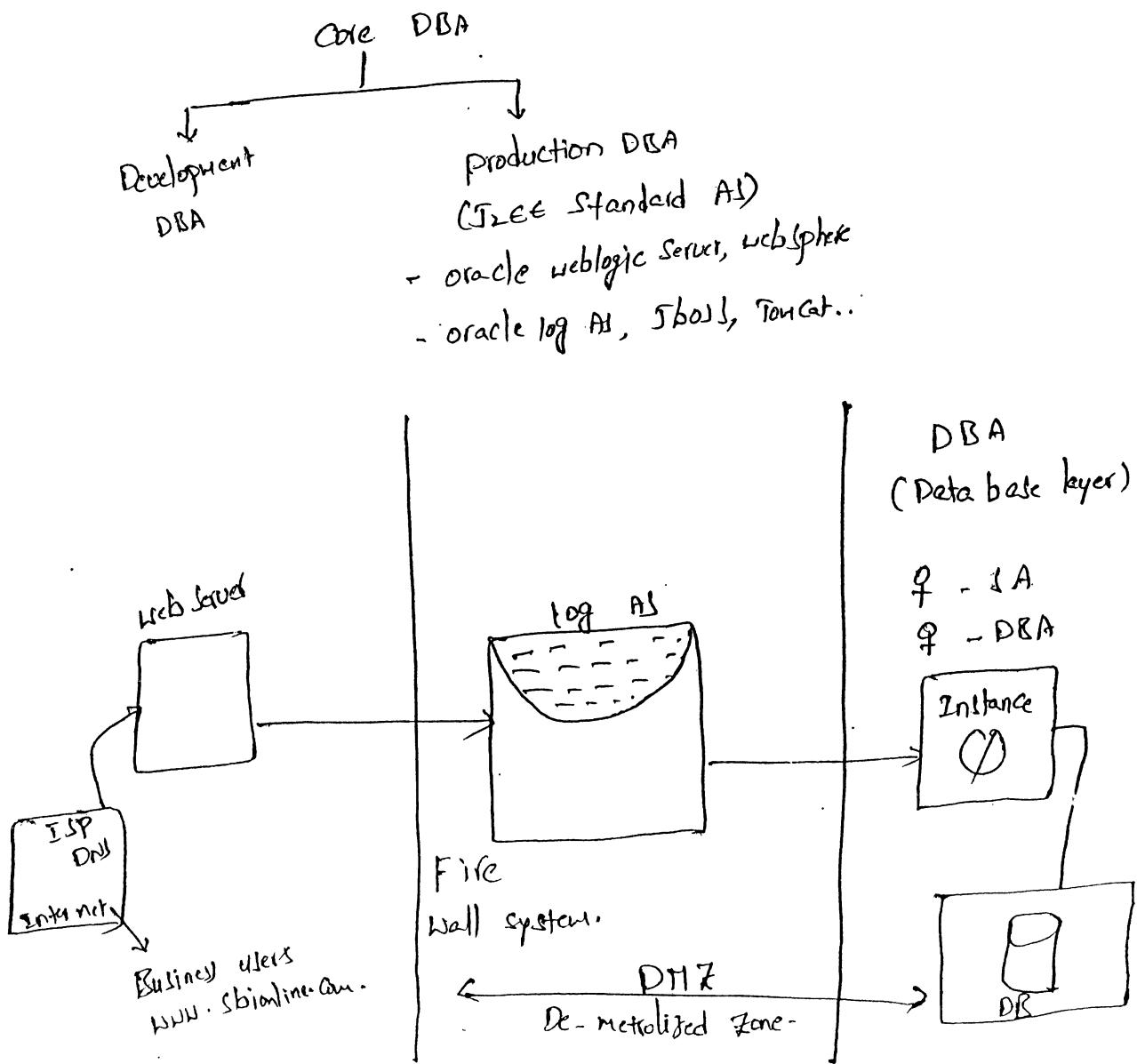
limitations in flashback technologies

we can't enable block change tracking .. etc.

\* All patches & patch sets, we can download from  
www.support.oracle.com / www.metalink.oracle.com.

♀ → IT

- (1) O/S → Java \*
- (2) Front end → .Net  
programming language  
(Web based application)
- (3) Backend db (Oracle)
- (4) Some Application Server
- (5) Web server.



7/8/14

## o/s

UNIX is platform independent (D) 777 (F) 666

UNIX is virus-free  
In UNIX even a single file can't UMASK  
execute \$ on its own  
\$ touch student

$$\begin{array}{r} 022 \\ + 755 \\ \hline 644 \end{array}$$

\$ ls -lH  
 indicates r w - r -- - t -  
 owner Group others

read - 4  
write - 2  
execute - 1  
 $\frac{1}{7}$

Microsoft SQL Server is platform dependent and it's runs on  
only windows

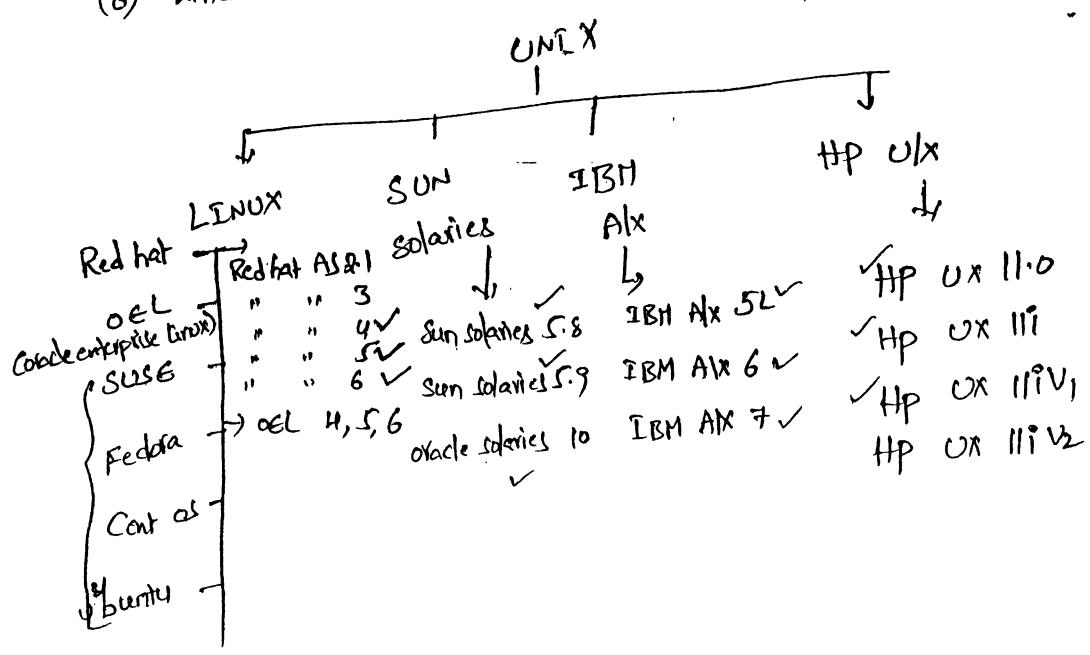
(1) What are various o/s available in the market?

Ans Windows Server o/s

(1) Windows NT

- (2) Windows 2000 Server client o/s
- (3) Windows 2003 Server
- (4) Windows 2005 Server - vista
- (5) Windows 2008 Server - windows 7
- (6) Windows 2012 Server - windows 8

IBM ✓  
Oracle ✓  
Accenture  
TCS  
SAP  
!;  
HP ✓  
CSC



(2) front end programming languages available?

Java      }  
.Net      } (web applications)

### Middle ware Technologies

Web Sphere

Web logic

Oracle log AS

Jboss

Tomcat ... etc.

### ERP

① SAP

② Oracle e-business suite

③ People soft

④ JD Edwards

⑤ BAN

### Data base

① Oracle

② Microsoft SQL Server

③ IBM DB2

④ Sybase, MySQL, Ingres, Postgress, Informix.

### Oracle Versions

Oracle 6                  Oracle 11g \* → (Grid)

Oracle 7                  Oracle 12c → (Cloud) → SAS, PAS, IAS.

Oracle 8                  Software as a service, Platform, infrastructure.

Oracle 8i ✓ → We can develop web applications from & i

Oracle 9i ✓ (Internet)

Oracle 10g \*

oracle 9i - 2000

oracle 10g - 2005

\* oracle 10g R2 - 2007 in the industry Release & only.

oracle 11g R1 - 2009

\* oracle 11g R2 - 2011

oracle 12c R1 - 2013

oracle 12c R2 - ?

3/8/14

10g R2

Base

DB Major  
Release number

version

DB Maintenance Release Number

Application server release NO)

Fusion middleware release NO.

Component Specific  
Release NO.

patch sets

- 10.2.0.2  
10.2.0.3  
10.2.0.4  
10.2.0.5

11g R1

Base version

patch set

- 11.1.0.6  
- 11.1.0.7

11g R2

Base version

patch sets

- 11.2.0.1  
- 11.2.0.2  
11.2.0.3  
11.2.0.4

Base versions  
We can download  
these from Support.  
oracle.com.

12c R1

Base version

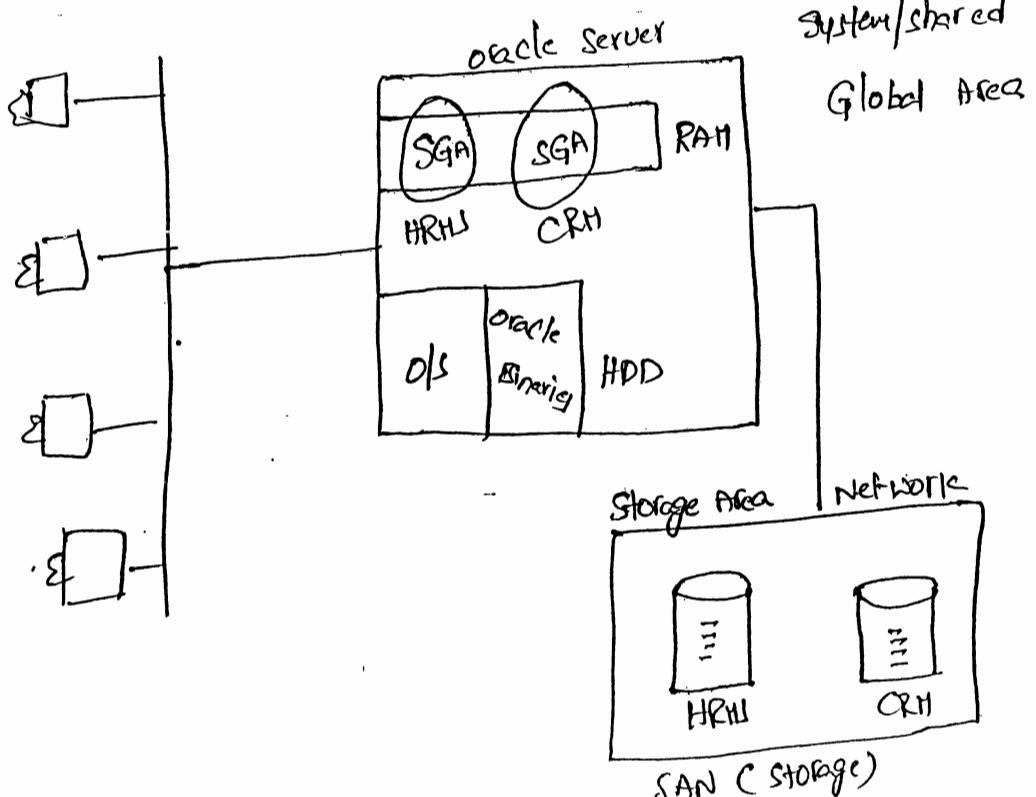
- (2.1.0.1)

## \* Oracle Server Architecture \*

- oracle server contains
- (1) oracle instance
  - (2) oracle database
  - (3) oracle binaries (S/W)
  - (4) Some mandatory files
  - (5) Some optional files
  - (6) oracle processes.

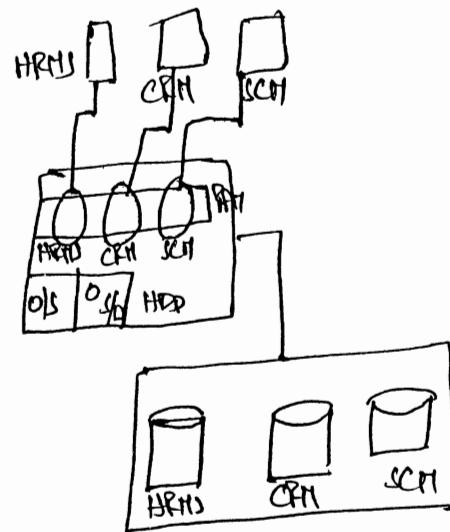
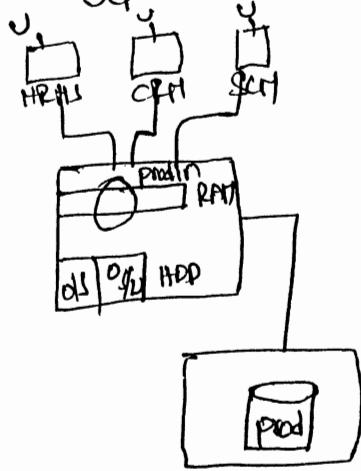
### (1) oracle instance :-

oracle instance is nothing but combination of "SGA" and background processes.



SQL> startup

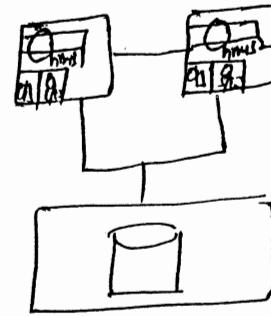
SGA is one of the memory structures of oracle.



- \* If instance is down, we can't access the data existing in the data base.

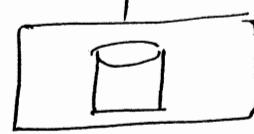
stand alone environment

DB instances  
1 : 1



RAC environment

1 : N(100)



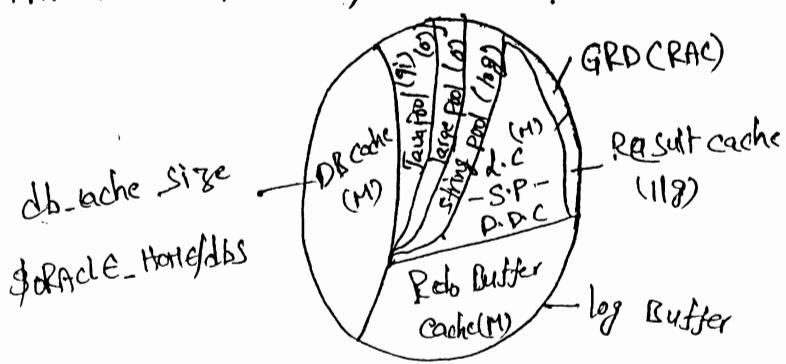
- \* To overcome instance level failures, oracle has reduced Multi instance databases, which we call them as RAC (Real Application Clusters) Data bases.

- \* Allocation of memory to SGA, purely depends on no.of concurrent users. but not on the size of the data base.

- SGA is one of memory structures of oracle.
- the two memory structures of oracle are
  - (1) SGA - System Global Area / Shared Global Area
  - (2) PGA - program Global Area.
- If the instance is down, we can't access the data existing in the database. To overcome this problem, oracle has introduced multi instance databases. which we call as "RAC" databases.

### structure of SGA:

- SGA is again subdivided into various sub components
  - M ① Database Buffer Cache
  - M ② Redo buffer cache
  - M ③ shared pool
  - ④ Java pool
  - ⑤ Large pool
  - ⑥ string pool
  - ⑦ Result cache
  - ⑧ GRD → Global Resource directory
- GRD is available only in case of RAC



stream\_pool\_size  
 large\_pool\_size  
 Java\_pool\_size  
 shared\_pool\_size  
 SQL\_start\_up  
 Pfile/sp file  
 Textfile/binaryfile

## ORACLE\_HOME/db

LINUX : \$ORACLE\_HOME/product/11.2.0/db-home

Windows : D:\oracle\product\11.2.0\db-home

init.ora

Sga\_max\_size = 12g

db\_cache\_size = 4g

shared\_pool\_size = 8g

pfile → init.ora

spfile → spfile.ora

→ If you have sufficient amount of memory allocated At least 40% of memory for SGA

→ the moment once we give start up command oracle searches for the existing  
of pfile or spfile. in \$ORACLE\_HOME/db directory

→ If the instance name is hrms, pfile name will be 'inithrms.ora' and Spfile  
name will be 'spfilehrms.ora'

→ If both files are existing, priority goes to spfile (Oracle reads spfile & starts the instance)

→ pfile is a text file whereas spfile partial text & partial binary in nature

→ edit spfile either with notepad, vi editor.

→ Based on the parameter in pfile or spfile oracle allocates the memory for  
SGA & its sub components

→ The mandatory sub components are ① DB buffer cache ② Redo buffer pool  
③ Shared pool

→ Background process are the further divided into ① Mandatory ② optional

### Mandatory BP Process

- ① Database writer (DBWR)
- ② log writer (LGWR)
- ③ System monitor (SMON)
- ④ process monitor (PMON)
- ⑤ Check point (CKPT)
- ⑥ RECO
- ⑦ MMON (Memory monitor)

**MANOJ ENTERPRISES**  
Plot: 40, Gasathri Nagar  
Ameerpet, Hyderabad

### optional BP process

- ⑧ MMAN (Memory Manager)
- ⑨ ARCH (Archiver)
- ⑩ CJAN (Co-ordinated Job Queue process)
- ⑪ CTWR (Change track writer)
- ⑫ RUWR (Recovery writer)
- ⑬ FBDA (Flashback data Archiver)
- ⑭ AHB (ASH background process)
- ⑮ RBAL (Re-balance master)

→ For the instance is up & running, all the mandatory background process  
should be up & running

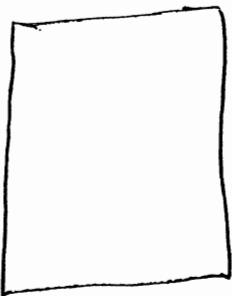
→ If any of the BP goes down, Instance get terminated

→ one of the memory structure of oracle called as - SGA

14/08/14

↳ oracle ->

oracle\_home/db directory



- \* the first file that oracle scans is either pfile/spfile. and the first file that oracle creates is Alert <SID>.log [SID - system identifier]  
[background - DESK]  
error: unable to identify.  
[Alert trns. log] eg:

\* Alert log is one of the important file for the DBA to Diagnose the problems with in the Data base

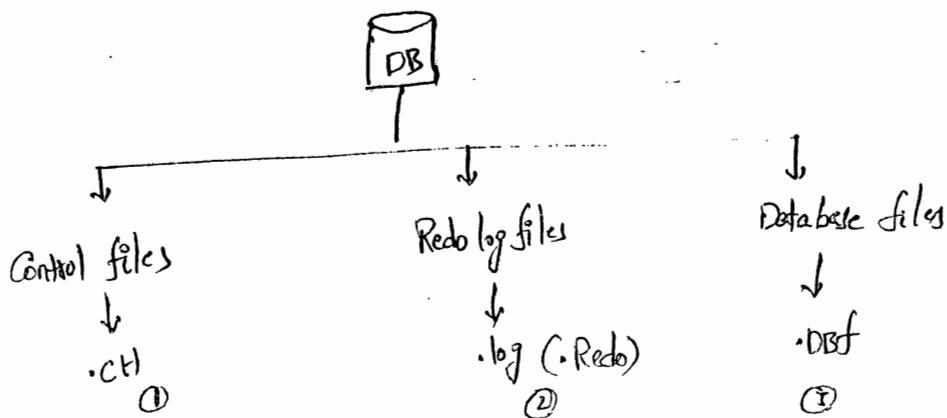
\* Alert log Contains all the initialization parameters, Administrative Commands that we execute as a DBA, log sequence numbers, warning messages and ora - related error numbers.

(1) Database is nothing but collection of 3 important files

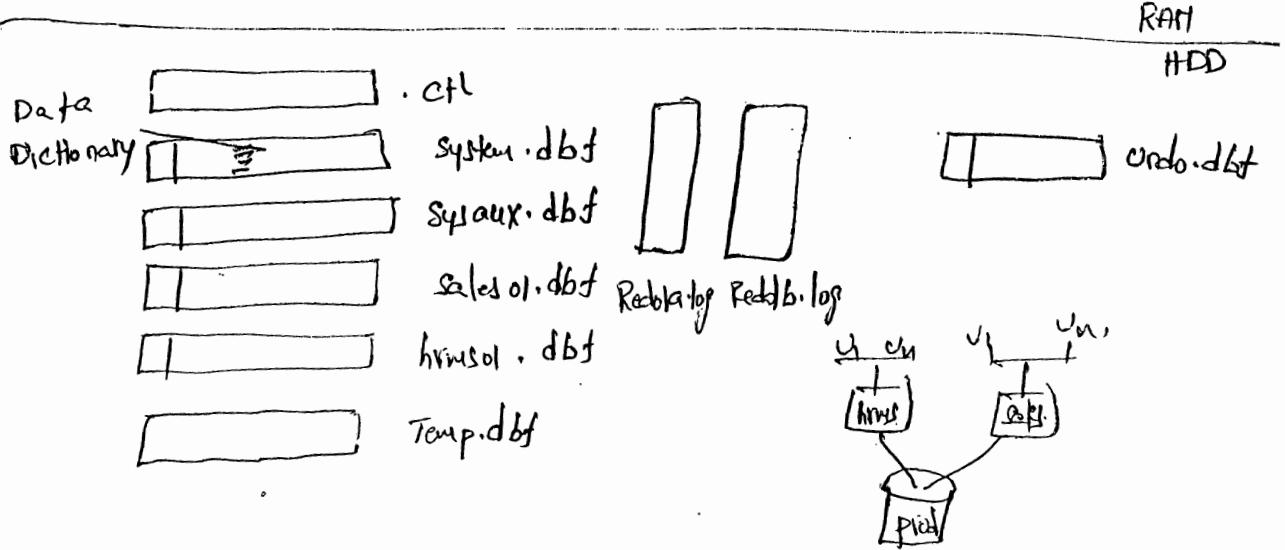
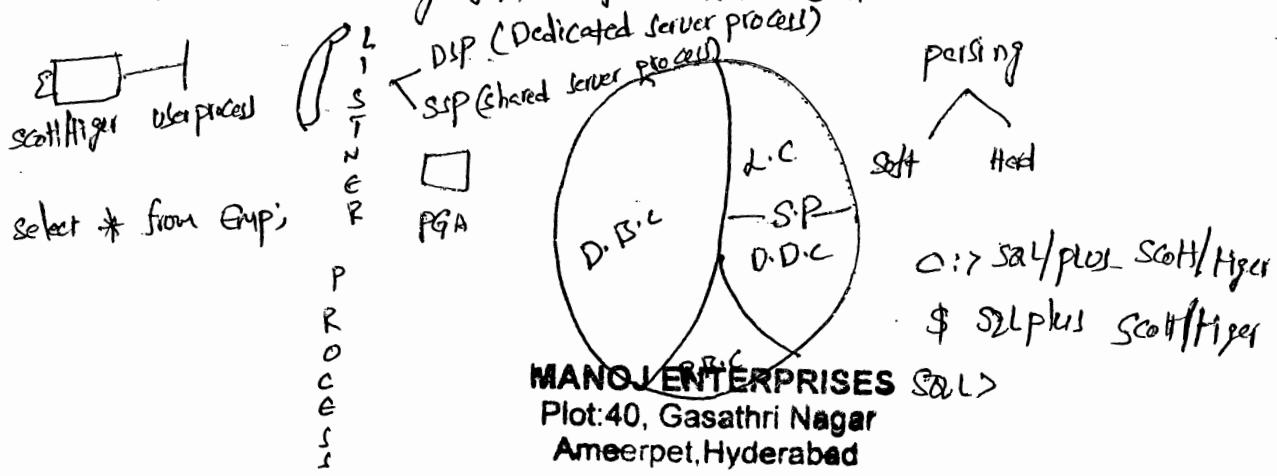
(a) Control files

(b) Re-do log files

(c) Data files.



- \* For the functionality of data base, minimum 1 Control file is required.
  - \* The minimum no. of online Redo log groups that are required to create a data base are 2.
  - \* The mandatory data files that are required to create a Data base are (1) system. dbf  
(2) sysaux. dbf
  - \* In version 9i, the minimum size of online Redo log file is 64K, whereas from log it is 4MB.  
If we create a data base using DBCA, oracle creates 2 Redo log groups.



\* parsing is nothing but checking for syntactical as well as semantical errors

\* there are 2 types of parsing. (1) soft parsing  
(2) Hard parsing

\* for the first time, every statement has to undergo hard parsing.

\* soft parsing means , no re-parsing

\* The parsed representation [parse tree] and execution plans will be stored in Library cache [L.C]

\* Library cache contains shared SQL Area and shared PL/SQL Area.

\* The shared SQL Area Contains parse tree and execution plan

\* The shared PLSQL area contains the compiled program units [procedures, functions, triggers ...etc]

\* During the parse tree, server process looks into Data Dictionary Cache to resolve the names of the objects and the access privileges.

\* Initially sorting operations takes place in PGA. If PGA

\* is not sufficient, sorting operations takes place in temp.dbf where oracle creates temporary segments

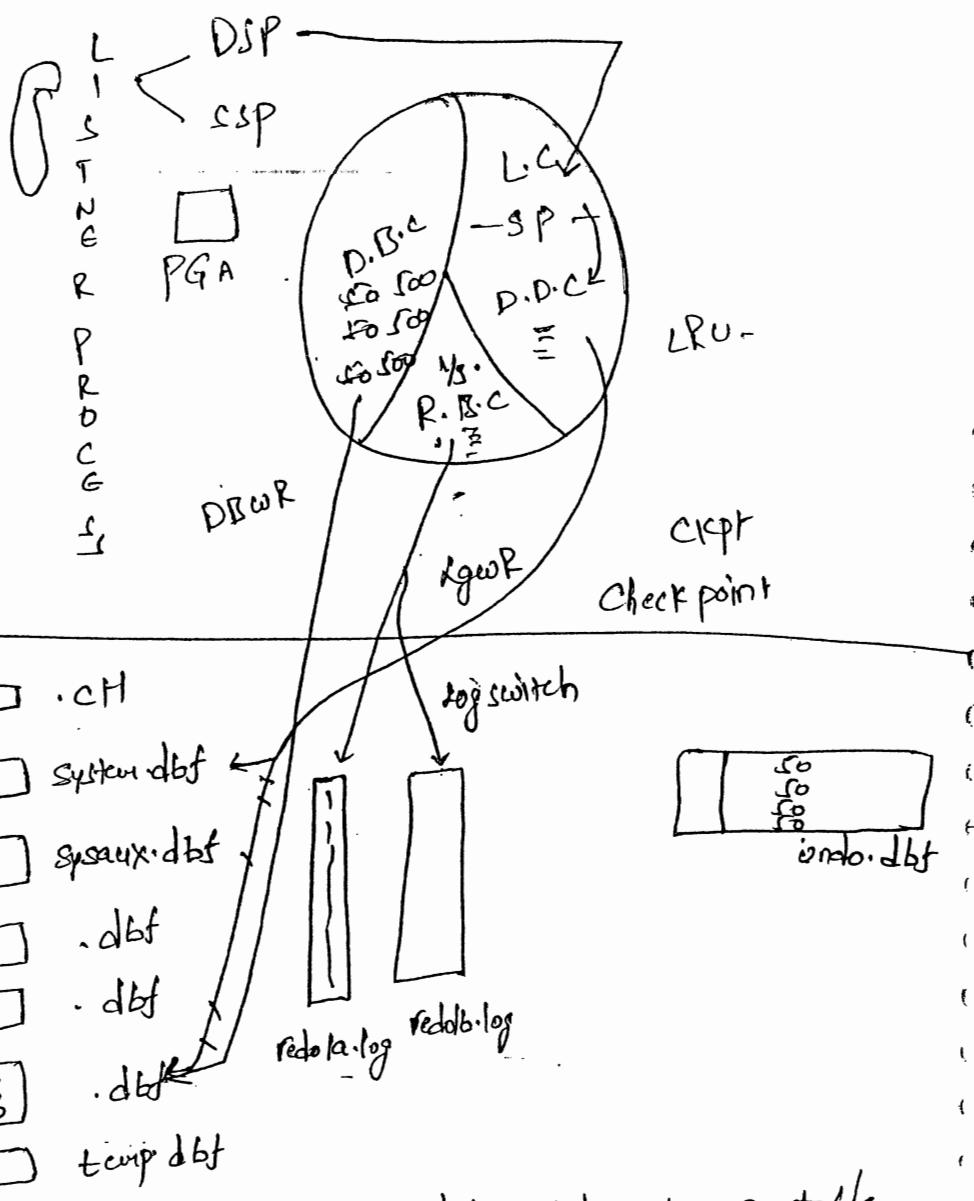
\* once the sorting operations have completed, temporary segments will be dropped automatically.

\* oracle uses LRU (Least recently used) algorithm to flush the data from database buffer cache as well as shared pool.

- \* Every Datafile contains header, which contains the size of the Data file, check point SCN, absolute file no., Relative file no.
- \* Absolute file no. uniquely identifies the Data file with in the data base.
- \* Relative file no. uniquely identifies the Data file with in the tablespace.
- \* for every 1 seconds, oracle generates one unique no. which is called SCN (System change number) number (prior to log).
- \* from .log generates for every second.
- \* In Dedicated Server Architecture, PGA Contains
  - (1) Sort Area
  - (2) Session information
  - (3) Cursor state
  - (4) Session variables.
- \* In shared server architecture, session information will be stored in large pool, if it is configured otherwise it will be stored in shared pool.
- \* The session information in the large pool, we call it as "User Global Area".
- \* Cursor state represents the progress in execution of every SQL statement.
- \* There are 3 types of segments
  - (1) permanent Segments (emp, dept, salary)
  - (2) Temporary segments (Created in temp.dbf)
  - (3) undo segments (Created in undb.dbf).

18/08/14

update emp  
set sal = 500;  
roll back;



- \* The moment once user gives an update statement, a table level lock as well as row-level lock will be held on that particular object.
- \* Row level lock will not allow other users from updating and table level lock will not allow other users from dropping.
- \* In-case of update, oracle keeps the past image in the undo segments.
- \* Undo provides read-consistency [in case of update and select, select users reads the data from undo segments].

- \* the purpose of undo is
  - (a) Read Consistency
  - (b) Transaction Roll-back.
  - (c) Transaction Recovery.
  - (d) Instance Recovery.

- whenever, instance crashes, oracle is not going to close the database headers. During, next startup "SMON" performs instance Recovery.
- Instance Recovery is nothing but Roll forward & Roll backward.
- SMON performs instance Recovery by scanning the Redo logs as well as the Undo segments.
- Roll backward means removing uncommitted data from the Data files.
- Roll forward means writing Committed data from online-Redo log files to the respective Data files.
- whenever, lock switch occurs check point event occurs which will be call by CKPT process.
- During check point, CKPT signals DBWR to write all the dirty buffers to the respective Data files.
- During check point, CKPT updates the check point change number to the Data files headers as well as the Control file.
- Log writer writes the data into online Redo log files under the following circumstances.

- (i) if Redo buffer cache is filled
- (ii) If the Data Reaches 1MBS.
- (iii) For every 5 seconds.
- (iv) If user commits
- (v) Before DBWR writes the data to the respective data files

PMON

→ PMON background process continuously monitors all the user processes. If any of the user process gets terminated without releasing the locks, 'PMON' releases the locks by roll back the transaction.

→ In case of shared server architecture, 'PMON' restarts the dead dispatches.

→ Initially sorting operations takes place in PGA. If PGA is not sufficient, sorting operations takes place in temporary segments. Once, the sorting operations are completed, SMON de-allocates the temporary segments.

→ Without undo, we can not execute any DML operation.

→ In case of update, oracle keeps the past image in the undo segments.

→ In case of insert, oracle keeps the rowid in the undo segments

→ ~~instead~~ of Rowid & Rownum are sub columns

→ Rowid is in 18 digit hexadecimal format

→ In case of Delete, oracle keeps the entire rowdata in the undo segments.

→ out of 5 DML statements, Delete requires more space in  
undo then update and then insert.

20/08/14

\* UNIX

RHEL AS 4 update 8

(4.8)

username : root

pwd : oracle DBA

→ Right click on mouse click on open terminal or goto ~  
applications click on applications click on system tools click on  
terminal (or) press Alt+F2 "gnome-terminal" click on enter.

(or) Ctrl+shift+N

→ To minimize the terminals use "Alt+F9"

→ To re-open the terminals we can use "Alt+tab"

→ To close the terminals we can use "Alt+F4" (or) Command  
line "exit"

→ "pwd" is the command for to know the present working  
directory.

→ "ls" is the command to display the files under the directory

→ "ls-a" is the command to display the hidden files in the  
directory (hidden files starts with ".gnome" and the  
rest of the files are normal files).

→ "ls -lrt" (or) "ls -ltr" (or) "ll" command to display the permission of all files. [directory - 777 , files - 666]

→ "umask"

→ "ls -al" (ls -a) is the command to display the permissions of hidden files

→ Creating Files :-

→ cat > f1

"Ctrl + C" is the

→ ls

particular (f1)

→ cat f1 - to show the content of the files.

→ cat >> f1 to add content of the existing file.

→ touch f2 - using touch command we can

create file with 0 bytes (without any data)

→ cp f1 f2 to copy the data from f1 to f2

cp: overwrite 'f2'? Y

→ cp f1 f2

## Creating Directories :-

→ `mkdir(d)` is the command to create directories

We can create individual directories at a time

Eg.: `mkdir d1 d2 d3 d4`

→ `mkdir -p d5/d6/d7/d8` to create nested directories.

→ `cd d1` to change the directory

→ `pwd`

- `/root/d1`

→ `cd ..` to go with the earlier directory

# `cd d5`

# `ls`

→ `d6`

→ `cd d6`

# `ls`

→ `d7`

→ `cd d5/d6/d7/d8/`

# `pwd`

`/root/d5/d6/d7/d8`

# `cd ..`

[root@sssl d7] # `cd .. / ..`

[root@sssl d7] # `cd -` is the command to go previous working directory

`/root/d5/d6/d7`

you have new mail in /var/spool/mail [root]

[root@sssl ~] #

[root@sssl ~] # cp f1 d1

[root@sssl ~] # cd d1

[root@sssl d1] # ls

f1

→ mv f1 d5 to move the files to directories

→ cd d5

→ ls

d6 f1

# mv d2 d5

# cp -r d2 d5

# cd d5

# ls

d2 d3 d6 f1

# ls -R d5 is the command to get all the sub directories

d5:

d1 d2 d3 d6 f1

d7

d5/d1:

f1

d5/d6/d7:

d8.

d5/d2:

d5/d3:

d5/d6 :

# rm f<sub>2</sub>  
rm: remove regular file 'f<sub>2</sub>'? y

# rm -rf f<sub>3</sub>

→ # rmdir d<sub>2</sub>

# rmdir d<sub>4</sub>

# rmdir d<sub>5</sub>

rmdir: 'd<sub>5</sub>

# rm -rf d<sub>5</sub>

→ cat /etc/issue (or) cat /etc/redhat-release.

Red Hat enterprise Linux AS release 4 (Enterprise Update 8)

→ uname -r - to know the kernel version

2.6.9-89.ELSmp (the interface between user and hardware)

→ # uname

Linux

→ # host name

sssl.oracle.com.

→ # ifconfig to know the interfaces

(or)

# host name -i

172.16.128.151

→ # uname -a

Linux sseli.oracle.com 2.6.9-89.EL5u1 SMP Mon Apr 20 10:34:11 EDT 2009 i686 i686 i386 GNU/Linux

# getconf LONG\_BIT (LONG-BIT)

32.

→ # fdisk -l

→ # free to know the Ram capacity  
(2064320)

→ # free -m  
(2015)

→ # free -g

(1)

(or)

→ - cat /proc/meminfo.

→ # file \*

→ # init 0 to shutdown the system.

# init 6 to open the system.

21/08/14

Editors  
To delete the file  
data from file  
vi  
pico  
nano

→ # vi /selfile(file name) using editors to open the file.

i - for insert mode

A - It will move to go to next character with insert mode

A - It will move to end of the line with insert mode

O - It's moving to next line with insert mode

O - It's moving to above line with insert mode.

S - It's deleting 1st character with insert mode.

S - It's deleting the entire line with insert mode.

To save the data

→ ESC + Shift : + wq!

(Save forcefully (l))

→ ESC + Shift ; q! (To not save the data).

→ To delete the lines without getting insert mode.

dd - To the 1<sup>st</sup> row

2dd - To the 2<sup>nd</sup> row

3dd - To the 3<sup>rd</sup> row.

↓  
→ H ← → L  
↓ J K

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

→ ESC + Shift # : / coord name - directly cursor goes to that word.

→ ESC + Shift % : / coord name

→ ESC + Shift : % S / u02 / u03 → in place of u02 - u03 will cursor

→ ESC + Shift : % S / u02 / u03 → in place of u02 - u03 will cursor

TO save the data → ESC + Shift : + wq!

→ To delete all the above line from Cursor position.

→ ESC + D + gg + press 'U' to get back the data

→ ESC + D + Shift G + To delete all the below lines from Cursor position.

→ ESC + Shift : & n To go the particular line number.

→ ESC + shift + G → to go to the end of the file.

→ ESC + gg - to go to the starting line of the file  
setting line no's

→ ESC + shift : + Set number (ESC + shift : + Sc n4)

To remove the nos

→ ESC + shift : + Set no number (ESC + shift : + Sc non4)

→ i) first set the line numbers

ii) esc + shift : 1,5 w >> /root/f1 → to move the particular lines  
from one file to another.

→ ESC + shift : 1,5 w /root/f2

\* \* (i) At os level every user should be attached to group.

(ii) At oracle level every user should be attached to Table space.

Creating users & Groups :-

→ # cat /etc/passwd

# cat /etc/group - To display all existing groups at os level.

→ # useradd username Raj

# cat /etc/passwd.

→ # groupadd groupname Ranga.

# id username

# useradd -g Ranga username

# id Ranga

# useradd -g Ranga Ranga

To change the Groups

→ # usermod -g raj hanuma

# id hanuma.

uid = 750 (hanuma)      gid = 729 (raj)      groups = 729 (raj).

MANOJ ENTERPRISES  
Plot 40, Gasathri Nagar  
Ameerpet, Hyderabad

MANOJ ENTERPRISES  
Plot 40, Gasathri Nagar  
Ameerpet, Hyderabad

MANOJ ENTERPRISES  
Plot 40, Gasathri Nagar  
Ameerpet, Hyderabad

22/08/14

- In version 9i, oracle has introduced "PGA\_Aggregate\_Target" parameter to allocate a consolidate PGA for all user processes.
- DBWR writes the data to the respective datafiles under the following circumstances.
- (i) when there are no free-buffers
  - (ii) when timeout occurs
  - (iii) when no. of dirty buffers reaches a threshold value
  - (iv) when check point occurs
  - (v) when we make the table space offline / read-only.
  - (vi) when you drop a table
  - (vii) when you truncate the table
  - (viii) when we keep the table space in begin backup mode.

### Data Dictionary:

- system.dbf contains Data dictionary
- Data dictionary contains meta data [Data about Database]
- Data dictionary is collection of read-only tables and views.
- Data dictionary contains all base tables [tab\$, col\$, priv\$, ind\$, AUD\$, etc.]
- \* → No user is authorized to create a DML statement on Data dictionary tables.
- \* → The only base table that is directly modifiable is AUD\$ which contains auditing information.

→ Once we create a database, Oracle internally calls SQL\*DBA and creates the base tables.

→ Data dictionary contains information about the logical structures, physical structures, users, user privileges, integrity constraints, information about objects, indexes etc..

→ Data dictionary views are further sub-divided into two categories  
- (a) static views      (b) dynamic performance views.

→ All views which are starting with dba-, user- & all- are called static views  
ex:- dba-tables, dba-segments, dba-extents, user-tables, all-tables etc..

→ All views which are starting with v\$ are called dynamic performance views.

→ Dynamic performance views gives accurate information because they get the data from the memory (SGA) as well as the control file.  
ex:- v\$ instance, v\$ database, v\$ control file etc..

### Initialization parameters:-

for Oracle\_HOME/db\$

→ Initialization parameters are further divided into two types  
① Implicit parameters    ② Explicit parameters.

→ Implicit parameters by default they assume the value as per Oracle.

→ Explicit parameters are those parameters which we specify explicitly  
ex:- SGA-Max-Size, dba-name, open-cursors, sessions etc..

→ explicit parameters are further divided into two categories  
① static parameters      ② dynamic parameters.

→ In case of static parameters, we can not change those values while instance is up & running. In order to change the values, we need to bounce the instance. (down & up)

ex:- SGA-HAR-SIZE , sessions etc..

→ Dynamic parameters are those parameters whose values we can change while instance is up & running

ex:- SGA-Target , open-cursors , shared-pool-size , db-cache-size etc..

\* \* → once you start the instance, the first file that oracle creates is "alert<SID>.log"

→ the default location of alert log file is "bdump directory" or "Diagnostic-Dest" location.

→ prior to 11g, diagnostic data is spread across multiple file systems & multiple directories.

background\_dump\_dest = /u01/app/oracle/bms/bdump.

user\_dump\_dest = /u01/app/oracle/bms/udump.

Core\_dump\_dest = /u01/app/oracle/bms/cdump.

→ from 11g, All the above 3 parameters have become absolute and oracle has introduced a single parameter that is "Diagnostic-Dest".

→ During day-to-day operations, as a DBA we need to monitor the alert log file to understand the diagnostic related information and error messages.

→ Alert log file contains (i) The time stamp when the db was started.  
(ii) The time stamp when the db was brought down  
(iii) The thread being used by the instance.

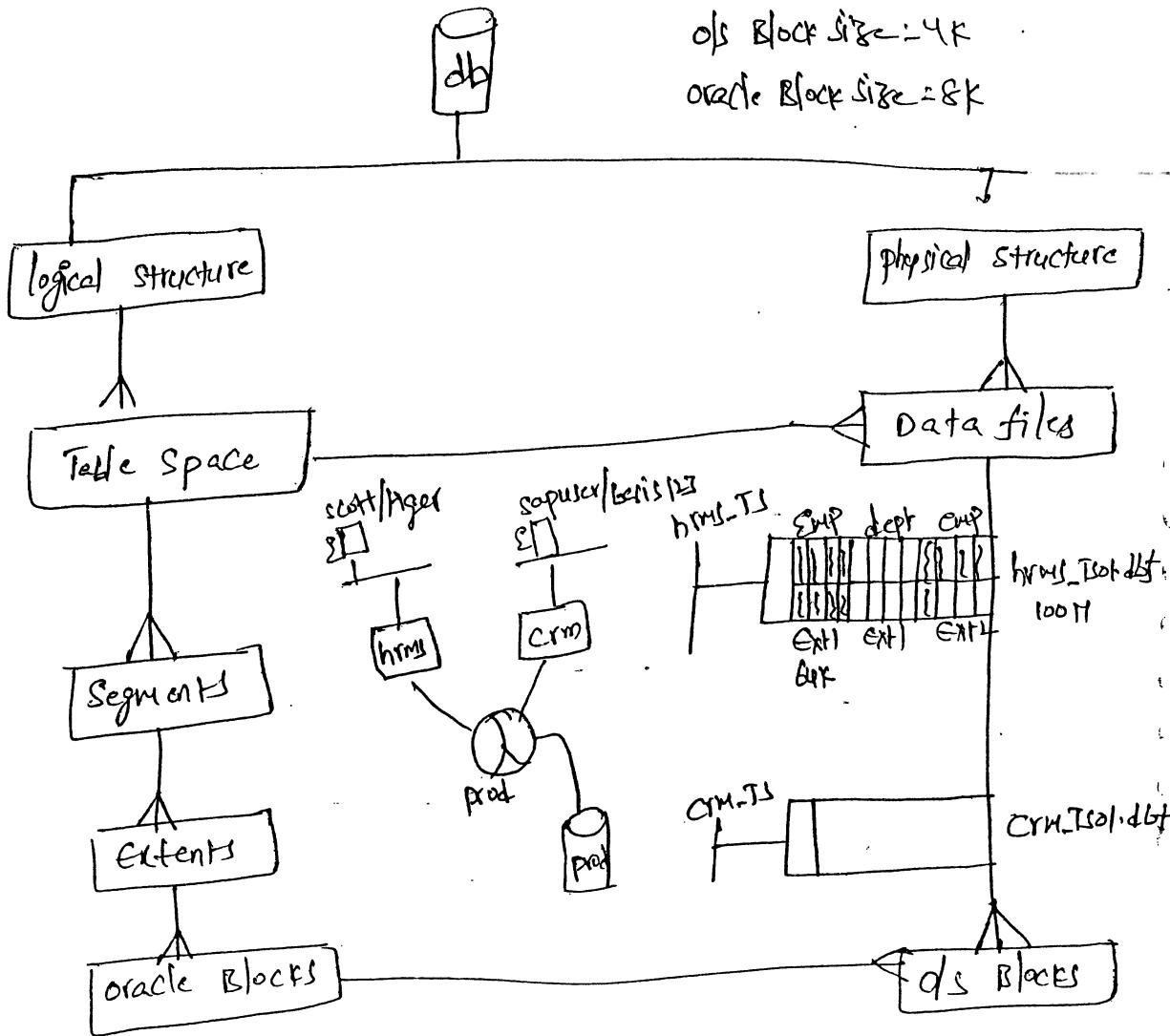
- (iv) the log sequence no.
- (v) Arch log information.
- (vi) FRA usage
- (vii) non-default initialization parameters.
- (viii) startup of all background processes
- (ix) warning messages.
- (x) ora related error no's.

ex: ORA - 60 , ORA - 600 etc..

- (xi) Administrative Commands being issued by DBA.

### Structure of the Database:-

- Database has got logical structure as well as physical structure.
- the physical structure of the database represents C, R, D files.
- logical structure of the database represents
  - ① Table Spaces
  - ② segments
  - ③ Extends
  - ④ Oracle blocks



DBA) Create tablespace hrms\_TS datafile '/vol/app/oracle/hrms/hrms\_TS01.dbf' size 100M;

SQL) Create tablespace CRM\_TS  
datafile '/vol/app/oracle/crm/crm\_TS01.dbf' size 200M;

DBA) Create user scott identified by tiger  
default tablespace hrms\_TS;

DBA) Create user Sepuler identified by basis123  
default tablespace CRM\_TS;

DBAS grant connect, resource to scott;

DBAS grant connect, resource to Sepuler;

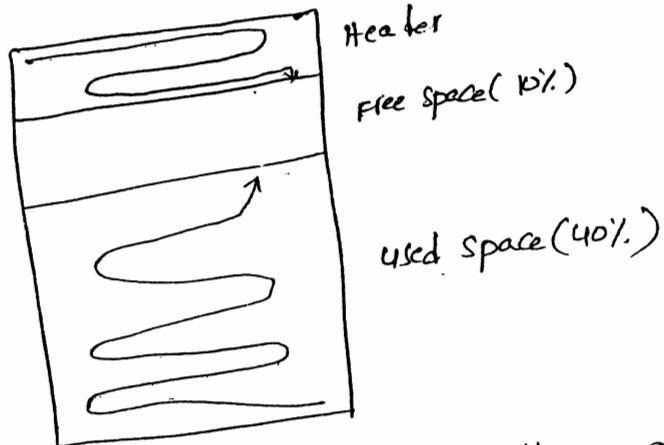
```

scott> create table EMP (C);
scott> insert into EMP (100);
scott> create table dept (C);
scott> Insert into Emp (200);

```

- In version 9i, the default block size is 8K
- In version 9i, oracle has introduced different block size concepts by introducing the 'init.ora' parameter 'db-nk-cache-size', where n=2,4,8,16,32

### Data base block structure:-

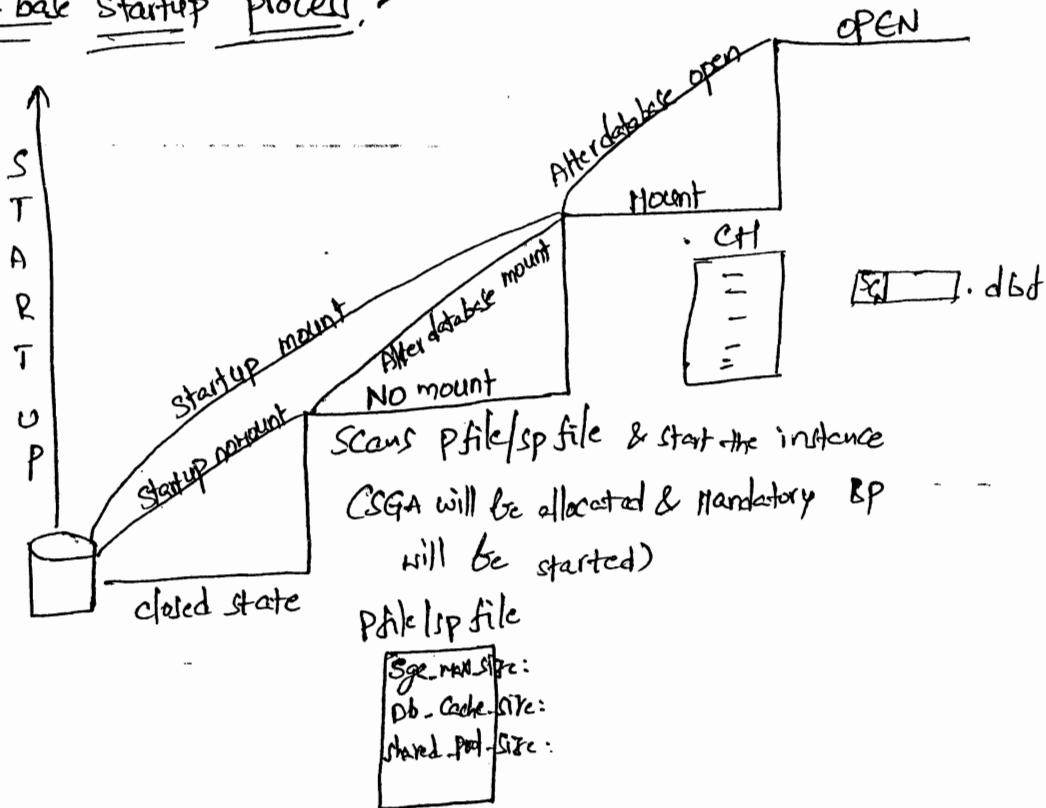


- Header of the block contains ① Data block Address ② Table directory ③ Row directory ④ Transaction slots.
- Transaction slots will be used by the users, when ever they perform some transactions.
- 10% of Space in a block is reserved as free for future updations.

### Block space utilization parameters:-

INITTRANS	(1)
MAXTRANS	(255)
PCTUSED	(40%)
PCTFREE	(10%)

## Data base Startup Process:-



- In order to go to nomount state, oracle scans the content of either pfile/spfile
- In the nomount state, instance will be started.
- In the nomount state, it starts the instance by allocating the memory for 'SGA' and by starting all the mandatory background processes.
- In the nomount state, oracle creates alert<SID>.log in the bldump directory or Trace directory.
- Mounting is nothing but associating the previously started instance with the database
- In the mount state, oracle scans the content of Control file for the physical locations of Data files & Redo log files.
- In the mount state, oracle ~~not~~ will not check for the existence of Data files & Redo log files.
- From mount to open, oracle performs sanitary check

- Sanitary check is nothing but, checking for the synchronization of SCN number
- In the open state, oracle opens the Datafiles & Redo log files
- In order to create a brand new database (or) In order to re-name the Database we have to be in 'nomount' state
- In the mount state, we carry out some administrative activities like Configuring the database in Archive log mode (or) Flashback mode etc..
- In order to perform database crash recovery, we have to be in 'Mount' state.

Note:- If you lose your pfile/spfile, go to 'alert <SID>.log' file  
Copy all the parameters and create one 'pfile' and start the Instance.

### Database shutdown modes:-

- (1) shut down normal
- (2) shut down transactional
- (3) shut down immediate
- (4) shut down ~~exit~~ abort.

#### ① shut down normal:-

this is a graceful (clean) shutdown. Next startup doesn't require instance Recovery

- No new connections will be allowed
- Oracle waits until all users disconnect their sessions explicitly.
- Dismounts the database, releases the memory to the OS. and closes the datafiles.

## ② shutdown Transactional:-

- this is also a graceful shutdown. Next startup doesn't require instance recovery. No new connections will be allowed.
- Oracle waits until all ongoing transactions are completed either by commit/ rollback.
- Dismounts the database, releases the memory & closes the Datafiles.

## ③ shutdown immediate:-

- this is also a graceful shutdown
- Oracle disconnects all the sessions by roll back any ongoing transactions
- Dismounts the database, releases the memory & close the datafiles.

## ④ shutdown Abort:-

- this is not a graceful shutdown. Next startup requires instance Recovery
- immediately releases the memory to the O/S. without closing the datafile headers.

Note:- Generally in the event of loss of full database (or) loss of system.dbf (or) loss of Control files (or) loss of Redolog files, we execute "shut abort".

\* startup force = shut abort + startup

- sometimes, shutdown command takes more time to bring down the database, in this scenario and if there are connections, we execute "startup force".

28/08/14

## steps to install oracle on linux :-

→ In windows , in order to install the software we execute ~~the~~ setup.exe

→ In unix environments , we in work 'runinstaller' . to install the software

① → As per OFA (Oracle Flexible Architecture) , oracle recommends us to create the following rules.

(i) oinstall

(iv) dba

② → oinstall is the primary group and dba is the secondary group.

# groupadd oinstall

# groupadd dba

→ Create one separate user account for oracle

# useradd -m -goinstall -Gdba oracle ↳

# passwd oracle ↳

\$ cat /etc/group

oracle ↳

③ → As per OFA, oracle recommends us to create ORACLE-BASE & ORACLE-HOME .

Note:- ORACLE-BASE is optional . Recommended to create ORACLE-BASE if u are installing multiple versions of oracle in the same server.

# mkdir -p /u01/app/oracle/product

q1 # mkdir -p /u01/app/oracle/product/9.2.0/db-home

log # mkdir -p /u01/app/oracle/product/10.2.0/db-home

11g # mkdir -p /u01/app/oracle/product/11.2.0/db-home

# chown -R oracle:oinstall /u01/app/oracle/product/11.2.0/db-home

o. g others # chmod -R 775 /u01/app/oracle/product/11.2.0/db-home

⑥ Configure kernel parameters as per the installation Doc:-

# vi /etc/sysctl.conf.

Note: The values of the parameters varies from o/c to o/s.

kernel.shmall = <value>

kernel.shmmax = <value in bytes>

kernel.shmmni = <value> <4096>

kernel.sem = <value>

fs.file-max = <value>

net.ipv4.ip\_local\_port\_range = <value>

net.core.rmem\_default = <value>

net.core.wmem\_default = <value>

net.core.rmem\_max = <value>

net.core.wmem\_max = <value>

→ 'shmmax' represents the size in bytes of a single shared memory segment. oracle recommends to allocate as much as possible to accommodate the entire 'SGA' in a single shared memory segment.

Note: Inadequate value for this parameter throws an error message unable to attach shared memory segment during database creation.

→ 'shmmni' represents the total no.of shared memory segments system wide.

\* we specify 'shmall' in pages. The sum of sizes of all SGA's should be less than 'shmall'.

→ file\_max represents the maximum no.of data files supported by oracle

→ `net.ipv4.ip\_local\_port\_range` represents the range of ports that can be used by oracle services.

(5) Set shell limits for oracle user account:-

# vi /etc/security/limits.conf

oracle soft nproc	<value>	nproc-(no.of processes - lower limit)	soft
oracle hard nproc	<value>	(upper limit)-hard	↑
oracle soft nofile	<value>	(no.of files - nofile)	↑
oracle hard nofile	<value>		↑

(6) Configure .bash\_profile:- (optional)

# su - oracle

\$ vi .bash\_profile (in sun solaris .profile).

\$ export ORACLE\_BASE = /u01/app/oracle/product

\$ export ORACLE\_HOME = \$ORACLE\_BASE/11.2.0/db-home

\$ export ORACLE\_SID = hrms

\$ export PATH = \$ORACLE\_HOME/bin:\$PATH/bin

Environment  
Variables

optional

\$ . .bash\_profile

Note: (i) Configuring .bash-profile is optional

(ii) in cook runinstaller and install the software

→ Execute the scripts as root user

/home/oracle/orainventory/orainstroot.sh

/u01/app/oracle/product/11.2.0/db-home/root.sh.

Note: In the industry we execute those scripts as 'Sudo'

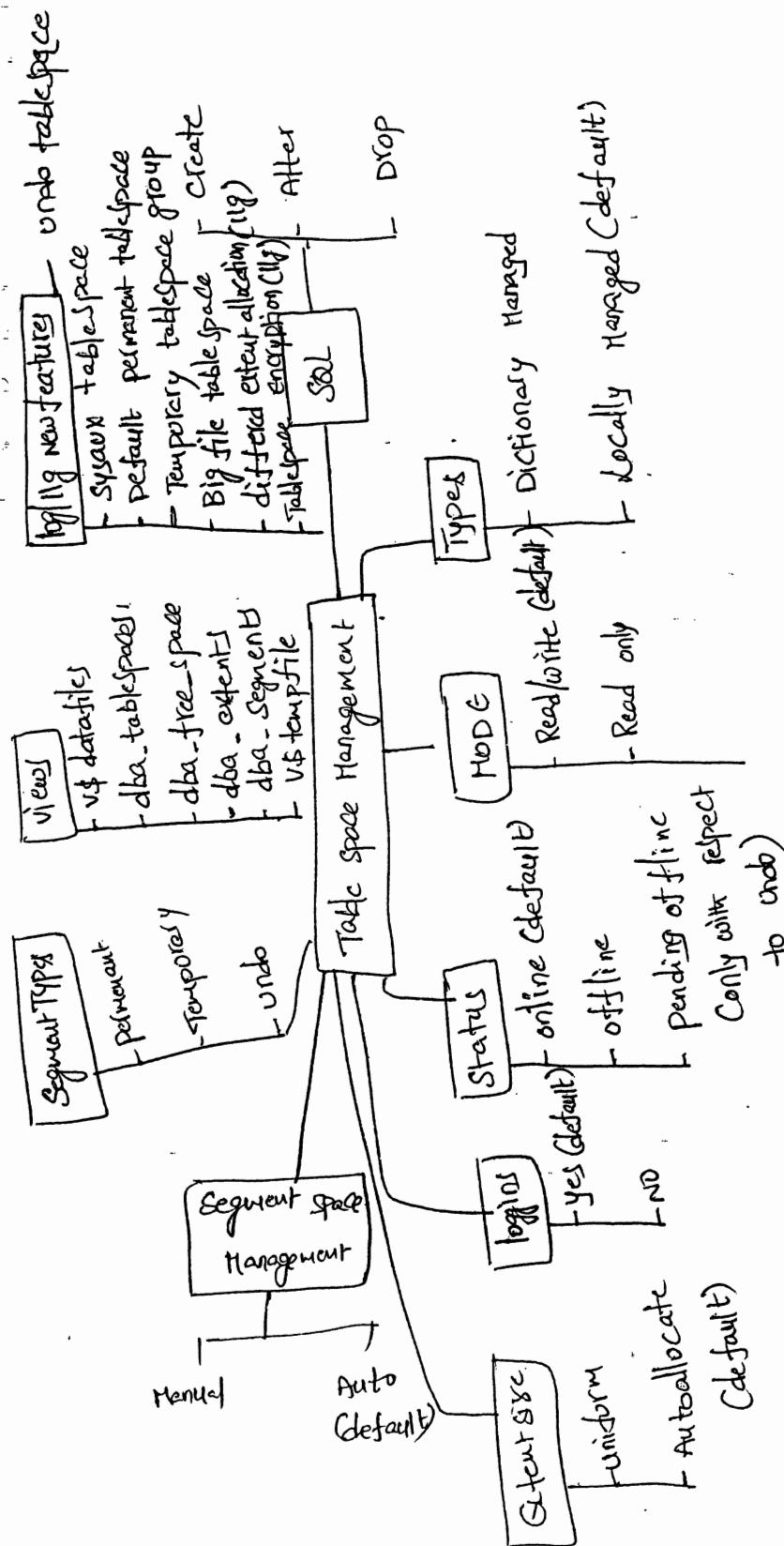
\$ sudo /home/oracle/orainventory/orainstroot.sh ↵  
password : oracle ↵

MANOJ ENTERPRISES  
Plot:40, Gajathri Nagar  
Ameerpet, Hyderabad

- The first script is going to change the inventory group to `oinstall` and it is going to proper permissions on "inventory" file
- The second script is going to create 'oratab' under /etc,
- \*→ When we create a database using 'DBCA', DBCA is going to place an entry in the 'oratab' file w.r.t. to database.
- \*→ If we create a database manually, we need to place an entry in the oratab.
- \*→ Inventory contains 'XML' files which represents the components that are installed with that particular installation.
- During patching activity, inventory gets modified.

### \* Table Space Management:-

- Table space is one of the logical structure of a database.
- Table space contains one (or) more datafiles.
- \*→ In 10g, oracle has introduced Bigfile Tablespace which contains only one datafile
- Bigfile tablespace datafile can grow upto terabytes.
- \*→ If the block size is 8k, a datafile can grow upto 32GB.
- If the block size is 16k, a datafile can grow upto 64gb.
- In the industry, recommended to have small size datafiles for better performance and also takes less time to restore & recover in the event of loss.



→ Recommended to create Bigfile tablespace if 3rd party volume manager

(or) ASL is in place

Advantages of having separate table spaces:-

→ we can segregate the metadata with the business data

→ we can make individual tablespace offline (or) read only

if required application users will face the

→ In the event of loss, only that application users will face the down time.

→ we can backup table space by tablespace.

→ ① the minimum & mandatory tablespaces that are required to create a database are 'System' & 'SYSAUX'

→ SYSAUX table space was introduced in 10g version of oracle.

→ SYSAUX tablespace is considered as an auxiliary tablespace for the system. Some of the components metadata have been moved from 'System' to 'SYSAUX'

→ To know the occupancy of 'SYSAUX', issue the following query.

SQL> select \* from V\$ SYSAUX\_OCCUPANTS;

→ we can't rename system & SYSAUX table spaces.

→ we can't offline system tablespace but we can offline SYSAUX table space

→ In oracle, every user has to be assigned to a tablespace.

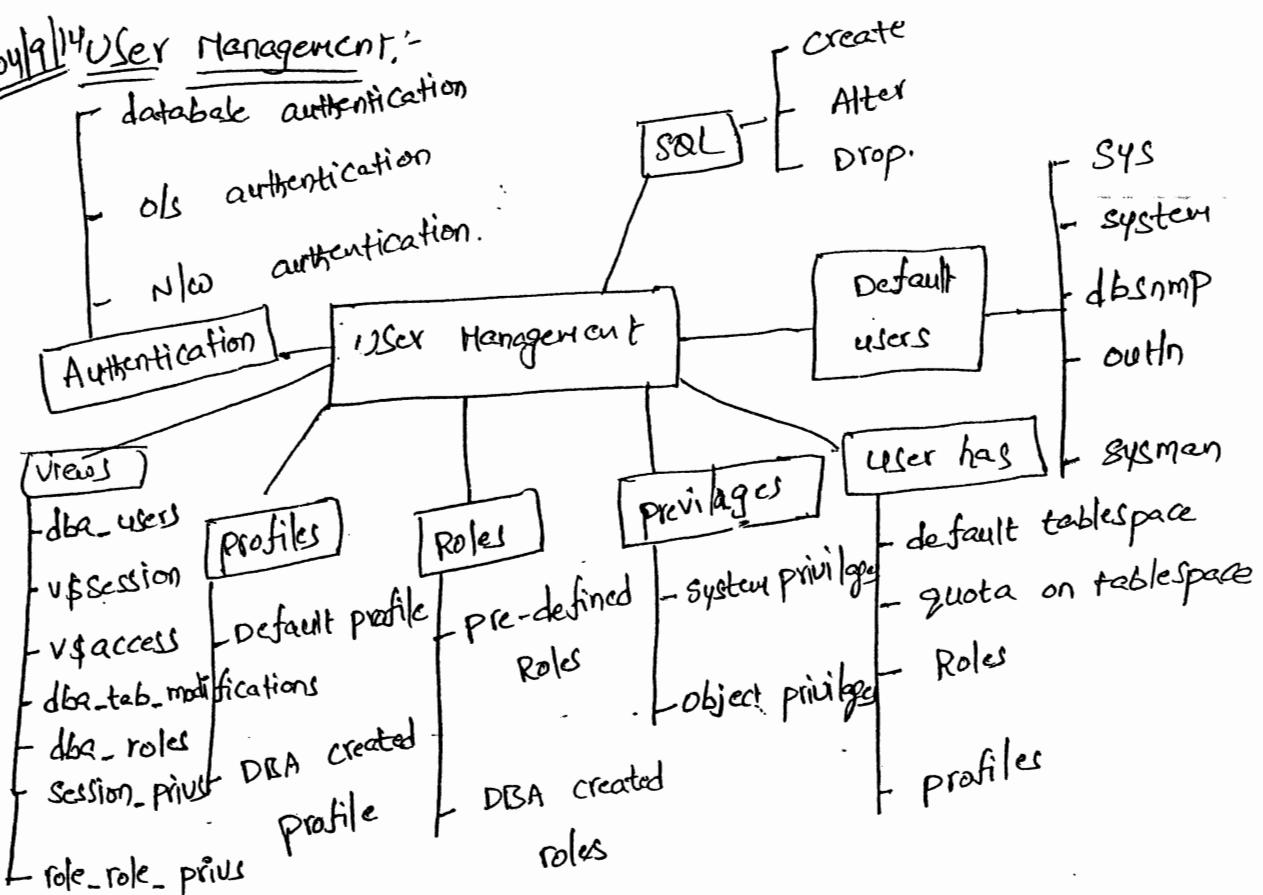
→ prior to 10g, if we create a user without assigning a tablespace, user will be assigned to System tablespace.

- But from log, user will be assigned to default permanent table space.
- A database can have one default temporary tablespace, one default undo tablespace and one default permanent tablespace
- Tables spaces are of two types ① Dictionary Managed  
② Locally Managed
- In case of dictionary managed tablespace, information about the free extents will be maintained in Data dictionary
- In case of locally managed tablespace, information will be maintained within the tablespace itself.
- Once u create a tablespace, it will be created in read/write mode and by default logging is enabled.
- \*\*→ Generally, no logging is enabled in case of bulk load operations & in case of index tablespace creations. Once the bulk load operations is completed, immediately take the backup of database.
- Logging means, every change will be recorded into online - Redo log files.
- If u create a tablespace with no logging, only minimal information will be recorded into online Redo log files.
- Extent size can be uniform (or) Auto allocate. By default it is auto allocate.
- Segment space management is of two types (1) Manual  
(2) Auto

- In Case of Manual, oracle uses free lists and free lists groups which is outdated concept.
- In Case of Auto, Some of the blocks will be Considered as BMB's (Bit map blocks) and each Bit indicates the status of the block
- In 11g, oracle has introduced default extent allocation which means oracle will not allocate an extent once we create a segment. oracle allocates an extent once we insert the data.
- \*\*→ Differed extent allocation Concept is more useful in case of new application development environment. In these environments, we can create n no.of tables even if there is no space in the data files.
- In 10g, oracle has introduced temporary tablespace group which contains one (or) more temporary tablespaces.
- A Segmient Spans across data files belongs to a tablespace.  
(If will not span across the tablespaces)
- A temporary segment created in temporary tablespace group Spans across the temporary tablespaces.

04/09/14 User Management:-

- database authentication
- o/s authentication
- N/W authentication.



→ user and schema both are interchangeable words

→ schema is nothing but collection of database objects like

tables, views, materialized views, synonyms, packages, procedures, functions, triggers, etc.,

→ If u drop a user, All the objects are under that particular schema will be dropped automatically.

→ the default users that are created once we create a database are

- ① sys - change-on-install

② system - Manager

③ dbsnmp - dbsnmp (required for network functionality)

④ outln - outln (contains stored outlines if nothing but execution plans)

⑤ sysman - sysman (required for enterprise Manager)

→ 'sys' is the super user which contains administrative privileges.

→ once we create a database, recommended to change the passwords of all default users.

→ '/ as sysdba' means connecting to 'sys' through os authentication.

→ privilege is the right to execute a specific command

→ there are two types of privileges ① system level privileges

ex:- create table, create any table, create index, create user, create tablespace, drop tablespace etc.,

② object level privileges

ex:- update on emp, delete on emp, select on emp, etc.

→ 'Role' is nothing but collection of privileges

→ there are two types of Roles ① pre-defined Roles

ex:- sys dba, dba, exp\_full\_database, connect, resource, select\_catalog\_role etc.,

② DBA created Roles

ex:- clerk\_role, manager\_role etc.,

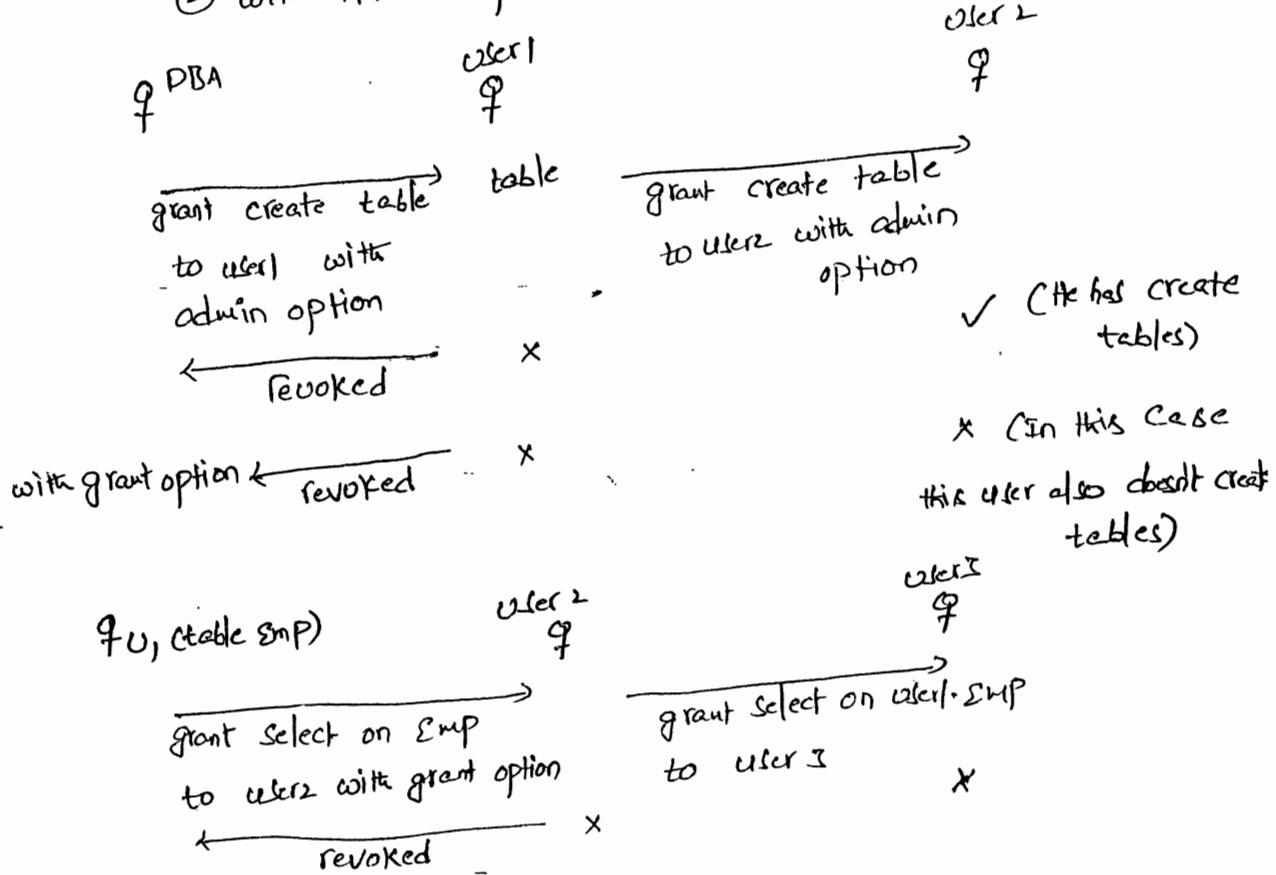
→ sysdba, sysoper, dba contains administrative privileges and we should not grant these roles to normal database users.

Sysoper -	start up shut down Alter database mount Alter database open Alter database Archivelog Alter database backup control file, recover database
-----------	---

Sysdba	Sysoper privileges with admin options Create database Restricted session Recover database until	Alter database begin backup Alter database end backup
--------	--	--

→ we can grant a privilege in two ways

- ① with grant option.
- ② with Admin option.



→ profile is a collection of resource limits

→ There are two types of profiles

① default profiles

② PBA created profile

→ Default profile will be created by `sqlplus /nolog`

→ & profile contains some resource limits like password lifetime, password re-use time, idle time, failed login attempts etc.,

\*\* → If u want to change the password of 'sys', we have to use 'orapwd' utility.

05/09/14

### Storage parameters:-

The following are the storage parameters that we specify either at tablespace creation level (or) object creation level.

(1) Initial

(2) Min extents

(3) Max extents

(4) Pctincrease

→ If u specify the storage parameters at tablespace level as well as object level, oracle considers object level storage parameters. (Priority goes to object level).

DBA  
SQL> create tablespace hrms\_TS

datafile '/u01/app/oracle/hrms/hrms\_TS01.dbf' size 100M

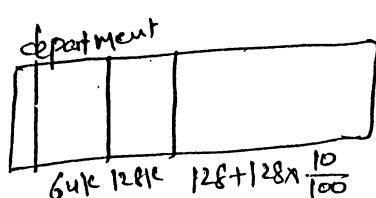
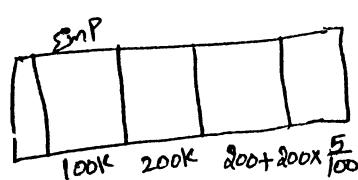
default storage (initial 100K  
next 200K  
minextents 2  
maxextents 10  
pctincrease 5);

DBA  
SQL> create user test identified by test;

default tablespace hrms\_TS;

Test  
SQL> create table employee ( );

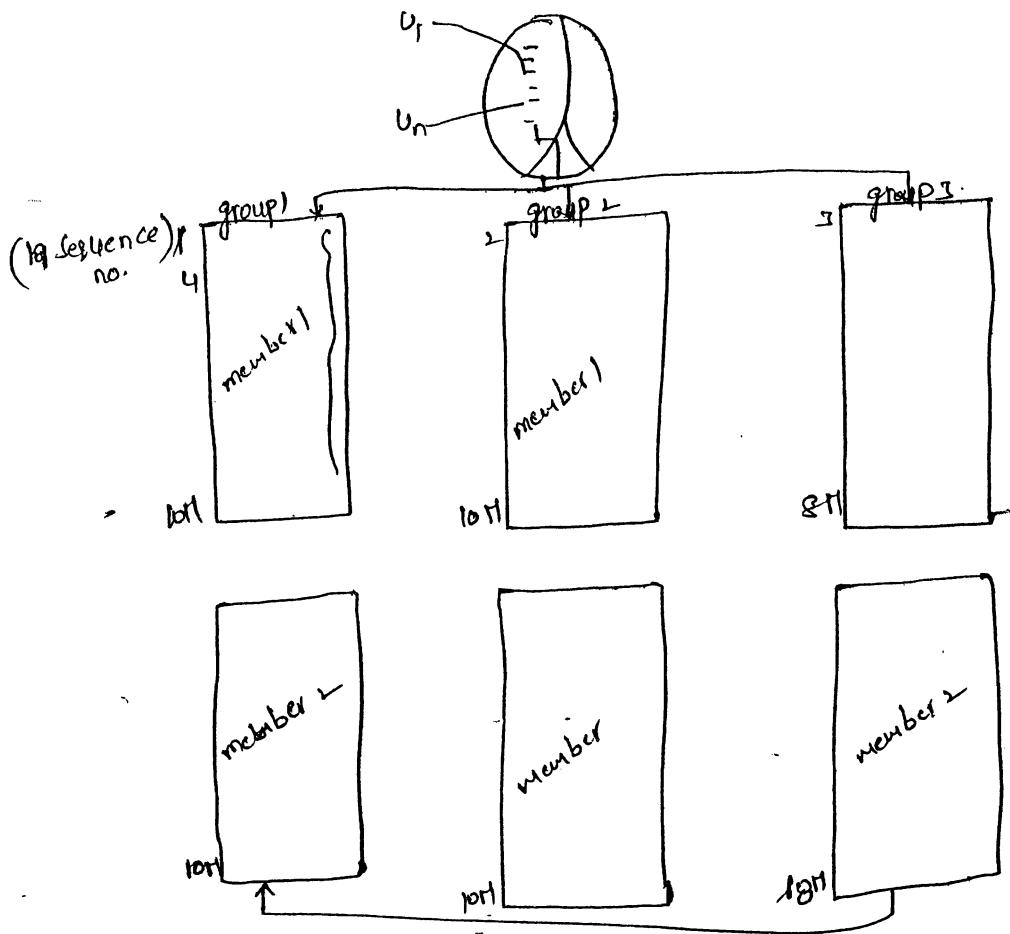
SQL> create table department ( )  
storage (initial 64K  
next 128K  
minextents 2  
maxextents 10  
pctincrease 10);



- 'initial' represents the size in bytes of first extent
- 'next' represents the size in bytes of second extent
- 'minextents' represents the minimum no. of extents
- 'maxextents' represents the max. no. of extents
- 'pctincrease' represents the % by which the 3rd & subsequent extents has to grow over the preceding extents.

### \*on-line Redolog file Management:-

- the minimum no. of online Redolog groups that are required to create a database are two
- prior to log, the minimum size of online Redolog file is 64K.
- from log, it is 4MB.
- every change initially has to be recorded into the online Redolog files.
- the transactional data in the online Redolog files, we call it as 'change vectors'.
- if u create a tablespace in nologging mode, information will not be recorded into ORLF's (only minimal information will be recorded)
- ORLF's are required for instance recovery as well as database recovery.
- Since these files are very important for instance recovery as well as database recovery, recommended to mirror them to avoid single point of failures.
- See that mirrored copies resides in a separate file system. to avoid single point of failures.



- logwriter writes into onlineRedolog files in cyclic manner
- identical Copies of ORLF's within a group are called members
- members within a group will have the same lock sequence no.
- all members within a group should be of same size
- members across the group can be of varying sizes, but it is not recommended
- ~~lock switch occurs~~, logwriter writes into ORLF's under the following circumstances
  - (1) If  $\frac{1}{3}$  of redo buffer cache is filled
  - (2) If the data reaches 1MB
  - (3) If user commits
- lock switch occurs under the following circumstances
  - (1) If an ORLF's is filled
  - (2) If the DBA switches Manually

SQL> Alter system switch logfile;

- 'ORLF' exists in any one of the following states
  - (1) Current
  - (2) Active
  - (3) In Active
  - (4) Unused
  - (5) stale
- If logwriter is pointing to an ORLF, then it exists in current state.
- An 'ORLF', which is required for recovery exists in Active state.
- One can't drop an Active Redolog file.
- An 'ORLF', which is not required for recovery exists in InActive state.
- 'ORLF' with partial data exists in stale state.
- \*→ Sometimes we see a message 'checkpoint not complete' in the Alert log. To overcome this problem add one more online redolog group (Also check for the availability of space in archive destination)  
(If there is a requirement to enable more DBWR processes, enable them).

### Ideally Archive log file Management:-

- Database may run either in noarchive log mode or archive log mode.
- It is one of the important responsibility of a DBA to configure the database in Archive log mode.
- In the industry every production database has to be configured in archive log mode.
- If the database runs in no archivelog mode, there will be no recovery.
- \*→ process of applying the archive logs to the previous backup is called recovery.
- During recovery, Oracle applies the archive logs in the same sequence.
- There is no concept of skipping & applying the archive logs.

→ Now we try, in order to configure the database in archive log mode we need to set the following parameters.

- ① Log\_Archive\_Start = TRUE
- ② Log\_Archive\_Dest = <PATH>
- ③ Log\_Archive\_Format = %t%u.%s.%r.arch

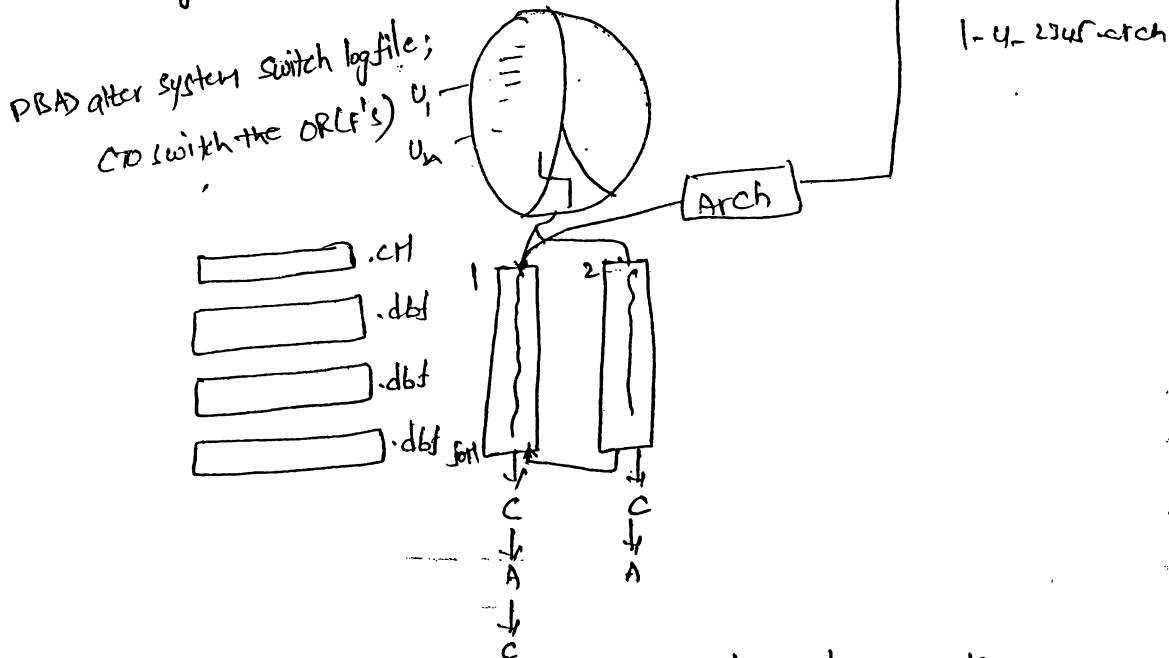
T - Thread no.      S - Log Sequence no.      R - Reset log ID  
 (Instance no.)

→ Reset log id is also called as incarnation number.

(Once we create a database oracle will give a database number that we call it as incarnation no.)

→ From log, Log\_Archive\_Start have become debigated. (Now more)

Log\_Archive\_Dest = /vol/app/oracle/crm/arch → 1-1-2345.arch  
 Log\_Archive\_Format = %t%u.%s.%r.arch



→ once, we configure the database in archive log mode, Archiver

background process gets enabled and it's picks the content of active online Redolog file and creates an archive log file in the archive destination.

→ Archive log files are Content Copies of ORLF's but not image copies.

→ In RAC system, each instance has its own archive logs

- \* If the archive destination is 100% filled, database goes to hung state.
- \* and it will not allow the business users to establish a connection.
- As a DBA, continuously we need to monitor for the availability of space in archive destination
- Along with the database backup, we also need to take the backup of archive log's and delete the archive log's from the archive destination.

### Flash-recovery Area / Fast Recovery Area (FRA) :-

- ~~In order~~ Introduced in 10g version of oracle and renamed into fast recovery area in 11g version of oracle.
- In order to configure FRA, we need to set the following parameters
  - ① DB\_recovery\_file\_dest\_size
  - ② DB\_recovery\_file\_dest
- once, we configure FRA, by default the following files will be created in FRA
  - ① Multiplexed control files (mirrored control files)
  - ② Multiplexed online redo log files (mirrored ORLF's)
  - ③ Archive logs
  - ④ RMAN backup sets
  - ⑤ flashback logs
- If we configure FRA, we need to monitor for the availability of space in FRA.
- If FRA space is 100% utilized, database goes to hung state.
- If database is in hung state, it will not allow business users to establish a connection and it gives archive error.

SQL> select \* from v\$flashrecovery\_area\_usage;

MANOJ ENTERPRISES  
Plot: 40, Gasathri Nagar  
Ameerpet, Hyderabad

## Control file Management:-

→ Control file is a binary file and oracle scans the content of the control file in the mount state.

→ For the functionality of the database, at least we require one control file.

→ Control file is linked to only one database

→ Loss of Control file needs recovery.

→ In order to avoid single point of failure, recommended to multiplex (mirror) Control files.

→ See that mirrored copies resides in separate file system.

→ Control file contains ① the name of the database

② Time stamp when database was created

③ character set of the database

④ TableSpace names

⑤ Names & locations of Data files & Redlog files

⑥ Log sequence no.

⑦ Check point information.

⑧ Archive log information.

⑨ Thread being used by the instance.

→ Control file contains ① Non-Reusable section

② Reusable section

→ Reusable section of the Control file contains RMAN backup information.

→ Control file is the binary file and we can generate a text based control file by issuing the following command

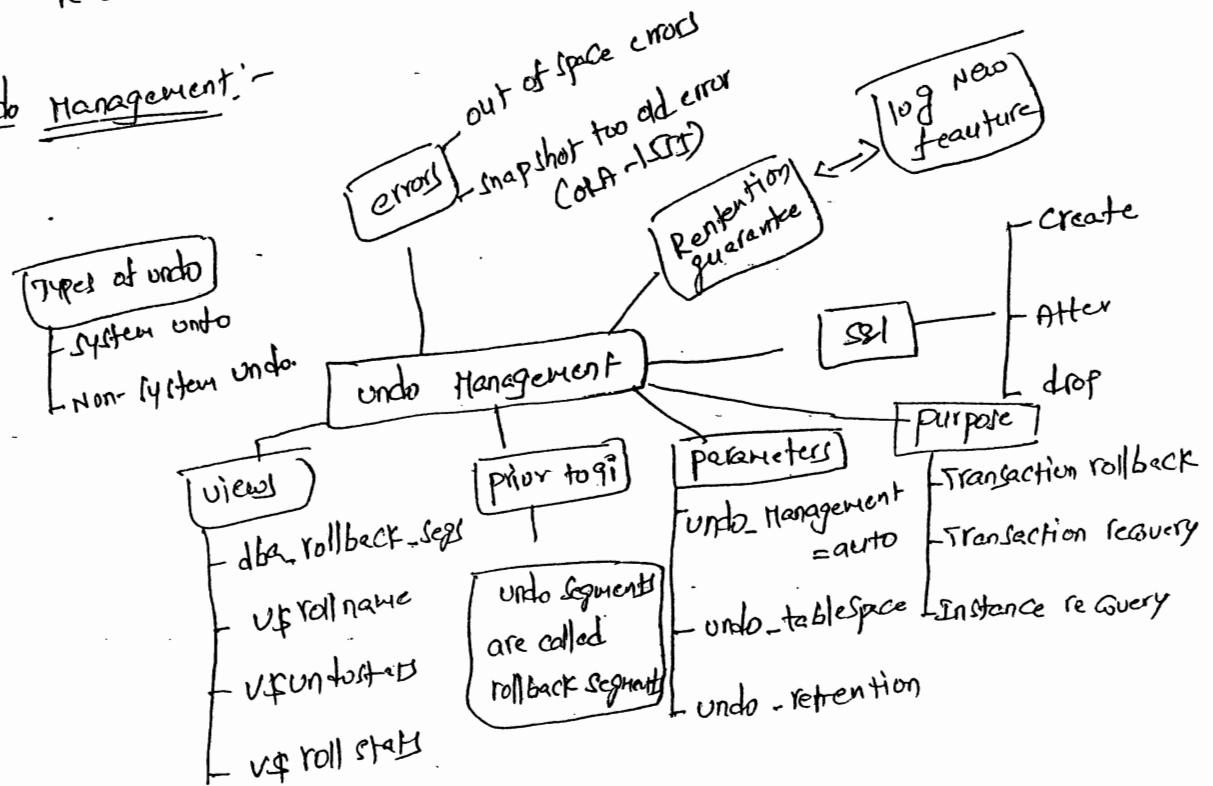
SQL> alter database back Controlfile to trace;

→ The above command generates a trace file in the udump directory

(or) diagnostic\_dest location.

- the above command creates a trace file which contains SQL command (create control file) to recreate a binary control file.
- oracle maintains the database name in 2 locations
  - ① P file or) Sp file
  - ② Control file.
- If you want to change the name of the database, we need to re-create the control file after changing the 'db' name

### Undo Management:-



→ prior to q1,

use to create the required no of roll back segments and we use to monitor the growth of the roll back segment

→ with respect to roll back segments, lot of DBA intervention is required

→ From q1, roll back segments are called undo segments

→ whether it is roll back segment (or) undo segment, for every DML

we require them.

as a DBA we

→ there are two types of undo segments  
 ① System undo segment  
 ② Non-system undo segment

\* → once we create a database, by default oracle creates one undo segment in system.dbf which is called system undo segment

→ In case of insert, oracle keeps the rowid in the undo segments.

→ In case of update, for it keeps the past image and in case of delete, it keeps the entire raw data.

prior to q1

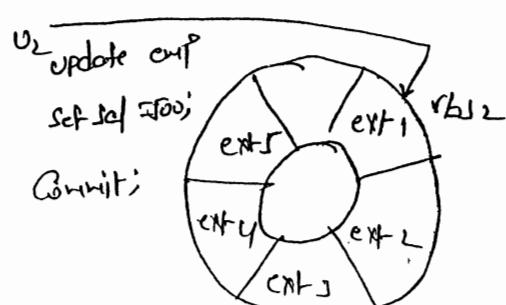
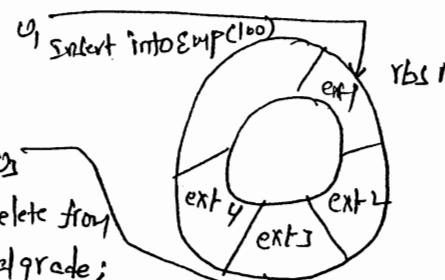
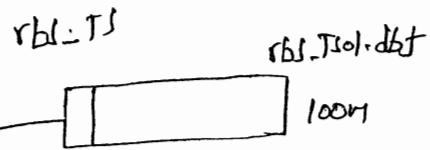
> create tablespace rbs-TS  
 datafile '100\applorecle\hrms\rbs-TS01.dbf' size 100M  
 extent management dictionary;

> create roll back segment rbs1  
 tablespace rbs-TS

storage (initial 100K  
 next 100K  
 minextents 2  
 maxextents 10  
 optimal 200K);

> create roll back segment rbs2  
 tablespace rbs-TS

storage (initial 400K  
 next 400K  
 minextents 4  
 maxextents 15  
 optimal 800K);



→ the minimum no. of segments

→ pct increase is not applicable to rollback segment

\*→ op\_time represents to what extent a rollback segment has to shrink once the transactions are completed

\*→ oracle is not going to shrink the rollback segment immediately once the transaction is completed. It shrinks if there are contiguous inactive extents (a) if the size exceeds the optimal parameter value.

\*→ once we create a rollback segment by default it will be in offline state

→ we need to make them online by executing the following parameter command.

DBA>alter rollback segment rbs1 online;

→ once we bounce the database, again it goes offline. To overcome this problem we use to set a init.ora parameter in the pfile.

i.e., rollback\_segments = rbs1,rbs2.

→ with respect to rollback segments, there are more chances of getting out of space errors and snapshot too old errors. To overcome all these problems, in version 9i oracle has introduced automatic undo management.

→ In case of automatic undo management, we need to set the following parameters in the init.ora

undo\_management = auto

undo\_tablespace = undotbs

undo\_retention = 900 seconds (default value).

→ Once we create a undo tablespace by default oracle creates 10 undo segments with the naming convention `sysmu$1` to `sysmu$10`.

→ Undo-retention represents how long the committed transactional data has to be retained in the undo segments.

\* → Oracle tries to retain the data as per the parameter value (`undo_retention`) but it is not guaranteed. If there is no space for the other incoming transactional data, simply it overwrites.

### Snapshot too old error: - (ORA-1555)

→ In case of parallel update and select, if the consistent part image is not available for a user who has already started reading the data from undo segments, such type of user is going to get snapshot too old error.

→ To minimize the occurrence of snapshot too old error, create bigger undo tablespace or commit more frequently.

→ In log, in order to retain the completed transactional data.

→ In log, in order to retain the completed transactional data, oracle has introduced. Creating undo tablespace with retention guarantee.

→ The purpose of undo is ① Transaction rollback  
② Transaction recovery  
③ Instance recovery

→ An undo segment will exists in any of the following states.

① online      ② offline      ③ pending offline

→

## 15/09/14 Oracle Networking :-

→ oracle networking facilitates accessing of data existing in a remote server.

→ oracle networking is also called as oracle net service

→ whether it is client (or) server the process of installation is same.

→ the important configuration files of oracle networking are

① listener.ora

② tnsnames.ora

③ sqlnet.ora

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

→ we can configure the above files either by invoking netca (Network Configuration assistant) (or) manually.

→ By default, oracle has provided some sample configuration files in \$ORACLE\_HOME/network/admin/samples directory

\*\*→ By default, oracle looks for network related files in \$ORACLE\_HOME/network/admin directory

→ we can also place network related files in some other location.

In this scenario we need to export the following environment variable

\$ export TNS\_ADMIN = \$HOME (path location)

→ listener.ora has to be configured at the server side and tnsnames.ora has to be configured at the client side.

→ sqlnet.ora is generally used in case of third party authentication (Kerberos) and to restrict / as sysdba

→ In order to restrict / as sysdba, we need to set the following parameters in sqlnet.ora

\* `sqlnet.authentication\_services = none`

up → Web Server → appl. Server → DB Server.

oracle client & oracle clients

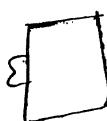
tnsnames.ora

to - hrms =

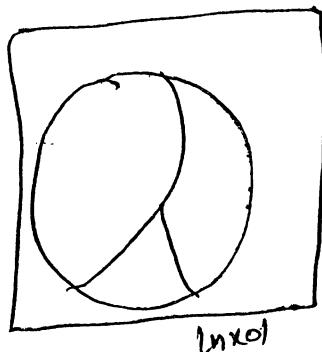
(Description =

(Address (list = (protocol=tcp) (host = (lnx01) (port=1521)))

(SID : hrms)



oracle server

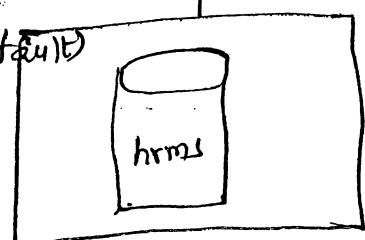


Listener.ora

listener\_name = LISTENER

protocol = TCP

hostname = lnx01/IP



ORACLE\_HOME

port = 1521 (default)

SID = hrms.

Commands

\$ lsnrctl stop

\$ lsnrctl start

\$ lsnrctl reload.

→ A part from maintenance activity, we also need to perform some maintenance activity on the listener log otherwise size becomes unmanageable.

- steps to perform maintenance activity on the listener log
- the default location of listener log is \$ORACLE\_HOME/network/log

```
$ lsnrctl  
lsnrctl> set log_status off
```

```
lsnrctl> exit
```

```
$ mv listener.log listener.log.bkp
```

```
$ touch listener.log
```

\$ lsnrctl  
lsnrctl> set log\_status on

Note: \* In order to perform the above activity, first we need to set logging-listener = off in the listener.ora

\* If the listener is down, we can't establish a connection to the instance.

\* Once the listener is down, nothing will happen to the existing connections but no new connections will be allowed.

### pfile/spfile:-

→ pfile is a text file whereas spfile is partial binary & partial text in nature

→ spfile was introduced in 9i version of oracle

→ Recommended not to edit spfile either with notepad (or)

vi editor

\* The advantage of having a spfile is, whatever the changes that we do for the dynamic parameters are persistent (permanent) across next reboots.

→ If the instance is started with spfile, we can specify the following:

scope = both

scope = Spfile

scope = memory (pfile)

→ If the instance is started with pfile, we can specify only scope = memory.

e.g.: SQL> alter system set open\_cursors = 500 scope = both;

\* Sometimes in the industry we get the following error messages

① Max. Cursors exceeded

② Max. Sessions exceeded

- For the first problem, ask the developer to close the cursors and dynamically increase the parameter value "open-cursors".
- For the second problem, increase the value of the parameter "sessions".  
This is a static parameter

### \* \* DB Links & Materialized Views

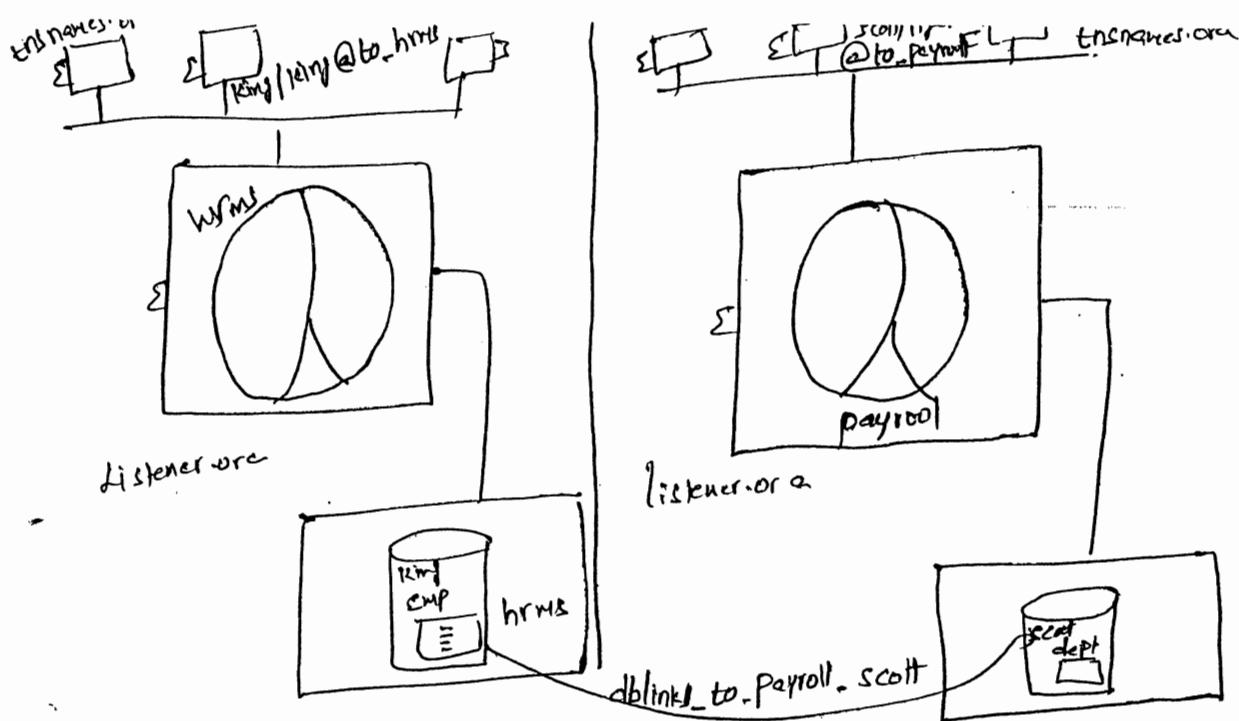
- Establishing a link from one database to another database is called a DB Link.
- If an user wants to create a DB link, user should have create database link privilege.
- Database links internally depends on oracle Networking concept.
- There are two types of DB links
  - ① private DB link
  - ② public DB link
- In order to create a public DB link, user should have create public database link privilege.
- Using DB links, we can perform all sorts of DML & Select operations but not DDL operations.

King> select e.empno, e cname, d.deptno, d.dname, d.location from Emp e,  
dept d where e.deptno = d.deptno;

DBA> grant create database link to King;

King> create database link dblink\_to\_payroll\_scott  
connect to scott identified by tiger  
using 'to\_payroll';

King> select \* from dept@dblink\_to\_payroll\_scott;



DBA> grant create synonym to king;

king> create synonym dept for dept@dept' dblink\_to\_Payroll\_scott;

king> select \* from dept;

- As the volume of data is increasing, accessing the data through dblinks degrades the performance. To overcome this problem, oracle has introduced materialized views

→ In older versions of oracle, materialized views are called "snapshots"

→ Normal view is a stored query and it is not a space demanding object.

ex:- > Create view emp\_view\_dept\_no10 as

select \* from emp where dept\_no=10;

> select \* from emp\_view\_dept\_no10;

- Materialized view is a local copy of remote database object and it is a space demanding object

- Sometimes we create materialized views on local database objects to enable query rewrites.
- If an user wants to create a materialized view, he should have create materialized view privilege.
- At the time of creating a materialized view, we can specify the refresh interval as well as the refresh mode.
- Refresh interval means how frequently the materialized view has to be refreshed.
- The different types of refresh modes are ① Complete ② fast ③ force ④ None.
- In case of Complete, oracle truncates the entire materialized view and refreshes the data. This is a time consuming process.
- In case of fast, oracle refreshes the data based on primary key.
- In order to create a materialized view with fast option source data base object must and should have primary key and at source side we need to create "materialized view log".
- Prior to log, in order to refresh materialized view we use to set the following parameter "Job\_queue\_process".
- From log, by default this parameter is set a value '10': CJOQ CJOQ1 CJOQ2 ... CJOQ9
- Force means, initially it tries to refresh based on primary key. If primary key is not available oracle truncate the materialized view and refresh entire data.
- None means, oracle will not refresh automatically and depending on our requirement we need to refresh manually by executing the following package.

SQL> execute dbms\_mview.refresh ('mvicename');

18/07/14

4. Support

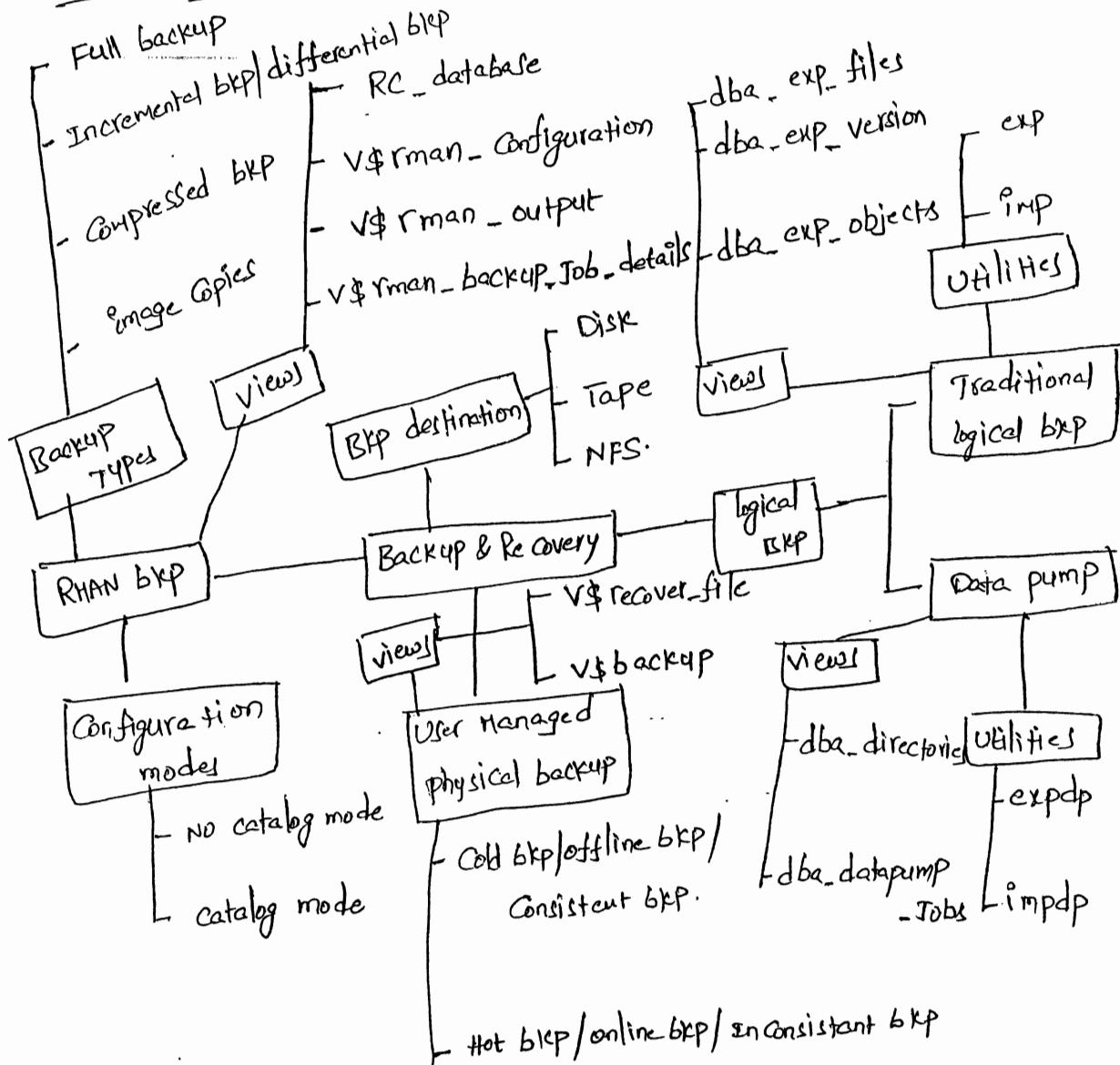
Q. Support

Ans:

180 - D - 582 - 218

60 - S

## Backup & Recovery :-



- The reasons why oracle has introduced a different backup options are
- ① Logical backup can't be considered as a main backup option since, we don't have the concept of recovery
  - ② We can't implement cold backup strategy for 24/7 customers
  - ③ If the database size is in Terabytes, every day we can't take full database backup. In this scenario, we need to have incremental backup.

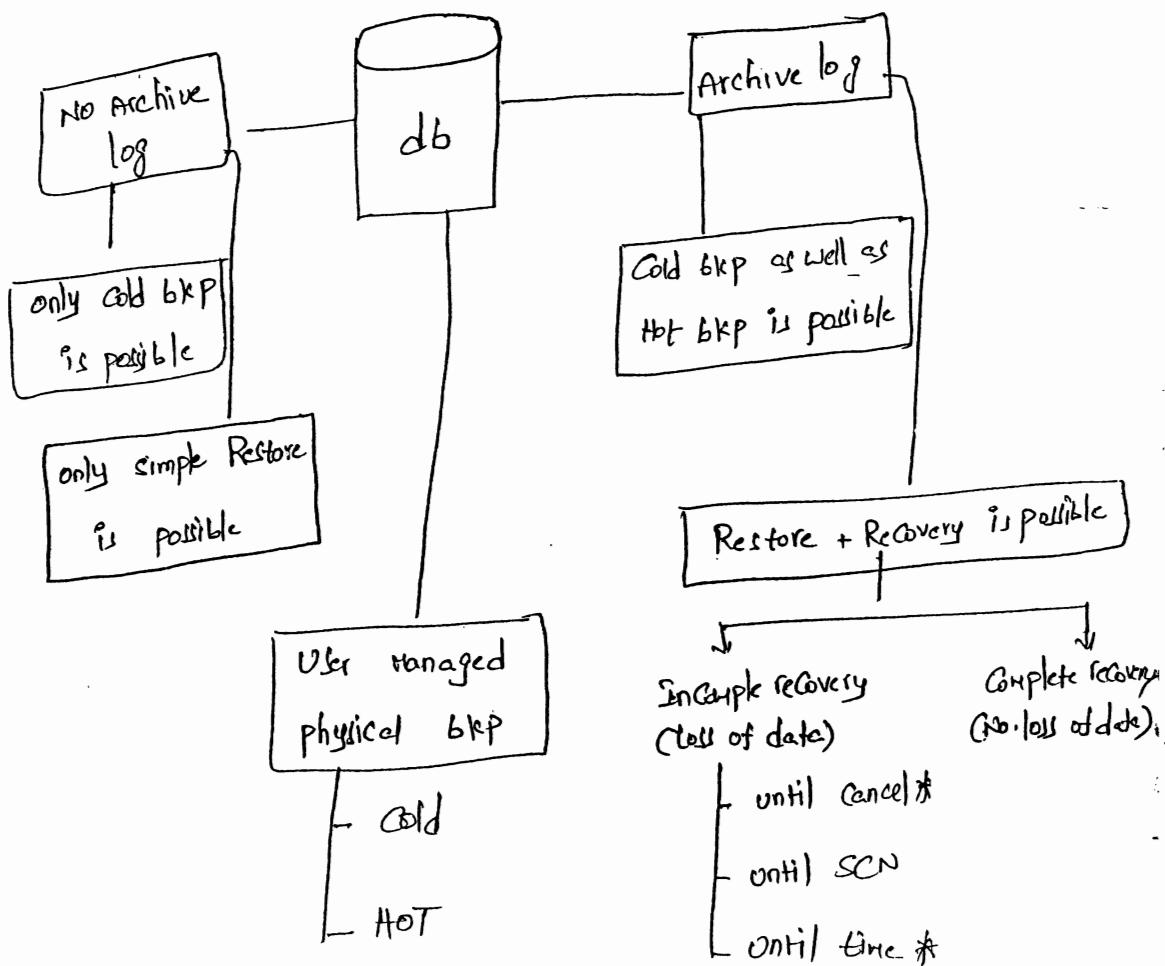
④ In case of Tera bytes data base environments, every day we take incremental backup.

### User managed physical backup:-

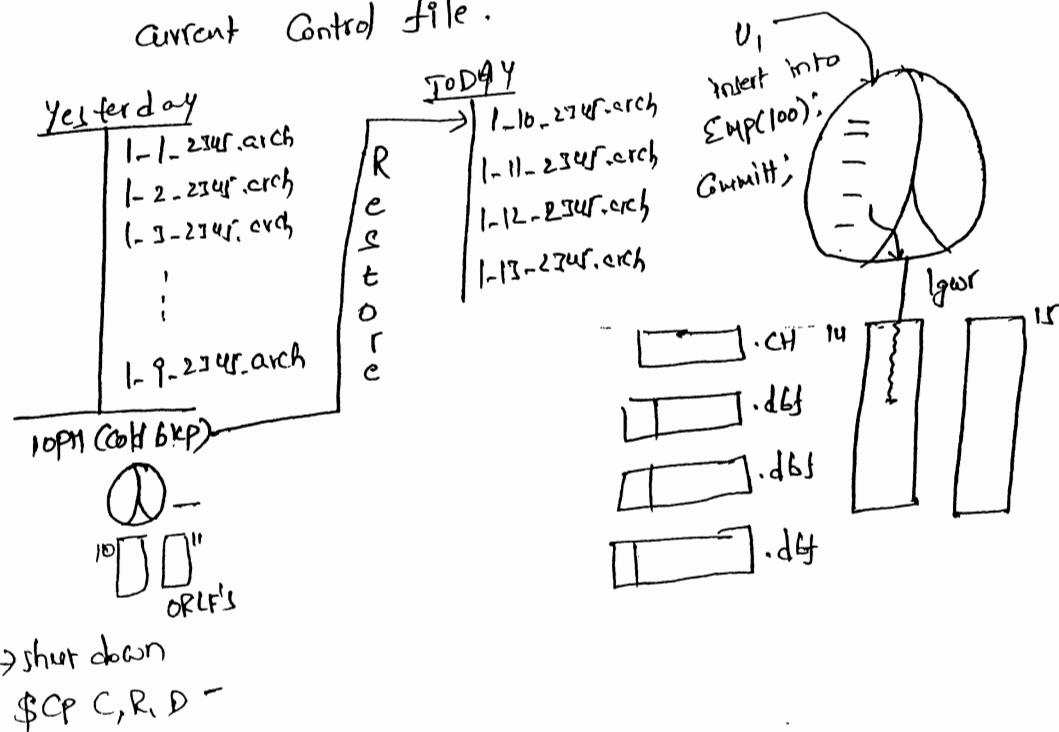
#### ① Cold Backup:-

- Cold backup is also called as offline backup (or) consistent backup.
- In order to take cold backup, we need to shutdown the database gracefully by executing any one of the following command.

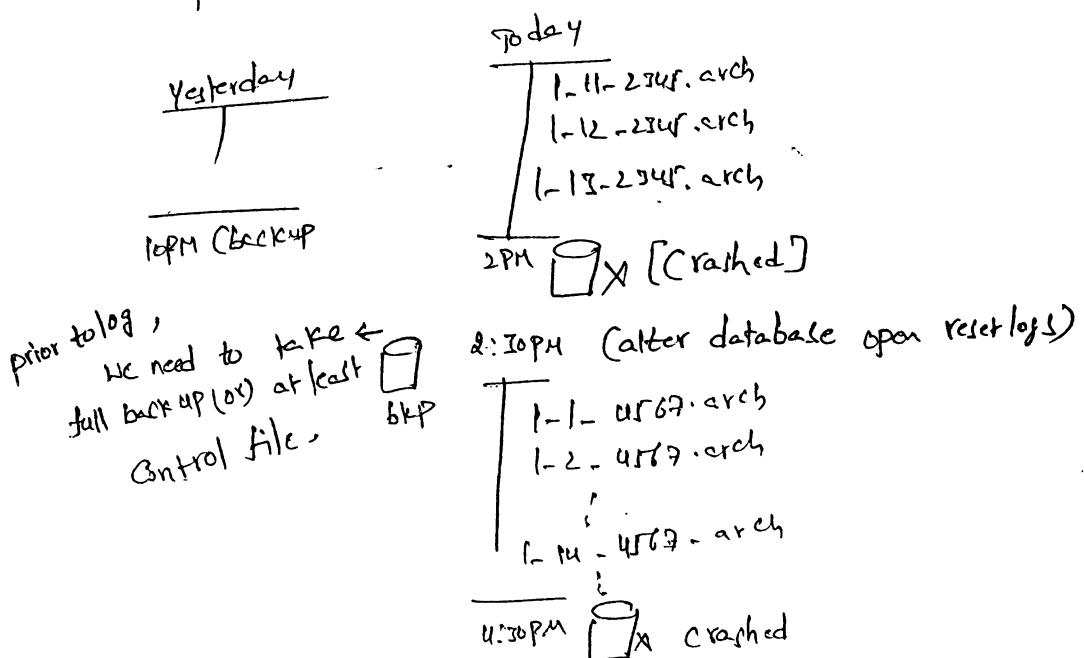
- (a) shutdown normal
- (b) shutdown transactional
- (c) shutdown immediate



- If DB runs in no archive log mode, only cold backup is possible and only simple restore is possible.
- If DB runs in archive log mode, both cold & hot backup is possible.
- Recovery is of two types
  - ii) Incomplete recovery
  - iii) Complete recovery
- In case of incomplete recovery, there will be loss of data
- \*→ whenever we perform incomplete recovery, we open the database with "reset logs".
- Whenever we open the database with reset logs, oracle generates new incarnation no. (reset log ID) and creates brand new online redo logs starting with log sequence "1".
- \*→ Loss of current online redo logs leads to incomplete recovery
- \*→ the only scenario where we open the database with reset logs but still it is complete recovery is in the event of loss of current control file.



- prior to log, child incarnation archive logs can't be applied to parent incarnation.
- prior to log, whenever we open the database with resetlogs, recommend to take the backup of full database (or) at leasts backup of control file.
- \*→ From log, child incarnation archive logs can be applied to the parent incarnation.



### Steps to take cold backup:-

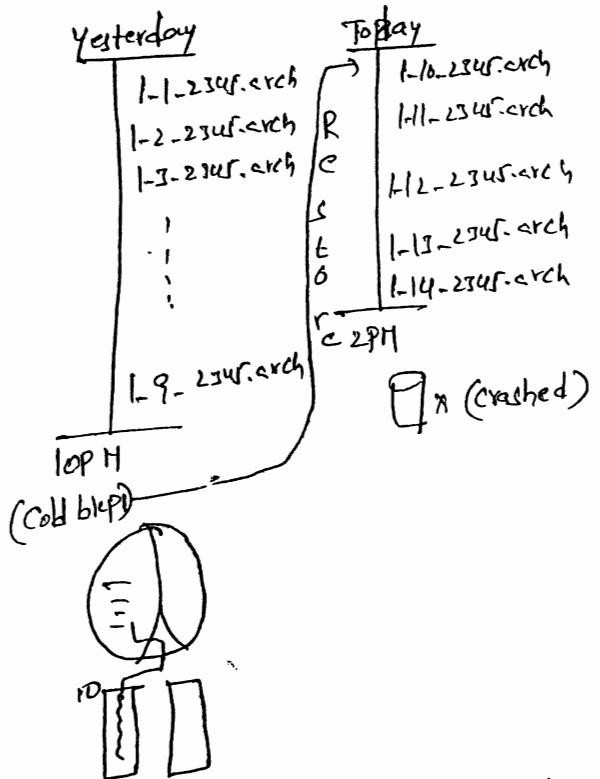
- ① shut down the database gracefully
- ② Copy CRD files to backup destination.

Note:- Along with the database backup, it is also recommended to take the backup of archive logs & pfile (or) spfile.

### loss of full database (CRD):-

## Steps to restore & recover:-

- ① shut abort
- ② Restore the latest backup
- ③ startup mount
- ④ Alter database recover automatic using backup controlfile until cancel;
- ⑤ Recover Cancel
- ⑥ Alter database open Resetlogs;



Note:- After performing full database crash recovery, oracle will not allow us to open the database because of log sequence mismatch. To overcome this problem, we open the database with "Resetlogs".

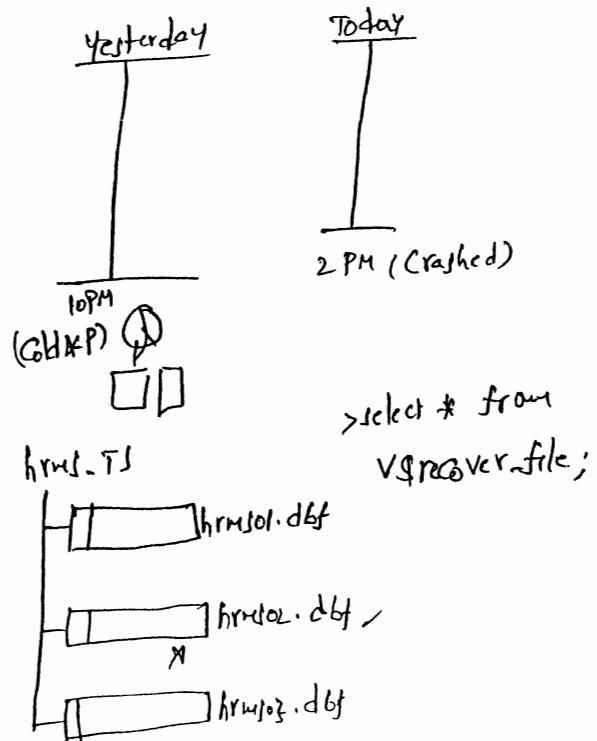
## S-II Loss of non-system datafile:-

### Method -I (Not Advisable)

- ① offline Tablespace (hrms-TS)
- ② Restore the lost datafile from the latest back up
- ③ Recover the lost datafile
- ④ online the tablespace (hrms-TS)

### Method -II (Recommended) :-

- ① offline the lost datafile
- ② Restore the lost datafile from the latest back up
- ③ Recover the lost datafile
- ④ online the datafile.



### S-III Loss of system datafile (System.dbf) :-

- ① shut abort
- ② Restore system.dbf from the latest backup
- ③ startup mount
- ④ Recover database
- ⑤ Alter database open.

### S-IV Loss of system datafile (System.dbf) :-

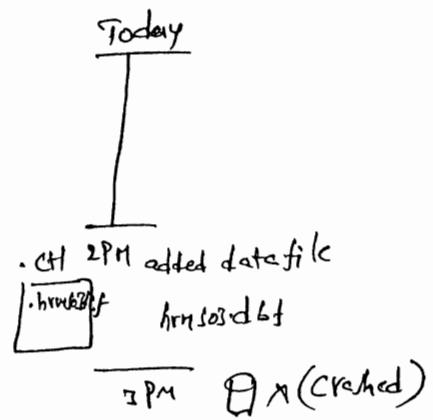
- ① offline sysaux.dbf
- ② Restore sysaux.dbf from the latest bkp.
- ③ Recover datafile <no>(or) <path/sysaux.dbf>
- ④ online sysaux.dbf

### S-V lost all Control files :-

- ① shut abort
- ② Restore the Control files from the latest bkp
- ③ startup mount
- ④ alter database recover automatic using backup control file until cancel;  
only active
- ⑤ Recover database using backup control file until cancel;  
(or)
- ⑥ Recover Cancel:
- ⑦ Alter database open resetlogs;

## S-VI Loss of Datafile which was not in backup:-

- ① offline the lost datafile
- ② create the lost datafile  
SQL> alter database create datafile  
'/u01/app/oracle/hrms/hrmso3.dbf':
- ③ recover datafile <no> or <path/hrmso3.dbf'.
- ④ online the datafile.



## S-VII Loss of temp.dbf:-

→ Not necessary to perform restore & Recovery (no entry in the Control file), just create one more temporary tablespace and make it as default for the database. (or) simply drop it and create the temp.dbf.

MANOJ ENTERPRISES  
Plot:40, Gosthani Nagar  
Amarpet, Hyderabad

## \* \* HOT backup:-

- Hot backup is also called as online backup (or) in consistent backup.
- This is the type of backup strategy that we implement for 24/7 customers.
- \*\*→ The pre-requisite for hot backup is database must & should run in archive log mode.
- In case of hot backup, there is no concept of simple restore, we must & should perform recovery.
- Whether it is cold or hot, we take the backup using cp command.
- With respect to cold backup, the way how we take the hot backup differs, but the process of restore & recovery is same.
- In case of hot backup, we backup tablespace by tablespace by keeping the tablespace in a special mode called begin backup.

→ once the backup of all tablespaces is completed we are going to backup the controlfile at oracle level by executing the following command

```
SQL> alter database backup controlfile '<path/control01.ctl>';
```

- \* → once the backup of all tablespaces and controlfile is completed, we switch the log file manually.
- switching the log file manually causes a checkpoint and the checkpoint change number will be updated to. Control files as well as data files.

### Steps to take Hot BACKUP :-

- ① SQL> alter tablespace hrms\_ts begin backup;
- ② \$ cp hrms.\* /opt/hotbkp/
- ③ SQL> alter tablespace hrms\_ts end backup;
- ④ Repeat step 1 to step 3 for all tablespaces;
- ⑤ SQL> alter database backup controlfile to '/opt/hotbkp/control01.ctl';
- ⑥ SQL> alter system switch log file;

Note:- In case of hot backup, we backup only Data files & Control files but not Redolog files.

\* → In log, oracle has introduced a single command to keep the entire database in begin backup mode. Recommended to use this command during half-peak hours (Non-business hours)

\* → If u keep the tablespace in begin backup mode, headers of datafiles of that particular tablespace will be freezed. and oracle will not update the SCN number that was generated.

→ oracle updates the SCN number once we execute end backup of the tablespace

~~→ prior to log, in order to refresh the data in log.~~

~~① cold after database begin backup;  
② MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad, S. A. 500012  
③ .bak file~~

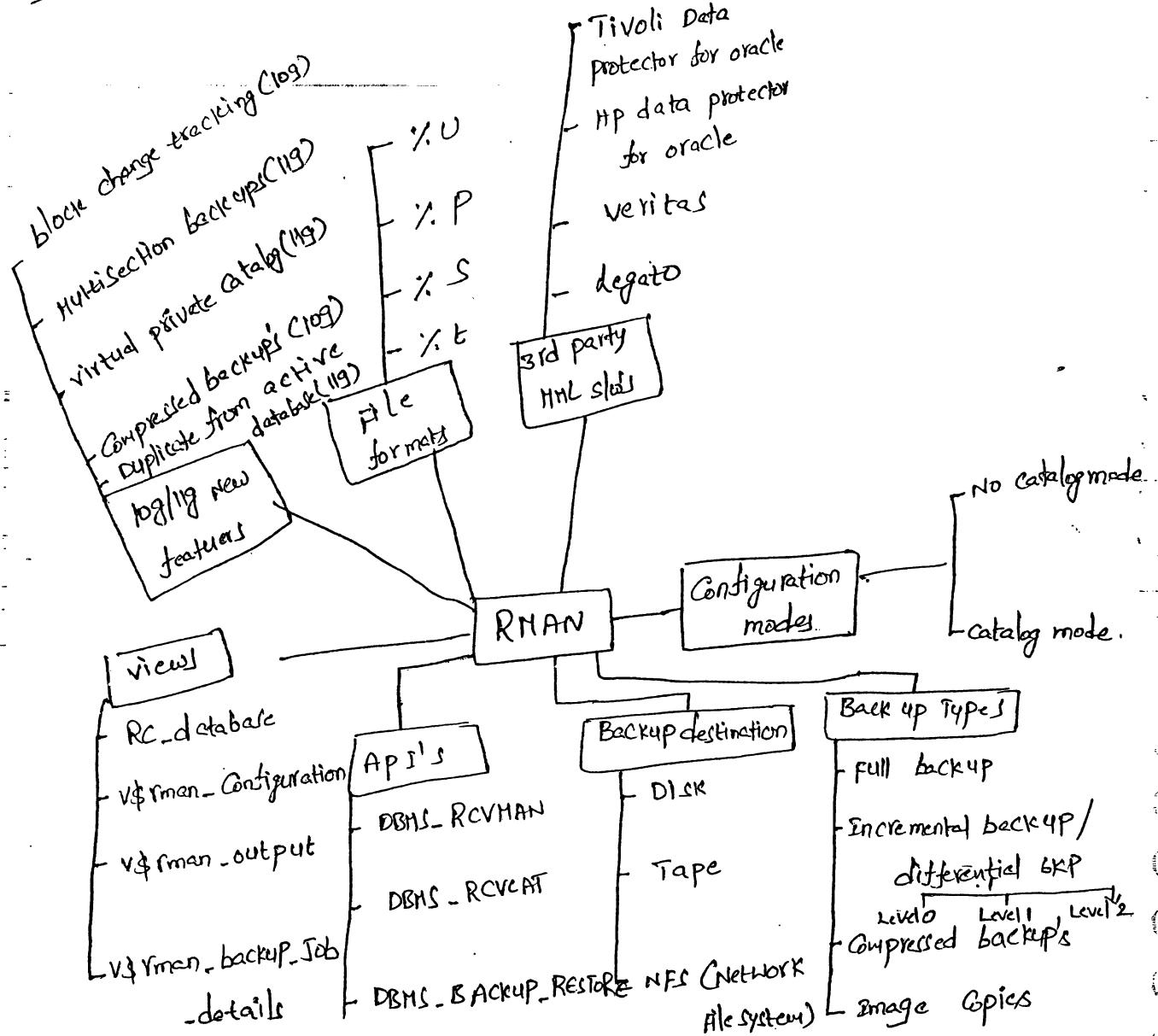
- \* Recommended to keep the tablespace in begin backup mode for shorter duration.
- \* If we keep the tablespace in begin backup mode for longer duration, excessive redo information will be generated which leads to more no.of log switches and more no.of archives in the archive destination.  
(If the archive destination is 100% filled, database goes to hung state)
- \* The reason why excessive redo information will be generated is because, for the first time oracle brings the entire block into database buffer cache which leads to more no.of log switches.
- In case of hot backup, there are chances of getting split blocks (inconsistent block) because of this reason, we must & should perform recovery.

Note:- while backup is going on, if the instance goes down, even if we (backup becomes useless), even if we execute startup it will not allow us to open database. execute end backup of the tablespace, open the database & schedule the backup again.

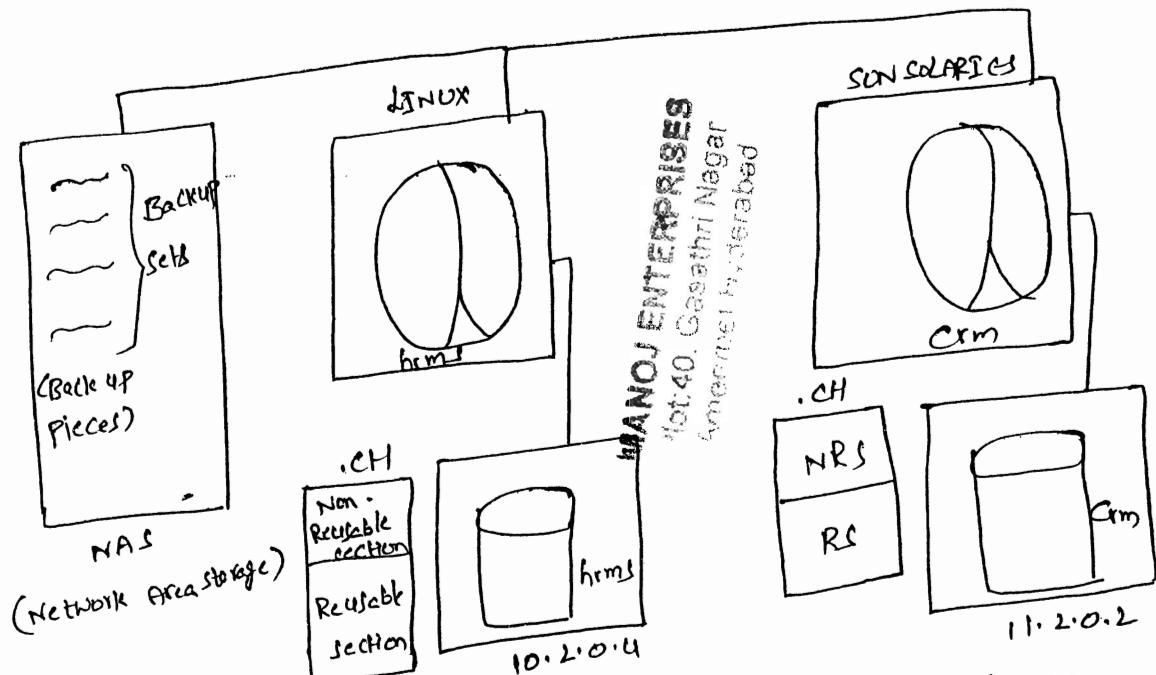
**MANOJ ENTERPRISES**  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

25/09/14

## \* \* RMAN (Recovery Manager) \* \*



- RMAN backup is a block level backup (Oracle block 8k)
- We can configure RMAN in two modes ① No Catalog mode  
② Catalog mode
- Recommended to Configure RMAN in Catalog mode.
- In no catalog mode, RMAN keeps the Configuration information & back up information in the reusable section of the target database control file.



No Catalog mode

```
$ export ORACLE_SID=hrms
$ rman target / <
rman> backup database;
```

```
$ export ORACLE_SID=crm
$ rman target / <
rman> backup database;
```

→ The retention of the backup information in the target database control file depends on the parameter Controlfile\_record\_keep\_time (default value 7 days). even if we specify a bigger value (365 days) for the parameter, retention may not be guaranteed. since, Control file grows up to o/s limitation beyond that it overwrites the existing backup information.

- RMAN is the centralized backup, restore and recovery utility provided if we configure RMAN in Catalog mode
- In No Catalog mode, we can't create stored RMAN scripts which can be used across the enterprise database.
- RMAN backup retention is based on two factors ① Retention based on no.of backups  
parameter: Configure retention policy to redundancy <value>

## ② Retention based on no.of days

parameter:- Configure retention policy to recovery window of <x days>;

C R p to R w of (x days);

→ In nocatalog mode, we can't specify "Keep until time" clause to retain a backup for a specific time interval

ex:- Backup & retention policy retention

weekly full bkp - 31 days

monthly full bkp - 365 days

Configure retention policy to Recovery window at 31 days;

random backup database;

random backup database keep until time 'sysdate+365' log;

→ Execution of every RMAN backup command creates backup sets in the backup destination.

→ A backup set is nothing but collection of backup pieces.

→ RMAN is capable of taking the backup to "disk" but not to "tape"

library

→ In order to take the backup into tape library, we need to integrate RMAN with 3rd party MHL (Media management layer) soft's like

Tivoli Data protector for oracle, HP Data protector for oracle, Veritas, Legato

(OR) OSB (Oracle Secure Backup).

→ OSB is proprietary of oracle and it is a separate licensed product.

→ In case of tape backup's, 3rd party administrator is going to install

& configure 3rd party software on all the database servers.

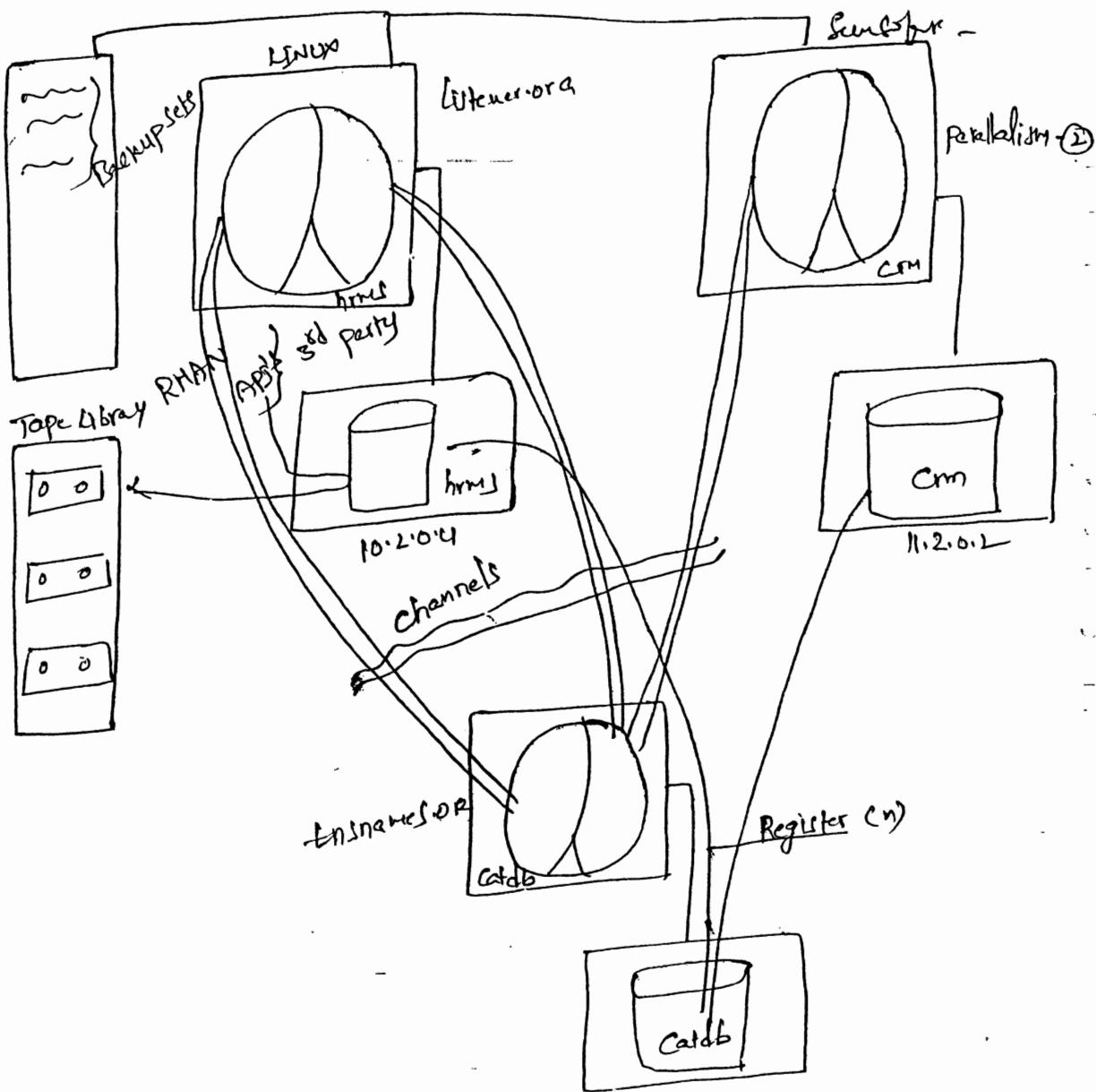
→ once 3rd party software is installed & configured, we need to create RMAN scripts by specifying 3rd party specific environment variables.

- we are going to get all 3rd party specific environment variables from support.oracle.com (os) 3rd party installation documents
- If the instance is started with Spfile , by default RMAN takes the backup of spfile.
- A backup piece may contain controlfiles as well as datafiles but not archive logs.
- Separate backup pieces will be created for archive logs.
- If backup strategy is RMAN, Recommended to Configure Large pool otherwise RMAN uses some space from shared pool.

26/9/14

#### Pre-requisites to Configure RMAN in Catalog mode:-

- Note:- Catalog is a repository of all target databases. We can register any no of databases in Recovery Catalog.
- ① Identify a host and server & install the same (or) higher version of Oracle binaries.
  - ② Create Catalog Database
  - ③ Create a separate tablespace to hold RMAN metadata tables.
  - ④ Create a user and grant Connect, Resource, RECOVERY\_CATALOG\_OWNER Roles.
  - ⑤ Create catalog
  - ⑥ Configure tnsnames.ora
  - ⑦ Register target databases into Recovery Catalog.
  - ⑧ Configure RMAN parameters
  - ⑨ Create backup, restore and recovery scripts.



- Since catalog is a separate database, it can hold the backup information as long as we require
- A channel is Communication session from catalog database to target database
- No of channels generally depends on the size of the target database and the no of processors within the server.
- We can configure multiple channels with the help of parallelism parameter.
- If the parallelism value is ~~not~~ 2 and if we allocate 4 channels in the run block, priority goes to run block.

ex: \$ vi rmanfull.bkp.rcv

run {

```
allocate channel c1 device type disk;  
" " c2 " " ;  
" " c3 " " ;  
" " c4 " " ;
```

backup database plus archive log delete all input;

### RMAN incremental backup:-

- If the database size is in terabytes, every day we can't take full backup. In this scenario, we need to implement incremental backup strategy.
- In case of incremental backup, RMAN takes the backup of only modified blocks, which minimizes the backup time and the space required in the backup destination.
- In order to minimize the incremental backup time, in log, oracle has introduced block change tracking.
- once we enable block change tracking, 'CTWR' (change track writer) gets enabled and it keeps track of the modified blocks information and updates to a separate file at the os level.
- SQL> alter database enable block change tracking using file '/opt/rhome/change\_track\_file';
- \*→ Snapshot Control file is a consistent point-in-time control file created by RMAN in the /dbs directory (at the target side)
- The different levels of incrementals are ① Level 0 (~~Full~~ Complete bkp)  
② Level 1 (Incremental bkp)  
③ Level 2 (Cumulative bkp)

→ Apart from target database, it is also equally important to have a proper backup strategy for catalog database (every day schema level logical backup, weekly once cold bkp).

RMAN Catalog start with /opt/rman bkp;

\*→ From 10.2.0.4, levels greater than 0 & 1 have become absolute (no more)

→ Level 0 is the base backup for incremental backup (level 1).

\*→ without Level 0, we can't take Level 1 backup.

\*→ without Level 0, If we initiate Level 1, by default RMAN takes Level 0 backup.

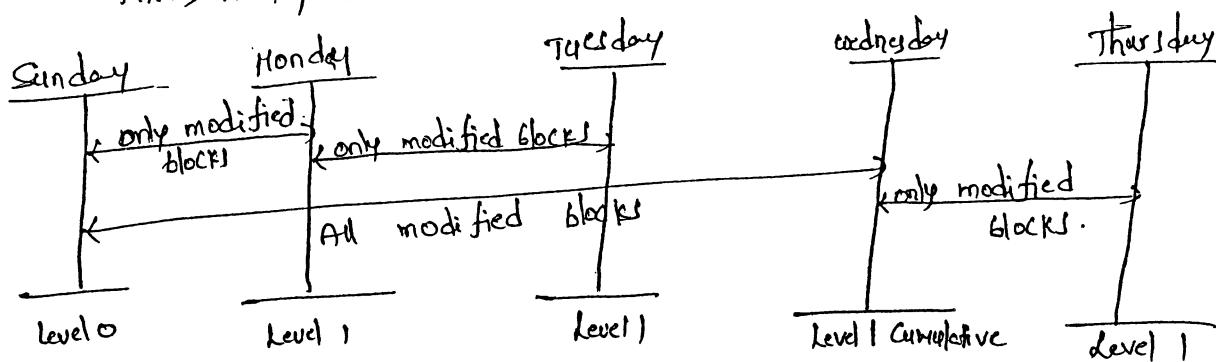
RMAN> backup incremental level 0 database;

RMAN> backup incremental level 1 database;

RMAN> backup incremental level 2 database;

(or)

RMAN> backup incremental level 1 cumulative database;

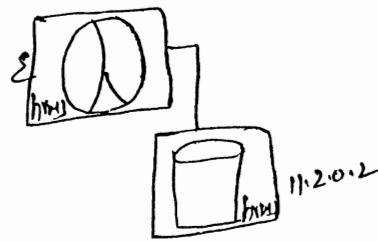


### RMAN Cloning:-

→ The process of duplicating the entire database from production to dev/test environments is called "cloning".

→ "Refresh" is the word that we use whenever we are performing schema level refreshes.

Procedure



- ① Create pfile from spfile

Change db\_name to duphrms

If file system naming convention is different set

db\_file\_name\_convert = ('Source path', 'target path')

log\_file\_name\_convert = ('Source path', 'target path')

- ② Take Rman backup along with archive logs

(11gR2 not required)

- ③ \$cp pfile, rman backup to test server

- ④ Configure tnsnames.ora (to duphrms)

- ⑩ \$rman target / <|

rman> Connect auxiliary sys /pwd@to\_duphrms

log rman> duplicate target database to 'duphrms'  
nofile name check;

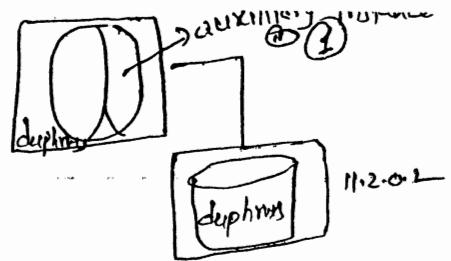
11g rman> duplicate target database to 'duphrms'  
from active database nofilename check;

### MultiSection backup's :-

→ In 11g, oracle has introduced multi-section backup's where we can split out the backup of huge datafile into multiple sections. so that, multiple channels are going to act on the same datafile.

→ Multi-section backup's concept is more useful in case of huge datafiles and bigfile tablespaces.

Result



- ④ check whether Listener is up & running or not

- ⑥ Copy initdphrms.ora to initdphrms.ora & place it in \$ORACLE\_HOME directory

- ⑦ Create directory structures as per initdphrms.ora

- ⑧ Create password file using orapwd utility

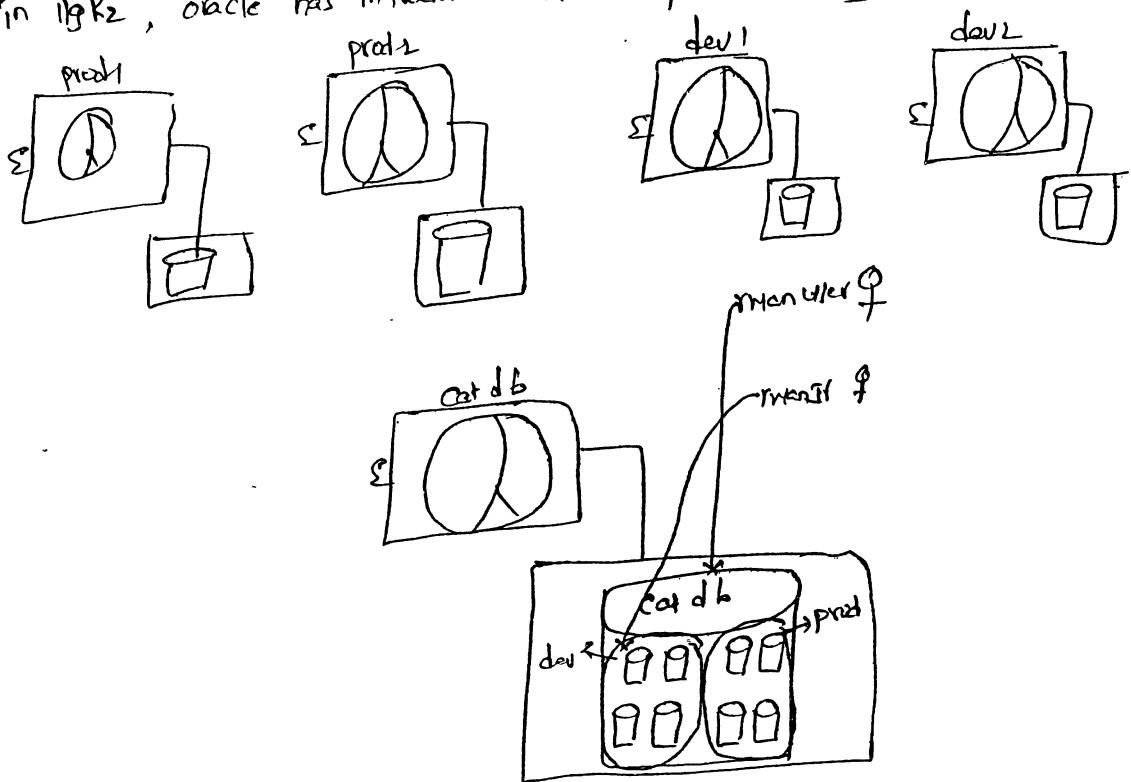
- ⑨ \$ export ORACLE\_SID = duphrms  
\$ sql startup nomount

- ⑩ \$sql> select name, open\_mode,  
log\_mode from v\$database;

- \$sql> select instance\_name, status  
from v\$instance;

## Virtual private Catalog:-

→ prior to 11g, A Single recovery catalog can be used by multiple users which may lead to security concerns. To overcome this problem, in 11gR2, oracle has introduced virtual private catalog.



## \* \* Backup strategy:-

EE

Daily → incremental bkp + archive logs & delete the archive logs from archive destination.

3 days once → cumulative bkp + archive logs & delete the archive logs from archive destination.

Weekly one → Full bkp + archive logs & delete the archive logs from archive destination.

SE :-

→ If we have the license of the standard edition we can't enable block change tracking. (not supported). In this scenario incremental bkp takes more time. To overcome this problem, in some customer environments, we follow the following backup strategy.

for every 12 hours }  
(ex) }  
for every 24 hours } ⇒ Take the backup of archive logs & delete the archive logs  
from archive destination.

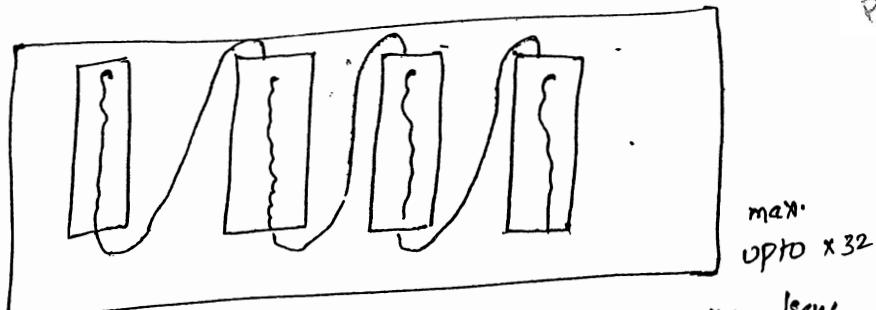
3 days once (or)  
weekly once }  $\Rightarrow$  Full backup + Archive logs & delete the archive logs from archive destination.

## RAID (Redundant array of independent disks (or) Inexpensive disks)

$\rightarrow$  The purpose of RAID is ① Fault tolerance  
② Mirroring  
③ parity

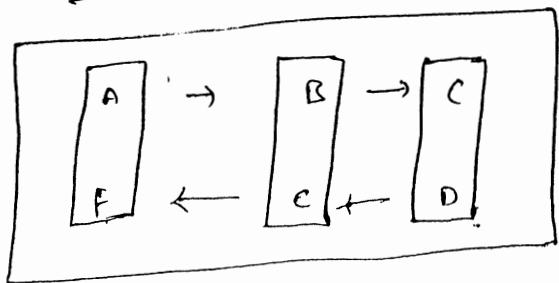
- $\rightarrow$  we can implement RAID at software level as well as hardware level.
- $\rightarrow$  Since H/W level RAID is very expensive, most of the customers they depend on software RAID.
- $\rightarrow$  In Linux, we can implement RAID with the help of "MDADM" tool (Metadata administration tool)
- $\rightarrow$  In 1960's, in the mainframe era, IT industry has introduced "Linear RAID".

Linear RAID :-      minimum disks - 2  
                              maximum disks - 32



If we lose one disk we can't recover the disks. So, IT industry not accepted.

RAID "0" also called as (striping)



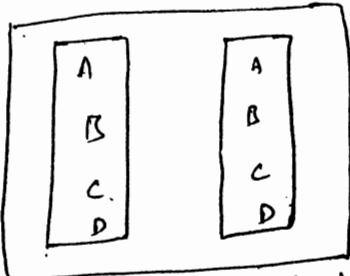
**MANOJ ENTERPRISES**  
Plot:40, Gesathri Nagar  
Ameerpet, Hyderabad

**MANOJ ENTERPRISES**  
Plot:40, Gesathri Nagar  
Ameerpet, Hyderabad

→ striping means equally distributes data among all disks there is no concept of mirroring, if we loose we don't recover.

### RAID 1 (Mirroring)

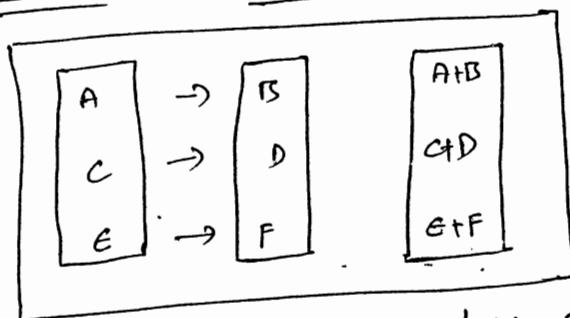
there is mirroring but not striping (disadvantage)



$$\text{space used} = \frac{\text{used disk}}{\text{No. of disks}} \times 100$$

$$= \frac{1}{2} \times 100 = 50\% \quad (\text{i.e., if at all out of } 3^2, \text{ we use } 16 \text{ only})$$

### RAID 4 (striping, mirroring with parities)

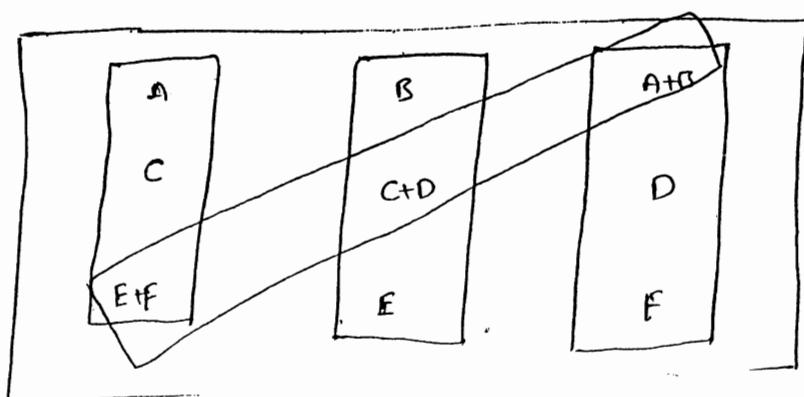


Disadvantage is if at all 3rd disk parity goes to 3<sup>rd</sup> disk

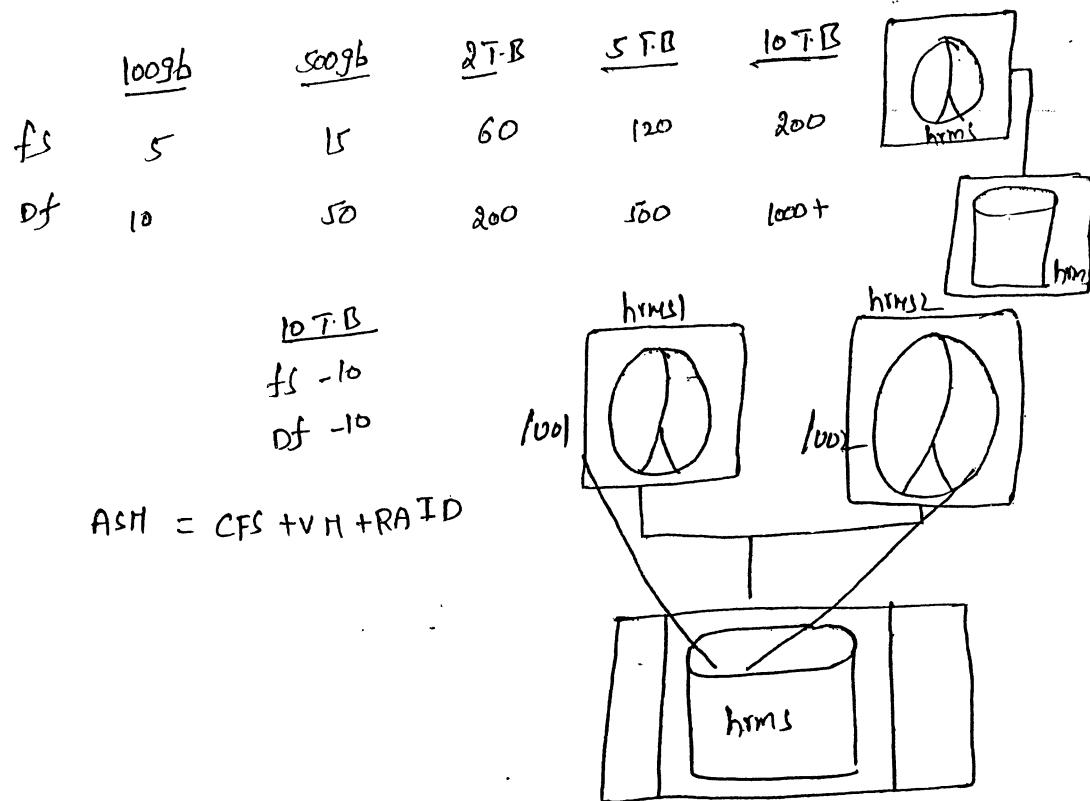
(more I/O's on 3<sup>rd</sup> disk) may cause failure.

### RAID 5 (striping, mirroring with parities)

The popular RAID implementation in the industry is RAID-5

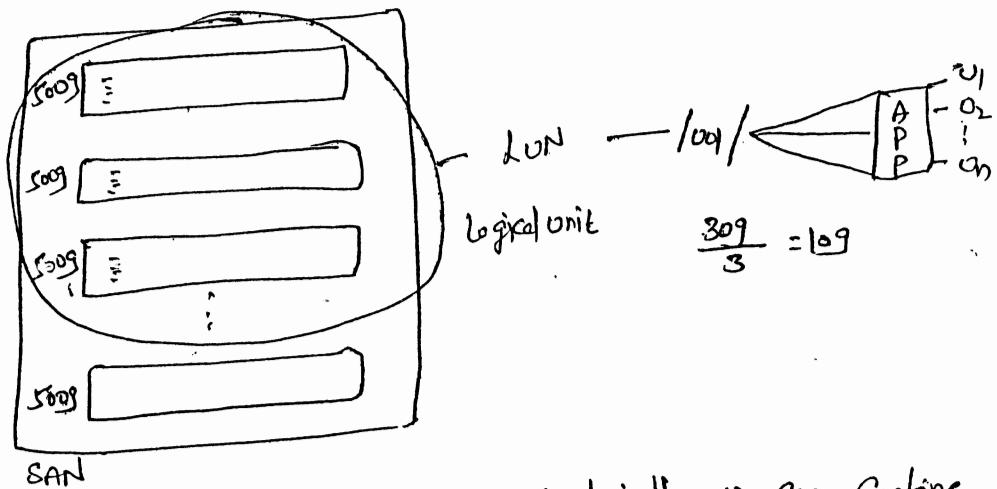
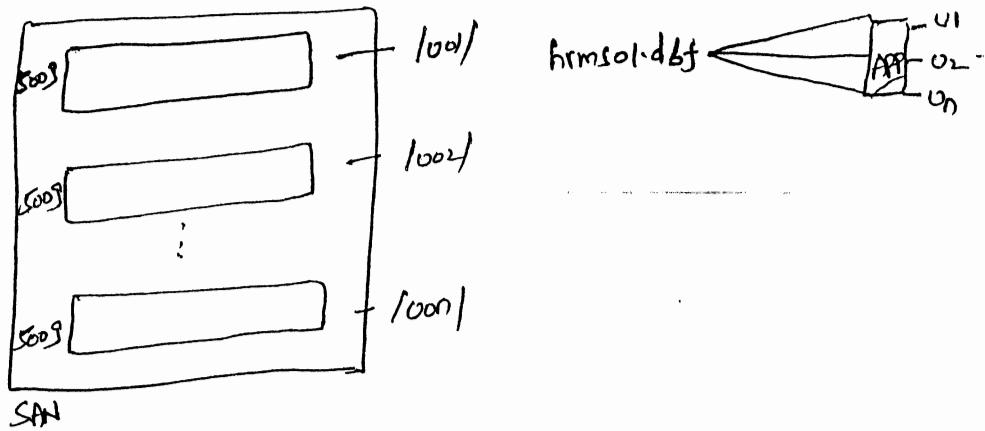


→ RAID-5 Configuration, parity will be maintained in all the disks



→ As the database size grows into terabytes (or) petabytes. Space management becomes a challenging task.

- To overcome space management challenges, in the industry most of the customers they use to depend on third party volume managers like veritox, tivoli volume manager etc.,
- The pre-requisite of RAC is database should be placed in the shared storage.
- In order to place the database in the shared storage, we need to mount the shared disk into multiple nodes (servers)
- In order to mount the shared disk in multiple nodes we depend on some third party softwares which we call them as cluster file system (CFS)
- In version 9i, RAC becomes an expensive solution to the customer, since customer has takes the license of third party CFS & volume manager (VM)
- In order to eliminate third party cluster file system (CFS) & volume manager (VM) in managing terabyte database environment, oracle has introduced ASM in 10g version of oracle.



- with the help of third party volume managers, logically we can combine two or more disks and we can create a LUN (logical unit)
- The LUN will be mounted a file system. even if we want re-size the file system, we can add a disk to the existing LUN.

### \* \* ASM (Automatic storage Management) :-

- ASM is the new technology introduced in 10g version of oracle.
- ASM is free of cost either the standard edition (or) enterprise edition.
- ASM is the foundation for RAC
- ASM can act as CFS, VM and it also supports RAID Concepts.
- In version 10g, ASM binaries are integrated with oracle binaries.
- From 11GR2, ASM binaries are separated from database binaries. and shifted along with clusterware binaries together we call it as grid infrastructure services.

log & ligr :-

oracle s/w = ASM binaries + oracle binaries

ligr & l2cr :-

Grid infrastructure service = ASM binaries + clusterware binaries

oracle s/w : oracle binaries

→ once we invoke the run installer of the grid infrastructure services, we get options like

- (i) install grid infrastructure services for stand alone
- (ii) Install grid infrastructure services for cluster (RAC)

→ striping is the default behaviour of oracle ASM. ASM equalizes striped the data among all the disks

ASM Terminology :-

- (1) ASM DISKS
- (2) ASM DISK groups
- (3) ASM instance
- (4) ASM background process
- (5) ASM files

→ In LINUX we can implement ASM either by using Raw device mode (or) ASM lib

→ ASM lib option is available only for Linux.

→ If we want to implement ASM using ASM lib, we need to download and install ASMRPMs (Read had package managers) based on the kernel version of os.

ASM DISKS :-

→ the disks which are labelled are used for ASM disk group creation are called ASM disks

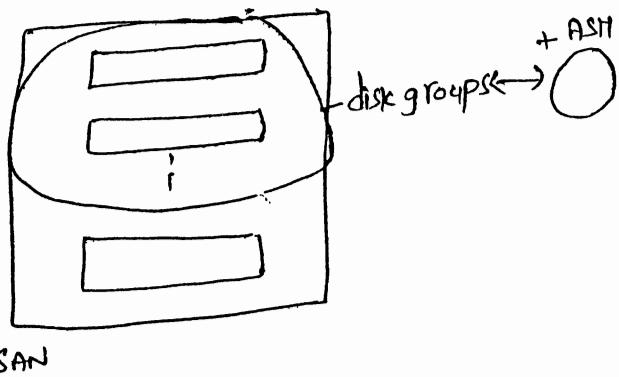
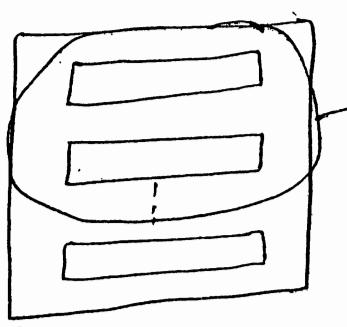
## ASH Disk group:

→ this is something like a LUN which we create using third party volume manager.

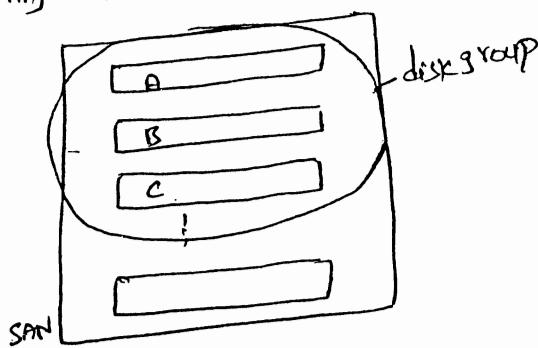
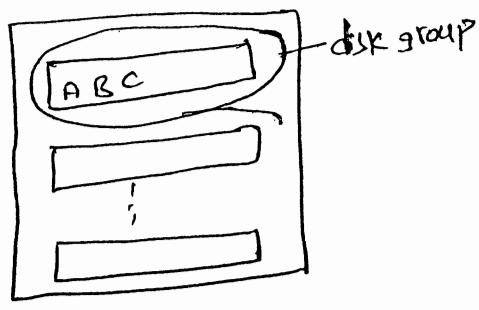
→ ASH supports RAID Concepts like RAID-0 & RAID-1

→ ASH supports three types of Redundancy.

- ① External Redundancy
- ② Normal Redundancy
- ③ High Redundancy

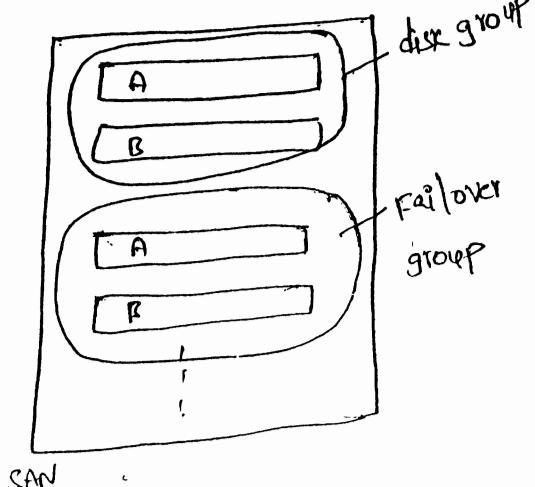
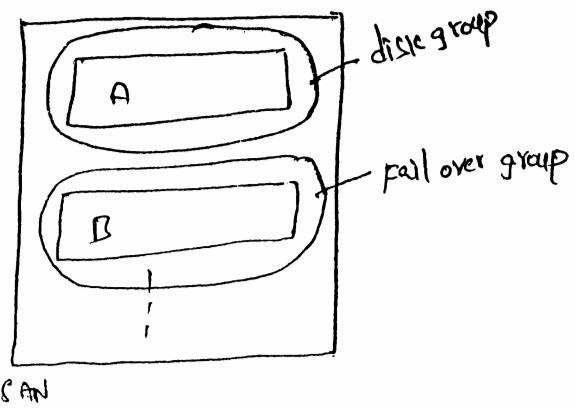


→ External Redundancy means no mirroring at ASH Level

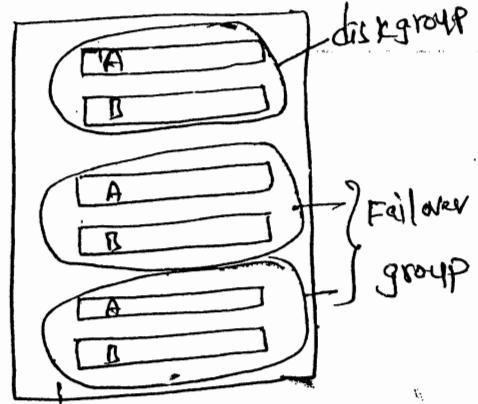
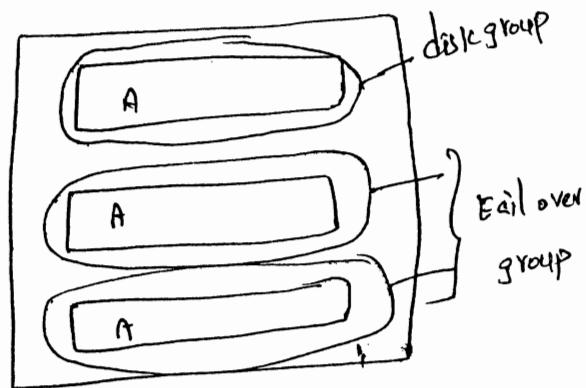


→ Normal Redundancy is also called as two way mirroring

→ Inorder to create a disk group with normal redundancy, minimum no of disks required are "2".

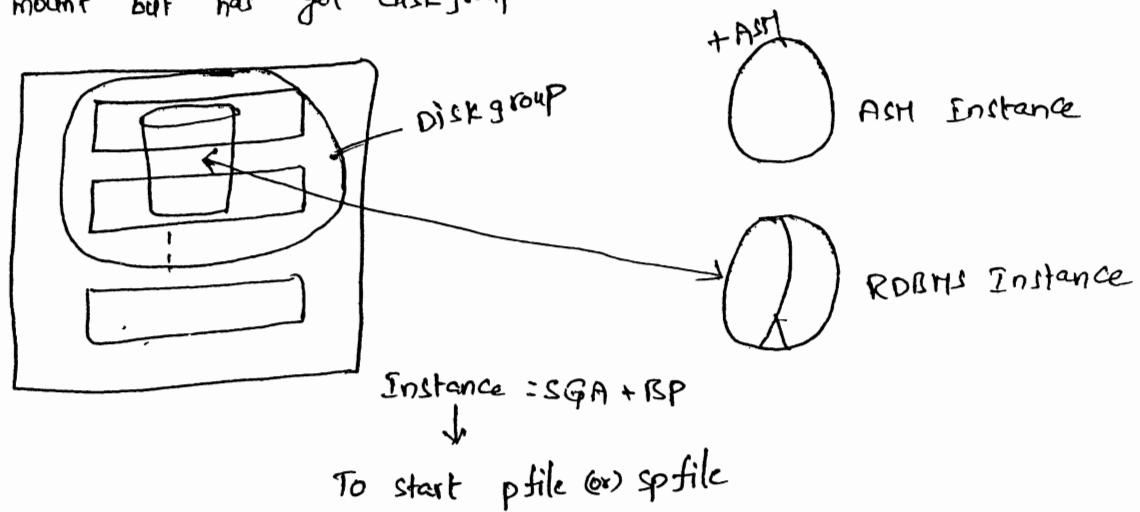


- High redundancy is also called as three way mirroring
- In order to create a disk group with high redundancy minimum no. of disks required are "3".



### ASM Instance:-

This is something like RDBMS instance but as no database to mount but has got diskgroup to mount.



→ In a Server, we can have only one ASM instance (limitation)

init.ash.ora

instance\_type = ASM

background\_dump\_dest = <path>  
user\_dump\_dest = <path>  
core\_dump\_dest = <path>

diagnostic\_dest = <path>

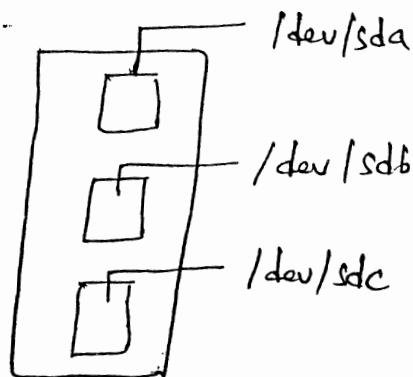
ASH\_POWER\_LIMIT = <1 to 1024>

ASH\_DISK\_STRING = '/dev/sda\*', '/dev/sdb\*'

ASH\_DISK\_GROUP = 'ASH-DG-DATA', 'ASH-DG-FRA'

Remote-login-password-file = 'Shared'

Large-pool-size = cvalue



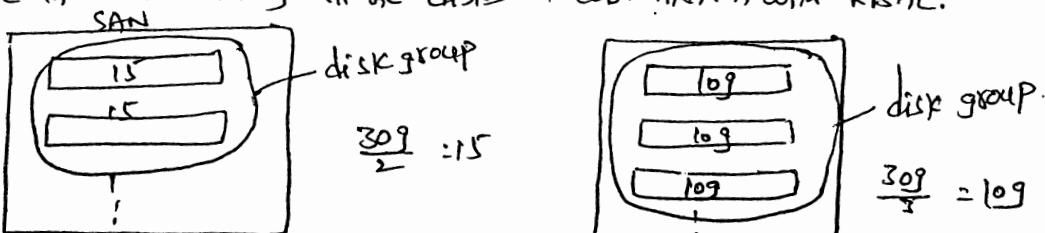
→ Apart from normal background processes, ASM has its own specific background process like ASMB, ARBN, RBAL.

\* → In log, Before starting ASM instance, we need to start OCSSD (Oracle cluster synchronization service domain) by executing the following script as root user

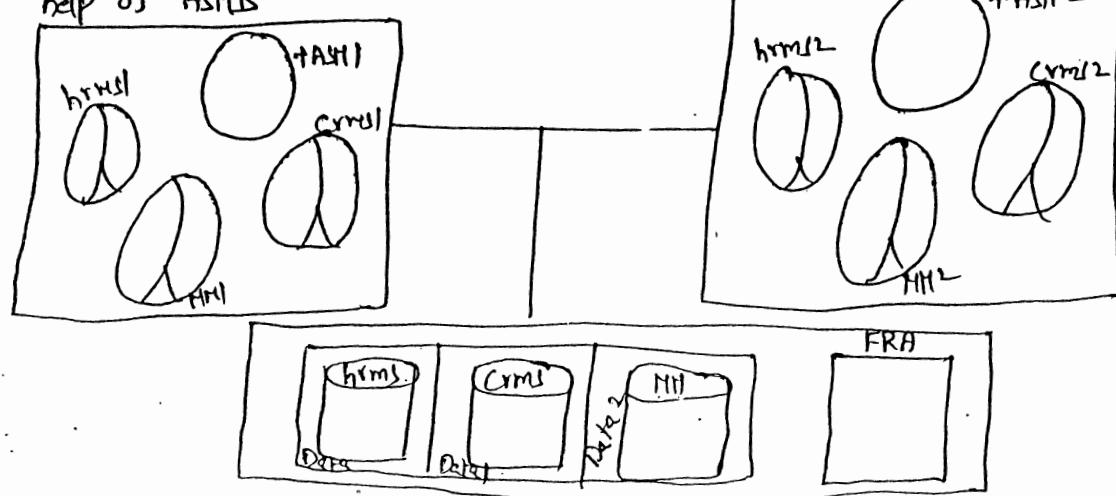
```
# cd $ORACLE_HOME/bin
```

```
# ./localconfig add <!--
```

→ whenever we add a disk to the existing disk group, ARBN dynamically rebalance the data among all the disks in coordination with RBAL.



→ In the foreground, ASM instance communicates with RDBMS instance with the help of ASMB



### Steps to bring down the environment:-

- (1) first bring down all RDBMS instances
- (2) then bring down ASM instance.

### Steps to bring up the environment:-

- (1) First bring up ASM instance
- (2) then bring up all RDBMS instances

### oracle Restart:-

- oracle has introduced oracle restart in 11gR2 version of oracle
- oracle Restart improves the availability of oracle databases.
- In order to Configure the oracle Restart we need to install grid infrastructure services.
- when we install oracle Restart the following oracle Components can be restarted automatically after a hardware (or) software failures occurs and when your database server restart.
  - (1) oracle instance
  - (2) oracle Listener
  - (3) oracle db services
  - (4) ASM
  - (5) ASM disk groups
- oracle restart runs out of grid infrastructure services home.
- oracle restart runs periodic checks to monitor the health of the Components, if any of Component goes down which we restart automatically.
- In oracle restart environments we can administer the Components with the help of "svrctl" (server control)

eg: \$ svrctl status database -d hrms  
\$ svrctl status ASM.

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

9/10/14

## \* MAA (Maximum Available Architecture) \*

→ Various technologies that are included under MAA are

- ① RAC
- ② Data Guard
- ③ ASM
- ④ Flashback technology

→ RAC is meant for instance level failures

→ In the industry, most of the time we come across instance level failures. To overcome this problem, Oracle has introduced multi instance database which is called RAC database.

\* \* → RAC supports two types of failovers ① Session fail over  
② Select fail over

\* \* → There is no concept of DML failover in RAC system

→ Since, ASM supports RAID Concepts because of this reason ASM is included under high availability (or) MAA

## \* \* DATA GUARD \*

→ Data guard alone are in combination with Real Application clusters (RAC) provides a robust high availability solution

→ The site where we configure the physical standby (or) DataGuard is called DR site (Disaster Recovery site)

→ There are two types of standby's ① Logical standby  
② Physical standby

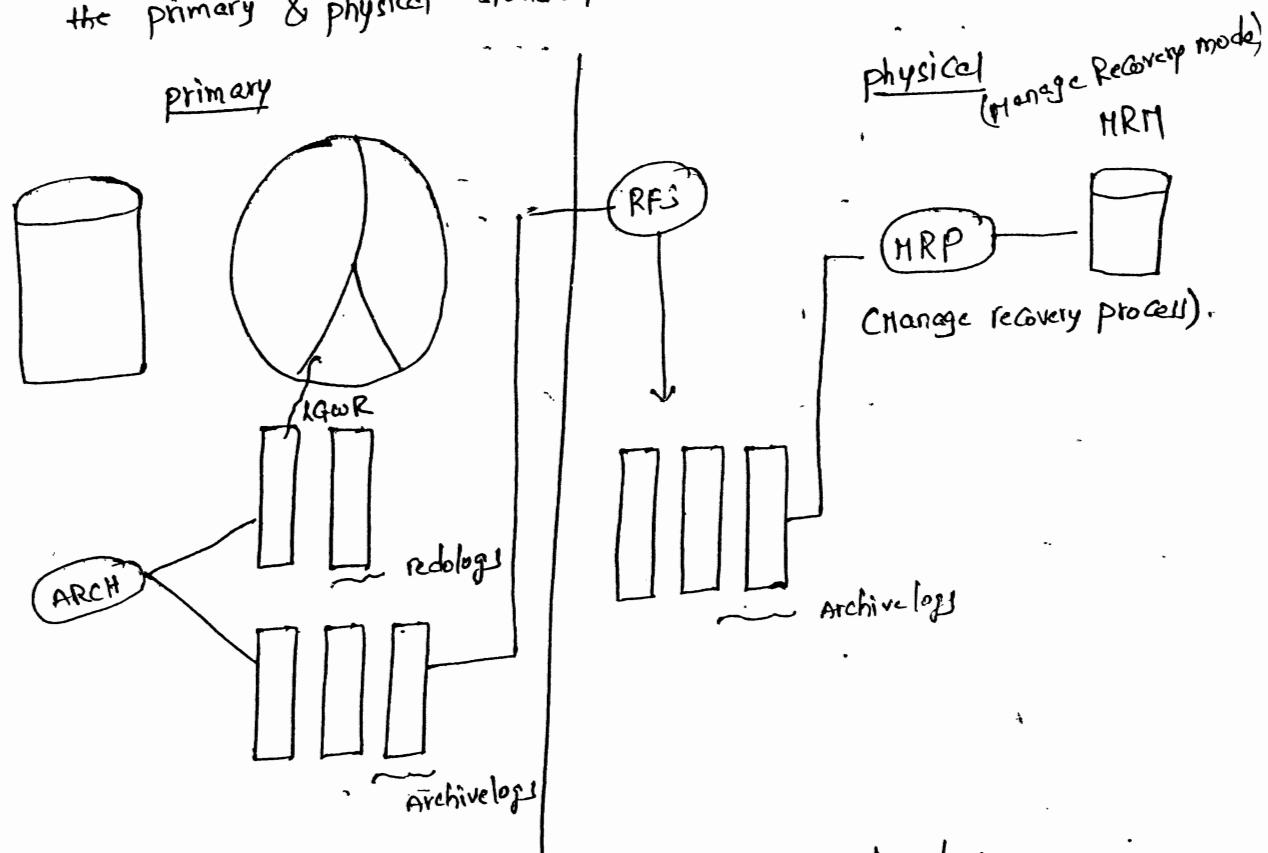
→ Logical standby is generally used for reporting purpose and it uses the concept of log minor and SQL apply

\* \* → If u want to have a business continuity plan then it is recommended to configure the physical standby.

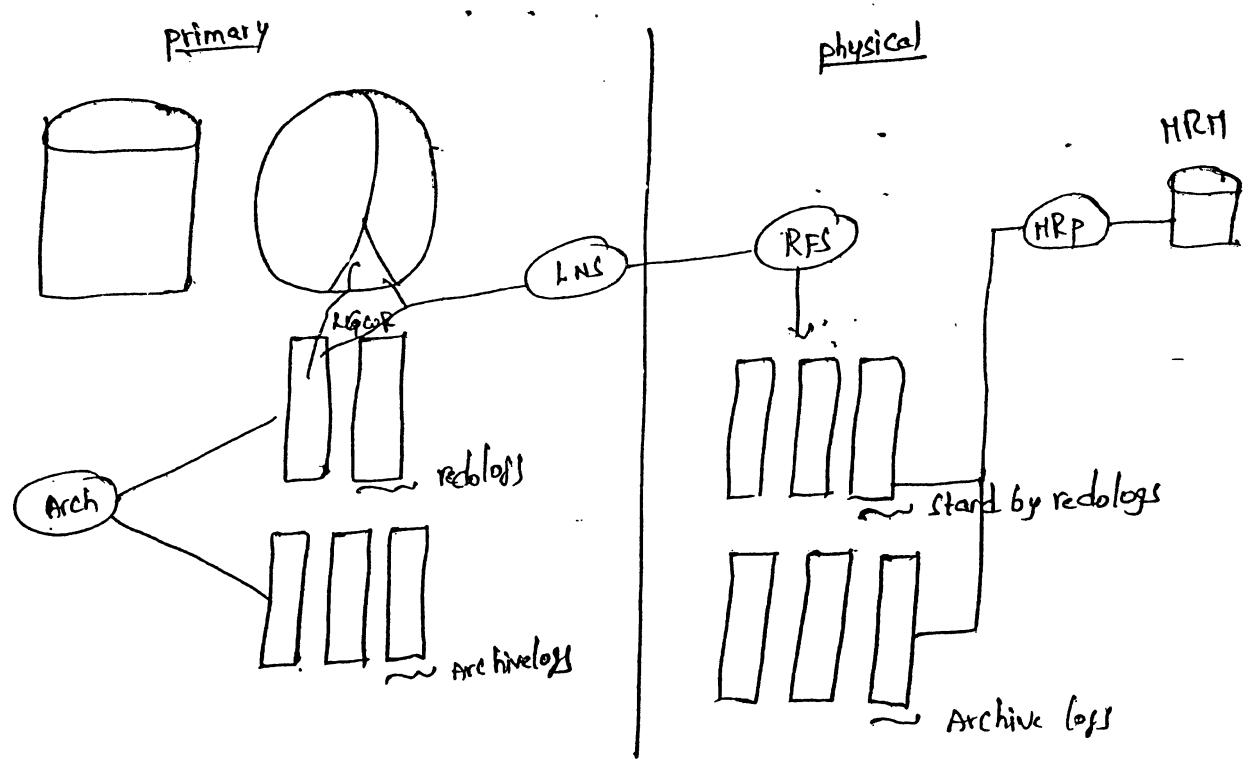
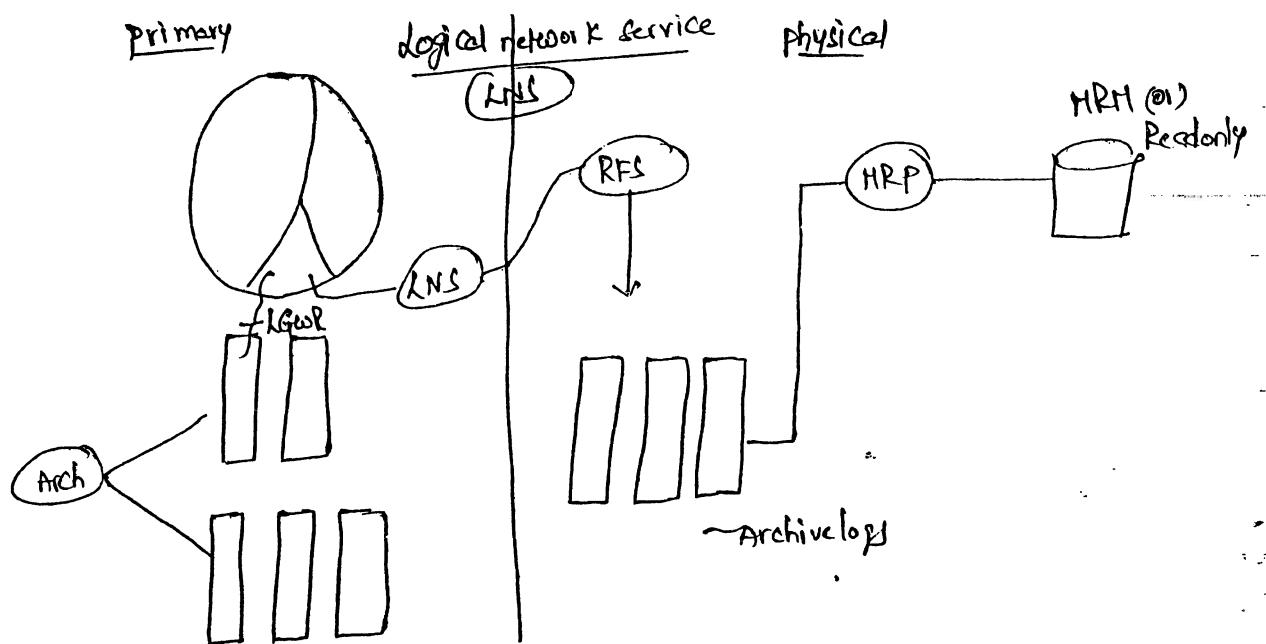
→ Prior to 9i, Data Guard is called Hot Standby

→ In version 9i, Oracle has made architectural changes and renamed the technology as physical standby database.

- If we configure physical standby with Dataguard broker then the configuration is called DataGuard.
- prior to 9i, the configuration mode is called delayed protection mode. In this mode, there will be lot of data divergence (mismatch) between the primary & physical stand by.



- RFS reads vector-by-vector and constructs the archive logs
- The constructed archive logs will be picked by MRP process and applies the data to the physical standby
- Standby runs either in MRM mode (or) read only mode.
- LNS:
  - In order to minimize data divergence, Oracle has introduced Log Network Service (LNS)
  - LNS search the data either from redo buffer cache (or) online Redolog files and hand-overs the data to RFS (Remote File System)
  - In the below architecture we can't achieve no data loss solution.
  - In order to achieve no data loss solution, Oracle has introduced Standby redo logs which needs to be created at the standby side



- In the above architecture, user gets Commit Complete acknowledgement once, the data has been written to the primary redo logs as well as the stand by redo logs.
- whenever there is an archive log gap, RFS pings back the primary archiver to shift the archive log to the physical stand by

→ In the above architecture, if primary is unable to shift the data to atleast one standby, primary goes down automatically without giving commit acknowledgement to the user.

→ up to 10g, we can configure 9 physical standbys & from 11g, we can configure 30 physical standbys

→ the formula to create Redologs in standby site.  
(instance)

$$(\text{no.of Redologs} + 1) * (\text{max no.of threads})$$

$$(2+1) * 1 = 3 \text{ (Redologs per Standby)}$$

(or)

$$(\text{min. no.of Redologs} + 1) * (\text{max no.of threads})$$

### Advantages of DataGuard:-

- ① High availability
- ② Disaster Recovery Solution
- ③ fail-over during unplanned downtime
- ④ switch-over during planned maintenance.
- ⑤ Standby can be used for reporting purpose.

### Pre-requisites to Configure DataGuard:-

- ① underline hardware architecture should be same.
- ② version of operating system, version of oracle including patch set level should be same.
- ③ primary must & should run in archive log mode.
- ④ In maximum protection mode we need to create standby Redologs which are of the same size of primary Redologs

### Architectural Components of DataGuard:-

#### ① Primary Database:-

→ The database that we wish to protect in the event of any disaster is called primary database.

**MANOJ ENTERPRISES**  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

**MANOJ ENTERPRISES**  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

## ⑤ Stand by:-

→ The database on which we depend in the event of primary collapse  
is called stand by database.

## ⑥ ORACLE NETWORKING:-

→ Facilitates movement of data from primary to stand by

## ⑦ Log transport service:-

→ Service at the primary which shifts the data to the physical stand by  
(Arch, LNS etc.)

## ⑧ Log apply service:-

→ Service at stand by which applies the data to the standby database.

## ⑨ Data Guard broker:-

→ It's an optional component of DataGuard Configuration. If we configure  
DGMON (DataGuard monitor) process gets enabled.

## ⑩ Archive log gap resolution:-

→ In order to resolve the archive log gap, Oracle has introduced 'FAL'  
(Fetch archive log) process. For this, we need to set the parameters  
"FAL-CLIENT & FAL-SERVER"

## ⑪ Network transmission modes:-

1. sync affirm

2. Async noaffirm.

→ In case of maximum protection mode, network transmission mode should  
be sync affirm.

## ⑫ DataGuard Configuration modes:-

### ⑬ maximum protection mode:-

→ If u want to architect a no data loss solution, we need to configure  
the physical standby in maximum protection mode.

→ In this mode, we need to create stand by redo-logs which are of the  
same size of primary redo-logs.

→ In this mode, user gets Commit Complete acknowledgement once the data  
has been written to the primary redo-logs as well as the stand by redo-logs.

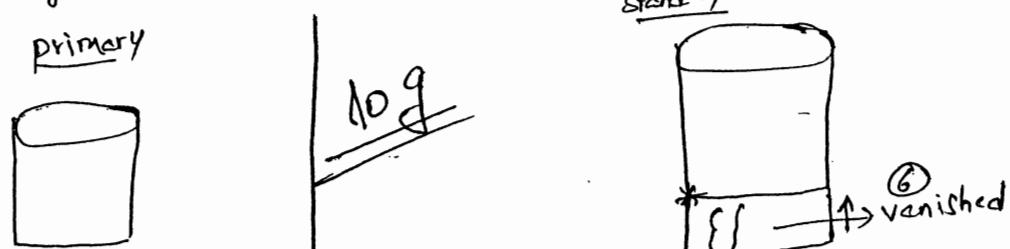
- In this mode, If primary is unable to shift the data to atleast one stand by, primary goes down automatically.
- In case of financial domains (BFSI), Recommended to configure Dataguard in maximum protection mode.

### ② Maximum performance mode:-

- In this mode, we can't achieve no data loss solution
- In this mode, it is not necessary to create standby redo logs
- In this mode, user gets Commit complete acknowledgement once the data has been written to the primary redo log

### ③ Maximum availability Mode:-

- Initially Configuration works in maximum protection mode, If primary is unable to shift the data to atleast one standby, primary will not go down & Configuration works in maximum performance mode.
- In 11g, oracle has introduced a primitive version of Snapshot standby where we need to configure most of the things manually. (Creating guaranteed restore point, Configuring the db in flash back mode etc..)



③ stop shipping the archive logs

④ start shipping the archive logs

- ① Configure the db in flash back mode
- ② Create a guaranteed restore point
- ③ Open it in Read/write mode
- ④ Use it as test instance (Create tables, procedures)
- ⑤ Flash back the database to guaranteed restore point
- ⑥ Put it in MRM (Manage Recovery mode) mode.

→ In 11g, oracle has introduced active Dataguard which can be used for

Real time reporting

→ We can also convert the Snapshot standby to active Dataguard but it requires a separate licensing.

### Snapshot Standby

- ① Can be open in Read/write mode
- ② Both will not be in sync
- ③ No extra licensing is required

### Active Dataguard

- ① can be open in read only mode
- ② Both will be in sync
- ③ Extra licensing is required.

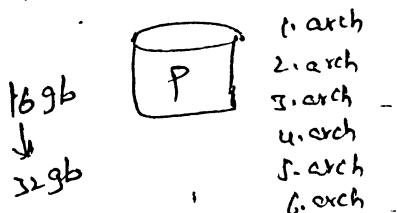
### Views to Data Guard:-

- ① v\$managed\_standby
- ④ v\$archive\_gap

- ② v\$database
- ③ v\$instance
- ⑤ v\$dataguard\_status
- ⑥ v\$dataguard\_config

### DR out of sync, steps to resync :-

Note:- If the primary database size is in terabyte, it is practically not possible to rebuild the DR right from scratch. In this scenario, we need to take the incremental backup(RMAN) and then we need to resync the DR.



- 1. RMAN> backup database plus archive log delete all input;
- 2. If it fails the statement up to apply all the archive logs.



primary



standby

④ RMAN> Run{

allocate channel c1 device type  
disk format '/vol/orabkp/%.0.bkp';

Backup incremental from SCN # ##  
database;

⑤ SQL> alter database create standby  
controlfile as '/vol/orabkp/standby.ctl';

⑥ SCP rman.bkp, standby.ctl to  
standby server

⑦ SQL> select current\_scn from v\$database;

① SQL> select current\_scn from v\$database;

② SQL> alter database recover managed standby  
database cancel;

③ SQL> shut immediate;

④ Replace the current control file with standby.ctl

⑤ SQL> start up mount

⑥ RMAN> catalog start with '/vol/orabkp/';

⑦ RMAN> Recover database;

⑧ SQL> select current\_scn from v\$database;

⑨ SQL> alter database recover managed standby  
database disconnect from session;

卷之三

## Patching & upgrading :-

patch :- patch is the bug-fix

patch set :- collection of bug-fixes is called a patch set

→ Different types of patches released by oracle are

① interim patch / one off patch (small bug fix)

② patch sets

③ CPU patches (critical patch update) (Security patches)

④ PSU's (Patchset updates)

→ All the above patches are installed using "opatch" utility except patchsets.

→ patch sets are installed by invoking "runInstaller".

→ For every quarter (Jan, Apr, July, Oct), oracle releases CPU patches <sup>11, 12, 13\*</sup> to overcome security threads.

→ Recently oracle started releasing PSU's

→ PSU's contains CPU & small other bug-fixes.

→ If PSU is installed, not necessary to install CPU

→ The default location of 'opatch' utility is \$ORACLE\_HOME/opatch

→ To know the version of opatch utility

\$ cd \$ORACLE\_HOME/opatch

\$ ./opatch -version

→ To know the list of patches that are installed in \$ORACLE\_HOME

\$ ./opatch lsinventory (or)

\$ ./opatch lsinventory -detail \$ORACLE\_HOME

<u>10gR2</u>	<u>Base version</u>	-	10.2.0.1
	patchsets	-	10.2.0.2 10.2.0.3 10.2.0.4

<u>11g R1</u>	<u>Base version</u>	-	10.2.0.5
	patchset	-	11.1.0.6 11.1.0.7

MANOJ ENTERPRISES  
Mohanirai Nagar  
Kompeta, Hyderabad

11gR2

Base version - 11.2.0.1

patchsets

- 11.2.0.2 }  
 11.2.0.3 } are also called as  
 11.2.0.4 } Base versions

12 CP1

Base version

- 12.1.0.1

Note:

patch (or) upgrade 10.2.0.1 database to 10.2.0.4 with minimal downtime



Current ORACLE\_HOME

ORACLE\_HOME = /u01/app/oracle/product/10.2.0/db-home/

Steps

- ① Install base version in /u02/app/oracle/product/10.2.0/db-home.
- ② Install the 10.2.0.4 on top of /u02/app/oracle/product/10.2.0/db-home
- ③ Shutdown the db
- ④ Copy initnew.ora from old home to new home
- ⑤ point to new home (export ORACLE\_HOME=new home path)

SQL> startup upgrade

SQL> @catupgrade.sql

SQL> @utlrp.sql

Downtime Consideration

① 15-20 min Installation

② 45 min (upgrade time)

③ 10-20 min (to recompile invalid objects)

1:15 min

Note: In 11gR2, oracle has introduced out of place patch set installation where we can install 11.2.0.2, 11.2.0.3, in a different home (Base version not required)

CN: 4287932.zip (interim patch)

\$ cp 4287932.zip \$ORACLE\_HOME/patch

\$ cd \$ORACLE\_HOME/patch

\$ unzip 4287932.zip

\$ ./patch apply 4287932

if any error occurred then

CN:

\$ ./patch rollback 4287932

## Pre-patch Considerations & Recommendations:-

- ① Take the backup of oracle inventory

Linux

\$ cat /etc/oraInventory

others

\$ cat /var/opt/oracle/orainst.loc

Inventory location: /home/oracle/orainventory

etc

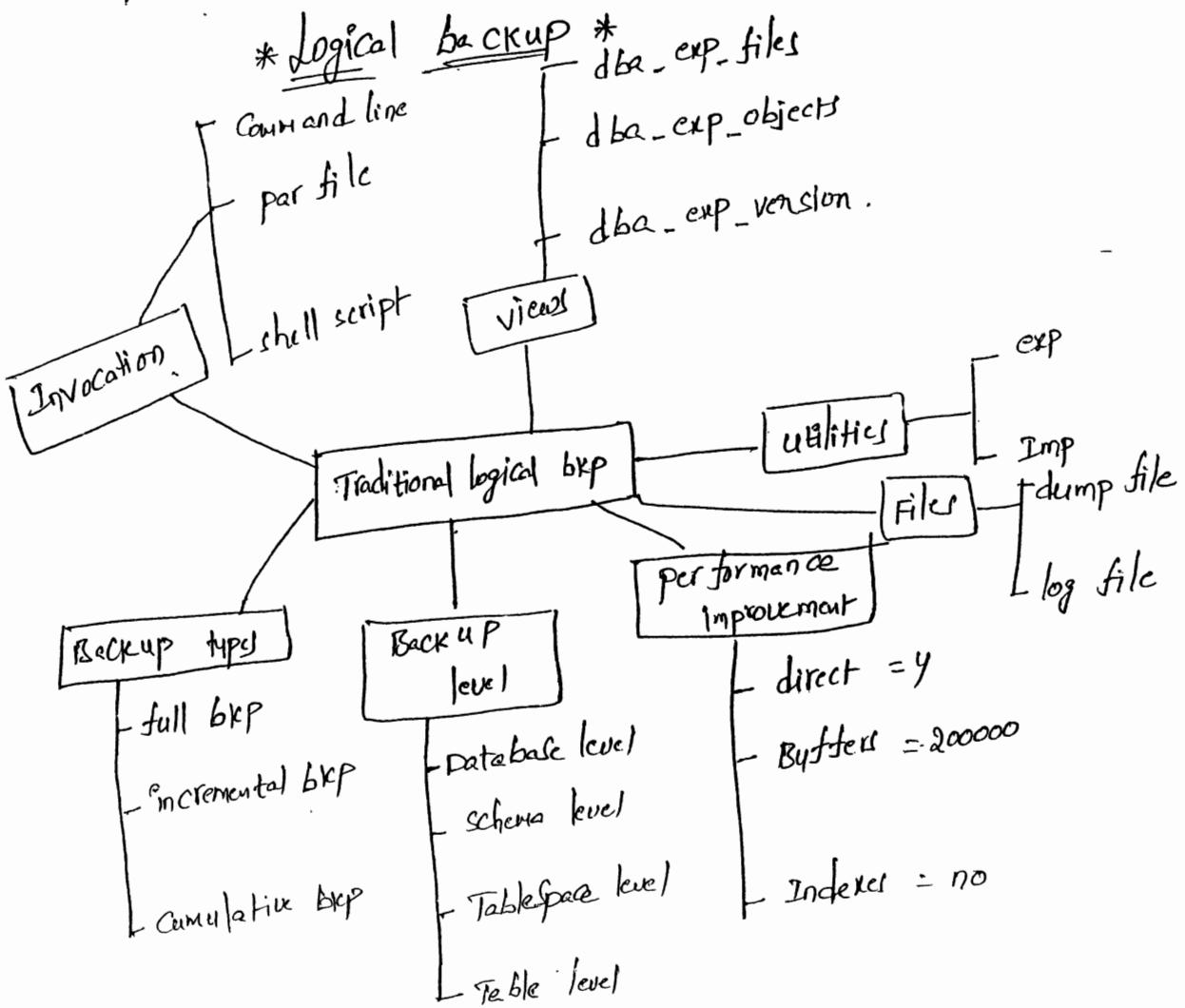
\$ tar -cvf oraInventory.tar /home/oracle/orainventory/ (to copy file)

\$ tar -xvf oraInventory.tar (to extract file)

- ② Take the backup of oracle binaries:

- ③ Take the backup of oracle database:

## ~~Performance & monitoring &~~



- We don't consider logical backup as the main backup option. since, we don't have the concept of Recovery.
- Logical backup's are more useful in case of database upgradations, database migrations
- Logical backups are platform independent.
- Migration means moving from one platform to another platform (Windows to LINUX).
- Upgradation means moving from one version to another version (11.2.0.2 to 11.2.0.3)
- Logical backups are more useful in case of schema level refreshes.
- \* → Logical backup is also useful in case of database re-org, schema level re-org & table level re-org.
- Files generated by logical backup are (1) dump file (2) log file
- Dump file is partial binary & partial text in nature.
- \* → Recommended not to edit the dump file either with notepad (or) vi editor. If there is any requirement take the backup & edit the file.
- Log file shows the progress of the Job. It contains warning messages, error messages that are encountered during the progress of the Job.
- By default, traditional logical backup is conventional (goes through SQL processing layer (SGA)), we can also bypass by specifying "DIRECT=Y"
- By default, an user can take the backup of his own objects.
- An user also can take the backup of entire database provided, he has the role "exp\_full\_database". In the same way in order to import the full database dump file, user should have imp\_full\_database role.
- \* → During import operations, some times we get error messages related to undb. To overcome this specify "Commit =Y"
- "Commit =Y" loads array of records and issues Commit frequently

→ To suppress DDL related error messages (Object already exist) specify  
" ignore = y "

→ Inorder to invoke exp (or) imp , database should be up & running.

→ " Consistent = y " internally issues set transaction read-only & performs  
the operation.

→ we can also take incremental backup using traditional logical backup.

\*→ In case of incremental backup it takes the backup of only modified  
objects.. the biggest disadvantage is, even if a single record is  
modified , it considers the entire object.

→ the different levels of incrementals are (i) Inc type = complete  
(ii) Inc type = incremental  
(iii) Inc type = cumulative

→ Inc type = complete is the base backup for next incremental backup.

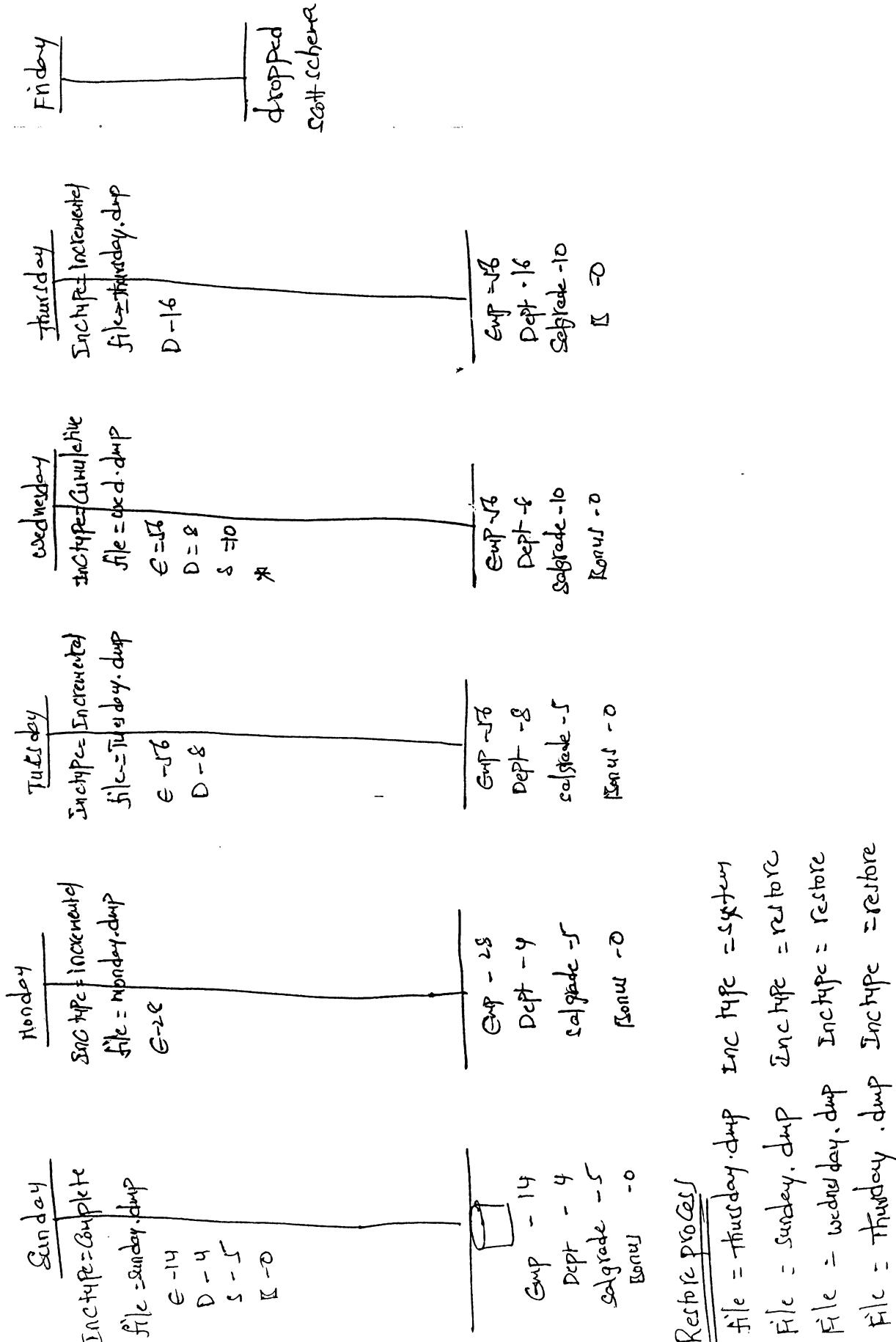
→ Inc type = complete takes the backup of entire database and it also  
updates the data dictionary tables and this information is required for  
next incremental backup.

→ Inc type = complete incremental takes the backup of all modified objects  
till the last incremental (or) cumulative (or) complete whichever appears  
first.

→ Inc type = cumulative takes the backup of all modified objects till the  
last cumulative (or) complete whichever appears first.

→ Inc type = system restores the most recent version of system objects but not  
user objects & user data.

→ Inc type = restore restores user objects & user data but not system objects



### Restore process

- (1) file = Thursday.dump IncrType = system
- (2) File = Sunday.dump IncrType = restore
- (3) File = Wednesday.dump IncrType = restore
- (4) File = Friday.dump IncrType = restore

## Data pump :-

- Data pump is the new technology introduced in 10g version of oracle
- Utilities of Data pump are (1) expdp (2) impdp
- Data pump is more useful in case of database upgradations, migrations & schema level refreshes
- Data pump is much faster compared to traditional logical backup, since, it uses the API's (Application program interfaces) like 'dbms\_datapump' & 'dbms\_metadata'. (packages)

\*→ The pre-requisite for datapump is we need to create a directory at oracle level as well as 'o/s' level

\*→ Data pump is a server side utility which means it generates the dumpfile at the serverside irrespective of the location from where we initiate the job.

→ A user can take the backup of his own objects provided user has read, write privilege on the directory

→ The biggest advantage of datapump is we can detach the job whenever there is a performance issue & we can attach the job at a later point of time.

\*→ Once we initiate datapump job, a master control process gets invoked and it creates a table by the job name in the user schema who ever has initiated the job.

\*→ Master control process keeps track of the status of the job in the table that was created

→ Once we re-initiate the stopped job oracle starts the job from the point where it has stopped

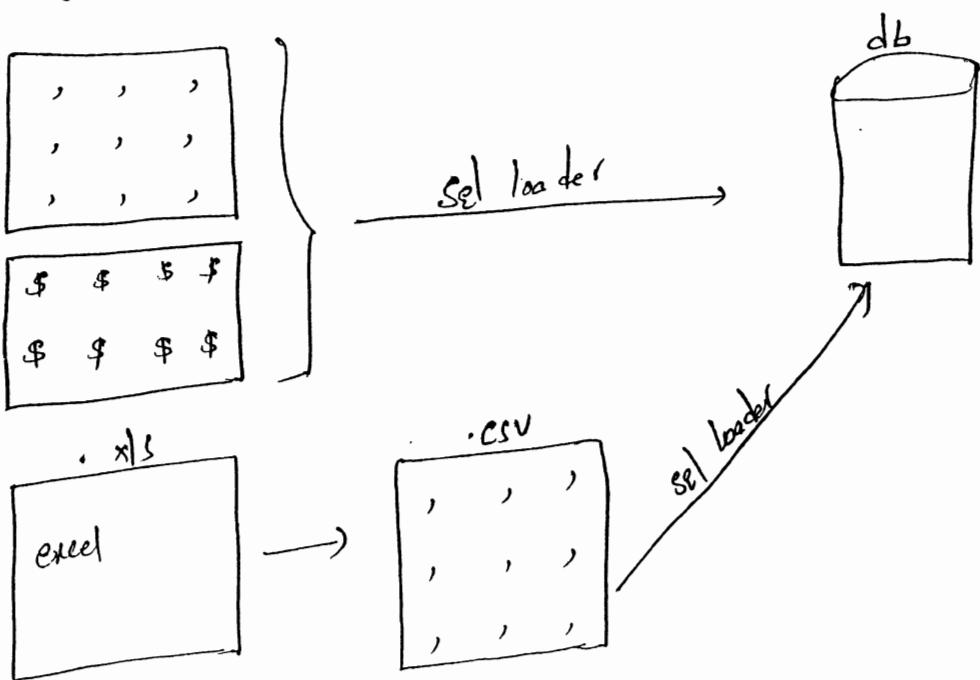
→ Once the job is completed, table gets dropped automatically

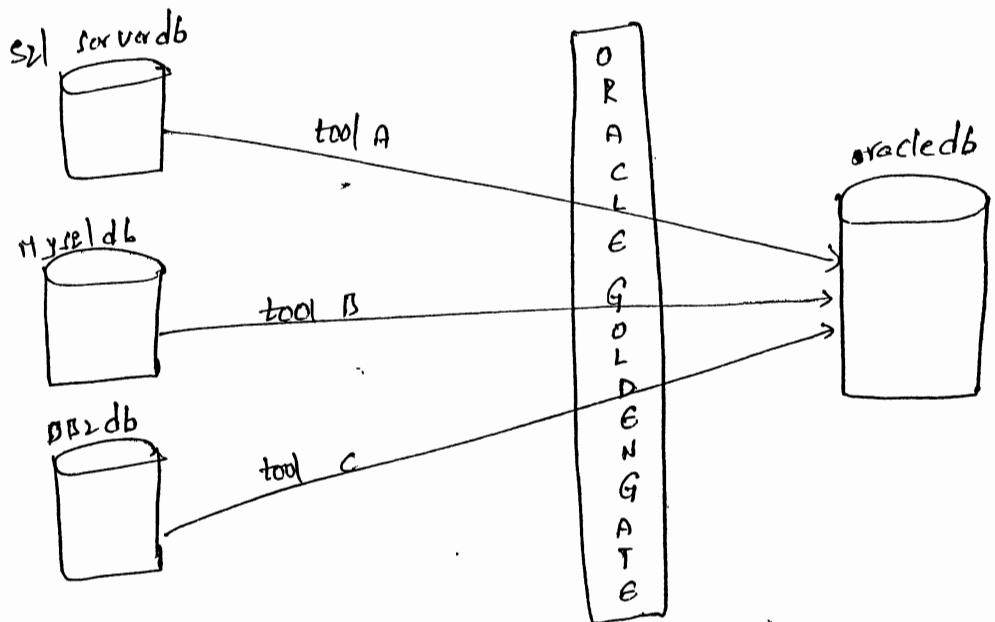
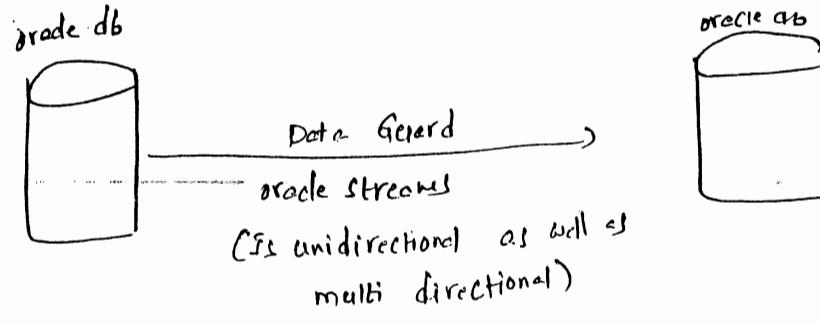
→ By default datapump is sequential. We can also initiate parallelly by specifying parallel parameter.

- we can also take the backup of only metadata if required with the help of Content parameter.
- we can also estimate the space with the help of estimate parameter  
(estimate = blocks/statistics)
- we can also take the backup using db-links with the help of the parameter "network-link"
- In case of datapump, if u specify the same name for the dump file, job fails with the error message : (file already exists)
- Recommended to specify a job name for every datapump job.
- "renap-schema" is equivalent to from user & to user parameters of traditional logical backup.
- we can also exclude some objects with the help of exclude parameter.

### Sql loader :-

- Sql loader is proprietary of oracle and it is used to load the data from flat files to oracle tables

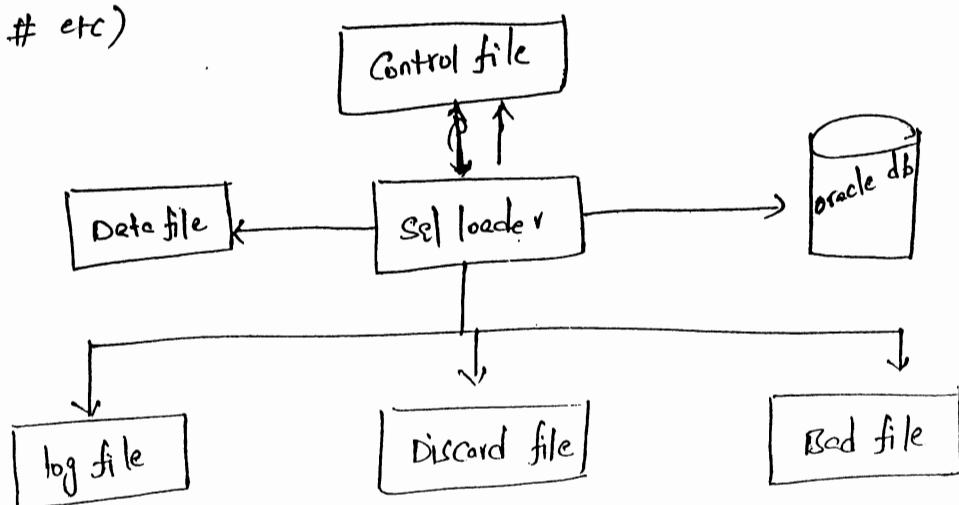




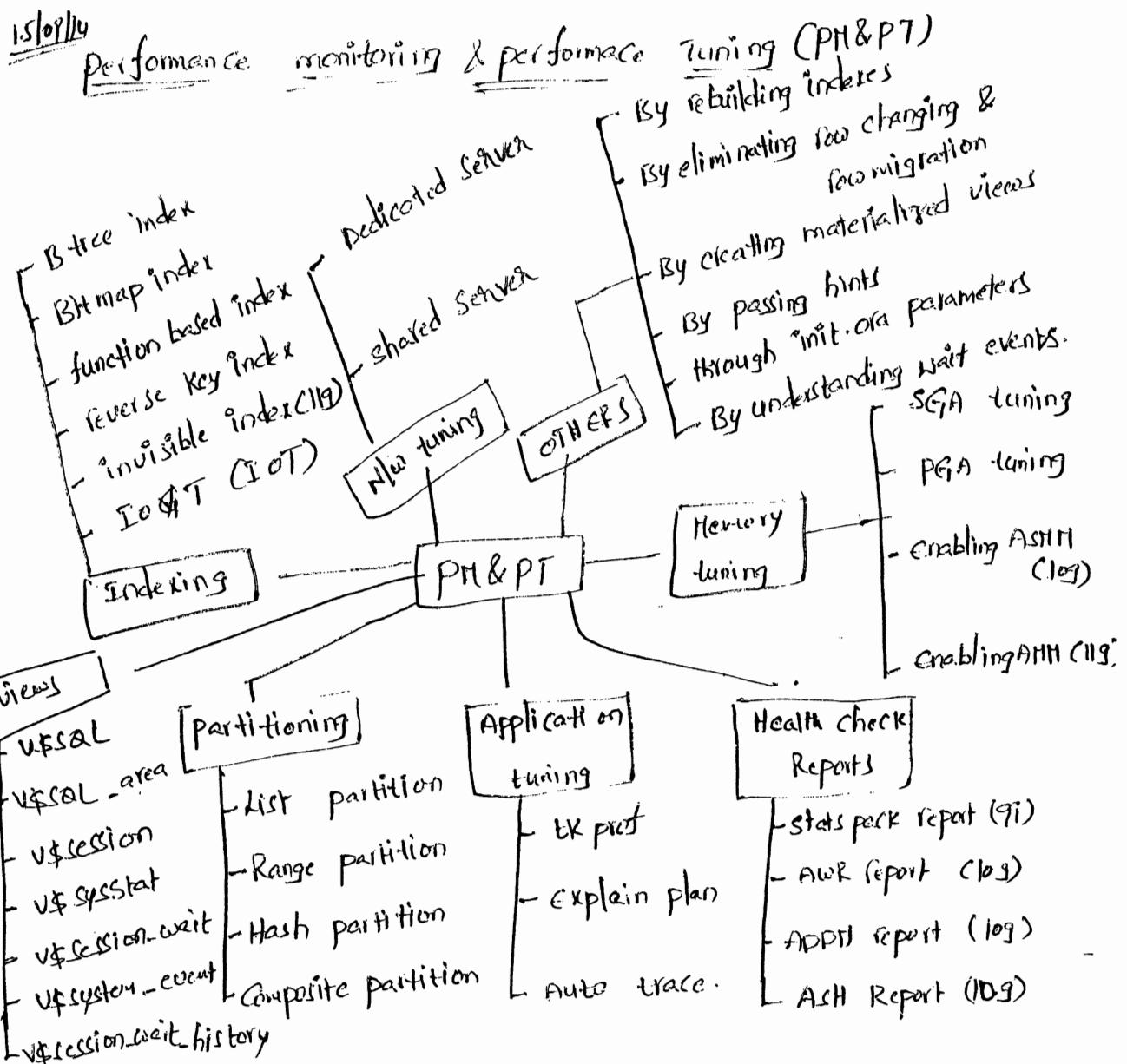
→ \* "oracle golden gate" is proprietary of oracle and it is a heterogeneous data replication tool.

→ Golden gate can replicate across heterogeneous environments.

→ sql loader can load comma separated values, delimited by special characters (\$, # etc)



- Control file describes how the data is in datafile and it also specifies into which tables SQL loader has to load the data.
- Data file contains actual data that needs to be loaded into oracle database.
- Log file shows the progress of the Job.
- Discard file contains the records that are rejected by SQL loader (Integrity constraint violations)



- Tuning starts at design phase.
- If the design is not proper it leads to redundancy, updation problems as well as deletion problems
- During the design phase, database architect is going to design a logical model (Entity relationship diagrams)
- Logical model will be converted into the physical model by using some third party data modeling tools like "ERWIN" data modeler.
- During the design phase, database architect will use a technique called normalization (1NF, 2NF, 3NF, BCNF)

- \* → Normalization is nothing but splitting a huge table into multiple entities.
- In most of the application's development environment, Data base architect will go upto 3rd normal form.
- As the level of normalization increases, we make it in more tables which also degrades performance.
- If n tables are involved in a query, we come across n-1 joining conditions
- Denormalization is nothing but combining multiple entities in to few entities.
- \* → In the industry, we depend on some 3rd party data modeling tools to convert logical into physical model (CERCOIN tool)
- We use normalization in case of OLTP systems & De-normalization in case of OLAP systems.
- The process of converting the logical model into physical model is called forward engineering.
- The process of converting the physical model into logical model is called reverse engineering.
- \* → The process of splitting an entity (table) into multiple rows entities is called Memory tuning :- Normalization
- The two memory structures of oracle are ① SGA ② PGA
- Memory tuning is one of the important aspect for a DBA
- If the memory is undersized (or) if the memory is oversized, it degrades the performance.
- If the memory is undersized, it degrades the performance because of, more no. of I/O's and excessive hard parsing.
- Prior to q1, memory tuning is a tough task for DBA because, all sub-components parameters are static in nature.

- In version 9i, Oracle has introduced dynamic parameters where we can increase the value of the parameter, while instance is up & running. (DBA intervention is required)
- In 10g, Oracle has introduced automatic shared memory management (ASMM) by introducing the parameter "SGA-Target"
- In order to enable ASMM, we need to set SGA-target to a non-zero value. (to disable set it to zero)
- once we enable ASMM, the following parameters are considered as auto tunable and not necessary to set explicitly in the init.ora.
  - ① db-cache-size
  - ② shared-pool-size
  - ③ large-pool-size
  - ④ Java-pool-size
  - ⑤ streams-pool-size
- the following parameters are not considered as auto tunable
  - ① log-buffer
  - ② db-nk-cache-size
  - ③ db-keep-cache-size
  - ④ db-recycle-cache-size  
(MAN background process gets enabled)
- once we enable ASMM, Oracle dynamically expands and shrinks the sub components depending on the work load.
- SGA-max-size represents the maximum allowable size for SGA, and
  - it is a static parameter
  - SGA-target, <sup>dynamic</sup> parameter value can't go beyond SGA.max.size

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

16/09/14

\* once, we enable ASMM, MMAN background process continuously monitors the SGA Sub Components and dynamically shrinks & expands the sub-components depending on the work load.

→ In version 9i, oracle has introduced "PGA-Aggregate-target" parameter to allocate a consolidate PGA for all users.

→ In version 11g, oracle has introduced Automatic Memory Management (AMM) by introducing the parameters "memory\_max\_target" & memory\_target.

→ In order to enable AMM, set memory\_target to a non-zero value

→ Memory\_max\_target is a static parameter.

→ Memory\_target is a dynamic parameter.

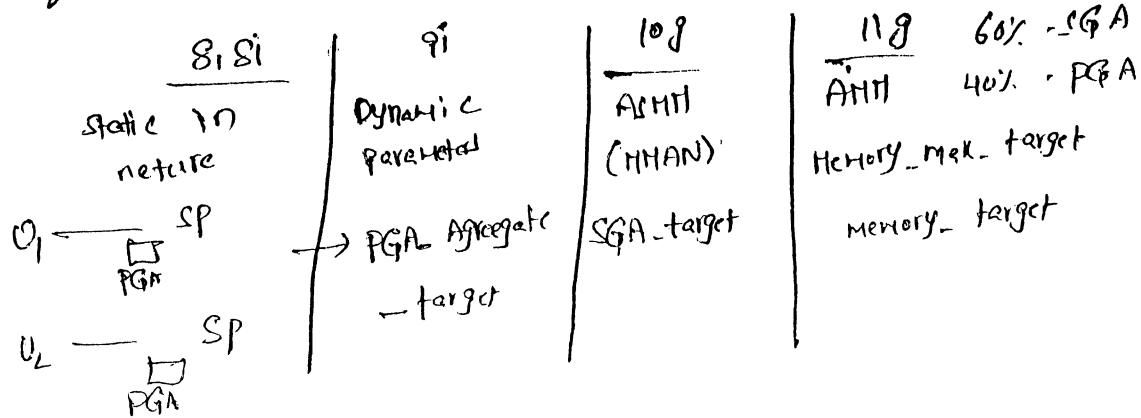
→ "Memory\_target" parameter value will not go beyond memory\_max\_target.

→ once we enable "AMM", oracle allocates 60% of memory for SGA & 40% of memory for PGA.

→ After enabling ASMM, if u set an auto tunable parameter, oracle considers the parameter value & oracle treats it as lower limit and if there is a requirement, it will increase the value for that.

\* Ideally oracle recommends to have the hit ratios  $\geq 90\%$ .

→ We can identify the hit ratios in AWR report (or) by writing some sql statements on data dictionary views



→ If u have sufficient amount of memory, allocate 40% of memory for SGA

### \* \* Indexing & partitioning:-

→ Index is an optional database structure generally created to improve the performance of select statements.

→ Index is the space demanding object.

\* \* → Index improves the performance of select statements and at the same time index also degrades the performance of DML operation (I,UD)

→ Recommended to create index on huge tables.

\* \* → Recommended to create index on columns which are most frequently used in the where-clause of a select statement.

→ The traditional index of oracle is B-tree index (Balanced-tree)

→ In the industry, generally developers are going to create the indexes if they have not created properly it degrades the DML performance.

→ As a DBA we need to monitor for the usage of indexes and if we notice any unused indexes just take a decision of dropping them.

EMP  
EMP\_index\_empno

> Create index EMP\_index\_empno on Emp(empno);

> Alter index EMP\_index\_empno monitoring usage;

> Select table\_name, index\_name, monitoring, used from V\$object\_usage;

### Structure of index:-

Key Column	key Column length	key Column value	Row id
			6363

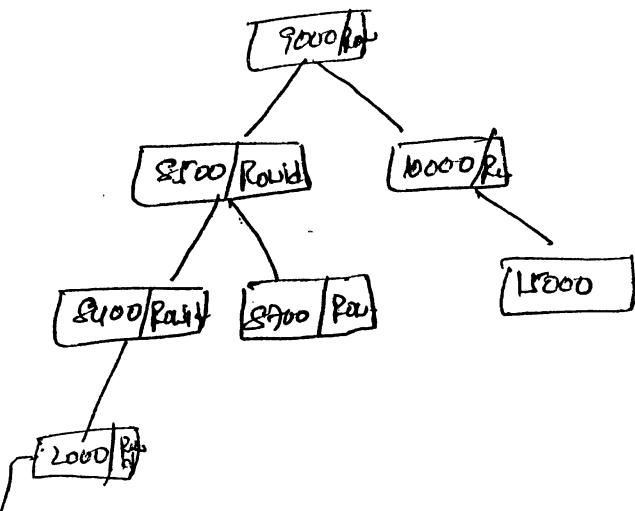
18 digit Hexadecimal in format.

> Create

> select rownum, rowid, empno, ename from emp;

- the column on which we create an index is called key column.
- we can also create indexes on morethan one column, such type of indexes are called composite indexes. (Max. 32 columns)
- Index minimizes the search criteria by almost 50%.

EMP		
EMPNO	ENAME	SAL
1	Ram	9000
2	Shaym	10000
3	Argun	8500
4	Laxmi	8700
5	Kalyan	8400
6	Raja	15000
7	Seipal	2000



> select \* from emp where sal = 2000;

- Different types of indexes are ① B-tree index ② Bit-map index  
③ Unique index ④ Reverse key index ⑤ Function based index  
⑥ Composite index ⑦ Index organization table.

17/09/13 → Whenever we have a sequence of integer values in an index

column, create reverse key index. So, that the data will be

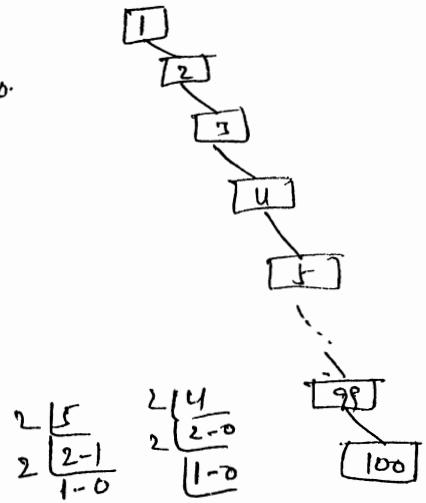
converted into binary format and will get a balanced tree.

> Create index customer\_details\_idx\_cno on customer\_details(cno) reverse;

**Customer-details\_idx\_cno**

**Customer details**

Cno	Cname	Bill Address	Bill Amount	Categno
1	Xyz	Hyd	5000	1-14
2				
3				
4				
5				
6				



> Create sequence cust\_details\_sq\_cno

start with 1

increment by 1;

> insert into customer\_details (Cno) values (sequence.nextval);

> select \* from customer\_details where cno = 99;

\* create bitmap index on low cardinality columns.

A column with repeated values is called a low cardinality column.

ex:- dept no in emp

> create bitmap index emp\_idx\_deptno on emp(deptno);

> Create index customer\_details\_idx\_cno

on customer\_details (Cno) reverse;

**EMP**

Empno	ename	deptno
1	Ram	10
2	Shyam	10
3	Bheem	20
4	Ravi	30
5	Antu	40
6	Ramesh	10
7	Balu	10

**Bitmap Index**

	10	20	30	40
1	1	0	0	0
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1
6	1	0	0	0
7	1	0	0	0

### Customer\_Details

Cno	Fname	Lname
1	Scott	Tiger
2	Ravi	Verma
3	Richard	Kumar
4	Sree	Vass
5	Rida	Ranga
6	Yuvraj	Singh
7	Raja	Sephar

✓ create index customer\_details\_idx\_fname  
on customer\_details (fname);

✓ create index customer\_details\_idx\_lname  
on customer\_details (lname);

✓ create index customer\_details\_idx\_fname\_lname  
on customer\_details (fname, lname);

> select \* from customer\_details where fname = 'sree';

> select \* from customer\_details where lname = 'Ranga';

> select \* from customer\_details where fname = 'yuvraj' and lname = singh;

→ If u are using more then one column in the where clause of a select statement then recommended to create Composite index.

→ In older versions of oracle, oracle considers Composite index if we use the leading edge column in the where clause.

\*→ In modern versions of oracle, oracle considers Composite index even if we don't use the leading edge column in the where clause. such type of feature is called "skip-scan index".

\*→ whenever we use a function in the where clause, oracle suppresses the usage of index. In this scenario recommended to create function based index.

> create index customer\_details\_idx\_fname  
on customer\_details (upper(fname));

> select \* from customer\_details where upper(fname) = 'Raja';

\*→ We can maximize the index creation speed by enabling <sup>tablespace with</sup> no logging (or) by increasing the degree of parallelism (or) by turning off rec-generation.

> create index emp\_indx\_sal  
on Emp (sal)  
no logging

tablespace hrms\_app\_indx\_ts;  
(or)

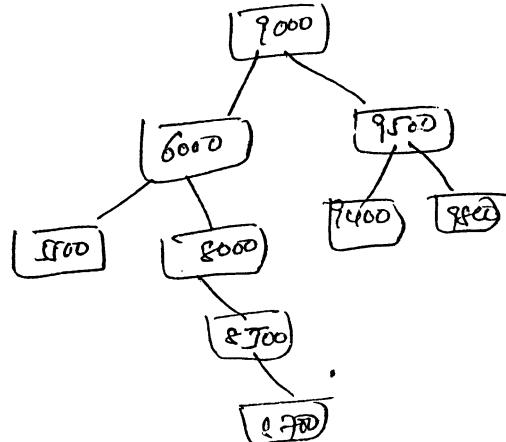
> create index emp\_indx\_sa1  
on Emp (sal)

parallel 4

tablespace hrms\_app\_indx\_ts;

→

EMP		
Emp no	ename	sal
1	Ram	9000
2	Ishreen	6000
3	xyz	8000
4	Rahcen	8500
5	Ramu	8700
	Sachin	4500
	John	9800
	Blake	9400
	Rosa	5500



MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

> delete from Emp where sal = 8000;

> delete from Emp where sal = 8500;

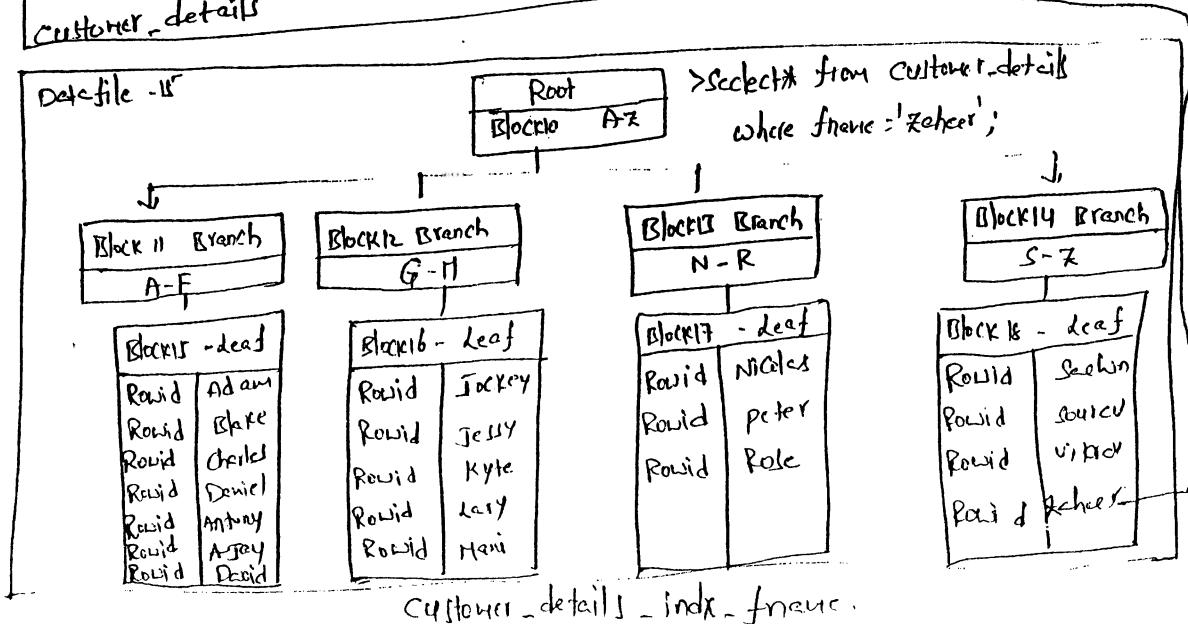
\*\* We can reclaim the unused space in an index by re-building indexes  
(or) by shrinking the space.

> alter index emp\_indx\_sal  
rebuild online;

> alter index emp\_indx\_sa1  
shrink space;

→ We can also Compress the usage of indexes.

> create index customerdetails\_indx\_finance\_Income on customerdetails(finance,income) Compress 2;



→ General guidelines to create indexes :-

- ① Define primary key constraint, wherever possible which results in index creation by default.
- ② Enable unique key constraint which results in unique index creation by default.
- ③ Create indexes on columns which are most frequently used in the where clause of a select statement.
- ④ Explicitly create indexes on foreign key columns
- ⑤ Create ~~traditional~~ <sup>traditional</sup> index (B-Tree) until unless there is a solid reason ~~to~~ to create a different type of index
- ⑥ Dedicate a separate tablespace for all indexes
- ⑦ Create Bit map index on low cardinality columns
- ⑧ Create Reverse Key index whenever we have sequence of integer values.
- ⑨ Create function based index whenever we use function in the where-clause
- ⑩ Occasionally perform some maintenance activity on the index structures. (rebuilding indexes & shrinking the space)

⑪ Understanding oracle optimizer, Data access methods & joining techniques:-

→ Optimizer is an intelligent component which determines the execution plan of every SQL statement.

→ There are two types of optimizers  
① Rule Based optimizer  
② Cost based optimizer.

→ Prior to 9i, the default optimizer is rule based.

→ Rule based optimizer totally depends on the pre-defined rules of Oracle which may not give efficient execution plan.

- From version 9i, the default optimizer is cost-based.
- In 11g, Rule based optimizer has become obsolete (no more).
- The parameter which determines the optimizer is "optimizer\_mode".

$\text{optimizer\_mode} = \text{ALL\_Rows}/$

$\text{First\_Rows}(n)/$   
 $\text{choose}/$   
 $\text{Rule}.$

- ALL\_Rows/ is meant for better true put.
- First\_Rows(n)/ is meant for better response time.
- "Rule" means oracle considers rule based optimizer.
- "choose" means oracle considers cost-based optimizer. If statistics are available otherwise it considers rule based optimizer.
- Cost based optimizer totally depends on the availability of statistics.
- If statistics are not available (or) if statistics are stale (outdated), optimizer may not give efficient execution plan, which may degrade the performance of select statement.
- Prior to 10g, as a DBA we use to gather optimizer statistics during half peak (non-business hours) either by using analyze command or by using dbms\_stats package.

> analyze table emp compute statistics;

> analyze table emp estimate statistics;

> exec dbms\_stats.gather\_table\_stats

(ownname => 'SCOTT',

table\_name => 'EMP', cascade => TRUE);

- Recommended to gather statistics using dbms\_stats package
- statistics are nothing but the exact & accurate information about database objects like no. of rows, average row length, indexes created etc..
- From log, by default oracle gathers optimizer statistics during the maintenance window i.e., 10PM - 6AM during week days and all day during weekends.
- Even though optimizer oracle gathers optimizer statistics by default, we also need to gather the statistics manually. depending on the requirements.
- oracle gathers optimizer statistics with the help of a scheduled job i.e., "GATHER\_STATS\_JOB"
- we can also disable automatic statistics gathering by executing the following package.  
`>exec dbms_scheduler.disable('GATHER_STATS_JOB');`
- oracle gathers the statistics if 10% of data is modified in an object.
- optimizer considers index if we are selecting less than 10% of data.
- optimizer determines execution plans and these execution plans will be stored in library cache.

### optimizer Date Access methods & joining techniques:-

- the following are the different types of data access methods used by the optimizer.
  - (1) Full table scan (FTS)
  - (2) Index look up
  - (3) Row id
- "Index look up" can be
  - (1) Index unique scan
  - (2) Index range Scan
  - (3) Index full Scan
  - (4) Index fast full Scan

- whatever the column values that we are retrieving are available in index then it is called index unique scan.
- In case of Index full scan, oracle scans the root, branch and the leaf nodes using single block read.
- Index fast full scan is an alternative to index full scan. In this case, oracle scans only the leaf nodes by skipping the root & the branches. by using multi block reads.
- optimizer considers index fast full scan if the entire data i.e., required is available in the index itself.

> select count(fname) from EMP;

- The different types of optimizer joining techniques are
  - ① Nested-loop Join
  - ② Sort-Merge Join
  - ③ Hash Join
  - ④ Partitioned product.

### ① Nested loop join:-

- optimizer considers nested loop join if you are joining two small tables and if the data is very less.

→ Loop within a loop is called nested loop.

- For every record in the outer loop, oracle searches for a record in the inner loop

> select e.empno, e.empname, e.sal, d.deptno, d.dname from EMP e,dept d  
where e.deptno = d.deptno;

result:

for i in (select \* from EMP)

loop

for j in (select \* from dept where i.deptno = j.deptno);

loop

display the result;

## ② Sort-merge Join:-

→ optimizer considers sort-merge join if the volume of data is moderate

## ③ Hash Join:-

→ optimizer considers hash join if we are joining a huge table with the smaller table.

→ Hash join works in two phases ① build phase ② probe phase.

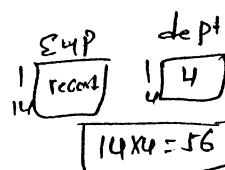
→ In the build phase, optimizer creates a hash table in the hash memory by considering the smallest table.

→ In the probe phase, for every record in the bigger table, it searches for a record in the hash table.

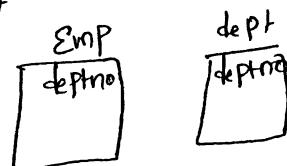
## ④ Cartesian product:-

→ If there is no joining condition, oracle evaluates to Cartesian product.

> Select ename, dname, sal from EMP, dept;



> Select ename, dname, deptno, sal from EMP, dept  
invalid statement  
where emp.deptno = dept.deptno;  
error: ambiguity error



> Select e.empno, e.ename, e.deptno, d.dname from  
EMP e, dept d where

→ It is always recommended to use the alias name for the tables, this minimizes the length of the code and also the parsing time.

### Application Tuning:-

→ In order to fine tune the application, first we need to understand the execution plan given by the optimizer.

→ We can generate the execution plan by the following options

① Explain plan

② TK prof

③ Auto trace.

→ Explain plan is a pre-execution utility

→ Explain plan can be used either by a developer (or) by DBA to generate the execution plan.

→ Execution plan generated by the explain plan contains optimizer data access methods, joining techniques and the cost involved in executing the statements.

→ The primary goal of a developer (or) a DBA is to minimize the cost involved in executing the statements.

→ Prior to log, in order to use explain plan, we need to create "plan\_table" by executing the following script.

```
SQL> @ $ORACLE_HOME/rdbms/admin/utlxplan.sql
```

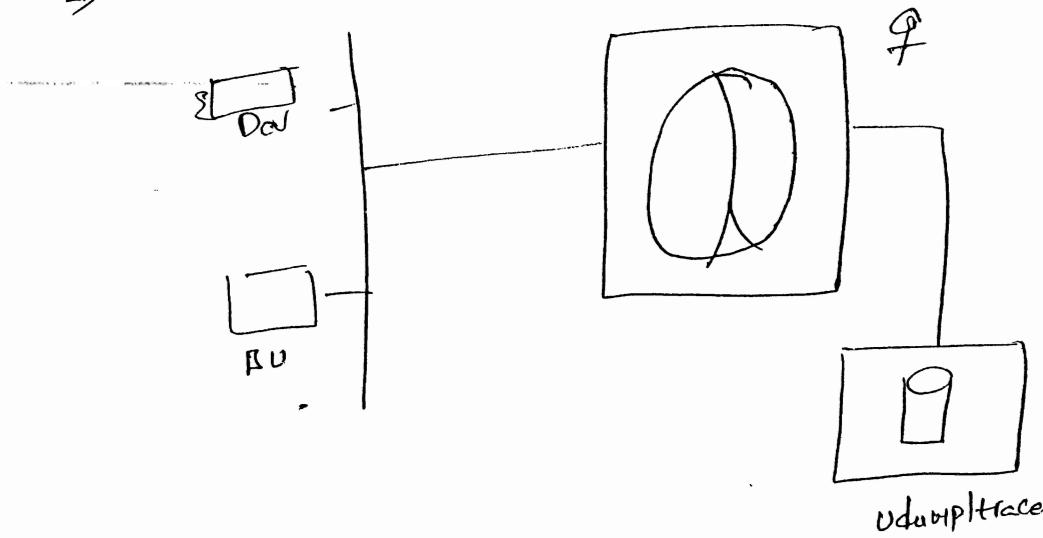
→ Plan\_table contains the execution plans given by the optimizer.

→ We can retrieve the execution plans from plan\_table by executing the following script.

```
SQL> @ $ORACLE_HOME/rdbms/admin/utlxpls.sql
```

→ tkprof is a post execution utility

→



→ whenever, business user complaints about the slowness of the session , we can enable the trace of the user session by executing the following package.

SQL> exec dbms\_system.set\_sql\_trace\_in\_session (sid, serial#, true);

→ once, we enable the trace of user session , it will generate a trace file in the udumptrace directory.

→ the trace file generated in the udumptrace directory is in raw format and is not in proper understandable format.

→ To convert the trace file into proper readable format, we use tkprof utility.

→ Trace file contains the logical reads , physical reads , Data access methods, joining techniques, time spent on parsing , execution etc..

→ we can identify session specific trace files by joining v\$session and v\$process views.

→ we can also enable the trace at instance level by setting the following

init.ora parameter.

SQL\_Trace : True

- If u enable the trace at instance level, it will generate a trace file for every user session
  - Enabling trace at instance level is not recommended
  - ① → Auto trace can be used either by developer or DBA to generate the execution plan
  - A developer can use auto trace provided dba grants plus trace role
  - As a DBA we can create plustrace role by executing the following script.
- SQL> @ \$ORACLE\_HOME/sqlplus/admin/plustrce.sql.

### Understanding wait events and minimizing System Contention:-

- The crucial performance indicator of an application is the response time.
- Response time = processing time + Wait time.
- If we minimize the wait time, application gives better performance.
- There are hundred's of wait events in oracle but hardly we concentrate on 10 to 12 wait events to minimize the system contention.
- All wait events are categorized into different clauses like
  - ① Application (related to locks)
  - ② user I/O (tb file sequential read, db files)
  - ③ Commit (log file sync)
  - ④ system I/O
  - ⑤ idle
  - ⑥ Concurrency (RAC related wait events)
  - ⑦ Network
  - ⑧ Administrative

### views:-

- ① v\$session-wait
- ② v\$session-event
- ③ v\$system-wait-class
- ④ v\$session-wait-history
- ⑤ v\$system-event
- ⑥ v\$system-wait-

## Result Cache

→ In 11g, oracle has introduced a new memory sub component called Result cache  
To cash the results of most frequently executed SQL statements.

→ In order to enable result cache oracle has introduced some parameters

① Result\_cache\_mode = Manual / Auto / Force (default is Manual)

- If it is manual, oracle will not cache the results of the SQL statements

→ In order to cache the results, we need to pass "Result-cache hint"  
along with the SQL statements

> select /\*+ result-cache \*/

→ In case of Auto, oracle caches the results based on frequency of execution, cost of the execution etc.

→ In case of Force, oracle caches the results of every SQL statements.

→ If Automatic memory management (AMM) is enabled, result cache occupies 0.25% of "memory-target" parameter value

→ If Automatic shared memory management (ASMM) is enabled, result cache occupies 0.5% of "SGA-Target" parameter value.

→ If shared pool is set manually, result cache occupies 1% of shared pool.

② Result-cache\_max\_size

③ Result-cache\_max\_result = 5 (default value)

→ '5' means a single query result maximum it can occupy 5% of Result Cache.

## Row chaining & Row Migration:-

- Generally row chaining & row migration occurs in case of insert, and update.
- If oracle is unable to accommodate the entire row in a single block then oracle stores partial data in one block & partial data in some other block.
- Generally row migration occurs in case of update.
- We can eliminate row chaining & row migration
  - ① by exporting the table , dropping the table & importing the table.
  - ② By moving the table to a tablespace of bigger block size.

SCOTT> desc user\_tables

SCOTT> select table\_name, num\_rows, last\_analyzed, chain\_count from user\_tables;

SCOTT> exec dbms\_stats.gather\_schema\_stats (ownname=>'SCOTT');

SCOTT> select table\_name, num\_rows, last\_analyzed, chain\_count from user\_tables;

scott> @\\$ORACLE\_HOME/rdbms/admin/utlchained

scott> select \* from tab;

scott> desc chained\_rows;

scott> analyze table emp list chained rows;

scott> select count(\*) from chained\_rows where TABLE\_NAME='EMP';

Note:-

If u move the table to a tablespace of bigger block size , indexes becomes invalid. In this case , we need to rebuild the indexes manually.

## Database Health Check Reports

- ① AWR (Automatic Workload Repository)
- ② ADDM (Additonal Diagnostic Monitor)
- ③ ASH (Active Session History)
- In 80, Oracle has introduced utlbstat & utltestat
- In 81, Oracle has introduced a primitive version of Statspack report and enhanced the capabilities in version 91.
- From 10g, Statspack has been evolved to AWR
- In order to generate database health check report, we require database performance statistics which we call them as snapshots.
- In 9i, we use to gather the performance statistics manually by installing Statspack & by executing the following package.

```
SQL> exec statspack.snap();
```

- From 10g, by default Oracle gathers performance statistics with the help of MMON (Memory monitor) background process & stores in SYSAUX table space.
- The default frequency is one hour & the default retention is 7 days
- We can also modify the default frequency & the default retention by executing the following package.

```
SQL> exec dbms_workload_repository.modify_snapshot_settings  
(Retention => 2880,  
interval => 120);
```

- \*→ In order to generate the health check report, we require 2 snapshot's between these 2 snapshot's, if at all if the instance is down we can't generate the report.

→ To know the list of Snap id's in the SYSAUX table space

SQL> select \* from dba\_hist\_snapshots;

→ once the retention period expires, by default oracle purges (deletes) the snapshots.

→ we can also drop the snap id's manually by executing the following package.

SQL> exec dbms\_workload\_repository.drop\_snapshot\_range

(low\_snap\_id => value,

high\_snap\_id => value);

→ Baseline is a collection of 2 snap id's that represents a specific period of usage.

→ once baselines have created, they can be used to compare current performance against similar period in the past.

→ we can create a baseline by executing the following package.

SQL> exec dbms\_workload\_repository.create\_baseline

(start\_snap\_id => value,

end\_snap\_id => value,

baseline\_name => 31<sup>st</sup> March 2010 baseline);

→ once we create a baseline, the snap id's associated with the baseline will not be purged even if the retention period expires.

→ Snap id's associated with the baseline will be dropped automatically once we drop the baseline

SQL> exec dbms\_repository.workload\_repository.drop\_baseline

(baseline\_name => 31<sup>st</sup> March 2010 baseline,

cascade => TRUE);

→ 'Cascade = TRUE' means oracle deleted the snap id's that are associated with the baseline.

→ To know the baselines that are created

SQL> select \* from dba\_hist\_baseline;

### Interpreting AWR Report:-

Note:- → we can generate a text based AWR report (or) html based AWR report by executing "Awrpti.sql".

→ In RAC system, AWR report is specific to instance (Awrpti.sql).

\*→ whenever we are analyzing AWR report, for the same period of time interval generate ADDM report and implement the solutions given by automatic database diagnostic monitor (ADDM).

→ Elapsed time represents the time interval between the begin snap id & the end snap id.

\*→ If DB time is much greater than elapsed time, we need to drill down further to understand the wait events within in the system.

→ To identify whether redo-buffer cache is properly sized (log-buffer), execute the following query. for multiple no.of times during peak load.

\*→ If "redo buffer allocation retries" value is increasing continuously then take a decision of increasing the parameter value i.e., log-buffer.

SQL> select name, value from v\$sysstat where name='redo buffer allocation

retries';

### load profile section:-

→ From this section, we can understand the logical reads, physical reads,

total no.of parses, hard parses and transactions per second (TPS)

### Instance sufficiency percentage:-

→ From this section, we can identify the hit ratios of sub-components

(Data base buffer cache hit ratio, library cache hit ratio, In-Memory sort % (PGA))

and soft parse %.)

→ execute to parse %. should be low

### Top 5 wait events:-

→ From this section, we can understand the top 5 wait events that are occurring within the database.

#### ① db file scatter read

→ This wait event indicates many full table scans

→ Identify the missing indexes for top 10 SQL statements

→ Tune the code wherever possible & try to cache small tables.

#### ② DB file sequential read

→ This wait event indicates many index scans which is quite natural.

→ Tune the Joining Conditions wherever possible.

#### ③ Free buffer

→ Increase the size of db\_cache\_size parameter.

→ If ASMM is enabled (or) AHH is enabled, check whether proper values are set for the parameters SGA-Target & Memory-target.

#### ④ Log file sync

→ Generally we come across this wait event if the Commit frequency is more.

→ Commit more records at a time and check whether online Redolog files are placed on faster disks.

#### ⑤ Log file switch :-

→ Check whether archive destination is full

→ Increase the no. of DB writers.

→ Add one more Redolog group.

## SQL Statistics:-

- From this section, we can identify the resource intensive SQL statements which are consuming significant amount of CPU time.
- (SQL ordered by CPU time) →
- Identify top 10 SQL statements, generate the execution plans and fine tune them.

## Tablespace IO stats and File IO stats:-

- From this section, we can understand which application is having more read & write activities.

## Advisory statistics:-

### ① SGA Target Advisory

- SGA size factor "1" indicates the current value for the parameter "SGA-Target".
- Based on estimated physical reads we can identify a proper value for the parameter "sga-target".

### ② PGA memory advisory

- Size factor "1" indicates the current value for the parameter "pga-Aggregate-target".

### ③ Undo Segment Statistics:-

- From this section, we can understand how many times we got snap shot too old errors and out of space errors. Based on these we can take a decision of increasing undo size.

## Bind peaking:-

- Oracle optimizer peaks at the bind variable value and generates that we passed for the first time and generates a execution plan. This execution plan will be used for multiple executions which may degrade the performance.

- > select \* from emp where deptno = deptno; Bind ✓
- > select \* from emp where deptno=10 → Literal
- > select \* from emp where deptno = deptno, substitution variable.
- To overcome this problem, in 11g Oracle has introduced adaptive cursor sharing
- In order to enable adaptive cursor sharing, we need to set the parameter "cursor\_sharing = Force"
- \* → In case of adaptive cursor sharing, Oracle generates multiple execution plans for multiple values of substitution variables and chooses an efficient execution plan.

### Flash back data Archive:-

In flash back data archive we are going to create separate tablespace...  
In this we can store the deleted information.

```
# su - oracle
$ export ORACLE_SID=hrms
$ sqlplus / as sysdba

@syd startup
sys> select name, open_mode, log_mode from v$database;
sys> select name, status from v$instance;
sys> select name from v$logfile;
sys> create tablespace fda_ts
  & datafile '/u02/app/oracle/hrms/fda_ts01.dbf' size 100 autoextend on;
sys> create flashback archive fda1
  & tablespace fda_ts
  & retention 1 year;
```

sys create flashback archive fda2

2 tablespace fda-ts

3 retention 1 day;

sys> create flashback archive fda3

2 tablespace fda-ts

3 retention 1 month;

sys> desc dba\_flashback\_archive;

sys select owner.name, flashback\_archive\_name, flashback\_archivett, retention;

sys select owner.name, flashback\_archive\_name, flashback\_archivett;

in days from dba\_flashback\_archive;

sys grant flashback archive on fda1 to scott;

sys> conn scott/tiger

scott> select \* from tab;

scott> alter table emp flashback archive fda1;

scott> desc user\_flashback\_archive\_tables;

scott> select table\_name, flashback\_archive\_name, status, archive\_table\_name from

user\_flashback\_archive\_tables;

scott> select \* from tab;

scott> delete from emp where deptno = 20;

scott> commit;

scott> select \* from tab;

scott> set time on;

scott> select systimestamp from dual;

scott> select \* from emp as of timestamp

as to\_timestamp('27-sep-2014 16:35:10', 'dd-mon-yyyy hh24:mi:ss');

scott> select \* from tab;

scott> select \* from sys.FBA\_HISI\_75568;

scott> set time off;

scott> conn / as sysdba

sys> grant flashback archive administer to scott;

scott> conn scott/tiger

scott> create flashback archive fday

& tablespace fdct\$

2 retention 4 month;

scott> drop flashback archive fday;

scott> alter table emp no flashback archive;

scott> conn / as sysdba

sys> alter flashback archive fdal purge all;

sys> alter flashback archive fdal

2 modify retention 2 year;

sys> alter flashback archive fdal set default;

---

x

### Dirty buffer

Normally, a dirty buffer is a buffer that has been changed in memory but not yet written to disk. Oracle only has to write dirty buffers to disk - a buffer that was read into memory and not modified can be discarded from memory without the expense of going to disk.

### Free buffer exists

This event occurs mainly when a server process is trying to read a new buffer into the buffer cache but too many buffers are either pinned or dirty and thus unavailable for reuse. The session posts to DBWR then waits for DBWR to create free buffers by writing out dirty buffers to disk.

DBWR may not be keeping up with writing dirty buffers in the following situations.

- (1) the I/O system is slow.
- (2) there are resources it is waiting for, such as latches.
- (3) the buffer cache is so small that DBWR spends most of its time clearing out buffers for server processes.
- (4) the buffer cache is so big that one DBWR process is not enough to free enough buffers in the cache to satisfy requests.

### Buffer pinning

In case of consecutive access on the same buffer within the same fetch call, Oracle pins the buffer so as to visit the buffer without acquiring the cache buffers chains latch and searching the cache buffers chains. The duration of pin is fetch-call scope, so we have a decreased overhead without the pin contention.

# vi /etc/charts in ssssl  
127.0.0.1

172.16.128.150	ssss1	ssss1.oracle.com
172.16.128.151	ssss2	ssss2.oracle.com

: w8!

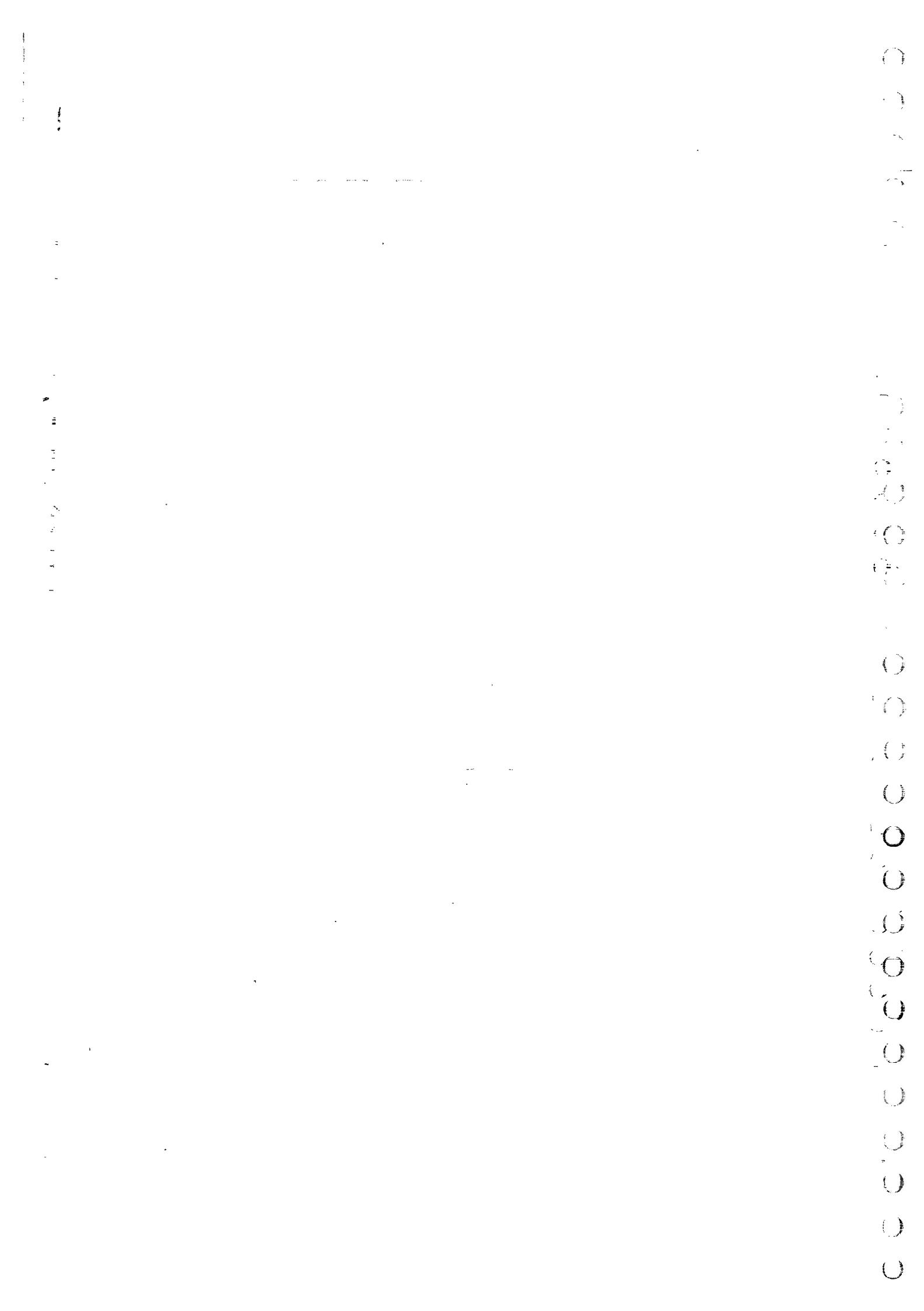
# network service restart  
# netst

IBM India

hr@vxi.in (3 years)

Rama Krishna Reddy CVXi IT Solutions  
Chinna Reddy

MANOJ ENTERPRISES  
Plot 40, Gaganthri Nagar  
Ameerpet, Hyderabad



22/08/14

## Installing Red Hat enterprise Linux AS 4 update 8 (4.8) (RHEL4.8)

# init 6 (or) reboot.

→ After re-starting or system press Enter ↵

→ click on skip , click on next → click on next → next

→ click on install RHEL AS click on next →

① Manually partition with Disk Druid → next

→ click on "new" → select '/' , size '10000' click on ok

→ click on "new" → select '/usr' (unique system resource)

The minimum size of '/usr' is 8GB.

size 10000 - MB click on OK.

→ click on "new" → select '/var' (variable) - mails & messages , size 10000 - MB → OK

→ click on "new" → '/opt' (optional) - third party media softwares , size - 1000 - MB → OK

### User-defined file systems

→ click on "new" → '/001' , 10000 - MB , ok.

→ New → '/002' , 10000 - MB , ok.

→ New → '/003' , 10000 - MB , ok.

→ New → file system type - swap , size - 4100 → ok.

→ Click on 'next' → next → click on edit → deselecting the

Configure using DHCP

Specify IP Address : 172.16.128.151

Net mask : 255.255.255.0 → OK.

→ Specify host name.

① Manually

scil.oracle.com

→ click on next → click on Continue → Continue

→ click on no firewall → click on enable selinux? Disabled

→ click on next → proceed. → next

→ select the nearest city - Asia/Kolkata. → next.

pack : oracle dba ✓

Confirm : oracle dba. ✓ → next

→ click on customize → next

→ click on everything → next → next

23/08/14

### \* \* ORACLE Installation \* \*

Pre requisites for "install oracle".

- (1) We need to create two groups install  
dba
- and one user - like oracle.

(2) Creating oracle home.

(3) Create oracle base. Cmdir -P /u01/app/oracle)

(4) Create oracle inventory (store oracle XML files)

(5) Configuring kernel parameters

(6) Configuring shell limits for oracle user

## ① creating required root groups & users :-

```
# groupadd oinstall
```

```
# groupadd dba
```

```
# useradd -m -g oinstall -G dba oracle.
```

```
# passwd oracle
```

changing pwd for user oracle

new unix pwd: Semi!

```
# id oracle
```

```
uid = 731(oracle) gid
```

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

## ② creating oracle base:-

```
# mkdir -p /u01/app/oracle
```

```
# chown -R oracle:oinstall /u01/app/oracle/
```

```
# chmod -R 775 /u01/app/oracle/
```

## ③ creating oracle inventory:-

```
# mkdir -p /u01/app/oraInventory
```

```
# chown -R oracle:oinstall /u01/app/oraInventory/
```

```
# chmod -R 775 /u01/app/oraInventory/
```

## ④ creating oracle home:-

```
# mkdir -p /u01/app/oracle/product/11.2.0/db_home
```

```
# chown -R oracle:oinstall /u01/app/oracle/product/11.2.0/db_home/
```

```
# chmod -R 775 /u01/app/oracle/product/11.2.0/db_home/
```

## ⑤ Configuring kernel parameters:-

# vi /etc/sysctl.conf

kernel.shmmni = 1056951840

kernel.sem = 250 32000 100 128

fs.file-max = 6815744

net.core.rmem\_default = 262144

net.core.wmem\_default = 262144

net.core.rmem\_max = 4194304

net.core.wmem\_max = 1048576

fs.aio-max-nr = 1048576

kernel.shmall = 4096

kernel.shmmax = 2097152

net.ipv4.ip\_local\_port\_range = 9000 65500

esc + shift : + wq!

## ⑥ Configuring shell limits for oracle user:-

# vi /etc/security/limits.conf

oracle soft nproc 2047

oracle hard nproc 16384

oracle soft nofile 1024

oracle hard nofile 65536

esc + shift : + wq!

[root@ ~] # xhost +

[root@ ~] # su - oracle

[switch user]

[oracle ~] \$ ~~password~~ vncpasswd

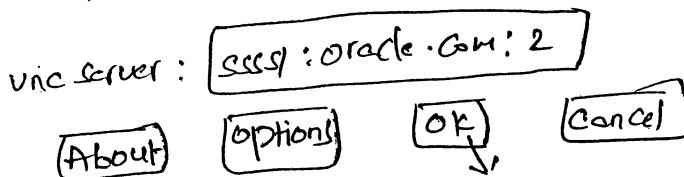
password : oracle

verify : oracle

\$ vncserver

You will receive a password to access your

\$ vncviewer



password : oracle

\$ cd /opt/

\$ ls

\$ cd database/

\$ ls

\$ ./runInstaller.

\* deselect  I wish to receive security updates via my oracle support

click on next → click on 'yes'

click on  skip software updates → next

→  install database software only → next

→  single instance database installation → next → next

→  Enterprise edition → next

→ oracle base - /u01/app/oracle

oracle home - /u01/app/oracle/product/11.2.0/db\_home

→ next

# cd /media

# ls

# cd /opt/

# ls

/opt # cp filename /opt

# cp fn1bf7.zip /opt

# cp fn2bf7.zip /opt

# cp fn3bf7.zip /opt

/opt # cd /opt

opt # unZip fn1bf7.zip

# unZip fn2bf7.zip

# unZip fn3bf7.zip

Inventory Directory - /u01/app/ora/inventory

oraInventory Group name : **oinstall**  → next → next

→ click on 'install'. **Fix & check again**

→ /tmp/cvu\_11.2.0.2.0\_oracle/runfixup.sh

→ open terminal

# Select script & Paste here.

# Select Script & execute

→ click on 'OK' → close.

\* \* Configuring Bash Batch profile:-

\$ vi . bash\_profile

Set your path here.

path = \$ PATH:\$ HOME/bin under we

export ORACLE\_HOME=/u01/app/oracle/product/11.2.0/db-home

export PATH = \$ ORACLE\_HOME/bin:\$ PATH:\$ HOME/bin.

:wq! and execute this file.

\$ . . bash\_profile

→ \$ sqlplus -v  
SQL\*plus: Release 11.2.0.3 production.

→ \$ echo \$ORACLE\_HOME  
/u01/app/oracle/product/11.2.0/db-home

→ \$ cat /etc/oraInst.loc  
inventory\_loc = /u01/app/orainventory  
inst\_group = oinstall

25/08/14

### Data base creation:-

We can create Data base in two ways

- (i) DBCA (Data base Configuration assistant)
- (ii) Manually

#### ① Creating Data base using DBCA:-

# su - oracle

\$ which dbca

/u01/app/oracle/product/11.2.0/db\_home/bin/dbca.

\$ exit

# df - lh

# mkdir -p /u02/app/oracle

# chown -R oracle:oinstall /u02/app/oracle/

# chmod -R 775 /u02/app/oracle/

# xhost +

# su - oracle

\$ dbca

click on next → ① Create database

click on next → click on General purpose → next

Data base name : crm

SID : crm

click on next → Deselect  Configure enterprise manager

click on next → ② use the same administrative password for all accounts

password : oracle

Confirm : oracle → click on next → click on Yes

→ click on use common location for all database files

: /u02/app/oracle Browse

→ click on next

→ click on  enable archiving (To create archive log mode database)

→ click on next → next → next → next

→ click on finish → click on ok

→ click on exit

To connect the database, we will use the following command

\$ export ORACLE\_SID = CRM

\$ sqlplus / as sysdba ( / indicates access privileges of database and we can login as sysdba)

SQL> shut immediate (To shutdown the database)

SQL> exit

\$ sqlplus / as sysdba

SQL> startup

SQL> select name, open\_mode from v\$database;

Name	OPEN-MODE
CRM	READ WRITE

SQL> select instance\_name, status from v\$instance;

Instance Name	Status
CRM	OPEN

SQL> select name from v\$datafile;

SQL> select name from v\$controlfile;

SQL> select member from v\$logfile;

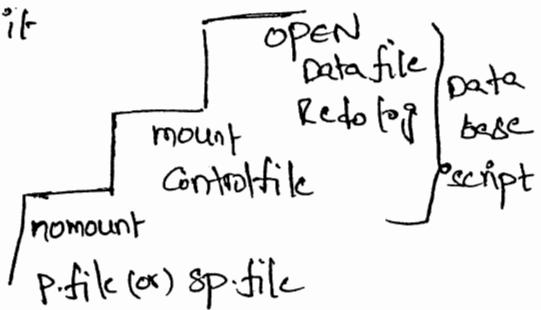
SQL> select name from v\$tempfile;

SQL> select name from v\$tablespace;

SQL> shut immediate

② Steps to Create Database Manually:-

- ① Create pfile using dummy file i.e., "init\_ora" (oracle\_home/dbs)
- ② Create required directory structure
- ③ Start the database ? in nomount state
- ④ Create Database script & execute it
- ⑤ Execute the following scripts
  - (i) catalog.sql
  - (ii) catproc.sql
  - (iii) pupbl1.sql



Note:- (i) If you execute catalog.sql oracle will create the default views, synonyms, indexes, tables, sequences etc..

- (ii) If u execute catproc.sql oracle will create the default procedures, functions, triggers etc..
- (iii) If u execute pupbl1.sql oracle will create the default profiles.

```

# su - oracle
$ cd $ORACLE_HOME/dbs
$ ls
$ cp init.ora initprod.ora
$ ls
$ vi initprod.ora
db_name = 'PROD'
audit_file_dest = '/vol/app/oracle/admin/prod/ademp'
db_recovery_file_dest = '/vol/app/oracle/fast_recovery_area'.
  
```

diagnostic\_dest = '/u01/app/oracle'

control\_files = ('/u03/app/oracle/prod/control01.ctl') upto ②

Compatible = '11.2.0'

exit; we!

③

\$ exit

# mkdir -p /u01/app/oracle

# mkdir -p /u01/app/oracle/admin/prod/adump

# mkdir -p /u01/app/oracle/fast\_recovery\_area

# mkdir -p /u03/app/oracle/prod

# chown -R oracle:oinstall /u01/app/oracle/

# chown -R oracle:oinstall /u01/app/oracle/admin/prod/adump/

# chown -R oracle:oinstall /u01/app/oracle/fast\_recovery\_area/

# chown -R oracle:oinstall /u03/app/oracle/prod/

# chmod -R 775 /u01/app/oracle/

# chmod -R 775 /u01/app/oracle/admin/prod/adump/

# chmod -R 775 /u01/app/oracle/fast\_recovery\_area/

# chmod -R 775 /u03/app/oracle/prod/

# su - oracle

\$ export ORACLE\_SID=prod

\$ sqlplus / as sysdba

SQL> startup nomount

SQL> exit

\$ exit

# df -kh

# mount -t tmpfs shmfs -o size=2g /dev/shm.

# df -kh

```
# su - oracle
```

```
$ export ORACLE_SID=prod
```

```
$ sqlplus / as sysdba
```

```
SQL> startup nomount
```

④ SQL> define editor = vi

```
SQL> ed dbscript.sql
```

To insert  
⑤

```
Create database prod.
```

```
datafile '/u03/app/oracle/prod/system01.dbf' size 10m autoextend on
```

```
sysaux datafile '/u03/app/oracle/prod/sysaux01.dbf' size 10M autoextend on
```

```
default tablespace prod_ts
```

```
datafile '/u03/app/oracle/prod/prod_ts01.dbf' size 10m autoextend on
```

```
undo tablespace undotbs1
```

```
datafile '/u03/app/oracle/prod/undotbs1.dbf' size 10m autoextend on
```

```
default temporary tablespace temp_ts
```

```
tempfile '/u03/app/oracle/prod/temp_ts01.dbf' size 10m autoextend on
```

```
log file
```

```
group 1 ('/u03/app/oracle/prod/redo01.log') size 4m,
```

```
group 2 ('/u03/app/oracle/prod/redo02.log') size 4m
```

```
Controlfile reuse;
```

```
:wq!
```

```
SQL> @dbscript.sql
```

Database created

```
SQL> select name, open-mode, log-mode from v$database;
```

NAME	OPEN-MODE	LOG-MODE
PROD	READ WRITE	NO ARCHIVELOG

SQL> cd run.sql

@ \$ORACLE\_HOME/rdbms/admin/catalog.sql

@ \$ORACLE\_HOME/rdbms/admin/catproc.sql

Conn system/manager

@ \$ORACLE\_HOME/sqlplus/admin/pupblc.sql

:ws!

SQL> @ run.sql

### Table space Management:-

# su - oracle

\$ cat /etc/oratab

\$ export ORACLE\_SID = SSI

\$ sqlplus / as sysdba

SQL> startup

SQL> select name, open\_mode from v\$database;

NAME	OPEN-MODE
SSI	READ WRITE

SQL> select instance\_name, status from v\$instance;

INSTANCE_NAME	STATUS
SSI	OPEN

SQL> select name from v\$tablespace;

NAME
SYSTEM
SYSAUX
UNDOTBS1
USERS
TEMP

SQL> Select name from v\$logfile;

cl scr

NAME  
/u01/app/oracle/sss/system01.dbf

SQL> Select name from v\$tempfile;

/u01/app/oracle/sss/temp01.dbf.

SQL> Create tablespace venkat

datafile '/u01/app/oracle/sss/venkato1.dbf' size 1m;

Error : No such file or directory

SQL> Create tablespace venkat

datafile '/u01/app/oracle/sss/venkato1.dbf' size 1m;

Table space created.

SQL> Select name from v\$tablespace;

SQL> Select name from v\$logfile;

SQL> desc dba\_tablespaces;

SQL> select TABLESPACE\_NAME, BLOCK\_SIZE, STATUS, LOGGING,

CONTENTS from dba\_tablespaces;

SQL> save t51.sql

Created file t51.sql

SQL> get t51.sql

SQL> @t51.sql (or) SQL> start t51.sql

SQL> desc dba\_tablespaces;

SQL> select tablespace\_name, Extent\_Management, Segment\_Space\_Management,

bigfile from dba\_tablespaces;

SQL> Save t52.sql

SQL> desc dba\_data\_files;

SQL> select Tablespace-name, File-name, file-id, Bytes from

dba\_data\_files;

SQL> col file-name for a30      <sup>format</sup> <sup>column</sup> (To reduce the column size)

SQL> /      (TO execute the previous query)

SQL> col Tablespace-name for "a10"

SQL> /

SQL> save tsi.sql

SQL> select File-name, file-id, sum(bytes/1024/1024) from

SQL> select File-name, file-id

dba-data-files group by file-name, file-id

Adding a datfile to existing tablespace:-  
SQL> @tsi.sql

SQL> alter tablespace venkat add

datafile '/u01/app/oracle/sss/venkator.dbf' size 2m;

Tablespace altered.

SQL> @tsi.sql

SQL> @tsi.sql

SQL> alter tablespace raj offline;

SQL> @tsi.sql

SQL> alter tablespace raj online;

SQL> alter tablespace raj nologging;

SQL> @tsi.sql

SQL> alter tablespace raj logging;

SQL> @tsi.sql

SQL> alter tablespace raj read only;

SQL> @ts1.sql

NAME	STATUS
Raj	Read only

SQL> alter tablespace raj read write;

SQL> @ts1.sql

SQL> alter tablespace raj rename to raj1;

SQL> @ts1.sql

Steps to rename datafile:

- ① offline the tablespace
- ② rename the datafile at os level.
- ③ issue alter tablespace rename file command.
- ④ online the tablespace.

SQL> alter tablespace raj offline;

\$ cd /u02/app/oracle/sss1

\$ ls

\$ mv sss01.dbf sss02.dbf (or) \$ cp sss01.dbf sss02.dbf

SQL> alter tablespace raj rename

datafile '/u02/app/oracle/sss/ss01.dbf' to '/u02/app/oracle/sss/ss02.dbf';

SQL> alter tablespace raj online;

SQL> @ts1.sql

MANOJ ENTERPRISES  
Plot:40, Gosthani Nagar  
Ameerpet, Hyderabad

## Creating Big file tablespace:-

SQL> create bigfile tablespace big\_ts

datafile '/u02/app/oracle/sss/big\_ts01.dbf' size 10m;

SQL> select tablespace\_name, bigfile from dba\_tablespaces;

Tablespace_Name	Big
-----------------	-----

BIG_TS	YES
--------	-----

SQL> @t53.sql

SQL> alter tablespace big\_ts resize 12m; *(only for bigfile)*

SQL> @t53.sql

SQL> alter database datafile # resize 4m;

Database altered.

## Dropping a datafile from existing tablespace:-

SQL> alter tablespace venkat drop

datafile '/u02/app/oracle/sss/venkato2.dbf';

## Creating temporary tablespace:-

SQL> create temporary tablespace temp2

tempfile '/u02/app/oracle/sss/tempo2.dbf' size 2m;

SQL> select name from v\$tablespace;

## Creating temporary tablespace groups:-

SQL> create temporary tablespace temp3

tempfile '/u02/app/oracle/sss/tempo3.dbf' size 2m;

tablespace group tempgrp;

SQL> select \* from dba\_tablespace\_groups;

SQL> alter tablespace temp2 tablespace group tempgrp;

SQL> select \* from dba\_tablespace\_groups;

SQL> alter tablespace temp2 tablespace group ' ';

SQL> select \* from dba\_tablespace\_groups;

SQL> desc dba\_free\_space;

SQL> select tablespace\_name, file\_id, bytes from dba\_free\_space

where tablespace\_name = 'VENKAT';

<u>NAME</u>	<u>FILE_ID</u>	<u>BYTES</u>
VENKAT	5	1048576

SQL> set wrap off;

SQL> select \* from database\_properties;

SQL> alter database default tablespace venkat;

SQL> alter database default temporary tablespace temp2;

SQL> select \* from database\_properties;

SQL> set wrap on;

\* SQL> drop tablespace venkat including contents and datafiles;

ORACLE

## \* \* User Management \* \*

# su - oracle

\$ cat /etc/oratab

\$ export ORACLE\_SID = hrms

\$ sqlplus / as sysdba

SQL> startup

sys> select name, open-mode from v\$database;

sys> select instance-name, status from v\$instance;

sys> select \* from all\_users;

sys> desc dba\_users;

sys> select username, default\_tablespace, temporary\_tablespace  
from dba\_users where username = 'SCOTT';

username	default_tablespace	Temporary_tablespace
SCOTT	USERS	TEMP

sys> select username, profile from dba\_users where username = 'SCOTT';

username	profile
SCOTT	DEFAULT

Creating a user:-

sys> create user ui identified by ui;

sys> select \* from all\_users where username = 'ui';

sys> conn ui/ui

sys> show user

sys> grant create session to ui;

sys> conn / as sysdba

sys> show user

\* sys> grant create session to u1;

\* sys> revoke create session from all;

Sys> grant create session to user

Sys Conn 41/41

vi) create table cmp (cid number);

ORA-01704 error: Insufficient privileges

sys> grant create table to all;

U1> Create table emp ( eid number);

UI> Select \* from tab;

U> insert into emp values (1);

SQL> insert into emp  
Error: no privileges on tablespace 'users'.

sys> alter user ut quota 5M on users;

UD insert into EMP values(1);

QD select \* from emp;

```
sys> desc dba_users;
```

```
sys> desc dba_users;
sys> select username, default_tablespace, temporary_tablespace from
      users where username = 'UI';
```

sys> Select Username;

dba-users where user\_name = Default\_tablespace

Username

Temporary-tablespace

Sys> desc dbq\_sys\_priv;

SQ1> desc dba\_sys\_privs where grantee='U';  
SQ1> select \* from dba\_sys\_privs where grantee='U';

sys> select \* from sysviews (0x)

U1> select \* from session\_priv;

```
ms> select * from user_sys_privs;
```

sub descend dba\_ts\_quotas;

sys> select Username, TableSpace\_name, Max\_Bytes from Hba\_TS\_Quotes

where `username = 'ui';`

U1> Select \* from user\_ts\_quoter;

SYS> desc db2-users;

SYS> desc dba-objs;  
SYS> select cffname, Account\_Status from dba-objs where cffname='UI';

sys> after user w/ account lock;

```
sys> select user_name, account_Status from dba_users where account='UN';
```

username status  
U1 docked

sys> alter user u1 account unlock; → username status  
u1 open

sys> alter user all identified by 42;

Sep 7 Conn 41/42

\$ sqlplus conn / as sysdba

```
sys> desc v$session;
```

```
sys> desc V$SESSION;  
sys> select sid, serial#, username from V$SESSION where username
```

not in ('sys');

not in ('sys');  
sys> # after System kill session '141,15'; (to kill the user session)

sys drop user all;

error: cascade must be specified to drop 'ui'

```
sys> drop user ui cascade;
```

## Creating roles:-

Sy<sub>5</sub>) select \* from dba\_roles;

```
sys> create role role1;
```

sys grant create session, create table to refel;

```
sys select * from dba_roles;
```

```
sys> desc role-sys-prius;
```

```
sys> select * from role-sys-privs where role = 'ROLE1';
sys> select * from role-sys-privs where role = 'CONNECT';
sys> select * from role-sys-privs where role = 'RESOURCE';
sys> select * from role-sys-privs where role = 'DBA';
sys> create user u1 identified by u1;
sys> grant role1 to u1;
sys> conn u1/u1
u1> select * from session-privils;
u1> conn / as sysdba
sys> desc dba-role-privils;
sys> select * from dba-role-privils where grantee = 'U1';
sys> revoke role1 from u1;
sys> create role role2;
sys> grant role1 to role2;
sys> select * from role-sys-privs where role = 'ROLE2';
sys> conn u1/u1
sys> conn / as sysdba
sys> grant role2 to u1;
sys> conn u1/u1
u1> conn / as sysdba;
sys> revoke role1 from role2;
sys> conn u1/u1
sys> grant role1 to role2;
sys> desc role-role-privils;
sys> select * from role-role-privils where role = 'ROLE2';
sys> revoke role1 from role2;
sys> drop role role1;           sys> drop role role2;
```

sys> desc dba\_profiles;

sys> select \* from profile, resource\_name, limit from dba\_profiles;

Creating a profile:-

sys> create profile prof1 limit  
failed\_login\_attempts 2  
sessions\_per\_user 2;

sys> alter user u1 profile prof1;

sys> select username, profile from dba\_users where username = 'U1';

sys> grant create session to u1;

sys> show parameter resource\_limit;

sys> alter system set resource\_limit = true;

sys> conn u1/u2

error: invalid username/password; logon denied

warning: you are no longer connected to ORACLE

sys> conn u1/u3

sys> conn u1/u4

error: the account is locked

sys> alter user u1 account unlock;

sys> conn u1/u1

sys> conn u1/u1

sys> conn u1/u1

exceeded simultaneous sessions\_per\_user limit.

with grant option & with admin option:-

sys> drop user u1 cascade;

sys> create user u1 identified by u1;

sys> create user u2 identified by u2;

{  
    | Sys> grant Connect, resource to U1 with admin option;  
    | Sys> select \* from dba\_sys\_privs where grantee = 'U1';  
    | Sys> Conn U1/u1  
    | U1> select \* from session\_privs;  
    | U1> grant creation session to U2;  
    |  
    | Sys> grant create session, create table to U1 with admin option;  
    | Sys> grant create session, create table to U1 with admin option;  
    | Sys> Conn U1/u1  
    | U1> grant create session to U2;  
    | U1> Conn / as sysdba  
    | Sys> revoke create session from U1;  
    | Sys> Conn U1/u1  
    | error: user U1 lacks create session privilege; logon denied  
    | Sys> Conn / as sysdba  
    | Sys> Conn U2/u2  
    | Connected  
    | with grant option:  
    | Sys> grant create session, create table to U1;  
    | Sys> Conn scott/tiger  
    | error: The account is locked  
    | Sys> Conn / as sysdba  
    | Sys> alter user scott account unlock identified by tiger;  
    | Sys> Conn scott/tiger  
    | SCOTT> select \* from tab;  
    | SCOTT> select \* from dept;  
    | SCOTT> Conn U1/u1

U1> select \* from tab;

U1> Conn scott/tiger

scott> grant select on dept to U1 with grant option;

scott> Conn U1/U1

U1> select \* from dept, scott.dept;

U1> grant select on scott.dept to U2;

U1> Conn U2/U2

U2> select \* from scott.dept;

U2> Conn scott/tiger

scott> revoke select on dept from U1;

scott> Conn U1/U1

U1> GRANT select \* from scott.dept;

U1> Conn U2/U2

U2> select \* from scott.dept;

error: Table or view does not exist.

### Redo log file management:-

# su - oracle

\$ expat ORACLE-SID=hrms

\$ sqlplus / as sysdba

Sys> start up.

Sys> select name, open\_mode from v\$database;

Sys> select instance\_name, status from v\$instance;

Sys> select member from v\$logfile;

Sys> desc v\$logfilc;

Sys> select group#, status, member from v\$logfilc;

sys> desc v\$log;

sys> select group#, members , status from v\$log;

sys> select a.group#, a.members , a.status , b.member from v\$log a , v\$logfile b where a.group# = b.group#.

<u>Group #</u>	<u>Members</u>	<u>Status</u>	<u>member</u>
1	1	Current	/u02/app/oracle/hrms/redo01.log
2	1	Inactive	/u02/app/oracle/hrms/redo02.log
1	1	Inactive	/u02/app/oracle/hrms/redo01.log

Adding a redo log group to existing Database:-

sys> alter database add logfile group 4 '/u02/app/oracle/hrms/redo04.log' size 4m;

sys> @ redo.sql

Adding a member to existing group:-

sys> alter database add logfile member '/u02/app/oracle/hrms/redo04a.log' size 4m to group 4;

sys> @ redo.sql

Dropping a member from group:-

sys> alter database drop logfile member '/u02/app/oracle/hrms/redo04a.log';

sys> @ redo.sql.

Renaming a member:-

- ① shut down the database
- ② re-name the member at 0/s level
- ③ start the database in mount state.
- ④ issue alter database rename file command
- ⑤ open the database

① sys > shut immediate

\$ cd /u02/app/oracle/hrms/

\$ ls

\$ cp redo04.log redo04b.log

sys> startup mount

sys> alter database rename file '/u02/app/oracle/hrms/redo04.log' to  
'/u02/app/oracle/hrms/redo04b.log';

sys> alter database open;

sys> @ redo.sql

dropping a group:-

sys> alter database drop logfile group 4;

sys> alter database drop logfile group 3;

sys> @ redo.sql

{sys> alter system switch logfile;

          } sys> @ redo.sql

11/09/14

Control file management:-

① Control file multiplexing

② Taking backup of text based control file

③ Re-naming Database.

Steps to multiplex Control files:-

① Shutdown your database

② Create required directory structure

③ Copy Control file from original location to new location.

④ Specify new Control file location into p-file.

⑤ Start the database.

① Create pfile from Spfile;

```
# su - oracle
$ export ORACLE_SID=hrms
$ sqlplus / as sysdba

sys> startup
sys> select name, open_mode, log_mode from v$database;
sys> select instance_name, status from v$instance;
sys> select name from v$controlfile;
sys> show parameter spfile;

sys> shut immediate

# mkdir -p /u02/app/oracle/hrms
# chown -R oracle:oinstall /u02/app/oracle/hrms/
# chmod -R 775 /u02/app/oracle/hrms/
# su - oracle
$ cd /u02/app/oracle/hrms/
$ ls
$ clear
$ cp control01.ctl /u02/app/oracle/hrms/control01.ctl
$ cd $ORACLE_Home/ dbs
$ vi inithrms.ora ←
Control_files = , '/u02/app/oracle/hrms/control01.ctl'
: wq!

sys> start up
sys> select name from v$controlfile;
sys> shut immediate.
```

```
$ cd /u01/app/oracle/hrms/
```

```
$ ls
```

```
$ rm control03.ctl
```

sys> start up.

sys> shut immediate.

```
$ cd $ORACLE_HOME/dbs
```

```
$ vi inithrms.ora <
```

\* control\_file = remove the 3rd control file location.

```
:wq!
```

sys> startup

```
sys> select name from v$controlfile;
```

\*\*\* sys> select \* from v\$diag\_info;

| Default Trace file

/u01/app/oracle/diag/rdbms/hrms/hrms/trace/

```
$ cd /u01/app/oracle/diag/rdbms/hrms/hrms/trace/
```

```
$ ls -lrt.
```

```
$ tail -f alert-hrms.log (If extension is .log then we need  
to open with tail command)
```

\$ ctrl+c to exit from the alert log file.

Taking backup of text-based Control file:-

```
sys> alter database backup controlfile to trace;
```

```
sys> alter database backup controlfile to trace as '/home/oracle/control';
```

```
sys> select * from v$diag_info;
```

```
$ pwd < /home/oracle
```

```
$ ls
```

\$ vi cont.srl

esc+d+gg to delete all above data

esc+d+G to delete all below data.

CREATE CONTROLFILE REUSE DATABASE "HRMS" RESETLOGS FORCE LOGGING ARCHIVELOG

Character Set WE8MSWIN1252;

:wq!

Recreating Control files:-

sys> shut immediate.

\$ cd /u01/app/oracle/hrms

\$ ls

\$ rm control01.ctl

\$ cd /u01/app/oracle/fast\_recovery\_area/hrms

\$ ls

\$ rm control02.ctl

sys> startup nomount

sys> ! pcd

sys> @ cont.srl ↵

19 ↵

control file created.

SQL> select open\_mode from v\$database;

SQL> alter database open;

SQL> alter database open resetlog;

Renaming a database:-

① Take the backup of text based Control file

② shut immediate

③ Rename the database name in P-file & text based Control file.

④ Remove the existing binary Control files

⑤ Start the database in nomount state & execute the script

⑥ open the database  
previous process

sys> shut immediate

\$ cd \$ORACLE\_HOME/dbs

\$ vi inithrms.ora

db\_name = 'hrms1'

:wq!

\$ cd

\$ vi cont.srl

CREATE CONTROLFILE SET Database "HRMS1" Resetlogs force logging Archivelog

:wq!

\$ cd /u01/app/oracle/hrms/

\$ ls

\$ rm control01.ctl

\$ cd /u01/app/oracle/fast-recovery-area/hrms/

\$ ls

\$ rm control02.ctl

sys> startup nomount

sys> @cont.srl

19 / ↵

sys> alter database open resetlogs;

sys> select name, open-mode from v\$database;

## Archive log management:-

# su - oracle

\$ export ORACLE\_HOME = crms

\$ sqlplus / as sysdba

sys> start up

sys> select name, open\_mode from v\$database;

sys> select instance\_name, status from v\$instance;

sys> select log\_mode from v\$database;

sys> select log\_mode from v\$database;

sys> archive log list;

sys> show parameter DB\_RECOVERY\_FILE\_DEST

sys> shut immediate

sys> startup mount

sys> alter database noarchivelog;

sys> alter database open;

sys> select log\_mode from v\$database;

sys> shut immediate

sys> startup mount

sys> alter database archive log;

sys> alter database open;

sys> Conn scott/tiger

sys> Conn / as sysdba

sys> alter user scott unlock identified by tiger;

sys> Conn scott/tiger

scott> select \* from tab;

scott> select insert into salgrade select \* from salgrade;

```
scott> /
scott> /
scott> /
:
scott> /
scott> commit;
scott> Conn / as sysdba;
sys> archive log list;
sys> show parameter DB-RECOVERY-FILE-DEST
$ cd /u01/app/oracle/fast_recovery_area/
$ ls
$ cd crms/
$ ls
$ cd 2009-04-06/
$ ls
```

Generating archive logs in separate location:-

- steps:-
- (1) Create pfile from spfile and rename spfile as well as shutdown database
  - (2) create required directory structure for archive log destination
  - (3) specify the new archive destination into pfile
  - (4) comment FRA location pfile . //
  - (5) start the database

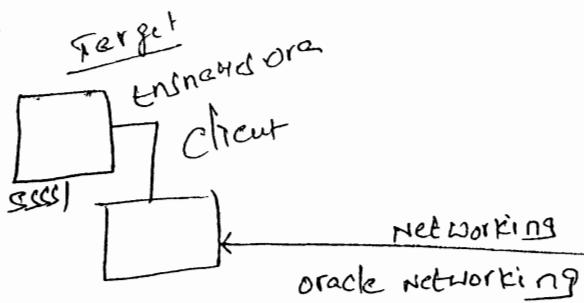
```
sys> create pfile from spfile;
sys> shut immediate
sys> exit
$ cd $ORACLE_HOME/dbs
$ ls
```

```
$ mv Spfile crms.ora Spfile crms.ora.bkp  
$ exit  
# mkdir -p /u03/arch  
# chown -R oracle:oinstall /u03/arch/  
# chmod -R 775 /u03/arch/  
# su - oracle  
$ cd $ORACLE_HOME/db  
$ vi initcrms.ora  
log_archive_dest = '/u03/arch'  
# db_recovery_file_dest = '/u01/app/oracle/fast_recovery_area'  
# db_recovery_file_dest_size = 41964001152  
; wq!  
$ cd  
$ export ORACLE_SID=crms  
$ sqlplus / as sysdba  
sys> startup  
sys> archive log list;  
sys> alter system switch logfile;  
sys> /  
sys> /  
$ cd /u03/arch/  
$ ls  
sys> show parameter log_archive_format;  
sys> show parameter log_archive_max_...  
sys> show parameter log_archive_dest
```

MANOJ ENTERPRISES  
Plot:40, Gajathri Nagar  
Amarpet, Hyderabad

## \*\* Networking \*\*

etc hosts



172.16.128.151

ssss1.oracle.com (hostname)

ssss1

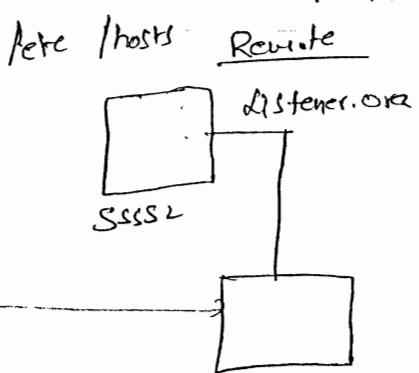
Specify the other server IP address  
& host name in /etc/hosts file.

vi /etc/hosts

172.16.128.152 ssss2.oracle.com ssss2

172.16.128.151 ssss1.oracle.com ssss1

etc hosts



172.16.128.152

ssss2.oracle.com

ssss2

vi /etc/hosts

172.16.128.151 ssss1.oracle.com  
ssss1

172.16.128.152 ssss2.oracle.com ssss2

```
[ssss1:~]# vi /etc/hosts
127.0.0.1      localhost.localdomain - local host
172.16.128.151 ssss1                  ssss1.oracle.com
172.16.128.152 ssss2                  ssss2.oracle.com
```

: wq!

[root@ssss1 ~]# ssh ssss2

root@ssss2's password: oracledb2

```
[root@ssss2 ~]# vi /etc/hosts
127.0.0.1      localhost.localdomain - local host
172.16.128.151 ssss1
172.16.128.152 ssss2
```

ssss1.oracle.com.  
ssss2.oracle.com.

: wq!

[root@sec1 ~] # ping 172.16.128.152 (or) ssss2 (or) sss2.oracle.com.

ping 172.16.128.152 (172.16.128.152) 56(84)

[root@sec1 ~] # ping 172.16.128.151 (or) sec1.oracle.com.

64 bytes from : 151 : icmp\_seq=0 ttl=64 time=0.032 ms

[root@sec1 ~] # ssh ssss2

root@ssss2's password: oracle66

[root@ssss2 ~] # exit..

[root@sec1 ~] # ssh ssss2

[root@ssss2 ~] # su - oracle

\$ ps -ef | grep Smon

oracle	1418	1	0	12:21?	00:00:00	ORCL-SMON-PROD
						↓ db

PROCESSID

\$ export ORACLE\_SID = prod

\$ sqlplus / as sysdba

sys> exit.

[root@sec1 ~] # xhost +

# su - oracle

\$

First invoke in first machine

[root@sec1 ~] # xhost +

# ssh -Y oracle@sec2

oracle@ssss2's password : oracle

[oracle@ssss2 ~] \$

[88882 ~] \$ which netca

\$ netca  
click on listener configuration → next  
① Add → next  
Listener name **LISTENER** (or) names → next → next  
② use standard port no. of 1521 → next  
→ next → next → finish.  
③ NO

[88882 ~] \$ cd \$ORACLE\_HOME/network/admin/

\$ ls

\$ vi listener.ora

:2!

[admin] \$ ls

\$ cd samples/

\$ ls

\$ vi listener.ora

:se ny

:w, w >> \$ORACLE\_HOME/network/admin/listener.ora.

:q!

\$ cd ..

\$ vi listener.ora

LISTENER =

(DESCRIPTION\_LIST =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 88882.oracle.com)(PORT = 1521))

# (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))

)

)

ADR\_BASE\_LISTENER = 1001/app/oracle

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(SID\_NAME=prod)

(ORACLE\_HOME = /001/app/oracle/product/11.2.0/db-home)

)

)

: w!

\$ lsnrctl start listener

\$ lsnrctl status listener

\$ lsnrctl stop listener

[root@sscs1 ~] # xhost +

# su - oracle

\$ netca

① Local Net Service Name Configuration → next

② Add

Service Name : prod

Host Name : sscs2.oracle.com

③ Use the standard port number 1521 → next

④ No, do not test

Net Service Name : tns-prod → next

⑤ NO

→ next

→ next

→ next → next

→ next

→ next

→ next

→ next → next

→ finish

\$ cd \$ORACLE\_HOME/network/admin

\$ vi tnsnames.ora

TO-PROD =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (Protocol=TCP) (Host=ssss1.oracle.com) (Port=1521))

)

(CONNECT\_DATA =

(SERVICE\_NAME=prod)

)

)

: w8!

[root@ssss1 ~] # tnsping to-prod

[root@ssss1 ~] \$ sqlplus scott/tiger@to-prod

SCOTT> select \* from tab;

[root@ssss1 ~] \$ cd \$ORACLE\_HOME/dbs

\$ orapwd file=orapwprod password=rec entries=3 force=y

\$

[root@ssss1 ~] \$ sqlplus sys/rac@to-prod as sysdba.

sys>

DB links :-

# su-oracle

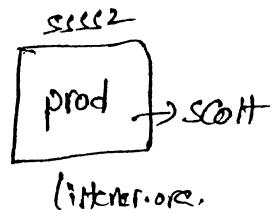
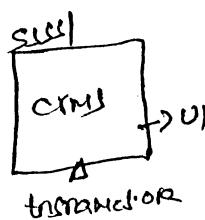
\$ export ORACLE\_SID=cms

\$ sqlplus / as sysdba

sys> start up

ssss1] # ssh ssss2

prod : oracle dba



```
scott]# su - oracle  
scott]$ export ORACLE_SID=prod  
scott]$ sqlplus / as sysdba  
sys> startup  
sys> alter user scott account unlock identified by tiger;  
sys> select * from tab;  
sys> Conn scott/tiger  
scott> select * from tab;
```

### Client Machine:-

```
sys> grant connect,resource to u1 identified by u1;  
sys> Conn u1/u1  
u1> Conn / as sysdba  
sys> grant create database link, create public database link to u1;  
sys> Conn u1/u1  
u1> Create database link priv_link  
      Connect to scott identified by tiger using 'to_prod';  
u1> select * from tab; | scott]$ lsnrctl start listener  
u1> select * from dept@priv_link; |  
      error: no listener  
  
u1> select * from tab@priv_link;  
u1> select * from dept@priv_link;  
  
u1> create public database link pub_link  
      Connect to scott identified by tiger using 'to_prod';  
u1> select * from dept@pub_link;  
u1> Conn / as sysdba  
sys> grant connect, resource to u2 identified by u2;
```

sys> Conn u2/u2

or> select \* from dept@pub-link;

or> select \* from dept@priv-link;

Error:

ORA-02019: Connection description for remote database not found

or> Conn / as sysdba

sys> desc dba-db-links;

sys> select \* from dba-db-links;  
(OR)

sys> Select \* from dba-db-links where owner = 'U1';

sys> desc all-db-links;

sys> select owner, db-link, username from all-db-links

where username = 'U1';

No rows selected

sys> select owner, db-link, username from all-db-links where username = 'SCOTT';

sys> select owner, db-link, username from all-db-links where owner = 'D1';

sys> select owner, db-link, username from all-db-links;

sys> Conn U1/U1

U1> desc user-db-links;

U1> select \* from user-db-links;

Creating materialized views & synonyms:-

sys> grant create materialized view to U1;

sys> Conn U1/U1

U1> Conn / as sysdba

sys> grant create synonym to U1;

sys> Conn U1/U1

SQL> Create synonym dept for dept@priv-link

SQL> select \* from tab;

SQL> select \* from dept;

SQL> Create materialized view mvview

refresh fast

with primary key

start with sysdate

next sysdate +1(24\*60\*60)

as select \* from dept;

SQL> save mvsel replace;

Remote Server  
SSSS2] \$ ~~sqlplus~~ & \$1plus scott/tiger

SCOTT> select \* from tab;

SCOTT> select create materialized view log on dept;

SCOTT> select \* from tab;

SCOTT> select \* from MLOG\$-DEPT;

Client Server

SQL> @mv.sel

SQL> select \* from tab;

Remote:

SCOTT> insert into dept values(50, 'oracle', 'hyd');

SCOTT> select \* from MLOG\$-DEPT; SCOTT> commit;

C-Server:

SQL> select \* from mvview;

## Undo Management:-

sys> select name , open-mode , log-mode from v\$database;

sys> select instance\_name , status from v\$instance;

sys> select name from v\$tablespace;

sys> Create tablespace undo

datafile '/u02/app/oracle/crms/undob1.dbf' size 10m;

sys> select tablespace\_name , contents from dba\_tablespaces;

sys> Create undo tablespace undotbs2

datafile '/u02/app/oracle/crms/undotbs2.dbf' size 10m

auto extend on;

sys> desc dba\_rollback\_segs;

sys> select segment\_name , tablespace\_name , status from dba\_rollback\_segs;

sys> desc v\$rollstat;

sys> desc v\$rollname;

sys> select a.usn , a.xacts , a.status , b.name from v\$rollstat a ,

v\$rollname b where a.usn = b.usn;

sys> save undo.sql

\* SCOTT> Select \* from tab;

SCOTT> ~~Select \* from salgrade Select \* from salgrade;~~

SCOTT> insert into salgrade Select \* from salgrade;

sys> @ undo.sql

sys> alter system set undo\_tablespace=undotbs2;

sys> @ undo.sql

SCOTT> Commit;

SYS> select tablespace\_name , retention from dba tablespaces;

SYS> alter tablespace undotbs2 retention guarantee;

SYS> select tablespace\_name , retention from dba tablespaces;

SYS> alter tablespace undotbs2 retention noguarantee;

SYS@ undob:~

SYS> alter system set undo\_tablespace = undotbs1;

SYS> show parameter undo -

Type

\* \* Cold Backup:-

\$ export ORACLE\_SID = hrms

\$ sqlplus / as sysdba

SYS> startup

SYS> select name , open\_mode , log\_mode from v\$database;

SYS> select instance\_name , status from v\$instance;

SYS> select name from v\$datafile;

SYS> select name from v\$controlfile;

SYS> select member from v\$log\_file;

SYS> shut immediate.

# mkdir -p /u03/coldbkp

# chown -R oracle:oinstall /u03/coldbkp/

# chmod -R 775 /u03/coldbkp/

# su - oracle

\$ cd /u02/app/oracle/hrms/

\$ ls

\$ cp \*.\* /u03/coldbkp/

\$ cd /u01/app/oracle/fast\_recovery\_area/bms

\$ ls

\$ cp Control02.CTL /u03/coldbkp/

sys> start up

sys> archive log list;

sys> conn scott/tiger

sys> alter user scott account unlock identified by tiger;

sys> conn scott/tiger.

scott> select \* from tab;

scott> insert into salgrade select \* from salgrade;

scott> /

scott> /

scott> / - - - - -

scott> commit;

scott> select count(\*) from salgrade;

COUNT(\*)  
10485760 → 1310720

scott> conn / as sysdba

scott> sys> archive log list;

sys> alter switch system switch logfile;

sys> archive log list;

S-5 loss of full database:-

sys> select name from v\$logfile;

sys> select name from v\$controlfile;

sys> select member from v\$logfile;

\$ cd /u02/app/oracle/bms/

\$ ls

\$ rm \*

\$ cd /u01/app/oracle/fast\_recovery\_area/hrms/

\$ ls

\$ rm control02.ctl

sys> shut abort

\$ cd /u03/cold\_bkp/

\$ ls

\$ cp \*.\* /u02/app/oracle/hrms/

\$ cp control02.ctl /u01/app/oracle/fast\_recovery\_area/hrms/

sys> start up nomount

sys> select file#, checkpoint\_change# from v\$datafile; 808761

sys> save df.srl

sys> select file#, checkpoint\_change# from v\$datafile\_header;

sys> save dfh.srl

sys> alter database recover automatic using backup controlfile until cancel;

sys> recover cancel;

sys> select open\_resetlogs from v\$database;

sys> alter database open resetlogs;

sys> select count(\*) from scott.emp; → sys> @df.srl → sys> @dfh.srl 814214

sys> shut immediate.

\$ cd /u02/app/oracle/hrms/

\$ ls

\$ cp \*.\* /u03/cold\_bkp/

\$ cd /u01/app/oracle/fast\_recovery\_area/hrms/

\$ cp control02.ctl /u03/cold\_bkp/

### S-F loss of non-system datafile

sys> startup

sys> select name from v\$logfile;

\$ cd /u02/app/oracle/hrms/

\$ rm users01.dbf

sys> conn scott/tiger.

scott> insert into salgrade select \* from salgrade;

Error: error in opening database file 4

scott> conn / as sysdba

sys> alter database datafile 4 offline;

sys> select file#, status, error from v\$datafile\_header;  
(or)

sys> Select \* from v\$recover\_file;

\$ cd /u02/cold\_bkp/

\$ cp users01.dbf /u02/app/oracle/hrms/

sys> recover datafile 4;

sys> alter database datafile 4 online;

### S-F loss of system datafile (system.dbf)

sys> select name from v\$logfile;

\$ cd /u02/app/oracle/hrms/

\$ rm system01.dbf

sys> shut abort

\$ cd /u02/cold\_bkp/

\$ cp system01.dbf /u02/app/oracle/hrms/

sys> start up nomount.

sys> recover database; (or) recover datafile,  
sys> alter database open;

~~S-12~~ ~~\* \* loss of control file~~ Control files:-

sys> select name from v\$controlfile;

\$ cd /u02/app/oracle/hrms/

\$ rm \*.ctf

\$ cd /u01/app/oracle/fast\_recovery\_area/hrms/

\$ rm control02.ctf

sys> shut abort

\$ cd /u02/coldbkp/

\$ cp control01.ctf /u02/app/oracle/hrms/

\$ cp control02.ctf /u01/app/oracle/fast\_recovery\_area/hrms/

sys> start up mount

- sys> recover database using backup control file until cancel;

auto ↳ (manually specify)

sys> recover cancel;

sys> select member from v\$logfile;

sys> recover database using backup control file until cancel;

↳ /u02/app/oracle/hrms/reredo1.log ↳

sys> recover cancel;

sys> recover database using backup control file until cancel;

↳ /u02/app/oracle/hrms/reredo2.log ↳

ORA-01547: warning:

sys>

sys> recover cancel;

sys> alter database open resetlogs;

sys> shut immediate.

\$ cd /u02/app/oracle/hrms/

\$ cp \*.\* /u03/coldbkp/

\$ cd /u01/app/oracle/fast-recovery-area/hrms/

\$ cp control02.ch /u03/coldbkp/

sys> startup;

Loss of redo log files :-

sys> select member from v\$logfiles;

\$ cd /u02/app/oracle/hrms/

\$ rm \*.log

sys> shut abort

\$ cd /u03/coldbkp/

\$ cp \*.dbf /u02/app/oracle/hrms/

sys> startup mount

sys> recover database until cancel;

↳ (Automatically it will take Auto option)

sys> recover cancel;

sys> alter database open resetlogs;

sys> shut immediate;

\$ cd /u02/app/oracle/hrms/

\$ cp \*.\* /u03/coldbkp/

\$ cd /u01/app/oracle/fast-recovery-area/hrms/

\$ cp control02.ch /u03/coldbkp/

S-51 loss of datafile which was not in the backup:-

sys> start up;

sys> create tablespace raj  $\rightarrow$  gen1  
datafile '/u02/app/oracle/hrms/raj01.dbf'  $\rightarrow$  sunil01  
size 10M;

sys> create user u1 identified by u1@le fault tablespace raj;

sys> grant connect, resource to u1;

sys> conn u1/u1

u1> create table a(a number)

2;

u1> insert into a values(1);

u1> insert into a select \* from a;

u1> /, 1, 1,

u1> commit;

u1> Select count(\*) from a;

[256]

u1> conn / as sysdba

sys> \$ cd /u02/app/oracle/hrms/

\$ ls

\$ rm raj01.dbf

sys> alter database datafile 5 resize 12m;

sys> alter database datafile 5 offline;

sys> alter database create datafile '/u02/app/oracle/hrms/raj01.dbf';

sys> recover datafile 5;

sys> alter database ~~datafile~~ 5 online;

sys> select count(\*) from u1.a;

## Hot backup:-

```
# su -oracle
$ export ORACLE_SID=hrms
$ sqlplus / as sysdba
sys> startup
sys> select name, open_mode, log_mode from v$database;
sys> archive log list;
sys> select name from v$datatfile;
# mkdir -p /u01/hotbkp
# chown -R oracle:oinstall /u01/hotbkp/
# chmod -R 775 /u01/hotbkp/
sys> select name from v$controlfile;
sys> select name from v$tablespace;
sys> select * from v$backup;
sys> alter database begin backup;
sys> select * from v$backup;
$ cd /u01/app/oracle/hrms/
$ ls
$ cp *.dbf /u01/hotbkp/
sys> alter database end backup;
sys> alter database backup controlfile to '/u01/hotbkp/control.ctl';
sys> alter system switch log file;
sys> alter user scott account unlock identified by tiger;
scott> conn scott/tiger
SCOTT> insert into salgrade select * from salgrade;
SCOTT> commit;
SCOTT> select count(*) from salgrade;
```

10485760

SQL> Conn / as sysdba

SQL> archive log list;

SQL> alter system switch logfile;

SQL> archive log list;

S-I Loss of full database :-

SQL> select name from v\$logfile;

SQL> select name from v\$controlfile;

SQL> select member from v\$logfile;

\$ cd /u02/app/oracle/frmns/

\$ ls  
\$ rm \*

\$ cd /u01/app/oracle/fast\_recovery\_area/frmns/

\$ ls  
\$ rm control02.ctl

SQL> shut abort

\$ cd /u03/hotbkp/

\$ ls  
\$ cp \*.dbf /u02/app/oracle/frmns/

\$ cp Control.ctl /u02/app/oracle/frmns/Control01.ctl

\$ cp Control.ctl /u01/app/oracle/fast\_recovery\_area/frmns/Control02.ctl.

SQL> start up mount

SQL> select file#, checkpoint\_change# from v\$logfile;

SQL> save dfl-sql replace;

SQL> select file#, checkpoint\_change# from v\$logfile\_header;

SQL> save dfh-sql replace;

SQL> alter database recover automatic using backup controlfile until cancel;

SQL> recover cancel;

SQL> select open\_resetlogs from v\$database;

SQL> alter database open resetlogs;

MANOJ ENTERPRISES  
Plot 40, Gaganthi Nagar  
Ameerpet, Hyderabad

29/01/14

## RMAN

No Catalog mode:-

# su - oracle

\$ export ORACLE\_SID = hrms

\$ sqlplus / as sysdba

SYS> startup;

SYS> select name, open\_mode, log\_mode from v\$database;

SYS> select instance\_name, status from v\$instance;

\$ rman target / ↵

error : TNS: net service name is incorrectly specified.

\$ export ORACLE\_SID = crms

\$ rman target / ↵

RMAN> shutdown immediate

RMAN> exit

\$ rman target / ↵

Connected to target database (not started)

RMAN> startup

RMAN> exit

\$ rman target / ↵

Connected to target database: CRM17m-

~~\$ rman~~ RMAN> exit

\$ rman nocatalog target / ↵

RMAN> exit

\$ rman

RMAN> connect target / ↵

RMAN> exit

\$ rman target / ↵

RMAN> show all;

RMAN> Configure retention policy to redundancy 2;

RMAN> show all;

SYS> Select \* from v\$RMAN\_Configuration;

RMAN> Spool log to rman.log

RMAN> show all;

RMAN> Spool off;

RMAN> host 'pwd';

/home/oracle.

RMAN> exit

\$ vi rman.log

1 Configure retention policy to redundancy 2;

2 Configure backup optimization on; # default

4 Configure controlfile auto backup on; (in every backup offfile also taken)  
If it is off then we need to take manually

5 Configure controlfile autobackup format for device type disk to '/u03/rmanbkp/%F';

6 Configure devicetype disk parallelism 2 backup type to backupset;

15 Configure channel 1 device type disk format '/u01/rmanbkp/%U';

16 Configure channel 2 device type disk format '/u03/rmanbkp/%U';

② → optimization on means rman will take the read only table spaces for  
the first time, then it will ignore the next time.

:wq!

# mkdir -p /u03/rmanbkp

# chown -R oracle:oinstall /u03/rmanbkp)

# chmod -R 775 /u03/rmanbkp/

# su -oracle

\$ export ORACLE\_SID=ORCL

\$ rman target + ↪

RMAN> @rman.log

RMAN> show all;

```
RMAN> report schema;
RMAN> report need backup;
RMAN> list backup;      (to display the existing backup)
RMAN> backup datafile 4;
RMAN> report need backup;
RMAN> list backup;
RMAN> report obsolete;
RMAN> backup datafile 4;
RMAN> report need backup;
RMAN> report obsolete;
RMAN> backup datafile 4;
RMAN> report obsolete;
RMAN> delete obsolete;
specify : NO
RMAN> delete noprompt obsolete;
RMAN> report obsolete;
RMAN> list backup;
** RMAN> delete noprompt backup;
RMAN> list backup;
RMAN> backup datafile 4;
RMAN> host 'clear';
RMAN> list backup;
$ cd /u03/rmanbkp/          (.. if it is expired backup we can't
                           change to valid backup)
q }
$ rm -rf *
RMAN> list backup;
RMAN> crosscheck backup;
RMAN> list backup;
RMAN> delete noprompt expired backup;
RMAN> list backup;
```

```
RMAN> report schema;  
RMAN> backup tablespace users, UNDOTBS1;  
RMAN> backup current controlfile; (to take manually control/file)  
RMAN> backup spfile; (to take spfile manually)  
RMAN> backup database plus archivelog;
```

### S-I Loss of full database:-

```
RMAN> report schema;  
$ cd /u02/app/oracle/crms/  
$ ls  
$ rm *  
$ cd /u01/app/oracle/fast-recovery-area/crms/  
$ ls  
$ rm control02.CH
```

```
RMAN> validate database;
```

```
RMAN> shutdown abort
```

```
RMAN> startup nomount
```

```
RMAN> Restore controlfile from '/u02/rmanbkp/C-15619817-20140929-04';
```

```
RMAN> alter database mount;
```

```
RMAN> restore database;
```

```
RMAN> recover database;
```

```
RMAN> S2) 'alter database open resetlogs';
```

### S-II Loss of Non-System datafile:- (log)

```
RMAN> report schema  
$ cd /u02/app/oracle/crms/  
$ ls  
$ rm c01err01.dbf
```

```
RMAN> validate database;
```

```
RMAN> S2) 'alter database datafile 4 offline';
```

```
RMAN> restore datafile 4;
```

```
RMAN> recover datafile 4;
```

RMAN> Sql 'alter database datafile 4 online'; (or)

S-II  
(Hg)

```
RMAN> Report schema;
$ cd /u02/app/oracle/crms/
$ rm user01.dbf
RMAN> validate database;
RMAN> list failure;
RMAN> advise failure;
RMAN> repair failure;
      yes (or)
```

RMAN> @ <PATH>

S-III

Loss of system datafile :-

```
RMAN> Report schema;
$ cd /u02/app/oracle/crms/
$ ls
$ rm system01.dbf
RMAN> validate database;
RMAN> shutdown abort;
RMAN> startup mount;
RMAN> restore datafile 1;
RMAN> recover datafile 1;
RMAN> Sql 'alter database open';
```

S-IV

Loss of Control file:-

```
sys> select name from v$controlfile;
$ cd /u02/app/oracle/crms/
$ rm control01.ctl
$ cd /u01/app/oracle/fast_recovery_area/crms/
$ rm control02.ctl.
```

RMAN> shutdown abort

```
RMAN> start up no mount
RMAN> restore controlfile from '/u01/rmanbkp/C-1516167-05';
RMAN> alter database mount;
```

RMAN> recover database;  
RMAN> set 'alter database open resetlogs';

S-1  
Loss of redo log files:-

sys> select member from v\$logfile;  
\$ cd /u01/app/oracle/  
\$ ls  
~~\*.log~~ ~~\*.ctl~~  
\$ rm \*.log

RMAN> shutdown abort

RMAN> startup mount;

RMAN> restore database;

RMAN> recover database;

RMAN> set 'alter database open resetlogs';

error: Resetlog option only valid after an incomplete database recovery

RMAN> set 'alter database open';

RMAN> list failure;

RMAN> advise failure;

RMAN> repair failure;

Specify?: Yes

7/9/14  
Steps to Configure RMAN in Catalog mode:-

Catalog server setup

MANOJ ENTERPRISES  
Plot: 40, Gasathri Nagar  
Ameerpet, Hyderabad

Target Server setup

- (1) Create one database
- (2) Create one tablespace
- (3) Create one user and grant appropriate privileges to that user
- (4) Connect to catalog database & execute the following command i.e., "Create catalog".  
to create rman metadata tables
- (5) Configure listener.ora
- (6) Configure password file

(9) Configure rman configuration parameters and take the backup of target db.

- (7) Configure tnsnames.ora
- (8) Connect to target database through catalog & register the target database into catalog db.

```
[ssss1 ~]# cat /etc/oratab.      ssh -Y oracle@SSSS1
# su - oracle
$ export ORACLE_SID=catdb
$ sqlplus / as sysdba.

SYS> startup
SYS> select name, open_mode, log_mode from v$database;
SYS> select instance_name, status from v$instance;
SYS> select name from v$tablespace;
SYS> select name from v$datafile;
SYS> create tablespace rman_ts
  2 datafile '/u02/app/oracle/catdb/rman_ts01.dbf' size 10m autoextend on;
SYS) create user rmanuser identified by rmanuser default tablespace rman_ts;
SYS) grant connect, resource, recovery_catalog_creator to rmanuser;
SYS> Conn rmanuser/rmanuser
RMAN USER> select count(*) from tab;
RMAN USER> exit
$ rman catalog rmanuser/rmanuser
RMAN> Create Catalog;
RMAN> exit
$ sqlplus rmanuser/rmanuser
RMAN USER> select count(*) from tab;
" > select * from tab;
# ssh SSSS2
SSSS2 password: oracle dba
SSSS2 ]# su - oracle
$ cat /etc/oratab/
$ export ORACLE_SID=hmu
$ sqlplus / as sysdba
SYS> startup.
```

```

$ cd $ORACLE_HOME/network/admin
$ ls
$ vi listener.ora

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL=TCP) (HOST=ccs2.oracle.com) (PORT=1521))
      # (ADDRESS = (PROTOCOL=IPC) (KEY=EXPR0C1521))
    )
  )
ADR_BASE_LISTENER=/u01/app/oracle
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME=hrms)
    )
  )

```

:wq!

```

$ lsnrctl start listener
$ lsnrctl status listener
$ cd $ORACLE_HOME/ dbs
$ orapwd file=orapwrm password=rac entries=3 force=y
$ su - oracle
$ cd $ORACLE_HOME/network/admin
$ vi tnsnames.ora.

```

TO\_HRMS =

```

  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL=TCP) (HOST=ccs1.2) (PORT=1521))
    )
  )
  (CONNECT_DATA =
    (SERVICE_NAME=hrms)
  )

```

:wq!

\$ tnsping to-hrms

sess1] \$ export ORACLE\_SID=catdb

\$ rman catalog rmanuser/rmanuser target sys/rac@to-hrms.

RMAN> list db\_unique\_name all;

RMAN> register database;

RMAN> list db\_unique\_name all;

RMAN> exit.

\$ sqlplus Rmanuser/rmanuser

RMAN[0]> select \* from rc\_database;

"> exit

~~\$ cat~~

\$ rman catalog rmanuser/rmanuser target sys/rac@ to-hrms

RMAN> show all;

RMAN> spool log to rman.log

RMAN> show all;

RMAN> spool off;

RMAN> exit

\$ vi rman.log → :se n4:

1, 2, 4, 6, 15, 16.

1: Configure retention policy to redundancy 2;

2: Configure backup optimization 'on';

3: Configure Controlfile Auto backup on;

4: Configure device type disk parallelism 2 Backup Type to backupset;

5: Configure channel 1 device type disk connect 'sys/rac@ to-hrms';

6: Configure channel 2 device type disk connect 'sys/rac@ to-hrms';

⋮ wq!

sess1] \$ rman catalog rmanuser/rmanuser target sys/rac@ to-hrms.

RMAN> @rman.log

RMAN> host 'clear';

RMAN> backup database plus archivelog;

If loss of full database :-

RMAN> report schema

sec2 \$ cd /u01/app/oracle/tns/

\$ ls

\$ rm \*

\$ cd /u01/app/oracle/fast-recovery-area/tns/

\$ rm control02.ctl

sec1 RMAN> validate database;

RMAN> shutdown abort

RMAN> startup nomount

RMAN> restore controlfile;

RMAN> alter database mount;

RMAN> restore database;

RMAN> recover database;

RMAN> set 'alter database reset logs';

Creating store script:-

sec1] RMAN> list script names;

RMAN> create script bkp

↳ { backup database 4; }

RMAN> create global script bkp1

↳ { backup database; }

RMAN> list script names;

RMAN> print script bkp;

RMAN> print script bkp1;

RMAN> run { execute script bkp; }

## Virtual private Catalog (VPC) :-

SSS1]\$ export ORACLE\_SID = catdb

\$ sqlplus / as sysdba

sys> create tablespace VPC\_TS

2 datafile '/u01/app/oracle/catdb/VPC\_TS01.dbf' size 10M autoextend on;

sys> create user vpcuser identified by vpcuser default tablespace VPC\_TS;

sys> grant connect,resource,recovery\_catalog\_owner to vpcuser;

sys> exit

\$ rman catalog rmanuser/rmanuser

RMAN> grant catalog for database RMS to vpcuser;

RMAN> exit

\$ rman catalog vpcuser/vpcuser

RMAN> create virtual catalog;

RMAN> exit

\$ sqlplus vpcuser/vpcuser

vpcuser> select count(\*) from tab;

## Enabling block change tracking:-

SSS2]\$ sys> alter database enable block change tracking using file

'/home/oracle/track.lst';

sys> desc v\$block-change-tracking;

sys> select status, filename from v\$block-change-tracking;

sys> alter database disable block change tracking;

## Cloning in RMAN NO Catalog Mode:-

## Target Server : SSS1

- ① Create pfile and send it to test server.
- ② Configure password file
- ③ Configure tnsnames.ora
- ④ Issue duplicate command to clone the database from target server to auxiliary server.

\*\* In password file same password should maintain in both the servers

```
SSS1] # su - oracle
$ export ORACLE_SID=SSS1
$ sqlplus / as sysdba
```

```
SQL> startup
SQL> select name, open_mode, log_mode from v$database;
SQL> select instance_name, status from v$instance;
SQL> select name from v$tablespace;
SQL> select name from v$datafile;
SQL> select name from v$controlfile;
SQL> select member from v$logfile;
```

→ SQL> show parameter spfile;  
→ SQL> create pfile from spfile;

```
SQL> exit
$ cd $ORACLE_HOME/dbs
```

```
$ ls
```

```
$ scp initsss.ora
```

SSS2:/home/oracle

SSS1's password : oracle

initsss.ora

SSS1] # scp initsss2

password : oracle

## SSS2 : Auxiliary server

- ① Create required directory structure.
- ② Start the database in nomount state.
- ③ Configure password file
- ④ Configure Listener.ora

```
** $ cd $ORACLE_HOME/sqlplus/admin
$ ls
$ vi glogin.sql
under glogin.sql
→ set sqlprompt '$-<user>'
description:
: LOB!
```

MANOJ ENTERPRISES  
Plot 40, Gasahri Nagar  
Ameerpet, Hyderabad

scrl] # See -oracle

```
$ pwd  
$ ls  
$ cp initscrs.ora initscrsdup.ora  
$ cp (mv) initscrsdup.ora $ORACLE_HOME/dbs  
$ cd $ORACLE_HOME/dbs  
$ vi initscrsdup.ora
```

eschift : %s/001/003

: %s/002/003

db-file-name-Convert = C:/002/app/oracle/scrs/, 'C:/003/app/oracle/scrs/')

log-file-name-Convert = C:/var/app/oracle/scrs/, 'C:/003/app/oracle/scrs /')

db-name = 'scrsdup'

: we!

\$ exit

```
# mkdir -P /u03/app/oracle  
# mkdir -P orcl/u03/app/oracle/admin/scrs/adump  
# mkdir -P /u03/app/oracle/fast_recovery_area/scrs  
# mkdir -P /u03/app/oracle/scrs  
# chown -R oracle:oinstall /u03/app/oracle/  
# chown -R oracle:oinstall /u03/app/oracle/admin/scrs/adump/  
# chown -R oracle:oinstall /u03/app/oracle/fast_recovery_area/scrs/  
# chown -R oracle:oinstall /u03/app/oracle/scrs/  
# chmod -R 775 /u03/app/oracle/  
# chmod -R 775 /u03/app/oracle/admin/scrs/adump/  
# chmod -R 775 /u03/app/oracle/fast_recovery_area/scrs/  
# chmod -R 775 /u03/app/oracle/scrs/  
# su - oracle  
$ export ORACLE_SID = scrsdup  
$ sqlplus / as sysdba  
SYS> startup nomount
```

ORA:00847: MEMORY\_TARGET not supported on /dev system

```
sys> exit
$ exit
# df -kh
# mount -t tmpfs shmfs -o size=2g /dev/shm.
# df -kh
# su - oracle
$ export ORACLE_SID =scott
$ sqlplus / as sysdba
sys> startup nomount
sys> exit
$ cd $ORACLE_HOME/db
$ oraconfig file=orapwscott password=rac entries=3 force=y
$ cd $ORACLE_HOME/network/admin
$ vi listener.ora
```

LISTENER =

(DESCRIPTION\_LIST =

(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = TCP) (HOST = scott.oracle.com) (PORT = 1521))

# (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))

ADR\_BASE\_LISTENER = /u01/app/oracle

SID\_LISTENER =

(SID\_LIST =

?!

\$ lsnrctl start listener

\$ lsnrctl status listener

scott] \$ cd \$ORACLE\_HOME/db

\$ oraconfig file=orapwscott password=rac entries=3 force=y

\$ cd \$ORACLE\_HOME/network/admin

\$ vi tnsnames.ora

```

TO-SIDUP =
  (DESCRIPTION =
    (ADDRESS-LIST =
      (ADDRESS = (PROTOCOL=TCP) (HOST=SSS2) (PORT=1521))
    )
  )
)

```

:wq!

sssl] \$ export ORACLE-SID=SSS  
\$ rman target sys/rac auxiliary sys/rac@to-sseedup  
RMAN> duplicate target database to seedup from active database;

Note: If we execute nofilenamecheck command, there are both the databases C, R, D file locations are same then we execute this command.

In Catalog mode

\$ rman catalog rmanuser/rmanuser target sys/rac@to-hrms auxiliary sys/rac@to-hrmseedup

\* \* ASM -

log log & Oracle Binaries  
+ ASM Binaries

- 1) Create required node groups & users
- 2) Create oracle home
- 3) Create kernel parameters
- 4) Configure shell limits
- 5) Create required node partitions
- 6) invoke dbca &  
create ASM & Home  
Create database

- 119
- oracle \$w = Oracle Binaries  
grid \$w = ASM Binaries + Clusterware Binaries
- 1) Create required no.of groups & users
  - 2) Create oracle base
  - 3) Create Oracle Inventory
  - 4) Create grid base
  - 5) Create grid ~~base~~ home
  - 6) Configure A Kernel Parameters
  - 7) Configure shell limits
  - 8) Create required no.of disk groups

```
# cat /etc/passwd (check for user)
# cat /etc/group (check for group)
# userdel oracle
# groupdel oinstall
# groupdel dba
```

### (1) Creating the required no of groups & users

```
# groupadd oinstall
# groupadd dba
# groupadd oper
# groupadd asmadmin
# groupadd asmdba
# groupadd asmoper
# useradd -m -g oinstall -G asmadmin,asmdba,asmoper grid
# useradd -m -g oinstall -G dba,asmdba oracle
# passwd grid
```

password : grid  
# passwd oracle

password : oracle

### (2) Create grid base:-

```
# mkdir -P /u01/app/grid
# chown -R grid:oinstall /u01/app/grid/
# chmod -R 775 /u01/app/grid/
```

### (3) Create grid home:-

```
# mkdir -P /u01/product/11.2.0/grid-home
# chown -R grid:oinstall /u01/product/11.2.0/grid-home/
# chmod -R 775 /u01/product/11.2.0/grid-home/
```

### (4) Create oracle base:-

```
# mkdir -P /u01/app/oracle
# chown -R oracle:oinstall /u01/app/oracle/
# chmod -R 775 /u01/app/oracle/
```

### (5) Create oracle home :-

```
# mkdir -P /u01/app/oracle/product/11.2.0/db-home
# chown -R oracle:oinstall /u01/app/oracle/product/11.2.0/db-home/
# chmod -R 775 /u01/app/oracle/product/11.2.0/db-home/
```

(6) Create oracle inventory:-

```
# mkdir -p /u01/app/ora/inventory  
# chown -R grid:oinstall /u01/app/ora/inventory/  
# chmod -R 775 /u01/app/ora/inventory/
```

(7) Configuring shell limits for grid user:-

```
# vi /etc/security/limits.conf
```

grid	soft	nproc	2047
grid	hard	nproc	16384
grid	soft	nofile	1024
grid	hard	nofile	65536

:wq!

(8) # vi /etc/selinux.conf

:wq!

(9) Create the required no of partitions:-

```
# fdisk -f
```

```
# fdisk /dev/sdb
```

press : m ↴

new partition press : n ↴

press : p ↴

partition number (1-4) : 1

First Cylinder (1-521, default): not specify ↴

using default value 1

Last Cylinder : +29 (Specify)

:n ↴

→ c ↴

partition number (1-4) : 2

First cylinder : not specify ↴

Last cylinder : not specify ↴

print free partition →: p ↴

:n ↴

l ↴  
First cylinder : not specify ↴

Last cylinder : +59

- ② : n ↘  
 ↳ ↘ (for logical partitions)
- First : ↘  
 Last : +59 ↘
- ③ : n ↘  
 ↳ ↘  
 First : ↘ ④.⑤.⑥  
 Last : +59 ↘ (up to 6 or 7 logical partitions)  
 : P ↘ (to display the partitions)

WQ

# partprobe (partx for other flavours)

# fdisk -l

#### ⑯ Deleting Content from partitions:-

# dd if=/dev/zero of=/dev/sdb5 bs=8192 count=2560

# dd if=/dev/zero of=/dev/sdb6 bs=8192 count=2560

like this sdb7, sdb8, sdb9, sdb10 ,sdb11

# rpm -qa | grep oracle\* (or) lnc -l

oracleasm-2.6.18-194.el5debug-2.0.5-1.el5

# cd /opt/

# rpm -ivh oracleasm-2.0.4-fc15.i386.rpm.

# cd

# rpm -qa | grep oracle\* lnc -l

8

#### ⑰ Configuring oracle ASM:-

# oracleasm configure -i

Default user to own the driver interface : grid

Default group to own the driver interface : asmadmin

: y

: y

: done

# df -ah

# oracleasm init

```
# df -ah  
# cd /dev/oracleasm/  
# ls  
# cd disks/  
# ls  
#
```

### Creating volumes:-

# oracleasm createdisk	VOL1	/dev/sdb5
# oracleasm createdisk	VOL2	/dev/sdb6
# oracleasm createdisk	VOL3	/dev/sdb7
# oracleasm createdisk	VOL4	/dev/sdb8
# oracleasm createdisk	VOL5	/dev/sdb9
# oracleasm createdisk	VOL6	/dev/sdb10
# oracleasm createdisk	VOL7	/dev/sdb11

```
# oracleasm scandisks  
# oracleasm listdisks  
# cd /dev/oracleasm/  
# ls  
# cd disks/  
# ls
```

✓ # oracleasm querydisk /dev/sdb5  
# b/kid

(to check the partitions with volumes)

### Installing grid software:-

```
# cd /opt  
# ls  
# unzip p10098816-112020-LINUX-1of7.ZIP  
# unzip ....." ....." -2of7.ZIP  
# unzip ....." ....." -3of7.ZIP  
# xhost +  
# scp -grid
```

```
[grid]$ vncpasswd  
passwd:oracle  
$ vnc Server  
$ vnc viewer
```

VNC Server: [lnx01.oracle.com : 1] → [OK]

password: oracle

\$ /opt/grid/runInstaller

○ Skip software updates → Next

○ Configure Oracle Grid Infrastructure for a Standalone Server

→ Next → Next

\* Disk group name [ASM - DATA]

Redundancy

○ External

ORCL: VOL1

→ Next

○ Use same password for these accounts

password: [oracle]

Confirm: [oracle]

→ Next → Yes → Next

(grid base) oracle base: /u01/app/grid

(grid base) software location: /u01/product/11.2.0/grid-home → next → Yes

oracle inventory: /u01/app/inventory

→ next → Install

Execute scripts

# /u01/app/inventory/orainstRoot.sh

# /u01/product/11.2.0/grid-home/root.sh

→ OK → Close

\$ # su - grid

\$ ps -ef|grep smon

Configuring bash profile:-

\$ vi .bash-profile

PATH=\$PATH:\$HOME/bin → under this

{ export ORACLE\_HOME=/u01/product/11.2.0/grid-home

{ export PATH=\$ORACLE\_HOME/bin:\$PATH:\$HOME/bin

:wq!

\$ . .bash-profile

\$ crsctl check has

\$ crsctl check css

\$ crs\_stat -t

## Installing oracle software:-

```
# xhost +  
# su - oracle  
$ vncpasswd  
password: oracle  
$ vncserver  
$ vncviewer
```

VNC Server: inx01.oracle.com :1

OK

password: oracle ↵

\$ /opt/database/runInstaller

I wish to receive security updates via oracle support → next → Yes

skip software updates → next

① Install database software only → next

② Single instance database installation → next → next

③ Enterprise edition → next

oracle base: /u01/app/oracle

oracle home: /u01/app/oracle/product/11.2.0/db-home → next → next

click on

Fix & check again

open terminal

# /tmp/cvU-11.2.0.2.0-oracle/runFixup.sh

click on 'ok' → Install

→ # /u01/app/oracle/product/11.2.0/db-home/root.sh

click on 'ok' → click on close

## Creating Disk groups:-

```
# xhost +  
# su - grid  
$ ps -ef | grep smon
```

```
$ export ORACLE_SID = +ASM  
$ sqlplus / as sysasm  
SQL> select * from v$diag_info;  
# cd /u01/app/grid/diag/asm/+asm/+ASM/trace/  
# tail -f alert+ASM.log  
$ which asmca  
/u01/product/11.2.0/grid-home/bin/asmca
```

At ole level  
to create diskgroup \$ asmca

Click on create

Disk group name : **ASM-DG-DATA**

External  
 ORCL:VOL2 → Click on 'OK' → 'OK' → Exit → Yes

```
$ export ORACLE_SID = +ASM  
$ sqlplus / as sysasm  
SQL> CREATE DISKGROUP ASM-DG-DATA FRA EXTERNAL REDUNDANCY  
DISK 'ORCL:VOL3';
```

SQL> EXIT

Adding a disk to existing disk group :-

\$ asmca  
Right click on **ASM-DG-DATA** → Click on add disk  
 ORCL:VOL4 → Click on 'OK' → OK → Exit → Yes

```
$ sqlplus / as sysasm  
SQL> ALTER DISKGROUP ASM-DG-DATA FRA ADD DISK 'ORCL:VOL5';
```

SQL> desc v\$asm\_diskgroup;

SQL> select group\_number, name, block\_size, state from v\$asm\_diskgroup;

SQL> desc v\$asm\_disk;

SQL> select group\_number, disk\_number, state from v\$asm\_disk;

```
$ asmcmd  
ASMCMD> ls  
ASMCMD> lsdg  
ASMCMD> spget  
+ASM-DATA/asm/asm parameterfile/registry.253.8671153.
```

```
ASMCMD> lsdsk  
ASMCMD> lsdsk --candidate -P (to check free disks(volumes))  
ASMCMD> lsdsk -p -G asm-dg-data (to check the disks under group)  
ASMCMD> exit
```

```
[oracle@nx01 ~]$ dbca
```

```
$ vi .bash_profile  
PATH=$PATH:$HOME/bin → under this  
export ORACLE_HOME=/u01/app/oracle/product/11.2.0/db-home  
export PATH=$ORACLE_HOME/bin:$PATH:$HOME/bin
```

:w!

```
$. .bash_profile
```

Creating a database:-

```
$ dbca
```

→ Next

① Create database

→ Next

② General purpose

→ Next

Global database name: [CRM] → next → next

Use the same administrative password for all accounts

password: [oracle]

→ next → yes

Storage type: [ASM]

Use common location for all database files

Click on [Oracle]

Select ② ASM-DG-DATA → OK → next

specify ASHSNTP password for : oracle

specify fast recovery area → click on Browse

select  ASM-DG-FRA

size : u001 : OK

enable Archiving → next → next → next → Finish

→ Click on 'OK'

→ Click on exit

Configuring ACFI ASM Cluster File System :-

[root@ln201 ~]# lsmod | grep oracle

oracle asm 4600 1

# cd /u01/product/11.2.0/grid-home/bin

# ./acfsload start -S

# cd

# lsmod | grep oracle

# su - grid

# asmca

click on volumes → click on create

volume name: acfs-volume

diskgroup name: ASM-DG-DATA

size : 1 GB

OK → click on OK

click on ASM cluster file system → click on create

General purpose file system

Mount point /acfs

Click on OK → click on OK

MANOJ ENTERPRISES  
Plot: 40, Gasathri Nagar  
Ameerpet, Hyderabad

```
# cd /
```

```
# mkdir /acfs
```

```
# acfs
```

Click on **Show mount all Command** and execute command as root user

```
# /sbin/mount.acf -o all
```

```
# df -kh
```

Click on **Show dismount all Command**

```
# /bin/umount -t acfs -a
```

```
# df -kh
```

[oracle@nx01 ~]\$ srvctl status database -d crs

\$ srvctl config database -d crs

\$ srvctl stop database -d crs

\$ su - grid

password : grid

\$ crsctl stop has

\$ crsctl start has (or)

\$ export ORACLE\_SID=+ASM

\$ sqlplus / as sysasm

SQL> start up

## \* DATAGUARD \*

Configuring standby database in MRP mode:-

sql> desc v\$archived\_log;

sql> desc v\$archive\_dest\_status;

sql> select recovery\_mode from v\$archive\_dest\_status;

sql> alter database recover managed standby database disconnect from session;

sql> select recovery\_mode from v\$archive\_dest\_status;

sql> select name from v\$tablespace;

SQL] sys> select name, protection-mode, controlfile-type, database-role, switchover-status from v\$database;

<u>NAME</u>	<u>protection-mode</u>	<u>Control</u>	<u>Db-role</u>	<u>switchover-status</u>
HRMS	Max. performance	Current	PRIMARY	To Standby

sys> save pri.sql replace;

SQL] sys> select name, protection-mode, controlfile-type, database-role, switchover-status from v\$database;

<u>NAME</u>	<u>protection-mode</u>	<u>Control</u>	<u>db-role</u>	<u>switchover-status</u>
HRMS	Max. performance	Standby	Physical Standby	Not allowed

sys> save std.sql replace;

Converting protection mode from performance mode to protection mode -

SQL(Standby)

(3) sys> shut immediate

sys> exit

\$ cd \$ORACLE\_HOME/dbs

\$ vi inithrms.ora ↵

18kt2=lgwr sync affirm

:wq!

\$ sqlplus / as sysdba

(4) sys> startup mount

sys> alter database set standby database

to maximize protection;

sys> alter database open;

(1) sys> alter database recover managed  
standby database cancel;

(2) sysalter database add standby logfile;

sys> alter database add standby logfile;

sys> alter database add standby logfile;

Converting protection mode from max. protection mode to max. availability mode :-

sssl1 (primary)

(2) `shut immediate`

(3) `sys> startup mount`

`sys> alter database set standby  
database to maximize availability;`

`sys> alter database open;`

sssl2 (standby)

(1) `sys> alter database recover managed standby  
database cancel;`

Converting protection mode from max. Availability to max. performance mode :-

sssl1 (primary)

(2) `shut immediate`

(3) `sys> startup mount`

`sys> alter database set standby  
database to maximize performance;`

`sys> alter database open;`

sssl2 (standby)

(1) `sys> alter database recover managed standby  
database cancel;`

switch-over process:-

sssl1 (primary)

(2) ~~start fast~~

(1) `sys> alter database commit to Switchover  
to physical standby;`

(2) `sys> shut immediate;`

`sys> exit`

(3) `$ sqlplus / as sysdba`

`sys> startup no mount`

sssl2 (standby)

(3) `alter database commit to switchover  
to physical primary;`

(4) `sys> shut immediate;`

`sys> exit`

`$ sqlplus / as sysdba`

`sys> start up`

5/11/14

## \* RAC \*

+ ~~clusterware~~

q1 RAC is very expensive solution.

To Configure q1 RAC customer need to take license of

① Certified version of OS - sun Solaris - q1

② oracle database

③ ORACLE RAC

④ cluster file system (CFS)

(Veritas CFS, GPF (General parallel file system) ... etc)

⑤ third party volume manager

(Veritas VM, Tivoli VM ... etc)

from log, we can avoid CFS, VM by introducing ACM

→ To minimize the cost to the customer, in version q1 oracle has introduced its own cluster file system (CFS) for LINUX & WINDOWS i.e., "OCFS" (oracle cluster file system)

→ OCFS is free of cost and the latest version is OCFS-2

	<u>db</u>	<u>RAC</u>	<u>No. of nodes</u>
DBE	\$7500	\$20000 \$21500 (Jan '13) \$14,000 (1 node), 11gR2 → 1	100
DSE	\$17500	free	4

Data Guard not supported  
Partitioning  
Bitmap index we can't create

Any Edition  
→ DB

- Exadata is an enterprise ready sun oracle db machine
- Engineered by oracle engineers.

→ volume of data is less

→ most of the time we execute

SQL statement

- ① oracle
- ② MS SQL server
- ③ DB2
- ④ MySQL

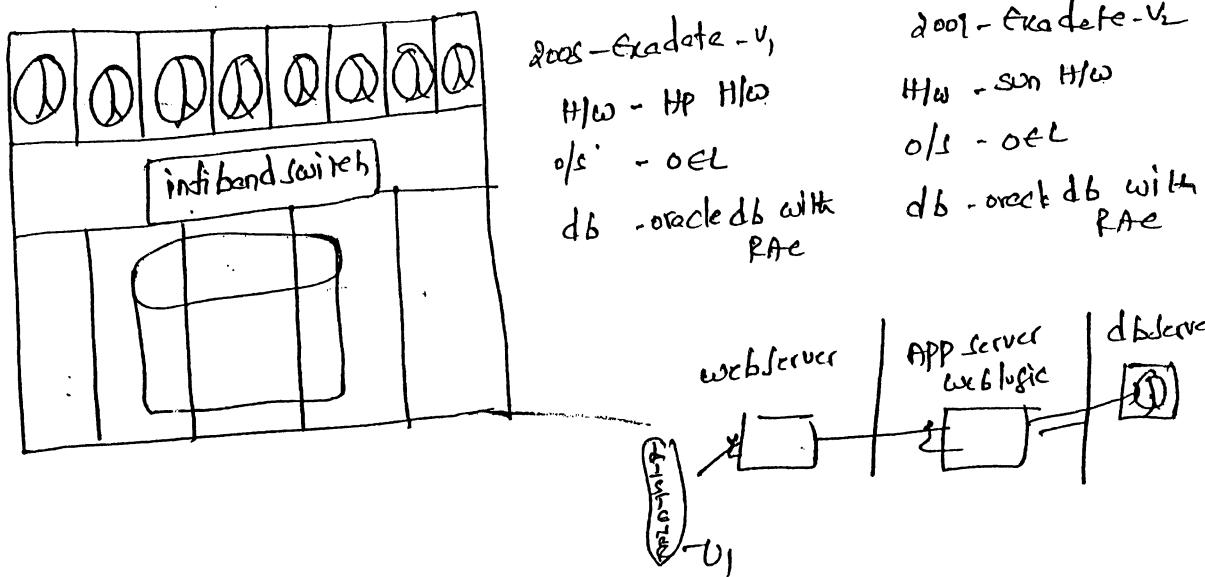
## OLAP/DW/DS

→ volume of data is huge (TB, PB)

→ most of the time we execute select

- ① Teradata
- ② Oracle
- ③ Netezza

→ Grid computing is a world joined ab machine.

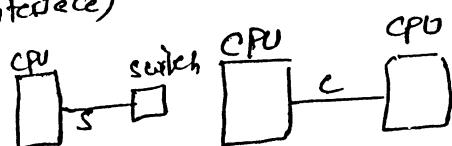


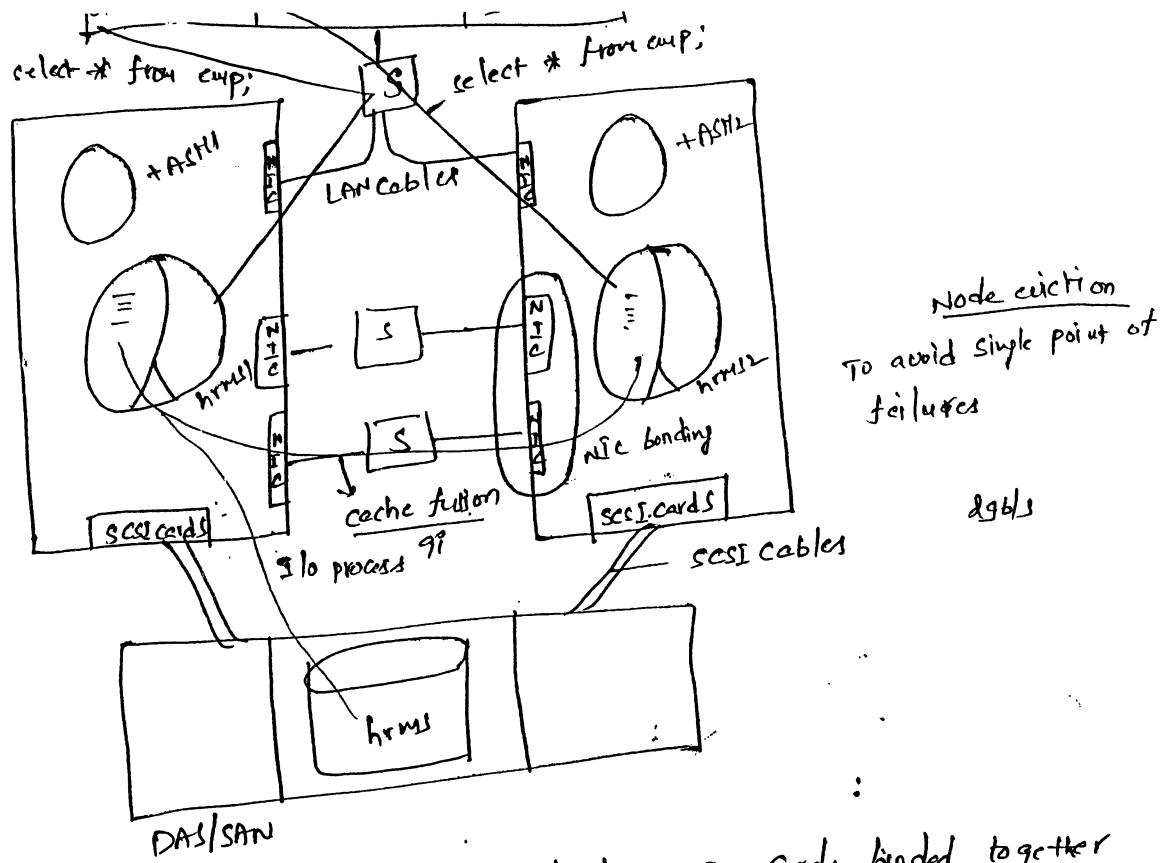
### Cluster:-

→ two (or) more nodes interconnected to each other and serving a common task is called a cluster.

### Cluster hardware Components:-

- ① two (or) more nodes
- ② shared storage (DAS, NAS, SAN)
- ③ Minimum 2 NIC cards per each node. (Network interface cards)
- ④ minimum 2 switches
- ⑤ SCSI Cards (Small Computer System Interface)
- ⑥ SCSI cables [straight cables]
- ⑦ LAN cables [cross cables]





- In some customer environments, we see redundant NIC cards binded together to improve the performance of cache fusion.
- In some customer environments, we see redundant NIC cards as cluster inter connects to avoid single point of failures.

softwares required to setup RAC environment:-

- ① Certified version of O/S
- ② Cluster software

third party cluster software available in the market:-

- ① Microsoft Windows cluster software
- ② Red Hat cluster software
- ③ HP Service Guard
- ④ HACMP (IBM) (High availability cluster multiplexing)
- ⑤ Veritas cluster fw.
- ⑥ Sun cluster fw.

### oracle version

9i

10gR1

[ 10gR2  
= 11gR1 ]

[ 11gR2  
= 12cR1 ]

### cluster software

cluster Manager

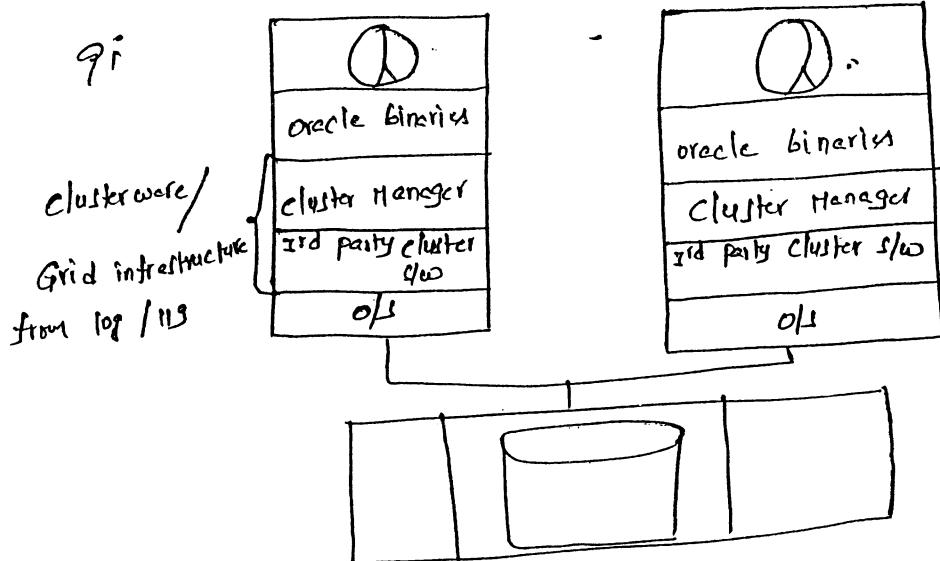
CRS s/w (cluster Ready Service)

clusterware

clusterware

Grid infrastructure services

Grid infrastructure services.



### Types of clustering Technology:-

1. Active/Passive
2. Active/Active

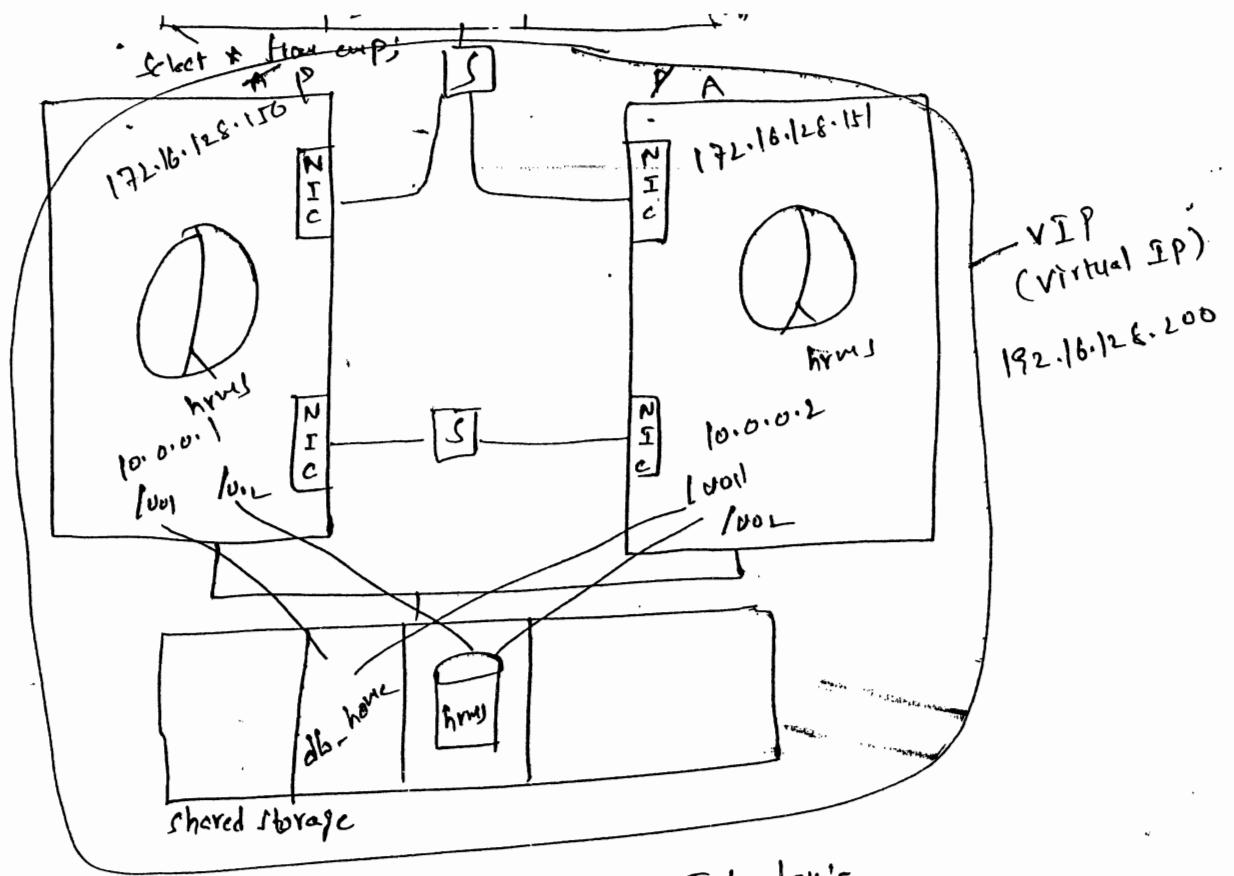
#### 1. Active/Passive CT:-

- ① Certified version of o/s - Sun Solaris
- ② cluster s/w - sun cluster s/w.

two hosts

CDP description :-

(Address list = (Protocol = TCP) (Host = 192.16.128.200) (Port = 1521))  
(SID = hrms)

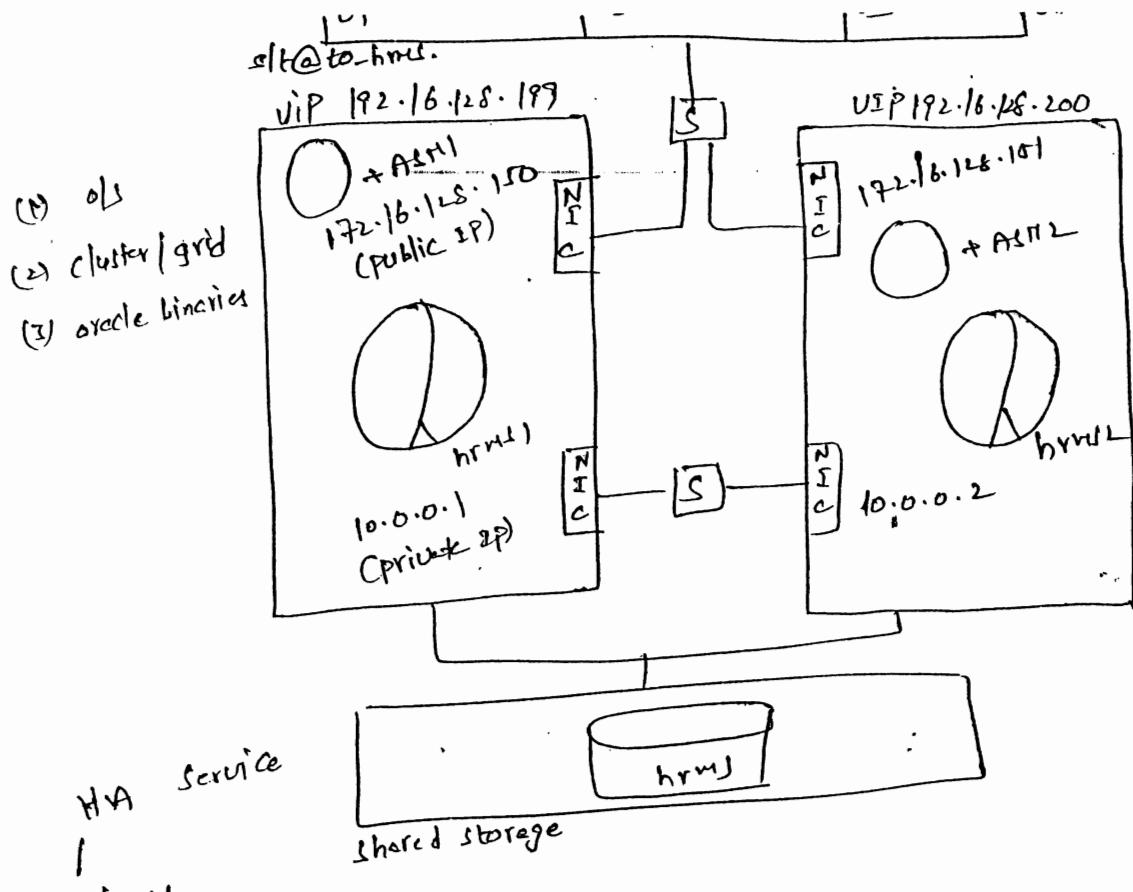


### Disadvantages of Active/Passive Clustering Technology:-

- It's an expensive solution with respect to hardware infrastructure.
- There is no load balancing.
- No scalability (adding a node)
- No better performance as the customer count increases.
- Note: To overcome the above disadvantages, vendors have introduced Active/Active C.T.
- In 10g, Oracle has introduced Active/Passive solution for Windows that is called "Oracle fail safe".
- In 11gR2, Oracle has introduced Active/Passive solution for Linux "RAC one node".

### 2. Active/Active:-

Ex:- RAC



to-hrm3 =

C Description =

C Address list = (Protocol = TCP) C host = 192.16.128.199 (PORT = 1521))

C Address list = (Protocol = TCP) C host = 192.16.128.200 (PORT = 1521))

C load\_balance = yes)

C Connect\_data =

C server = Dedicated)

C service\_name = to-hrm3)

C failover\_mode =

C type = select)

C method = basic)

C Retries = 150)

C delay = 5)

Advantages of Active/Active clustering Technology:-

- High availability
- scalability (Adding a node)
- load balancing
- Fail over
- It's not an expensive solution with respect to hardware infrastructure.

Note: If we use cross cable as a cluster interconnect for private inter connect, we are restricting the scalability of RAC & nodes.

→ whether it is active-passive (or) active/active, we can implement the clustering technology at the following layers

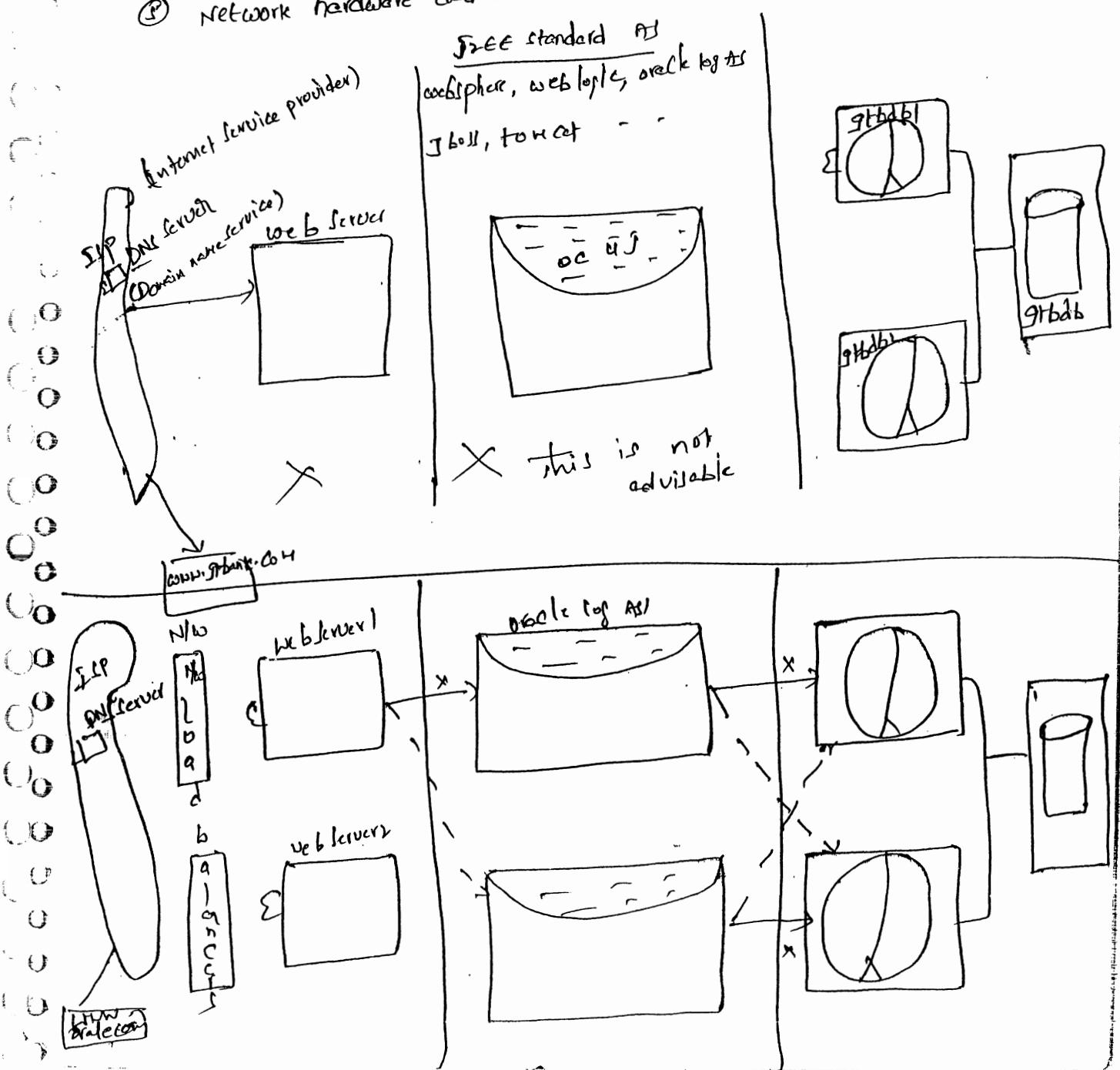
① O/S layer

② Database layer

③ Application server layer (Middle tier)

④ Web servers layers

⑤ Network hardware load balancer layer.

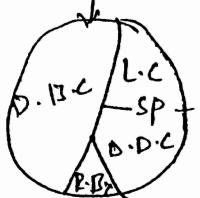


## Differences between stand alone db & RAC db:-

### stand alone db

- (1) this is a single instance database
- (2) one pfile/spfile required

$$(3) \text{ instance} = \text{SGA} + \text{BP}$$

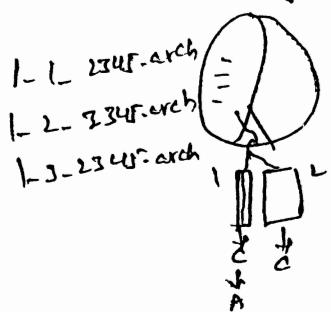


- (4) SMON, PMON, CKPT, DBWR, LGWR, Recovery

- (5) minimum no.of online redo log groups that are required to create a db are "2"

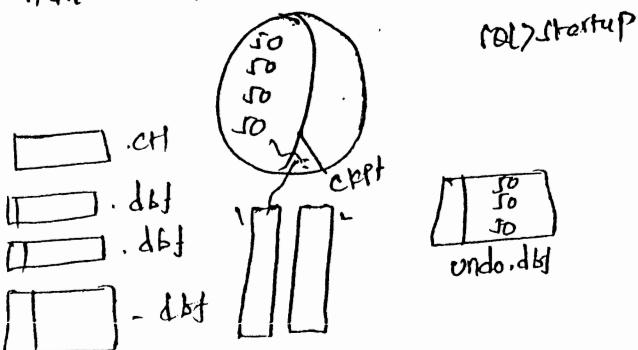
$$(6) \text{log\_archive\_format} = \text{Y.t Y.S Y.R . arch}$$

thread no.                                    redo logid:  
instance no.                                    log sequence no.



- (7) one default undo tablespace.

- (8) SHON performs instance recovery during next startup

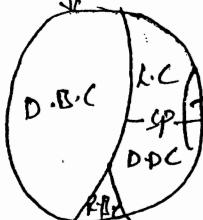


### RAC db

- (1) this is a multi instance database

- (2) each instance has it's own pfile/spfile

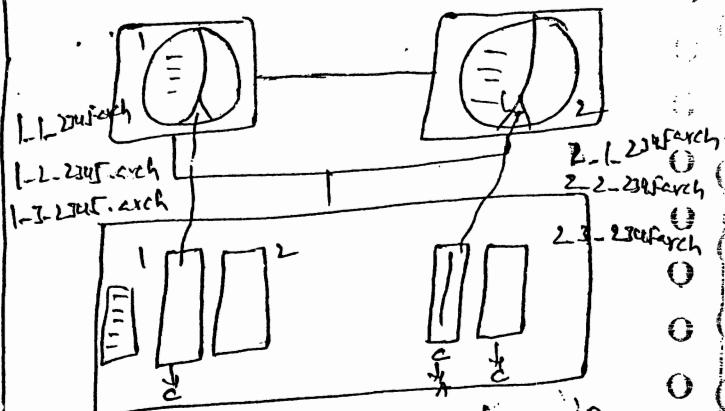
$$(3) \text{ instance} = \text{SGA} + \text{BP}$$



GRD  
(Global resource directory)

- (4) SMON, PMON, CKPT, DBWR, LGWR, Reco, LMON, LSTNM, LCK, LMD, diag.

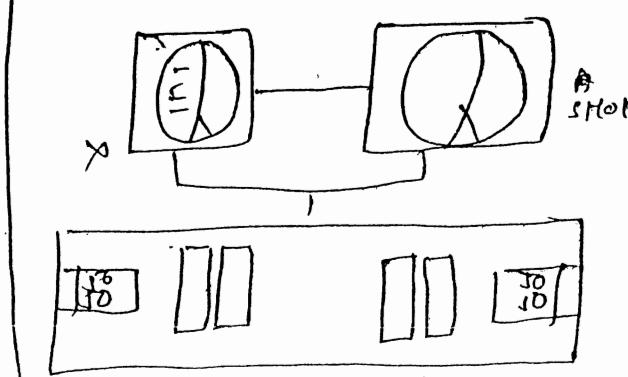
- (5) each instance has it's own set of online Redo log groups.



→ Recommended to place archive logs in shared storage not in local disk. Because whenever we are performing recover operations it will take maximum down time to the customer.

- (6) Each instance has it's own undo.

- (7) surviving instance SHON performs instance Recovery of dead instance.



Note: once the instance crashes, GKD gets freed and RAC system goes to hung state. RAC system will be in hung state until instance Recovery happens.

⑩ Alter system switch logfile;

⑪ x

| ⑩ Alter system archive log current;

| ⑪ HA, scalability, load balancing, failover & better performance

## RAC Database storage options:-

9i  
- RAW  
CFS

log - 11g R1  
RAW  
CFS  
ASM

(99.y.) RAC installation  
on ASM.

11g R2 - 12c R1  
CFS  
ASM.

### Raw partition:-

→ unformatted partition is called Raw partition.

### Limitations of Raw partitions:-

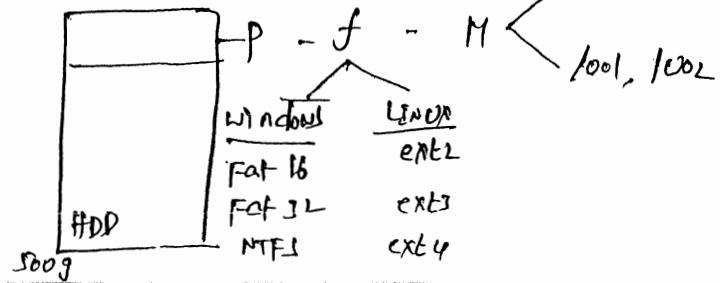
- In a disk the max. no. of Raw partitions that we can create are 256.
- In a single Raw partition, we can place only one file.
- once the Raw partition is created, we cannot shrink (or) we can't expand.
- placing CRD files in Raw partitions gives better performance but manageability wise it is very complex to manage the environment.

Note: To overcome the above limitations, in version 9i most of the customers they use to depend on third party cluster file systems like Veritas CFS, General parallel file system etc.

→ In version 9i, oracle has introduced its own cluster file system for Linux & windows i.e., OCFS (Oracle Cluster File System)

→ OCFS is free of cost and the latest version is OCFS 2

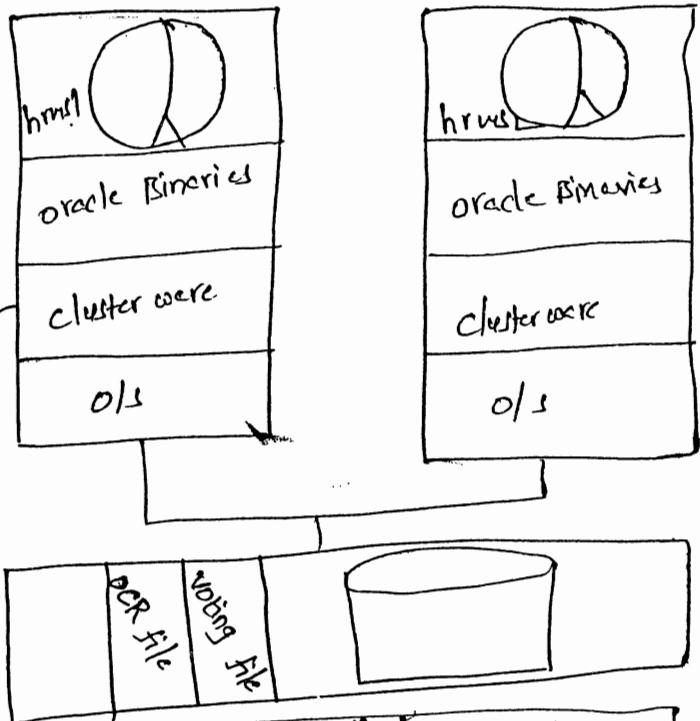
→ ASM = CFS + VM + RAID



→ The functionality of CFS is, we can mount the shared disk to multiple nodes.

- ① O/S
- ② clusterware
- ③ oracle binaries

- ① OCR file ✓
- ② Voting file/VFD ←
- ③ RAW ✓
- ④ CFS ✓
- ⑤ ASM ✗



	OCR file	voting file	CRD	pfile/spfile	Archivelogs
RAW	✓	✓	(but not recommended)	✓	(but not recommended)
CFS	✓	✓	✓	✓	✓
ASM	✗ <small>(ASM is possible from 11g RL)</small>	✗	✓	✓	✓

$$\frac{log R_2 - log R_1}{\text{Clusterware slo}} = \text{Clusterware binaries}$$

$$\text{oracle slo} = \text{ASM binaries + oracle binaries}$$

$$\frac{log R_L - log P_1}{\text{Grid infrastructure services}} = \text{ASM binaries + clusterware binaries}$$

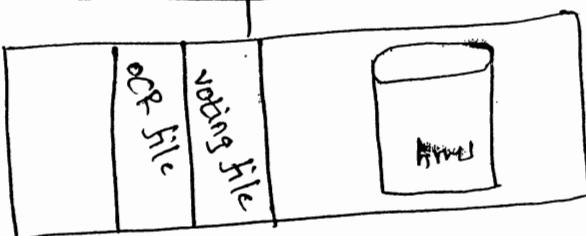
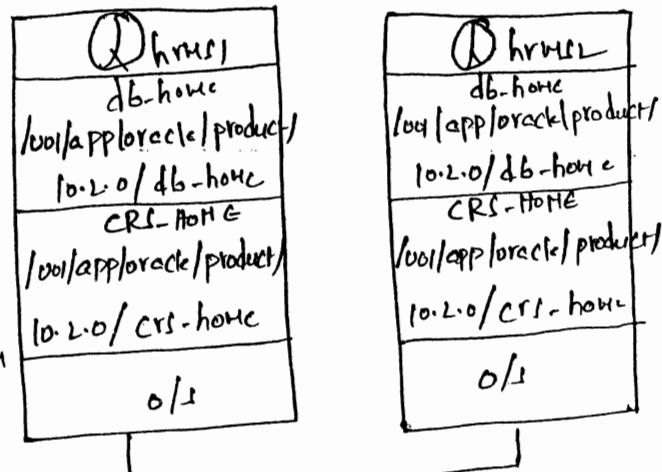
$$\text{oracle slo} = \text{oracle binaries}$$

- ① OIS
- ② clusterware - CRS-HOME
- ③ oracle binaries - db-home

prerequisite to

- ① Trust relationship between the nodes
  - (a) ssh
  - (b) rsh

- ② location should be same in both the nodes



### Configuring ssh for user equivalence & Trust relationship:-

lnx01

User : oracle

default directory: /home/oracle

\$ mkdir .ssh (.: indicates hidden directory)

\$ /usr/bin/ssh -keygen -t rsa  
id-rsa id-rsa.pub

\$ /usr/bin/ssh -keygen -t dsa  
id-dsa id-dsa.pub

\$ cat id-rsa.pub > lnx01

\$ cat id-dsa.pub >> lnx01

& scp lnx01 lnx02:/home/oracle/.ssh

pwd:

→ authorized\_keys

\$ chmod 644 authorized-keys

lnx02

User : oracle

/home/oracle

\$ mkdir .ssh

\$ /usr/bin/ssh -keygen -t rsa  
id-rsa id-rsa.pub

\$ /usr/bin/ssh -keygen -t dsa  
id-dsa id-dsa.pub

→ lnx01

\$ cat id-rsa.pub > lnx01

\$ cat id-dsa.pub >> lnx01

\$ cat lnx01 > authorized-keys

\$ cat lnx02 >> authorized-keys

\$ scp authorized-keys lnx01:/home/oracle/.ssh

\$ chmod 644 authorized-keys.

## High availability RAC implementation steps.

① Install the same version of O/S in all the nodes.

② Create the required root groups & oracle user account.

Note: group id & user id should be same in all the nodes

③ Create the required directory structures for CRS-HOME & DB-HOME.

(Naming convention should be same in all the nodes)

④ Configure kernel parameters as per the installation doc.

⑤ Set shell limits for oracle user account.

⑥ Edit /etc/hosts and specify public IPs, private IPs & virtual IPs.

⑦ Configure ssh for user equivalence and trust relationship.

⑧ Create the required number of partitions in the shared storage. for OCRfile, voting file and ASM diskgroups.

⑨ If we are implementing RAC on Linux using OCFS2 and ASM lib.

interface, download and install OCFS2 & ASM rpm's in all the nodes.  
(otn-oracle.com)

⑩ See that date & time is as closed as possible in all the nodes.

Note: If we are implementing RAC in a huge environment, generally system admin is going to configure "NTP" (Network time protocol) server to synchronize date & time in all the nodes.

⑪ Install clusterware binaries in the first node.

⑫ Install oracle binaries in the first node.

⑬ Invoke netca and configure the listener.

⑭ Invoke dbca and configure ASM and create the required number of ASM disk groups.

⑮ Invoke dbca & create RAC database.

Preliminary checks that needs to be carried out once the system admin handover the system :-

```
[nxo1] # ifconfig -a |more  
eth0 : IP: 172.16.128.150 (public IP)  
eth1 : 10.0.0.1 (private IP)  
# ping 172.16.128.150  
# ping 10.0.0.1  
# fdisk -l  
# ssh 172.16.128.151
```

```
[nxo1] # ifconfig -a |more  
# ping 172.16.128.151  
# ping 10.0.0.2  
# fdisk -l  
[nxo1] # ping 172.16.128.151  
# ping 10.0.0.2
```

## Implementation of LogRac on RHEL AS 4 :-

- Node 1      Node2
- ① Creating the required no.of groups & oracle user account
    - # groupadd dba
    - # useradd -m -g oracle
    - # passwd oracle
    - # id oracle
    - # id =500 (oracle) group=500 (oinstall), 501 (dba)
    - # id =500 (oinstall) group=500 (oinstall), 501 (dba)
  - ② # groupadd -g 500 oinstall
    - # groupadd -G 501 dba
    - # useradd -m -g 500 -g oinstall -G dba oracle
    - # passwd oracle
    - # id oracle
    - # id =500 (oracle) group=500 (oinstall), 501 (dba)
  - ③ Creating the required directory structures for CBS-HOME
    - # mkdir -p /u01/app/oracle/product/10.2.0/cbs-home
    - # mkdir -p /u01/app/oracle/product/10.2.0/db-home
    - # chown -R oracle:oinstall /u01/app/oracle/product/10.2.0/cbs-home
    - # chown -R oracle:oinstall /u01/app/oracle/product/10.2.0/db-home/
    - # chmod -R 755 /u01/app/oracle/product/10.2.0/cbs-home/
    - # chmod -R 775 /u01/app/oracle/product/10.2.0/db-home/
  - ④ Configure Kernel parameters as per the installation doc.
    - # vi /etc/sysctl.conf
    - kernel.shmall = 2097152
    - kernel.shmmax = 536870912
    - kernel.shmmni = 4096
    - kernel.sem = 256 1000 100 128
  - ⑤ # /sbin/sysctl -P

**MANOJ ENTERPRISES**  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

```
# net.ipv4.ip_local_port_range = 65536 65536
# net.core.rmem_default = 268435456
# net.core.wmem_default = 268435456
# net.core.rmem_max = 1048576
# net.core.wmem_max = 1048576
# net.core.commen_max = 1048576

: w2!
# /sbin/depmod -p   (to reflect the parameters into kernel)
# cd /etc/
# scp /etc/security/limits.conf 172.16.128.151:/etc/security/limits.conf
# pod: racdba

⑦ set shell limits for oracle user account.
# vi /etc/security/limits.conf
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
: w2!
# cd /etc/security/
# scp limits.conf 172.16.128.151:/etc/security/limits.conf
# pod: racdba
```

② To apply the law for all the shells

```
# vi /etc/profile
if [ $USER = "oracle" ]; then
if [ $SHELL = "/bin/ksh" ]; then
ulimit -p 16384
ulimit -n 65536
else
ulimit -u 16384 -n 65536
fi
umask 022
fi
```

```
:wq!  
# cd /etc/  
# scp profile 172.16.125.151:/etc/profile
```

⑨ Configuring hang check timer:-  
Note:- Applicable only for Linux, mandatory in version 9i, optional in 11g.

```
# vi /etc/rc.d/rc.local  
modprobe hangcheck-timer  
hangcheck_tick=50 hangcheck_margin=180
```

co2! # cd /etc/rc.d  
# scp rc.local 172.16.123.11 :/etc/rc.d/rc.local  
# cat rc.local

**MANOJ F**  
Plot. of  
Ar

## ⑩ Configuring /etc/hosts

# vi /etc/hosts

127.0.0.1 localhost.localdomain

# # PUBLIC IP's

172.16.128.150 lnx01  
172.16.128.151 lnx02

# # PRIVATE IP's

10.0.0.1 lnx01 -priv  
10.0.0.2 lnx02 -priv

# # VIP's

192.16.128.199 lnx01-vip  
192.16.128.200 lnx02-vip

:wq!

# cd /etc/  
# scp hosts 172.16.128.151:/etc/hosts

pod: racdb.e

Configuring ssh for user equivalence and trust relationship

⑪

# su - oracle  
# mkdir ~/.ssh  
\$ chmod 755 ~/.ssh

\$ /usr/bin/ssh-keygen -t rsa -f id\_rsa -N ""  
\$ /usr/bin/ssh-keygen -t rsa -f id\_dsa -N ""  
\$ cat id\_rsa.pub > lnx01

\$ cat id\_dsa.pub >> lnx02

⑫

# su - oracle  
\$ mkdir ~/.ssh  
\$ chmod 755 ~/.ssh  
\$ /usr/bin/ssh-keygen -t rsa -f id\_rsa -N ""  
\$ cat id\_rsa.pub > lnx01  
\$ cat id\_dsa.pub >> lnx02

```
$ cat id-rsa.pub > lnx01  
$ cat id-dsa.pub >>lnx01
```

```
$ cd .ssh  
$ ls  
$ Cat lnx01 > authorized-keys  
$ Cat lnx02 >>authorized-keys  
$ chmod 644 authorized-keys  
$ scp authorized-keys lnx01:/home/oracle/.ssh/  
? yes
```

Prod : oracle

(15) Establishing user equivalence & testing password less connectivity.

```
$ exec /usr/bin/ssh-agent $SHELL  
$ /usr/bin/ssh-add
```

(16) → \$ ssh 172.16.128.150

```
$ exit  
$ ssh lnx01  
$ exit  
$ ssh lnx01.oracle.com $exit  
$ ssh 172.16.128.151 $exit  
$ ssh lnx01 $exit  
$ ssh lnx01.oracle.com $exit  
$ ssh 10.0.0.1 $exit  
$ ssh lnx01.priv $exit  
$ exit
```

(17) Same as step no. 14

```
$ scp lnx02 lnx01:/home/oracle/.ssh  
? yes
```

Prod : oracle

```
$ cd .ssh  
$ ls  
$ chmod 644 authorized-keys  
$ ls  
$ chmod 644 authorized-keys  
? yes
```

(18) → \$ exec /usr/bin/ssh-agent \$SHELL

```
$ /usr/bin/ssh-add
```

(19) Same as step no. 14

\$ exit

10.0.0.2

such measures as possible.

(16) check whether date&time is as close as possible

\$ date

 Creating file required no. of partitions in the shared drive.

- ① occ file
- ② voting file
- ③ ACH - DG - DATA
- ④ ACH - DG - FRA

oui screen  
speci

Species	SCP	Location
deer	/day /ray /reef'	/daw /cedar x
moose	/deu /reef /reef6	/deu /reef x

near location	Bhuv / Pdhu / Chittagong	Mymen / Rangpur / Chittagong
---------------	--------------------------	------------------------------

→ \$date

others — /dev/rdsr /c1t1d0

1 /dev/rdsk/c1t1d0s1  
 1 /dev/rdsk/c1t1d0s2  
 1 /dev/rdsk/c1t1d0s3

卷二 屏幕

dev location	<code>[dev]/[disk]/[C:\disk\000]</code>
target	<code>[dev]/[disk]/[C:\disk\001]</code>

# fdisk -l

# fdisk /dev/sdb

: p (to print the partition table)  
in (new partition)

p (primary)

: 1  
First cylinder : ↗  
last cylinder : +19

: n

e (extended)  
partition number : 2  
First cylinder : ↗  
last cylinder : ↗  
: p

: n  
(logical)  
First cylinder : ↗  
last cylinder : +19

: n  
First cylinder : ↗ last cylinder : +19  
To do this up to 3 times. after that in the next &  
logical ~~with~~ partition enter last cylinder at : +89 in place  
of "+19"

: P  
!w. ↗

```
# part probe
```

(e)

Hopping & Binding Raw partitions to Raw device files

```
# vi /etc/sysconfig/rawdevices
```

```
/dev/raw/raw5 /dev/sdb5  
/dev/raw6 /dev/sdb6  
/dev/raw7 /dev/sdb7  
/dev/raw8 /dev/sdb8  
/dev/raw9 /dev/sdb9
```

```
:wq!
```

```
# cd /etc/sysconfig/  
# cp rawdevices Linux:/etc/sysconfig/rawdevices
```

```
# chmod +x rawdevices
```

```
# service rawdevices restart.
```

(f) Changing ownership & setting proper permissions on Raw partitions.

```
# chown -R oracle:oinstall /dev/raw/raw*
```

```
# chmod -R 777 /dev/raw/raw*
```

(g) Changing ownership & setting proper permissions on Raw partitions during system boot-up

```
# vi /etc/rc.d/rc.local
```

```
chown -R oracle:oinstall /dev/raw/raw*
```

```
chmod -R 777 /dev/raw/raw*
```

```
:wq!  
# cd /etc/rc.d  
# cp rc.local Linux:/etc/rc.d/rc.local chmod +x rc.local
```

**MANOJ ENTERPRISES**  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

## INSTALLING ORACLE CLUSTERWARE.

Node 1 :-

```
[lnx01] # xhost +
# su - oracle
$ cd /opt/clusterware/
$ ls
$ cd clusterfy/
$ ls
$ ./runclusterfy.sh stage -post hoover -n lnx01,lnx02 -verbose ↴
```

\$ cd /opt/clusterware/

\$ ls

\$ ./runinstaller

Next → Next

home details → path: /u01/app/oracle/product/10.2.0/crs-home → Next → Next

click on Add → public node name:

private node name:

virtual host name:

→ Next

click on eth0 → edit → choose ① public →  → Next

② Normal Redundancy

Specify OCR loc:

mirror :  → Next

③ Normal Redundancy

Specify Voting Disk Loc:

Additional表决 loc:  → Next

"表决"  → Next → Install

[lnx01] # /home/oracle/orainventory/ocrinstroot.sh then [lnx01] # same

[lnx01] # /u01/app/oracle/product/10.2.0/crs-home/root.sh then

root.sh: 2nd script

→ The above script takes the backup of inittab file as inittab.orig. and ..  
places 3 entries in the inittab file with respect to 3 daemons of cluster

i.e., CRSd, & a crsma.

- The above script created OCR file & voting file in the shared storage.
- It is going to start all the I daemons of the cluster

Click on **OK**

[lnx01] # cd /u01/app/oracle/product/10.2.0/crs-home/bin/

# ./vipca ↵

Next → Next → [lnx01]

→ Next → Finish

[lnx01] # su - oracle

\$ cd /u01/app/oracle/product/10.2.0/crs-home/bin/

\$ vi racquip

FAIL-WHEN-DEFAULT G0-NOT-FOUND=0

IP address

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet,Hyderabad 192.16.128.179

clickdown ↴

→

\$ scp racquip lnx01:/u01/app/oracle/product/10.2.0/crs-home/bin/racvip.

:yes!

change this)

click on **Retry** **ignore** ↴ →

click on **OK** → click on **exit** ↴ →

click on **OK** → click on **Retry** → exit. → yes.

To check the successful state of clusterware installation:-

[lnx01] \$ cd /u01/app/oracle/product/10.2.0/crs-home/bin/

\$ ./crsctl check crs

\$ crs\_stat -t ↴

\$ ./crs\_stat -t

\$ ./olsnodes

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet,Hyderabad

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet,Hyderabad

## Installing Oracle binaries.

```
$ cd /opt $ ls  
$ cd database-otn/ $ ls  
$ ./runInstaller
```

Next → Oracle Enterprise Edition → Next

Home details path: /u01/app/oracle/product/10.2.0/db-home → Next

Select all → Next → Next

① Install database software only → Next → Install

[lnx01]# /u01/app/oracle/product/10.2.0/db-home/root.sh ↵ ↵

[lnx02]# /u01/app/oracle/product/10.2.0/db-home/root.sh ↵ ↵  
click on OK → exit → yes

## Configuring .bash\_profile :-

[lnx01]# su - oracle  
\$ vi .bash\_profile

```
export ORA_CRS_HOME=/u01/app/oracle/product/10.2.0/crs-home.
```

```
export ORACLE_HOME=/u01/app/oracle/product/10.2.0/db-home.
```

```
export PATH=$ORA_CRS_HOME/bin:$ORACLE_HOME/bin:$PATH:$HOME/bin
```

unset ~~the~~ USERNAME

:wq!

\$ . .bash\_profile

\$ scp .bash\_profile lnx01:/home/oracle/.bash\_profile.

## Configuring Listener:-

lnx01] \$ netca  
 ○ cluster configuration  
 ○ Listener configuration  
 ○ Add  
 Listenername **LISTENER**  
 ○ PORT 1521

→ next → next  
 → next  
 → next  
 → next → next  
 → Next → Next → Next → Next → finish.

## Configuring ASM:-

lnx01] \$ dbca  
 ○ Oracle RAC database  
 ○ Choose configure ASM  
**Select all**

→ next  
 → next  
 → next

sys pwd : racdba  
 Confirm : racdba  
 ○ Create initialization parameter file → next → **OK**

Click on **Create New**  
 Disk group name **ASM-DG-DATA**

### Redundancy

○ External

/dev/req/raw0

→ Click on **OK**

Click on **Create New**

Disk group name **ASM-DG-FRA**

○ External

/dev/req/raw1

→ Click on **OK** → Finish. → No.

## Creating RAC database.

[lnx01] \$ dbca

① Oracle RAC database

→ Next

② Create database

→ Next

Select all

→ Next → Next

DB Name : hrms

→ Next → Next

syspwd : racdb@

→ Next

③ ASM

→ Next

ASM-DG- DATA

→ Next

④ Use common location for all Database files

→ Next

Specify Flash Recovery Area

Browse

Enable Archiving

⑤ ASM-DG-FRA

Click on Edit Archive Node Parameters

Delete Archive log destination → Click on OK → Next → Next → Next

→ Next → Finish. → OK

[lnx01] \$ ps -ef | grep smon.

\$ sructl status database -d hrms.

\$ export ORACLE\_SID=hrms

\$ sqlplus "/as sysdba"

SQL> show parameter cluster

SQL> desc gv\$instance;

SQL> select inst\_id, instance\_name, host\_name, state from gv\$instance;

SQL> select name from v\$logfile;

SQL> create tablespace hrms\_ts

2 datafile '+ASM-DG- DATA/hrms/hrmso.001.dbf' size 100M;

Steps to bring down the RAC environment:-

\$ export ORACLE\_SID=hrms1

\$ crsctl stop dbconsole

\$ srvcctl stop service -d hrms

\$ srvcctl stop database -d hrms.

\$ srvcctl stop asm -n lnx01

\$ srvcctl stop asm -n lnx02

\$ srvcctl stop nodeapps -n lnx01

\$ srvcctl stop nodeapps -n lnx02

\$ srvcctl stop nodeapps -n lnx02

Note:- Nodeapps includes (i) GSD (Global service daemon)  
(ii) ONS (Oracle notification service)  
(iii) VIP (Virtual IP)  
(iv) Listener.

Stopping Cluster:-

\* In order to stop (or) start the cluster, we require root privilege. In the industry we use sudo.

[lnx01] # cd /u01/app/oracle/product/10.2.0/crs-home/bin/  
# ./crsctl stop crs.

[lnx02] # cd /etc/init.d

# ./init.crs stop.

Steps to bring up the RAC environment:-

If o/s is already up&running, Just start the cluster.

[lnx01] # cd /u01/app/oracle/product/10.2.0/crs-home/bin/  
# ./crsctl start crs

[lnx02] # cd /etc/init.d

# ./init.crs start.

[lnx01] # cu - oracle

\$ crsctl check crs

\$ ps -ef | grep smon

\$ srvcctl status database -d hrms.

## Converting standalone database as RAC database.

[ux01] # xhost +

# su - oracle

\$ dbca

① Oracle single instance database

② Create database

Global db name

prod

③ Use the same password

pwd : racdba

→ next

→ next → next

→ next → next

→ next

→ next

④ ASM

pwd : racdba → [OK]

ASM-DG-DATA → next

⑤ Use common location for all database files. → next

[Previous] →  set ASM-DG-FRA → [OK]

Enable Archiving

Edit ArchiveMode parameters → remove the path

click on [OK] → next → next → next → next → finish. → [OK]

Click on exit.

lxr01] \$ cd \$ORACLE\_HOME/assistants/

\$ ls  
\$ cd config/

\$ ls  
\$ cd samples/

\$ ls

\$ pwd  
\$ cp convertToRAC.xml /home/oracle/

\$ cd

\$ vi convertToRAC.XML

Specify home of non-RAC database

source DB home /u01/app/oracle/product/102.0/db-home

| specify oracle home where the rac database would be configured:-

/u01/app/oracle/product/10.2.0/db\_home

| specify SID of non RAC database .

SID = "prod"

ORCL > sys </nolog

pwd > racdbac

Role > sysdba <

| ASM info element

SID = "+ASM1"

pwd > racdbac

Role > sysdba <

| specify the list of the nodes

< node name = "lnx01" />

< node name = "lnx02" />

| specify prefix for rac instances

prefix > prod <

| specify port for the listener

<n: Listener port = "1521" />

| specify db area location

<n: Target db area > +ASM-DG-DATA <

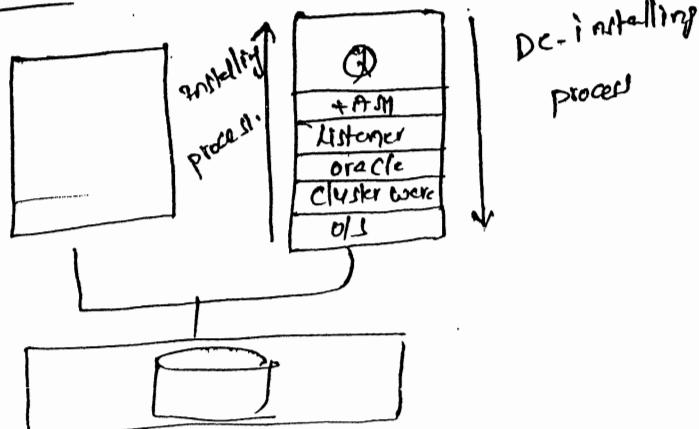
<n: Target FRA > +ASM+DG-FRA <

: w2!

\$ which rconfig

\$ rconfig convert to RAC.XML

De installing existing RAC environment:-



```

[lnx01] # Xhost +
# cu - oracle
$ dbca
① Oracle rec cluster database → next
② Delete a database → next
③ hints → finish → yes → no.

```

### Deleting a listener:-

```

$ netca
① cluster configuration → next → next
② Listener Configuration → next
③ Delete → next → next → yes → next → next
→ finish.

```

### stopping AVM instance:-

```

[lnx01] $ svchost stop avm -n lnx01
$ svchost stop avm -n lnx02

```

### Deinstalling oracle binaries:-

```

$ /opt/database-otn/runInstaller
click on Deinstall products (or) $ cd $ORACLE_HOME/oui/bin
$ ./runInstaller
oradbg_home1 $ ./runInstaller
Click on Remove → click on Yes → click on Close → cancel → yes.

```

### Deinstalling clusterware:-

#### Stopping nodeapps:-

```

$ svchost stop nodeapps -n lnx01
$ svchost stop nodeapps -n lnx02

```

### stopping cluster:-

```

[lnx01] # service init.ora stop.
[lnx01] # ssh lnx02 service init.ora stop
pwd : rec dba.

```

execute `rootdeletion.sh` & remove all the nodes. (2560)

Node1: # cd /u01/app/oracle/product/10.2.0/ctx\_home/install  
# ./rootdelete.sh.  
# ./rootdeinstall.sh.

Remove the files that are created at various run levels from all the nodes:-

[Lnx01] # rm -rf /etc/init.d/init.cssd  
# rm -rf /etc/init.d/init.css  
# rm -rf /etc/init.d/init.crsd  
# rm -rf /etc/init.d/init.evmd  
# rm -rf /etc/init.d/init.k96init.crc  
# rm -rf /etc/rc2.d/K96init.css  
# rm -rf /etc/rc3.d/K96init.css  
# rm -rf /etc/rc3.d/S96init.css  
# rm -rf /etc/rc5.d/K96init.css  
# rm -rf /etc/rc5.d/S96init.css  
# rm -rf /etc/oracle/scls.scr  
# rm -rf /etc/inittab.css  
# cp /etc/inittab.orig /etc/inittab.

[Lnx01] # rm -rf

If any CRS daemons are in up&running kill their process id from all the nodes.

[Lnx01] # ps -ef | grep crsd  
# ps -ef | grep CRS  
# ps -ef | grep crsd  
# ps -ef | grep evmd

[Lnx01] # kill or it is.

If there is no other oracle software is running remove the following files from all the nodes :-

[Lnx01] # rm -rf /var/tmp/.oracle/  
# rm -rf /tmp/.oracle/  
# rm -rf /etc/oracle/

Remove the files from ORACLE home until we run.

```
[Inx01] # cd /u01/app/oracle/product/10.2.0/crs-home/ | [Inx02] # same as it is  
# ls  
# rm -rf *
```

Removing CRS ~~service~~ home from our screen:-  
clusterware  
\$ /opt/clickteam/runinstaller.

- ☒ [Deinstall products]
- ☒ ora crs logo home → click on Remove → yes → yes → close → cancel
- ☒ ora crs logo home → click on Remove → yes

Deleting Content from RAC partitions:-

```
# fdisk -l  
# dd if=/dev/zero of=/dev/sdb5 bs=8192 count=2560  
# dd if=/dev/zero of=/dev/sdb11 bs=8192 count=2560
```

Removing mappings:-

```
[Inx01] # cd /etc/sysconfig/  
# vi rawdevices.
```

remove the mappings

```
# scp rawdevices [Inx02]:/etc/sysconfig/rawdevices.
```

[Inx02] # Same as it is

Removing permission:-

```
# cd /etc/rc.d  
# vi rc.local  
remove permissions :008!  
# scp rc.local [Inx01]:/etc/rc.d/rc.local
```

Note:- If you have implemented 10g RAC using OCF12 remove the following

file from all the nodes.

```
[Inx01] # rm -rf /etc/ocfs2 | [Inx02] # rm -rf /etc/ocfs2.
```

REMOVING THE GROWTH OF THE SYSTEM

Linux] # cd /u01/app/oracle/product/10.2.0/db-home/

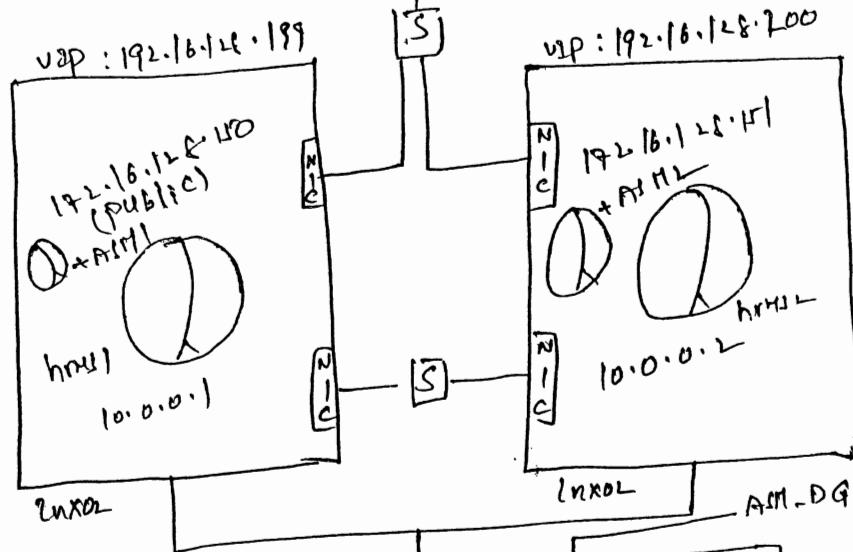
Linux] # same

# U

# rm -rf \*

U1

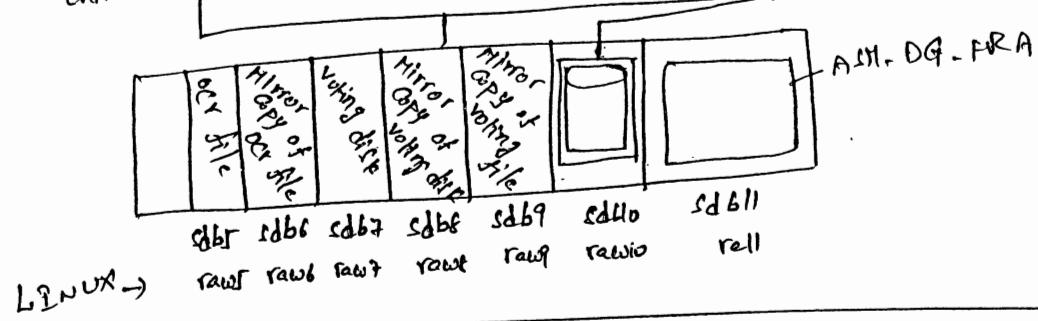
UN



/etc/fstab

chown

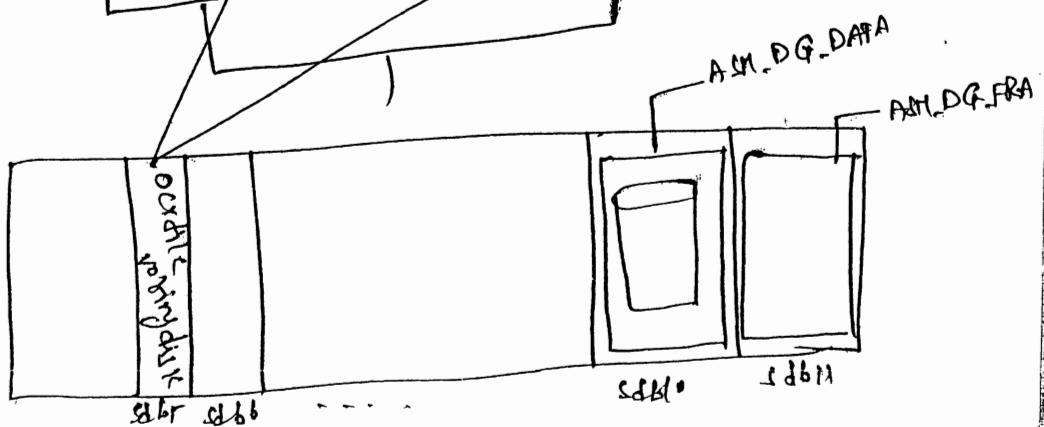
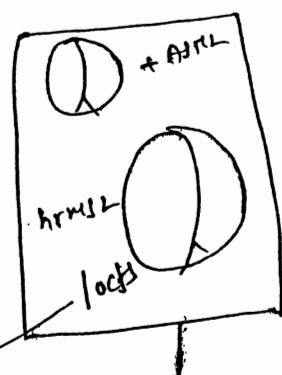
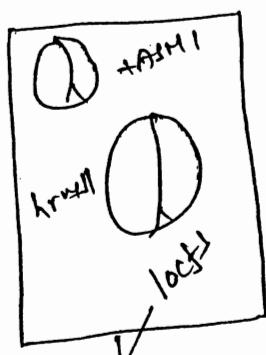
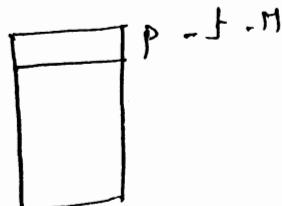
chmod



LINUX: only for

OCFS2 - LINUX

ASM Lib - LINUX



## Implementation of OCF2 RAC using OCF2 & ASM lib interface.

- (1) Create the required no.of groups & oracle user account
- (2) Create the required directory structures for crs-home & db-home
- (3) Configure kernel parameters as per the installation doc.
- (4) Set shell limits for oracle user account
- (5) Edit /etc/hosts and specify public IPs, private IPs & virtual IPs.
- (6) Configure ssh for user escalation & trust relationship.
- (7) Create the required no.of partitions in the shared storage for OCF2 file system & ASM diskgroup.
- (8) Download & install OCF2 and ASM rpm's based on the kernel version of the OS.

### Node1 & Node2:-

```
lnx01] # uname -r  
2.6.9-89.ELSMP.  
# cd /opt  
# ls  
# cd 26989/rpm/  
# ls  
# rpm -ivh ocf2* --nodeps --force  
# rpm -q grep ocf2  
# rpm -ivh oracleasm* --nodeps --force  
# rpm -q grep oracleasm
```

To copy the files [lnx01]  
lnx02 {  
# cd /opt  
# scp -r 26989/rpm/ lnx02:/opt

### Configuring OCF2:-

#### Node1:-

```
[lnx01] # ocf2 console  
Click on Cluster → choose Configure nodes → Close  
Click on Add → Name: lnx01  
EP : 172.16.128.150  
Port : 7777  
[OK]
```

Click on Add : Name : lnx01  
IP : 172.16.128.151 (OK)

Click on [Apply] → Click on [Close]

Click on [cluster] → choose propagate configuration

root lnx01 pwd : racdba

root lnx02 pwd : racdba

click on [Close]

Click on [File] → choose [Exit]

enabling OCFS2 during system bootup:-

Node1 & Node2:-

[lnx01] # cd /etc/init.d

# ./orcbs enable

Creating OCFS2 file system on the unused disk partition:-  
(CapC')

Node1:-  
[lnx01] # mkfs.ocfs2 -b 4K -c 32K -N 4 -L /ocfs /dev/sdb5

Creating a mount point for OCFS2 file system:-

Node1 & Node2:-

[lnx01] # pwd

# cd /

# pwd

# mkdir /ocfs

Mounting OCFS2 to file system:-

node1 & node2:-

[lnx01] # mount -t ocfs2 -L /ocfs -o datavolume /ocfs.

# df -h

To mount OCFS to file system automatically during system bootup we need

to place an entry in fstab:-

Node1 & Node2:-

[lnx01] # vi /etc/fstab

Linux = /etc  
/etc  
etc  
- netdev, data volume, mount, ..

Creating directories for the shared files :-

Node 1 :-

```
[nx01] # mkdir -p /ocfs/OCR
```

```
# mkdir -p /ocfs/voting.
```

Changing ownership & setting proper permissions on shared directories :-

node1 & node2 :=

```
[root@192 ~]# chown -R oracle:oinstall /ocfs1/OCR/  
[root@192 ~]# chown -R oracle:oinstall /ocfs1/VOTING/  
[root@192 ~]# chmod -R 775 /ocfs1/OCR/  
[root@192 ~]# chmod -R 775 /ocfs1/VOTING/
```

## Configuring AVM lib :-

node1 & node2:-

[root@nxo1]# cd /etc/init.d

# :oracleasm Configure

default user : oracle

default group : dba

: y

: y

34

Labeling discs for ASM usage :-

Node 1 :-

```
Node1:
[lnx01] # cd /etc/init.d
# ./oracleasm createdisk vol1 /dev/sdb10
# ./oracleasm createdisk vol2 /dev/sdb11
# ./oracleasm scandisks
# ./oracleasm listdisks
```

node:-

lnx02] # cd /etc/init.d

# ./orecclaim scandisks

#. Porcelain 1st disk

See that date & time is as ~~same~~ as possible.

Node1 & Node2:-

[lnx01] # date

Installing oracle clusterware :-

```
[lnx01] # su - oracle  
$ cd /opt/clusterware/  
$ ls  
$ ./runInstaller.
```

Next → Next →

Specify home details

Path: [lu01/app/oracle/product/10.2.0/crs-home]

→ next → next

Click on [Add] → Public Node Name : lnx02  
Private Node Name : lnx01-priv  
Virtual Node : lnx01-vip

→ [OK] → Next

Click on [Edit] → Click on [Edit] → ⚡ public → [OK] → next

① External Redundancy

Specify OCR loc: [ocfs1/OCR/ocr-file]

→ next

② External Redundancy

Specify Voting Disk loc: [ocfs1/VOTING/voting-file]

→ next → Install

```
[lnx01] # lhome /oracle/oraInventory/ocrInitRoot.sh
```

```
[lnx01] # /u01/app/oracle/product/10.2.0/crs-home/root.sh
```

```
[lnx01] # lhome /oracle/oraInventory/ocrInitRoot.sh
```

```
[lnx01] # /u01/app/oracle/product/10.2.0/crs-home/root.sh
```

Click on [OK]

```
[lnx01] # cd /u01/app/oracle/product/10.2.0/crs-home/bin/
```

# ./lwpca

Next → Next → lnx01

IP
192.16.128.199

→ Next → Finish

[lnx01] # su - oracle

```
$ cd /u01/app/oracle/product/10.2.0/crs-home/bin/
```

\$ vi racquip

[FAIL\_WHEN\_DEFAULTGW\_NOT\_FOUND = 0

:w!

\$ scp racquip Lnx02:/u01/app/oracle/product/10.2.0/crs-home/bin/racquip.

Click on [Retry] → Click on [OK] → Click on [Exit] →

Click on [OK] → Click on [Retry] → Click on [Exit] → Yes.

Installing oracle binaries:-

Lnx01 # cd /opt/database-0tn/

\$ ls

\$ ./runInstaller

Next → ① Enterprise edition → Next

Home details path: [u01/app/oracle/product/10.2.0/db-home] → next → [Select all] → next

→ next → ② install database software only → next → Install

Lnx01 # /u01/app/oracle/product/10.2.0/db-home/root.sh

Lnx01 # /u01/app/oracle

click on [OK] → Exit → Yes

Configuring .bash-profile:- → Done

Configuring Listener:-

Lnx01 # netca

o Cluster configuration

o Listener Configuration

o Add

Listener name **LISTENER**

Port **1521**

→ next → next

→ next

→ next

→ next → next

→ next → next → next → Finish.

## Configuring ASM.

[Inx01] \$ dbca

Oracle rac database

→ next

① Configure ASM

→ next

→ [select all] → next

sys pwd : [rac dba]

Confirm : [rac dba]

② Create initialization parameter file → next → [OK]

Click on [create new] → [change disk discovery]

[ORCL:VOL1] → [OK] (or) [/dev/oracle/disk#] → [OK]

[Inx01] # Service raw device restart & [Inx01] # service rawdevice restart.

Disk group name [ASM-DG-DATA]

③ External

ORCL:VOL1

→ Click on [OK] → This is not mounted for node 1

Click on [Create New]

Disk group name : [ASM-DG-FRA]

④ External Redundancy

ORCL:VOL2

→ Click on [OK]

Click on [Finish]

→ no

[Inx01] # su - oracle

If disk group mounted (1/2) then

\$ ps -ef | grep smon

\$ export ORACLE\_SID = +ASM2

\$ sqlplus "/as sysdba"

SQL> desc V\$ASM\_DISKGROUP;

SQL> select name, state, total\_mb, free\_mb from V\$ASM\_DISKGROUP;

SQL> alter diskgroup ASM-DG-Data mount;

SQL> select name, state, total\_mb, free\_mb from V\$ASM\_DISKGROUP;

SQL> exit

## Creating RAC Database

[ux01] \$ dbca

① Oracle RAC Database

② Create database

DB Name

hrms

sys pass: rac db\*

Confirm: rac db\*

③ A CM

A CM - DG - DATA

④ Use common location for all db files → next

specify FRA

enable Archiving

→ next

→ next →  [select all] → next → next

→ next → next

→ next

→ next

→ next

[cancel]

⑤  [cancel] → ⑥ A CM - DG - FRA → OK

edit Archive Mode page

↓  
Delete Archive desti →  [OK]

Next → Next → Next

Next → finish → click on  [OK]

[ux01] \$ sqlplus status database -d hrms

\$ sqlplus config database -d hrms

Types of init.ora parameters in RAC system:-

→ In RAC system, init.ora parameters are broadly categorized into

① Instance specific parameters

② Common parameters.

→ Parameters which are prefixed with instance name are called instance specific parameters.

→ Parameters which are prefixed with \* are called Common parameters.

```

[ux01] $ cd $ORACLE_HOME
$ ls
$ cat inithrms.ora
SPFILE='+ASM_DG_DATA/hrms/spfilehrms.ora'.
$ sqlplus export ORACLE_SID=hrms
$ sqlplus "/as sysdba"
SQL> Create pfile='/home/oracle/inithrms.ora' from spfile;
SQL> exit
$ pwd
$ vi inithrms.ora.

→ In oracle there are some '-' parameters which are also called as
hidden parameters. We should not use these parameters without the support
of oracle consultation team.

① - allow-open-resetlog-corruption = true
② - no-recovery=true-resetlog = true
③ - spin-count . . . etc.

```

Note: If u enable ASM and if we create pfile from spfile ,  
all autotunable parameters will be prefixed with "--" which means the  
values of these parameters are not constant and they may change depending  
on the workload.

### RAC specific parameters :-

- \* .cluster-database-instances = 2
- \* .cluster-database = true .

hrm1.instance-number = 1

hrm2.instance-number = 2

\* .remote-listener = 'LISTENER\_RHRMS' → This is meant for server side  
connect time load balancing.

hrm1.thread = 2

hrm1.thread = 1

hrm1.undo-tablespace = 'UNDOTBS1'

hrm2.undo-tablespace = 'UNDOTBS2'

[lnx01] \$ SQLplus / as sysdba

SQL> show parameter open\_cursors

SQL> alter system set open\_cursors=400 scope=both sid='hrms1';

SQL> show parameter open\_cursors

SQL> alter system set open\_cursors=500 scope=both sid='\*';

Troubleshooting oracle clusterware & collecting diagnostic information :-

→ The default location of alert log of cluster is

\$ cd \$ORACLE\_HOME/log/lnx01

\$ ls

\$ cat alertlnx01.log

Collecting diagnostic information from CRS\_HOME using diagcollection.pl :-

Note:- we need to run the script as root user in all the nodes. In the industry we use sudo.

\$ sudo crctch stop CRS

/etc/scdbers

oracle ALL:ALL expect root

[lnx01] # export ORACLE\_HOME=/u01/app/oracle/product/10.2.0/crs-home/

# cd \$ORACLE\_HOME/bin

# ./diagcollection.pl --collect --crs \$ORACLE\_HOME.

# ls \*.gz

# mv \*.gz \$HOME

# cd

# ls

# gunzip ocrData-lnx01.tar.gz

# ls

# tar -xvf ocrData-lnx01.tar

[lnx01] \$ ocrdump

\$ cat ocrdumpfile.

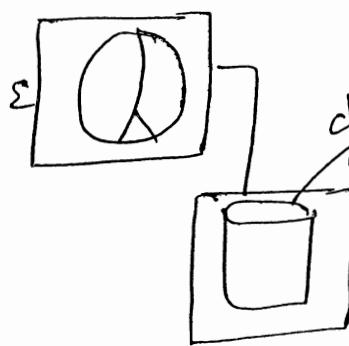
## Collecting Diagnostic Information from Oracle Home

```

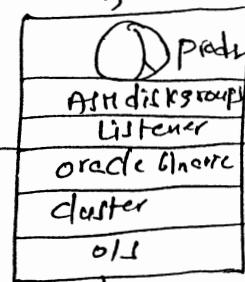
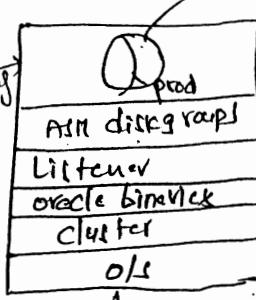
[oracle]# export ORACLE_HOME=/u01/app/oracle/product/10.2.0/db_1/home/
# cd $ORA_CRS_HOME/bin
# ./diagcollection.pl --collect --oh $ORACLE_HOME.
# ls *.gz
# mv *.gz $HOME
# cd.

```

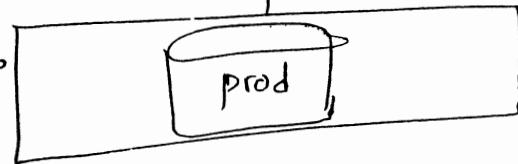
rConfig1 - to upgrade standalone to RAC :-



clone the db to  
one of RAC server using  
RMAN



- (a) if db size is small
  - Create RAC db
  - Create RAC db
- (b) if db size is huge
  - Clone the db to



oracle bin }> struct  
clust }> crctl  
ols

Administering cluster using crctl ! -

→ To check the status of the cluster  
log\$ng/ruc \$crctl check crs

11gR2/ruc \$crctl check cluster -all

→ To stop the cluster

```

# cd /etc/init.d
# ./init.crs stop

```

```

# cd $ORA_CRS_HOME/bin
# ./crctl stop crs

```

11gR2 #crctl stop cluster -all

- To start the cluster
 

```
# cd /etc/init.d
# ./init.crs start
```

(or)

```
# cd $ORACLE_HOME/bin
# ./crsctl start crs
```

OR

```
# crsctl start cluster -all
```
- To know the version of the cluster
 

```
$ crsctl query crs softwareversion
```

(or)

```
$ crsctl query crs activeversion
```
- To know the location of voting disk
 

```
$ crsctl query crs votedisk
```
- To know the location of OCR file & to check the integrity of OCR
 

```
$ocrcheck.
```
- To know the location of OLRA
 

```
$ocrcheck -local
```
- To disable the cluster
 

```
$ crsctl disable crs
```
- To enable the cluster
 

```
$ crsctl enable crs
```
- To check the status of all resources
 

```
$crs -stat -t
```
- To know the no.of nodes that are participating in the cluster.
 

```
$olcnodes
```
- To dump the Content of OCR into a text file.
 

```
$ocrdump
```
- To know the master node & the default OCR backup location.
 

```
$ocrconfig -showbackup
```

## Administering RAC USING ORACLE

- To know the configuration of the database  
\$ sructl config database -d hrms
- To check the status of all instances of a database  
\$ sructl status database -d hrms
  - options: start → to start
  - : stop → to stop

- To stop the database

\$ sructl stop database -d hrms -o immediate

- To check the status of a specific instance  
\$ sructl status instance -i hrms1 -d hrms
  - options: start
  - : stop

- To check the status of ACH

\$ sructl status akm -n lnx01

options: start  
: stop

### IGRL

\$ sructl status akm

- To check the status of nodeapps  
\$ sructl status nodeapps -n lnx01

options: start  
: stop

### IGRL

\$ sructl status nodeapps

- To disable the database

\$ sructl disable database -d hrms

- To enable the database

\$ sructl enable database -d hrms

- To check the status of all services that are registered with the cluster.

\$ sructl status service -d hrms

options: start  
: stop

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Amberpet, Hyderabad

- To create a specific service  
`$cruct add service -s test -d hrms -r hrms`  
`-a hrms -P basic`
- S - means service name
- d - db name
- r - preferred instance
- a - available instance
- P - TAF policy (Transparent application failover) parameter.

- To check the status of a specific service.

`$cruct status service -s test -d hrms`

options: start  
stop

- To register a database with the cluster

`$cruct add database -d hrms -o $ORACLE_HOME`

- To unregister a database with the cluster

`$cruct remove database -d hrms -o $ORACLE_HOME`

- To register an instance with the cluster

`$cruct add instance -i hrms -d hrms -n [nx0]`

- To unregister an instance with the cluster

`$cruct remove instance -i hrms -d hrms -n [nx0]`

- To register ASH with the cluster

`$cruct add ash -i +ASH -n [nx0]`

- To check the status of scan listener

118L  
`$cruct status scan-listener`

- To know the configuration of scan listener

118L  
`$cruct config scan-listener`

- To know the configuration of scan

118L  
`$cruct config scan`

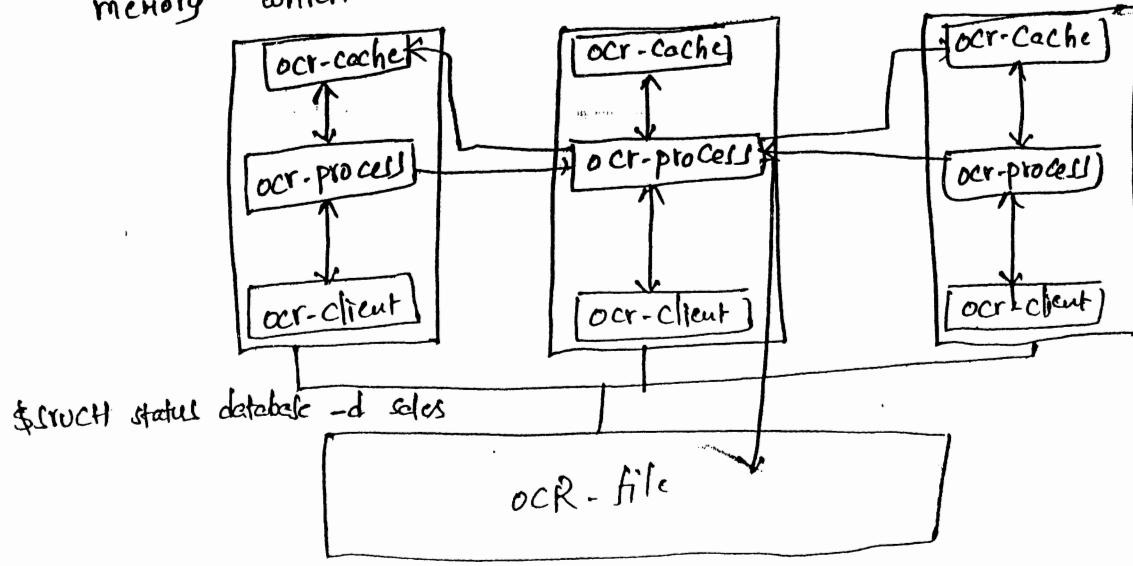
- To know the status of the cluster \$cructh status scan
- To know the configuration of all services of a database \$cructh config service -d hrms.

### Clusterware Architecture:-

- The two important files of oracle clusterware are <sup>(1) ocr file (2) voting file</sup> (Oracle Cluster Registry)
- The above two files needs to be placed in the shared storage either in Raw partitions (or) CFS
- From 11GR2, we need to place either in CFS (or) ASM disk groups  
(directly placing in raw has been de-supported)

### OCR File:-

- OCR file contains entire cluster configuration information like node names, IP's information, Databases, listeners, services etc. that are registered with the cluster.
- Various clients of OCR are <sup>(1) DBCA (2) netca (3) Enterprise manager console (4) VIPCA (5) SRSCTL.</sup>
- In order to optimize the performance of the queries on OCR, Oracle RAC follows distributed "OCR-cache" architecture. which means every node that is participating in cluster maintains a local copy of OCR file in the local memory which is called OCR-cache.



- Out of all the nodes, one of the node will be designated as master. Master node takes the responsibility of updating local ocr-cache & remote ocr-caches.
- At any point of time, only master node ocr-process will be in directly contact with the ocr-file.
- If the master node gets evicted (reboot), node with the lowest node membership becomes the master node.
- CSD provides (Cluster synchronization service daemon) node membership for every node that is participating in cluster.
- By default master node takes the backup of ocr-file for every 4 hours, at the end of the day & at the end of the week.
- To know the master node and the default backup location
 

```
$ ocrconfig -showbackup
```
- SQL select \* from gv\$gcr\_resource;

- Voting file/DISK:-
- This is something like quorum disk in 9i RAC (or) o/c clustering technology.
  - Every node that is participating in cluster has to vote in the voting disk for the specified default time interval. If any of the node fails to vote, the other node considers it as dead node.
  - In order to support Redundancy and to avoid split brain syndrome, oracle RAC supports redundant no. of voting disks and majority voting principle.
  - As per majority voting principle, a node should be in a position to vote in more than 50% of voting disks. otherwise, node gets evicted from the cluster. Because of this reason, oracle RAC maintains odd no. of voting disks.
  - Every clustering technology has split-brain syndrome problem. each vendor follows his own algorithm to avoid this.

→ Some third party vendors they follow STOPPING Algorithm.

(choose the other machine in the head)

→ CSSD monitors the liveness of the node through two communication channels

① Network heart beat

② Disk heart beat.

→ In 11gR2, every node that is participating in cluster maintains a local configuration file in the local hard disk. which is called "OLR".

→ OLR Contains cluster configuration information specific to node.

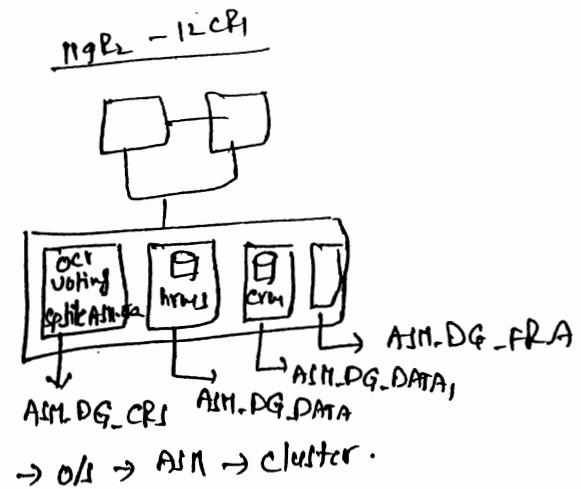
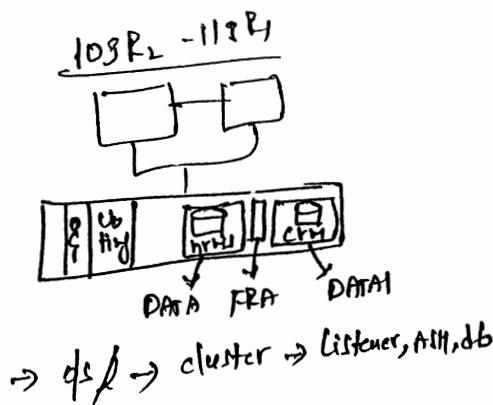
→ \$ocr check -local.

→ In 11g R2, with the help of OLR and Grid plug&play profile, node could able to start ASM instance

→ once the ASM instance is started, node can access the OCR file & voting file which is required to start the cluster.

→ The default location of Grid plug&play profile is \$GRID\_HOME/

\$GRID\_HOME/gpnp/profile/peers/profile.xml.



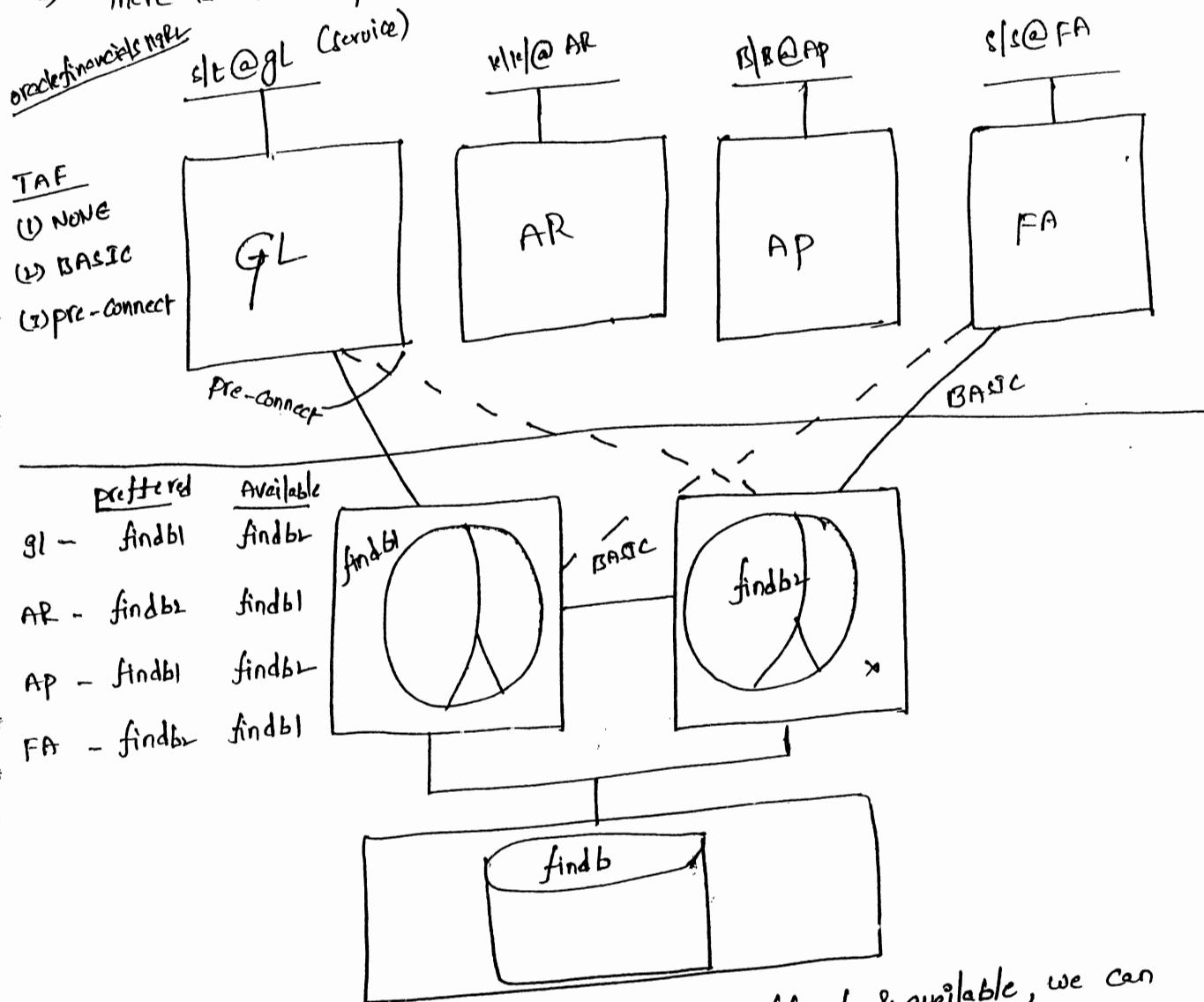
SERVICE MANAGEMENT

### Service :-

- logical grouping of sessions which are performing a single kind of task is called a service.
  - There are two types of services ① Internal Services  
② Application specific services.
  - Internal services are sys\$background & sys\$users.
  - once we create a database, by default internal services will be created automatically.
  - we can create a service with the following options
    - ① dbca
    - ② SRVCTL
    - ③ enterprise manager console
    - ④ dbms\_service package.
  - dbms\_service package is generally used to create a service in standalone system.
  - whenever we create a service, service information has to be updated at the following layers.
    - ① Database layer (system.dbf)
    - ② Clusterware layer (OCR-file)
    - ③ Network layer (TNSnames.ora)
  - when we create a service using dbca, service information will be updated at all the 3 layers and the service will be started automatically.
  - In 11gR2, we don't have the option of creating service using dbca.
  - when we create a service using "SRVCTL" (or) "enterprise Manager console", service entries will not be updated in the network layer and the service has to be started manually.
  - At the time of creating a service, we can choose one of the instance has preferred & the other as available.
  - preferred instance means user will establish a connection to that particular instance from where the statements will be executed. upon the failure of the preferred instance, user session & select will failover to the available instance.

- RAC supports two types of failovers
  - ① session failover
  - ② select failover

→ There is no concept of DML failover in RAC system.



- At the time of creating a service, not only preferred & available, we can also choose the TAF (Transparent Application failover) parameters like

① NONE    ② BASIC    ③ PRE-CONNECT.

→ If TAF policy parameter is NONE then there will be no failover

→ If TAF policy parameter is BASIC, user establishes a connection to

the preferred instance, upon the failure of the preferred instance user select will failover to the available instance from where the rest of the records will be fetched & displayed

- If the TAF policy parameter is pre Connect, we establish a connection to the preferred instance as well as a backup connection to the available instance, but the statements will be executed with the preferred instance only. Upon the failure of the preferred instance, since the backup connection is already available with a slight delay the records will be fetched and displayed.
- Delay will be more in case of basic & less in case of pre-connect.

### Service Entries in log & l1g R1 :-

GL =

(DESCRIPTION =

(ADDRESS\_LIST = (PROTOCOL = TCP) (HOST = Linux01 - VIP) (PORT = 1521))

(ADDRESS\_LIST = (PROTOCOL = TCP) (HOST = Linux02 - VIP) (PORT = 1521))

(load\_balance = yes)

(CONNECT\_DATA =

(Server = dedicated)

(Service\_name = GL)

(failover\_mode =

(Type = select)

(Method = basic)

(Retries = 180) 10/15/10 (default is '180' but we specify 10/15/10)

(Delay = 5)

Q1 RAC  
~~VIP~~

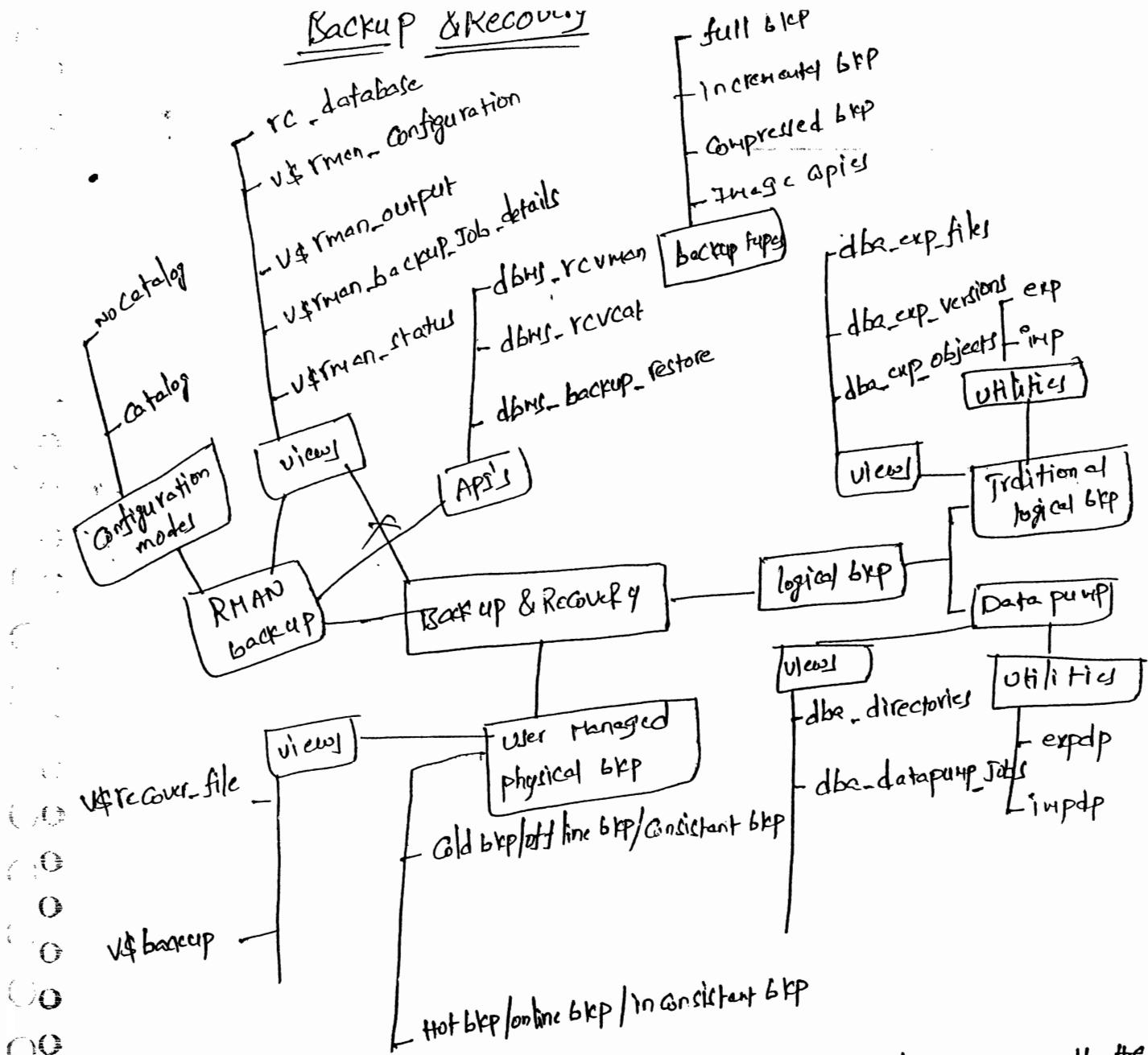
log RAC  
VIP ✓

Note:-

In log, oracle has introduced virtual IP to overcome the default TCP IP timeout delay in notifying the applications.

\* During node additions & deletions, we need to modify the service entries manually. To overcome this problem in l1g R1, oracle has introduced SCAN NAME and "SCAN IPs" (single client access name).

## Backup & Recovery



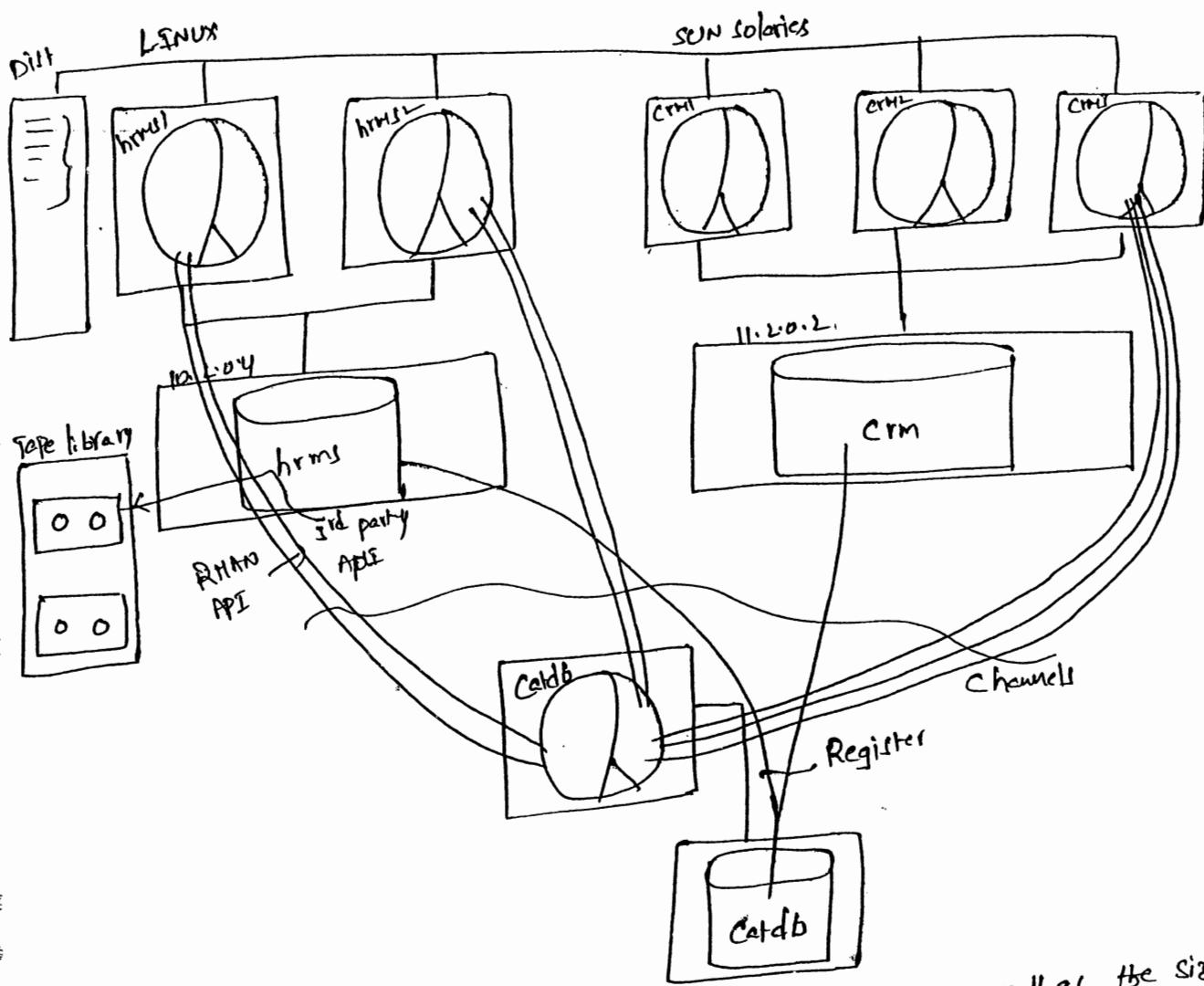
- If the database storage area location is a cluster file system, all the above mention backup's are possible.
- If the database storage area location is ASM, only logical & RMAN backup's are possible.

## RMAN & RAC Considerations:

- In RAC system, recommended to configure multiple channels. See that channels are equally distributed among all the instances. (For load balancing)
- In some customer environments, an instance will be dedicated only for RMAN backup's.
- In RAC system, from 11.2.0.2 snapshot control file has to be placed in the shared storage.
- In case of tape backup's, we need to integrate RMAN with 3rd party HML (Media Management layer) software like Tivoli data protector for oracle, HP data protector for oracle, veritas (or) legato etc.
- In case of tape backup's, 3rd party administrator will install 3rd party software on all the database servers. Once, 3rd party software is installed & configured, we need to create RMAN backup, restore & recovery scripts by specifying 3rd party specific environment variables.

## Pre-requisites to configure RMAN in Catalog mode:-

- Identify a local & cold server and install the same (or) higher version of oracle binaries.
- Create catalog database
- Create a separate Tablespace to hold RMAN metadata tables.
- Create a user and grant Connect, Resource, Recovery\_Catalog\_Owner roles.
- Create catalog
- Configure `tnsnames.ora`
- Register target database into Recovery catalog
- Configure RMAN parameters
- Create backup, restore & recovery scripts.



Note: No. of channels generally depends on no. of processors as well as the size of the database.

### RMAN Incremental backup:-

- In case of incremental backup, RMAN takes the backup of only modified blocks
- In order to minimize the incremental backup time, in 10g, Oracle has introduced block change tracking.

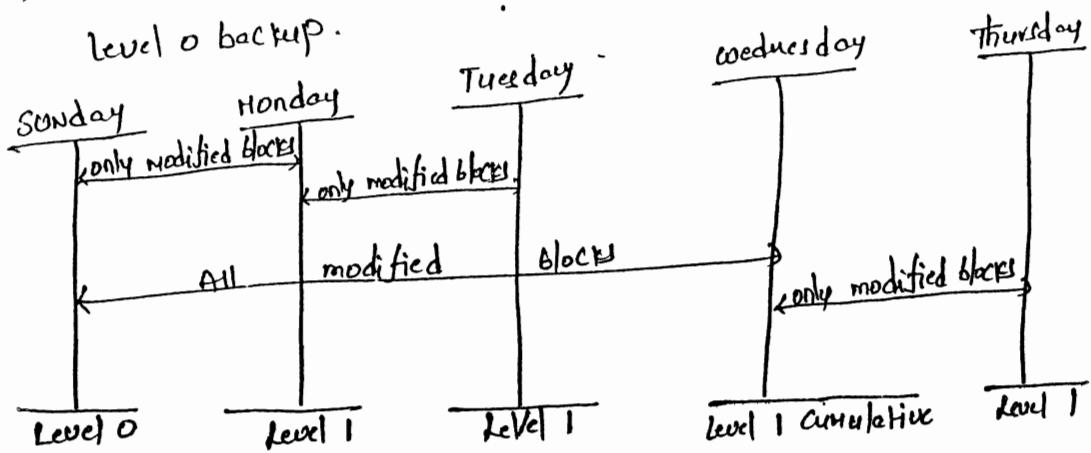
In RAC system, block change tracking file has to be placed in the shared storage.

```
SQL> alter database enable block change tracking
      using file '/u01/app/oracle/oradata/hrms/changetrack_file';

```

(or)  
`'/u01/app/oracle/oradata/hrms/changetrack_file';`

- once we enable block change tracking, CWT (change track writer) is enabled and it keeps track of the modified blocks information and updated to a separate file at off level.
- The different levels of increments are
  - ① Level 0 (complete)
  - ② Level 1 (incremental)
  - ③ Level 2 (cumulative)
- From 10.2.0.4, levels greater 0 & 1 have become obsolete.
- From 10.2.0.4, RMAN backup incremental level 0 database;
  - RMAN> backup incremental level 0 database;
  - RMAN> backup incremental level 1 database;
  - RMAN> backup incremental level 2 database; (From 10.2.0.4 it has no more)
  - (or)
  - RMAN> backup incremental level 1 cumulative database;
- Level 0 is the base backup for incremental backup.
- without initiating Level 0, If we initiate level 1, by default RMAN takes level 0 backup.



Note- If the instance is started with spfile, by default RMAN takes the backup of spfile.

- If we configure FRA, by default RMAN backup sets will be created in FRA

## \* RMAN \*

Steps to Configure RMAN in catalog mode:-

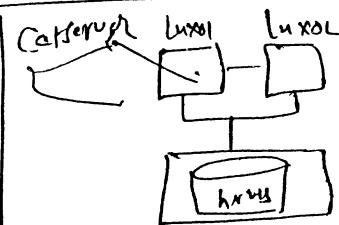
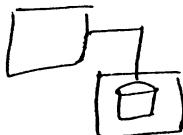
### Catalog Server

- (1) Create a database
- (2) Create a tablespace
- (3) Create an user & grant appropriate privileges to the user
- (4) Connect to catalog database and execute the following i.e., "Create Catalog" to create RMAN Metadata tables.
- (5) Configure tnsnames.ora
- (6) Connect to catalog database through Catalog database & Register target db into catdb.
- (7) Configure RMAN Configuration parameters and take the backup of target db.

### target server

- (5) Configure listener.ora
- (6) Create password file

### Catalog Server



```
[lux01] # df -lh
```

```
# mkdir -p /u01/app/oracle/
# chown -R oracle:oinstall /u01/app/oracle/
# chmod -R 775 /u01/app/oracle/
```

```
# xhost +
# su - oracle
```

```
$ dbca
```

① Oracle single instance database → next

② Create database → next → next

name: [Cat db] → next → next

③ Use same password for all accounts

pwd: [rac dba] → next

④ File system → next

⑤ Use common location for all SPFILE files

[/u01/app/oracle] → next

Enable archiving

→ next → next → next → next → finish → exit

Click on **exit**

[ux01] \$ export ORACLE\_SID=catdb

\$ sqlplus / as sysdba

sys> select name, open\_mode, log\_mode from v\$database;

sys> select instance\_name, status from v\$instance;

sys> select name from v\$tablespace;

sys> select name from v\$datafile;

sys> create tablespace rman\_ts

datafile '/u01/app/oracle/catdb/rman\_ts01.dbf' size 10M autoextend on;

sys> create user rmanuser identified by rmanuser default tablespace rman\_ts;

sys> grant connect, resource, recovery\_catalog\_owner to rmanuser;

sys> conn rmanuser/rmanuser

sys> show user

sys> select count(\*) from tab;

sys> host catalog rmanuser/rmanuser

RMAN> create catalog;

RMAN> exit

RMANuser> select count(\*) from tab;

-----  
PO (In Log) 102 (In 11g)

[ux01] \$ cd \$ORACLE\_HOME/network/admin.

\$ vi listener.ora

[ux01] \$ export ORACLE\_SID=catdb

\$ rman catalog rmanuser/rmanuser target sys/racdb@hrms

RMAN> Register database;

RMANuser> select \* from rc\_database;

RMAN> Unregister database;

RMANuser> select \* from rc\_database;

RMAN> Register database;

RMAN> show all;

RMAN> spool log to file.log

RMAN> show all;

RMAN> spool off;

RMAN> host 'prod';

1home/oracle

RMAN> exit

lux01] \$ vi rman.log

CONFIGURE RETENTION POLICY TO REDUNDANCY 2;

CONFIGURE BACKUP OPTIMIZATION ON;

CONFIGURE CONTROLFILE AUTOBACKUP ON;

CONFIGURE DEVICE TYPE DISK PARALLELISM 2 BACKUPTYPE TO BACKUPSET;

CONFIGURE CHANNEL 1 DEVICE TYPE DISK CONNECT 'sys/racdba@hrms1';

CONFIGURE CHANNEL 2 DEVICE TYPE DISK CONNECT 'sys/racdba@hrms2';

:wc!

lux01] \$ rman catalog rmanuser/rmanuser target sys/racdba@hrms1.

RMAN> @rman.log

RMAN> showall;

RMAN> backup database plus archivelog;

S-T desc of full database:-

RMAN> report schema;

lux01] \$ export ORACLE\_SID = +ASM1

\$ asmcmd

ASMCHD> ls

ASMCHD> cdasm-dg-data

ASMCHD> ls

ASMCHD> cdhrms1

ASMCHD> pwd

ASMCHD> ls

ASMCHD> rm \*.ctl ? yes

If the storage is ASM we can't remove CRD files while database is up & running. ASM provides security.

lux01] \$ sructl stop database -d hrms

ASMCHD> rm \*.ctl \*.dbf \*.log ? yes

ASMCHD> ls

```
lux01] $ export ORACLE_SID=hrms  
$ sqlplus / as sysdba
```

sys> startup nomount.

sys> exit

```
$ export ORACLE_SID=catdb
```

```
$ rman catalog rmanuser/rmanuser target sys/racdba@hrms
```

It will throw a error message : this is a bug in log

```
lux01] $ cd $ORACLE_HOME/network/admin
```

```
$ vi tnsnames.ora
```

```
(INSTANCE_NAME=hrms) (OR=A)
```

:wq!

```
lux01] $ export ORACLE_SID=catdb
```

```
$ rman catalog rmanuser/rmanuser target sys/racdba@hrms
```

RMAN> restore controlfile;

RMAN> alter database mount;

RMAN> restore database;

RMAN> recover database;

```
RMAN> S2) 'alter database open resetlogs';
```

RMAN> exit

```
lux01] $ crsctl stat -t
```

```
$ crsctl start instance -i hrms -d hrms
```

```
$ crsctl stat -t
```

### Create Store script:-

RMAN> list script names;

RMAN> Create script bkp

```
{ backup database 423 }
```

RMAN> Create global script bkp1

```
{ backup database; }
```

RMAN> list script names;

RMAN> print script bkp;

RMAN> print script bkp1;

```
RMAN> run { create script bkp; }
```

## Node EVICTION issues.

→ In RAC system a node gets evicted mainly under the following circumstances.

(i) If node is unable to access the shared storage (node is unable to vote in the voting disk)

(ii) If the cluster interconnect (or) private interconnect goes down. (node is unable to send a network heartbeat).

(iii) Sometimes node also gets evicted because of bug in the cluster. In this case, we need to download and install the latest CRS#bundle patch using opatch utility.

(iv) Sometimes node also gets evicted because of resource starvation. In this case, even if we look at the logs & traces, we don't find much information.

→ In this case, even if we raise a service request with oracle support team, oracle will ask us to download & configure "oswatcher".

→ oswatcher is a shell script (or) perl script given by oracle in the metalink. → oswatcher contains of monitoring commands like top, sar, iostat, vmstat, netstat etc..

→ If we configure oswatcher, we need to have enough space in the file system and we also need to have proper purging policy for the traces generated by oswatcher.

	Linux	Others
30sec.	60sec	

\$ crsctl get ccmicount

\$ crsctl set ccmicount <value>

\$ crsctl get crsdisktimeout 250 sec.

\$ crsctl set crsdisktimeout <value>

cluster startup process in log 8 NGR :-

Root.sh → /etc/inittab → /etc/inittab.orig

35 /etc/init.d/init\_crsd

35 /etc/init.d/init\_Cssd

35 /etc/init.d/init\_eumd

init

init\_cssd

Cssd

init\_crd

monitor

oprocd

init\_eumd

eumd

crsd → [Actions]

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Amarpet, Hyderabad

### Cssd (or) ocrd - Cluster synchronization service daemon.

- This daemon also comes into the picture in standalone system. If the database storage area location is "ASM".
- In 10g, before starting ASM instance, we need to start this daemon by executing the following script as root user.

```
# cd $ORACLE_HOME/bin  
# ./localconfig add ↵
```
- In RAC system, this daemon gets enabled automatically once we install oracle clusterware.
- In RAC system, this daemon provides node membership for every node that is participating in the cluster.
- This daemon monitors the liveliness of the node through network heart beat & disk heart beat.
- This daemon notifies other members in the cluster whenever a node joins (or) leaves the cluster.
- Unable to start this daemon will not allow the node to become part of the cluster.

\* If I kill Cssd, node reboots automatically

### Crd (cluster ready service daemon)-

- This daemon ~~acts as~~ acts as an engine for all high availability operations.
- This daemon maintains & manages all the resources that are registered with the cluster.
- This daemon runs under root account.
- This daemon gets restarted automatically, if it fails.

### Cssd monitor-

- Monitors the scheduling of CPU. If it detects the hung state of a node, it is going to reset the node to maintain data integrity. This process is called I/O fencing.
- Prior to 10.2.0.4, this task was carried out by oproc (Oracle process monitor daemon).

### evmd (ever running daemon).

- Scans callout directory and publishes cluster events and these events will be captured by ONS (Oracle notification service).
- \* In a scenario, where both nodes are alive, they could able to vote in the voting disk but they are unable to communicate with each other through Network heartbeat. In this scenario, the node with the lowest membership survive and the other node gets evicted from the cluster.
- \* In this scenario, a kill block will be written to the node voting disk slot in every voting file by the master node. After reading the kill block, the "ocssd-bin" process of the other node will commit suicide by resetting it.

### Creating and Administering High Availability Services:-

[nxo1] \$ dbca

Oracle RAC database  
Services Management

Click on **Add** → Enter service name **GL** → **OK**

prefferred	available
<input type="radio"/>	<input type="radio"/>
<input type="radio"/>	<input type="radio"/>

hrm1

hrm2

TAF Policy

basic

→ next

→ next → next

<input type="radio"/>	<input checked="" type="radio"/>
prefferred	available

**Add**

→ Enter service name

**AR**

→ **OK**

hrm1

hrm2

hrm1

TAF Policy

pre-connect

**Add**

→ Enter service name

**AP**

→ **OK**

hrm1

hrm2

TAF Policy

basic

→ Finish

→ **OK** → **NO**

1) [root] \$ export ORACLE\_SID = hrms

\$ sqlplus "as sysdba"

sys> desc gtservices

sys> select inst\_id, NAME, network\_name, creation\_date from gtservices;

sys> exit

i) \$ sructl status service -d hrms

ii) \$ cd \$ORACLE\_HOME/network/admin  
\$ vi tnsnames.ora

\$ sqlplus "as sysdba"

sys> alter user scott account unlock identified by tiger;

sys> desc gtsession;

sys> select INST\_ID, SID, Username, Status, Service\_Name from gtsession

sys> select INST\_ID, SID, Username, Status, Service\_Name from gtsession

where Username = 'SCOTT';

sys> save userinfo.sql

[root] # su - oracle

\$ export ORACLE\_SID = hrms

\$ sqlplus scott/tiger@gl

sys> @ userinfo.sql

scott> exit

\$ sqlplus scott/tiger@AP

sys> @ userinfo.sql

scott> exit

\$ sqlplus "as sysdba" stop service -s gl -d hrms

\$ sructl status service -s gl -d hrms

\$ sqlplus scott/tiger@gl

\$ lsnrctl status

\$ tnsping gl

\$ sructl start service -s gl -d hrms

\$ sqlplus scott/tiger@gl

SCOTT> EXIT  
 \$ sqlplus scott/tiger@ar  
 SCOTT> INSERT INTO salgrade SELECT \* FROM salgrade;  
 SCOTT> /, /, /, -- etc.  
 SCOTT> COMMIT;  
 SCOTT> SELECT COUNT(\*) FROM salgrade;  
 SCOTT> SELECT \* FROM salgrade;  
 SCOTT> SELECT COUNT(\*) FROM salgrade;  
 SCOTT> CANCEL ABORT.  
 SCOTT> EXIT  
 \$ SRVCTL STATUS SERVICE -D HRMS  
 \$ SRVCTL START INSTANCE -I HRMS1 -D HRMS.  
 \$ SRVCTL STATUS DATABASE -D HRMS  
 \$ SRVCTL STATUS SERVICE -D HRMS.  
 \$ SRVCTL STATUS SERVICE -D HRMS. (TO KNOW SERVICE CONFIGURATION)  
 \$ SRVCTL CONFIG SERVICE -D HRMS.  
 \$ SRVCTL STOP SERVICE -S GL -D HRMS } ONE WAY  
 \$ SRVCTL START SERVICE -S GL -D HRMS } ONE WAY  
 \$ SRVCTL STATUS SERVICE -D HRMS  
 \$ SRVCTL RELOCATE SERVICE -H } SECOND WAY TO BRING BACK THE  
 \$ SRVCTL RELOCATE SERVICE -D HRMS -S AR -I HRMS2 -T HRMS / SERVICES.  
 \$ SRVCTL STATUS SERVICE -D HRMS.

Q add a service using SRVCTL:-  
 \$ SRVCTL ADD SERVICE -S TEST -D HRMS -R HRMS1 -A HRMS2 -P BASIC  
 \$ SRVCTL STATUS SERVICE -S TEST -D HRMS  
 \$ SRVCTL START SERVICE -S TEST -D HRMS  
 \$ SRVCTL STATUS SERVICE -S TEST -D HRMS

\$ sqlplus scott/tiger@test.  
 Error: TNS: Could not resolve (while creating a service using SRVCTL it couldn't place an entry in tnsnames.ora. We need to place manually an entry).

\$ cd \$ORACLE\_HOME/network/admin  
 \$ vi tnsnames.ora

Copy the existing service config entry and paste

TEST =  
DESCRIPTION =

:02!

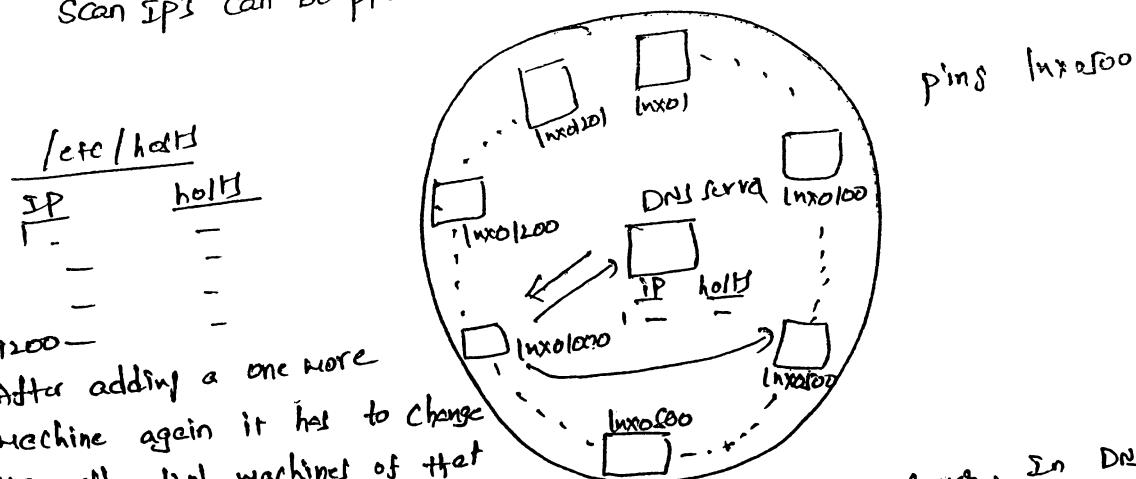
\$ sshplus scott/tiger@ tiger.

@scott\$ exit

\$ srch config service -d hmc

understanding scan name & scan IP's :-

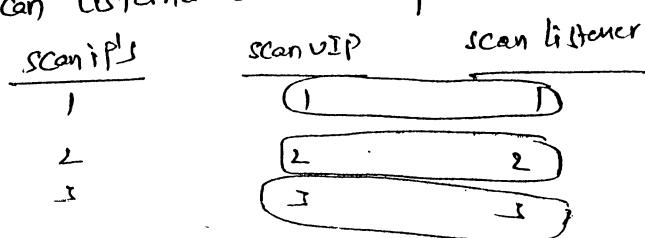
- During node additions & deletions, we need to modify the service entries manually. To overcome this problem, In RHEL, oracle has introduced scan name & scan IP's
- Irrespective of the no.of nodes, oracle recommends to have 3 scan IP's with the single scan name.
- Scan IP's can be placed either in /etc/hosts (or) DNS Server (Domain Name service).



After adding a one more machine again it has to change the all 1201 machines of that entry. To overcome this, IT has come up with DNS service. In DNS service we need to place only one entry.

- During grid infrastructure service installation, for every scan IP oracle creates one scan VIP and one scan listener

- Scan VIP & scan listener forms a pair



- If we place scan IPs in /etc/hosts, only one scan listener gets enabled.
- If we place scan IPs in DNS server, all 3 scan listeners get enabled.

11g RL

2 Node RAC

No. of nodes x no. of IPs per each node

$$2 \times 3 = 6$$

10 Node RAC

$$10 \times 3 = 30$$

11g RL

2 Node RAC

No. of nodes x no. of IPs per each node + 3 scan IPs

$$2 \times 3 + 3 = 9$$

10 Node RAC

$$10 \times 3 + 3 = 33$$

entries in /etc/hosts with respect to 11g RL RAC :-

/etc/hosts entries

IP

192.168.233.40

192.168.233.41

192.168.233.50

192.168.233.51

192.168.233.60

192.168.233.61

192.168.233.101

192.168.233.102

192.168.233.103

Local host

lxn01

lxn02

Local host

lxn01.oracle.com

lxn02.oracle.com

lxn01 - priv

lxn02 - priv

lxn01 - VIP

lxn02 - VIP

cluster - scan

cluster - scan

cluster - scan

with respect to 11g RL RAC :-

SOA

dnsoracle.com {

=

= }

Entries in DNS server

@ IN

NS

dns.oracle.com

Local host

A

127.0.0.1

dns

A

192.168.233.224

lxn01

A

192.168.233.40

lxn02

A

192.168.233.41

luxol-vip	IN	A	192.168.222.00
luxol-vip	IN	A	192.168.222.61
cluster-scan	IN	A	192.168.222.101
			192.168.222.102
			192.168.222.103

Note:- private IP is meant for node to node communication and it is also meant for cache fusion. Because of this reason, we have not placed private IPs in DNS server.

Service Entries in HgR2 :-

GL =

(DESCRIPTION =

(Address list = (protocol = TCP) Chat=cluster-scan.oracle.com) (Port = 1521))

(load\_balance = yes)

(Connect-data =

(server = dedicated)

(service\_name = GL)

(failover\_mode =

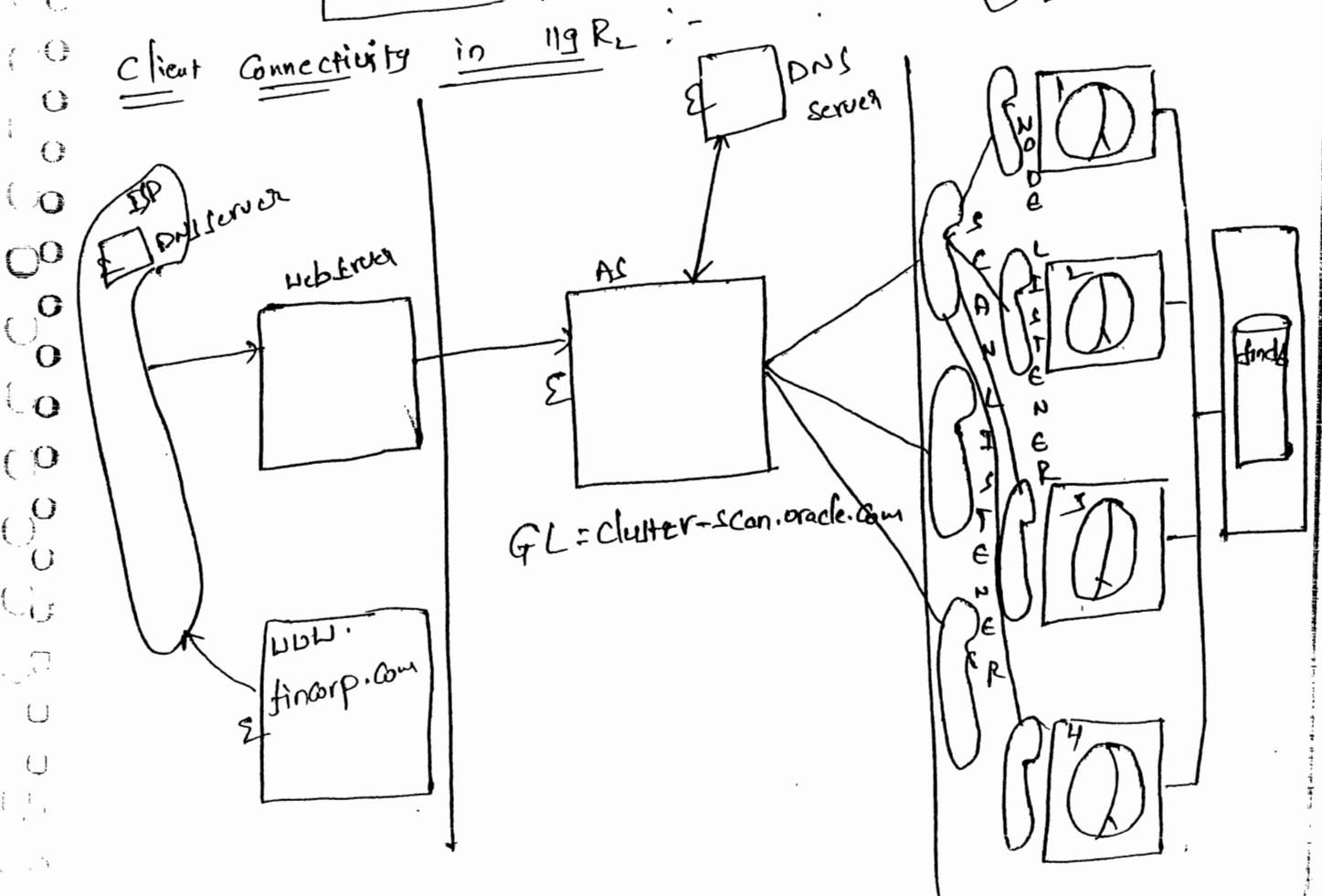
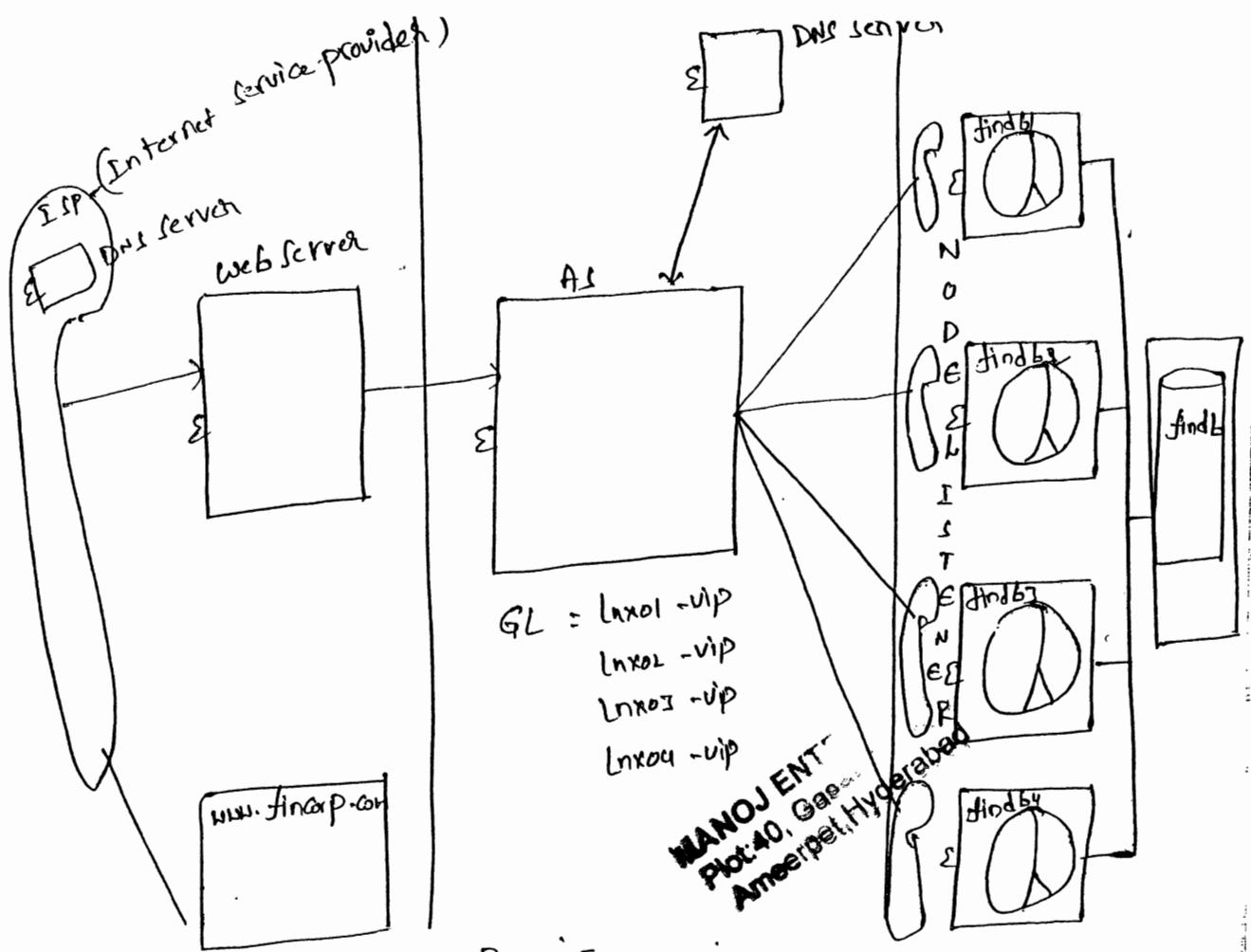
(ctype = select)

(method = BASIC)

(Retries = 180) 10/15/20

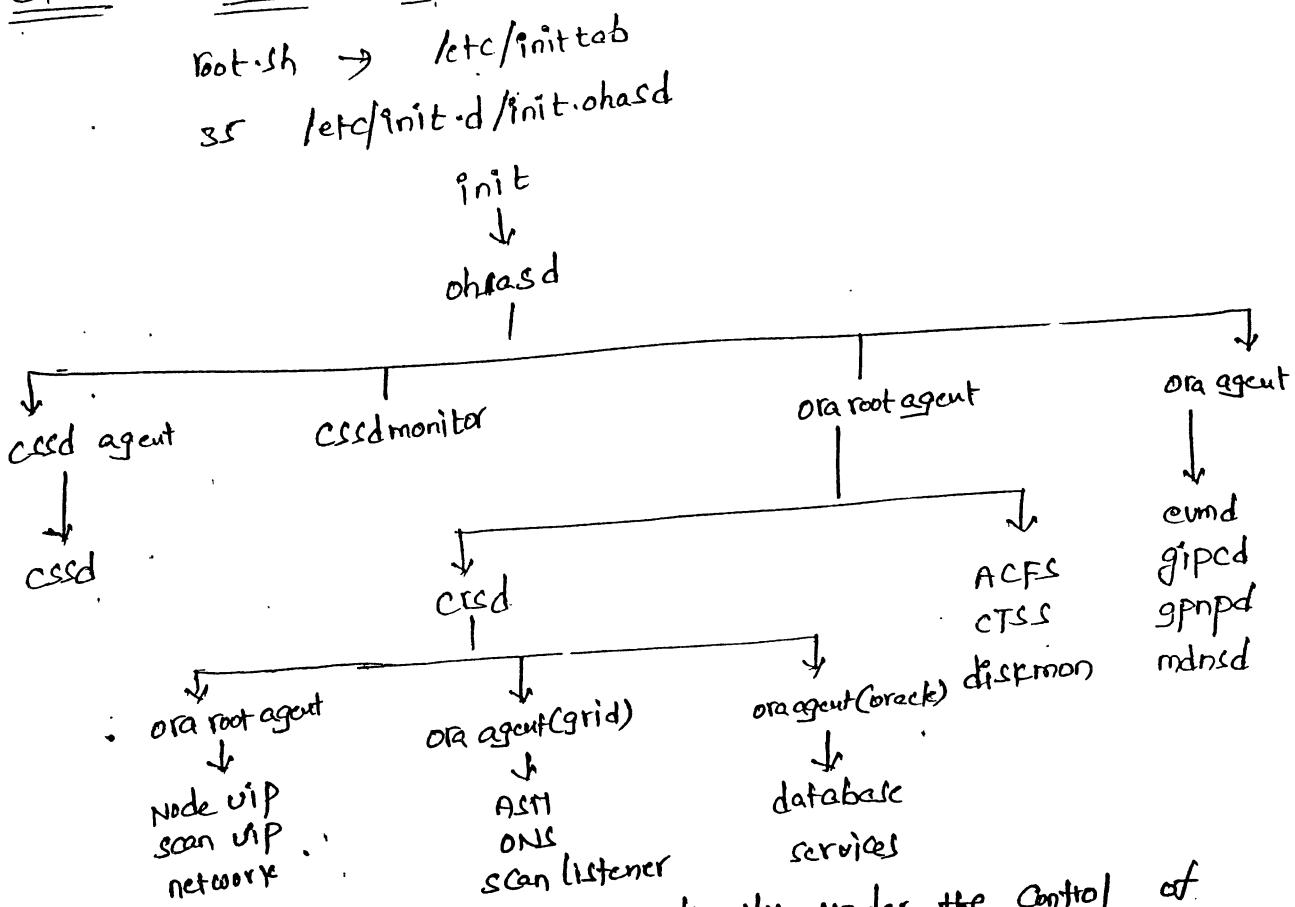
(delay = 5)

Client Connectivity in log & HgR1 :-



- Every instance will communicate with the scan listener.
- Scan listener knows the load on each instance listener.
- Scan listener redirects the connection to the least loaded instance listener.  
This is called "server side connect time load balancing".

### Cluster startup process in 11gR2 :-



- In 11gR2, the only daemon which is directly under the control of os-init process is **ohasd** (Oracle high availability service daemon).
- In 11gR2, Oracle has introduced the concept of agents and these agents are going to start their respective daemons.
- With the introduction of **ohasd**, we can perform cluster wide operations like starting the cluster, stopping the cluster and checking the status of the cluster.
- Oracle has re-architected grid infrastructure services in to two stacks
  - ① Oracle high availability service stack (High level stack)

## (i) cluster ready service stack (low level stack)

- one set of oraagent and orarootagent will be started by "ohasd" and another set of orarootagent & oraagent will be started by crsd.

### ACFS (A SM cluster file system):-

- This is a generic ASM based cluster file system
- with the introduction of ACFS, we can mount a chunk (portion) of a disk group as a normal file system.

### CTSS (Cluster time synchronization service):-

- Introduced in 11GR2 to synchronize the time in all the nodes.

- If NTP (Network time protocol) is already in place, CTSS runs in observer mode. otherwise it runs in active mode.

### Disk mon (Disk monitor):-

- Introduced for exadata systems, performs I/O fencing in exadata.

### GSpcd (Grid interprocess communication daemon):-

- This is meant for grid interprocess communication, supports various protocols like TCP, UDP etc..

### GPNPD (Grid plug& play profile daemon):-

- This is going to maintain grid plug& play profile.
- grid plug& play profile is an XML file which contains ASM disk strings and ASM Spfile location.

### Hdnsd (Multi Casted domain name service daemon):-

- This is also meant for name resolution.

patching & upgrading :- 10.2.0.2, 10.2.0.3, 10.2.0.4 - patch set

Applying patch set on Cluster home using Non-Rolling upgrade method :-

[nxo1] # cd /opt/

# ls

# cp README.html /root/Desktop/

? y

# cd

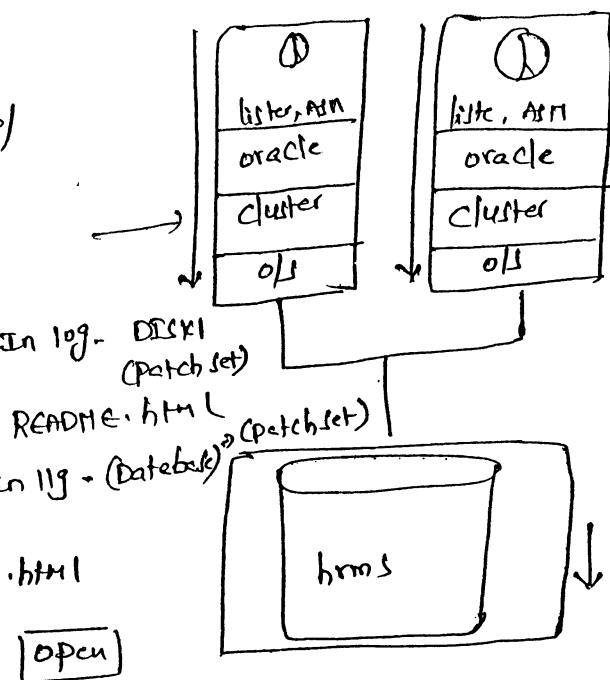
# firefox

Click on [File] → [open file]

↳ [Desktop]

↳ README.html

↳ [open]



[nxo1] # su - oracle

\$ crs\_stat -t (to check status of all resources)

\$ export ORACLE\_SID=hrms

① \$ emctl stop dbconsole (this is going to stop Enterprise Manager console)

\$ ssh nxo2

[nxo2] \$ export ORACLE\_SID=hrms

\$ emctl stop dbconsole

\$ emctl status dbconsole - It will show under which node it is running.

\$ exit

② [nxo1] \$ isqlplusCH stop

③ \$ svchd stop service -d hrms

④ \$ svchd stop database -d hrms

⑤ \$ svchd stop asm -n nxo1

\$ svchd stop asm -n nxo2

(6) \$ svckill stop nodeapps ...  
\$ svckill stop nodeapps -n lnx02  
\$ crs\_stat -t

(7) \$ crsctl query crs softwareversion (or)  
\$ crsctl query crs activeversion.  
\$ exit

[lnx01] # service init.crs stop

(To stop the cluster in both the nodes)

# ssh lnx02 Service init.crs stop

[lnx01] # xhost +

# su - oracle

\$ /opt/Disk1/runInstaller

Next

Select [oracle1 log-home]

→ next → next → next → install

[lnx01] # /u01/app/oracle/product/10.2.0/crs-home/bin/crsctl stop crs

[lnx01] # /u01/app/oracle/product/10.2.0/crs-home/install/root102.sh

[lnx01] # /u01/app/oracle/product/10.2.0/crs-home/install/root102.sh  
[Execute same scripts in all other nodes. # ssh lnx02 /u01/ --  
pwd: racdba.]

click on [exit] → yes

Applying patchset on oracle home:-

[lnx01] \$ ssh lnx02

[lnx02] \$ export ORACLE\_SID = hrms2

\$ emctl stop dbconsole

\$ sqlplusctl stop

\$ exit

[lnx01] \$ svckill stop service -d hrms

\$ svckill stop database -d hrms

\$ svckill stop asm -n lnx01

\$ svckill stop asm -n lnx02

\$ svckill stop listener -n lnx01

\$ svckill stop listener -n lnx02

```

$ ps -ef | grep crsctl
[lnx01] $ /opt/disk1/runInstaller
Next → next → next → next → next → Install

[lnx01] # /u01/app/oracle/product/10.2.0/db-home/root.sh
[lnx01] # ssh lnx02 /u01/app/oracle/product/10.2.0/db-home/root.sh
pwd: rac dba

click on exit. → yes

Upgrading database from 10.2.0.1 to 10.2.0.4!

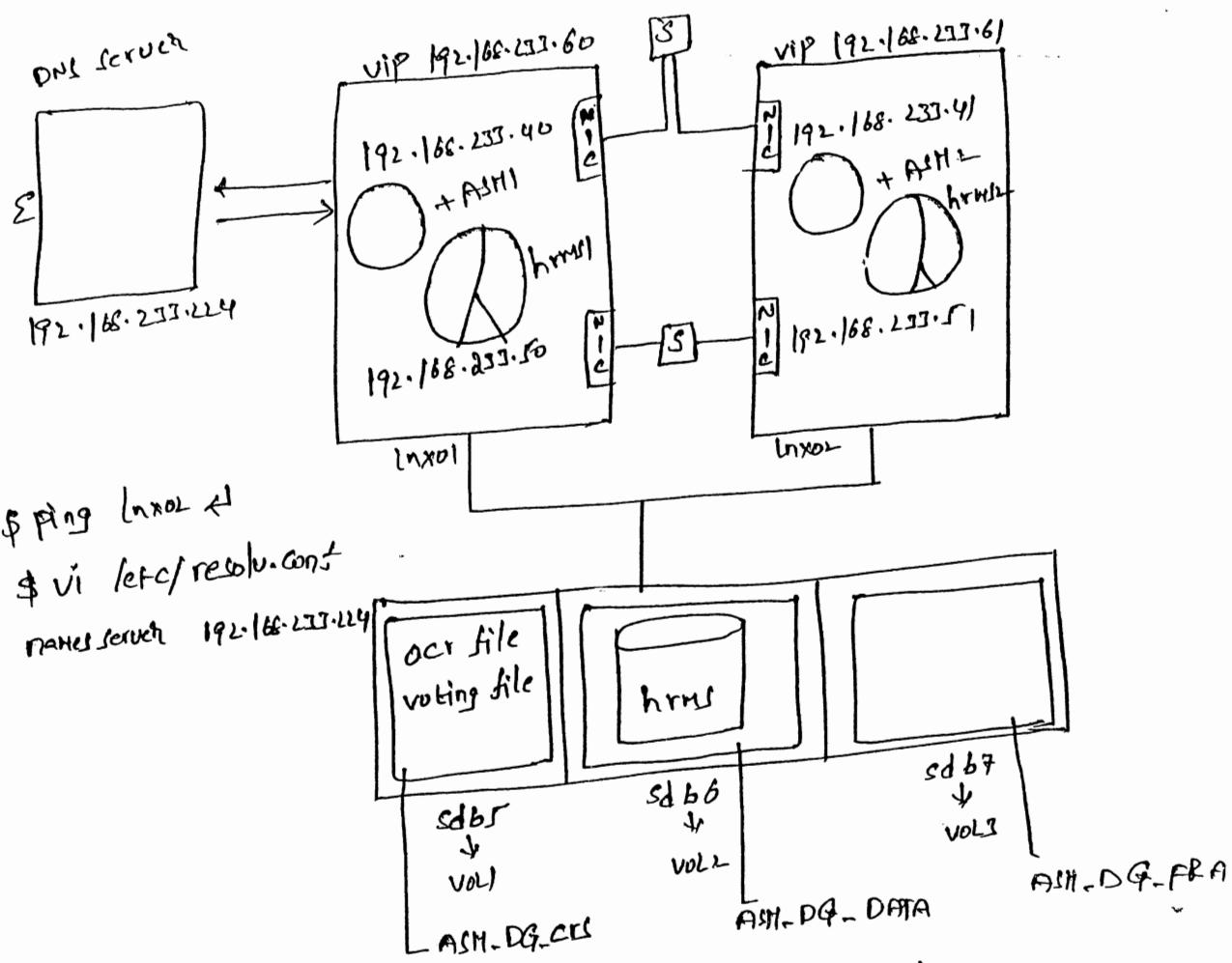
[lnx01] $ svrctl start listener -n lnx01
$ svrctl start listener -n lnx02
$ svrctl start asm -n lnx01
$ svrctl start asm -n lnx02
$ export ORACLE_SID=hrms1
$ sqlplus /as sysdba

sys> startup nomount
sys> alter system set cluster_database=false scope=spfile;
sys> shutdown immediate
sys> startup upgrade
sys> @?/rdbms/admin/catupgrd.sql ←
sys> @?/rdbms/admin/utlrp.sql ←
sys> select object_name, status from dba_objects where status='INVALID';
sys> select object_name, status from dba_registry;
sys> alter system set cluster_database=true scope=spfile;
sys> shutdown immediate
sys> exit

[lnx01] $ svrctl start database -d hrms
$ svrctl start service -d hrms
$ emca -upgrade db-cluster (For RAC)
$ emca -upgrade (In standalone to upgrade the db console)

```

## Implementation of 119K2 RAC on OEL AS 5 . -



Configurations that needs to be done at DNS server:-

→ In order to configure DNS server, we need to install the following rpm's  
 ① Bind                  ② N cache/Caching(OEL)  
 (CRHES)

→ After installing the above rpm's, we need to configure the following files.  
 ① named.conf            ② named.rfc1912.zones.

After Configuring the above files, we need to create one file for

forward look up zone & another file for reverse look up zone.

① for                  ② rev

```
$ # cat for
$ TTL 86400
@ IN SOA localhost root 42
```

→ After Configuring the above files, we need to start dns server

```
[root@dns ~]# service named start
```

```
# service named stop
```

```
# service named start
```

```
# dig -x 192.168.233.224
```

Configurations that needs to be done at RAC servers:-

Node1 & Node2:-

```
[lnx01]# vi /etc/resolv.conf
```

```
nameserver 192.168.233.224
```

:wq!

```
[lnx01]# ping lnx02
```

```
[lnx02]# vi /etc/resolv.conf
```

```
nameserver 192.168.233.224
```

:wq!

```
[lnx01]# nslookup lnx01
```

```
# nslookup lnx02
```

```
# nslookup cluster-scan
```

## node1

- ① Creating the required no of groups & oracle user account

```
lwp01# userdel -r oracle
```

```
# groupdel oracle
# groupdel dba
# groupadd -g 501 oracle
# groupadd -g 502 dba
# groupadd -g 503 oper
# groupadd -g 504 acadmin
# groupadd -g 505 asmdba
# groupadd -g 507 asmoper
# useradd -u 501 -g oracle -G asadmin,asmdba,asmoper grid
# useradd -u 502 -g oracle -G dba,asmdba,oracle
# useradd -u 503 -g oracle -G grid
# passwd grid
```

```
grid:~# passwd oracle
```

```
grid:~# passwd oracle
```

```
# id grid
```

```
# id oracle
```

- ② Creating directory for grid base  
 note:- grid base should not be under grid home

```
lwp01# mkdir -p /u01/app/grid
```

## node2

- ① Same as set no. ①

```
lwp02# userdel -r oracle
```

```
# groupdel oracle
# groupdel dba
# groupadd -g 501 oracle
# groupadd -g 502 dba
# groupadd -g 503 oper
# groupadd -g 504 acadmin
# groupadd -g 505 asmdba
# groupadd -g 507 asmoper
# useradd -u 501 -g oracle -G asadmin,asmdba,asmoper grid
# useradd -u 502 -g oracle -G dba,asmdba,oracle
# useradd -u 503 -g oracle -G grid
# passwd grid
```

```
grid:~# passwd oracle
```

```
grid:~# passwd oracle
```

```
# id grid
```

```
# id oracle
```

- ② Creating directory for grid base  
 note:- grid base should not be under grid home

```
lwp02# mkdir -p /u01/app/grid
```

- # chosen -R grid:install /u01/app/grid/  
 # chmod -R 775 /u01/app/grid/  
 (4) same as (i)
- (5) creating directory for grid home  
 [u00] # mkdir -P /u01/product/11.2.0/grid/home/  
 # chown -R grid:grid /u01/product/11.2.0/grid/home/  
 # chmod -R 775 /u01/product/11.2.0/grid/home/  
 (6) same as (5)
- (7) creating directory for oracle base  
 [u00] # mkdir -P /u01/app/oracle/  
 # chown -R oracle:oracle /u01/app/oracle/  
 # chmod -R 775 /u01/app/oracle/  
 (8) same as (7)
- (9) creating directory for oracle home  
 [u00] # mkdir -P /u01/app/oracle/product/11.2.0/db-home/  
 # chown -R oracle:oracle /u01/app/oracle/product/11.2.0/db-home/  
 # chmod -R 775 /u01/app/oracle/product/11.2.0/db-home/  
 (10) same as (9)
- (11) creating directory for oracle inventory  
 [u00] # mkdir -P /u01/app/oraInventory/  
 # chown -R grid:grid /u01/app/oraInventory/  
 # chmod -R 775 /u01/app/oraInventory/  
 (12) same as (11)

- ⑫ Configuring Kernel Parameters  
Note: In OEL is not required but in other flavours of O/S, we need to configure as per the installation doc.

⑬ set shell limits for oracle user account

```
luxoid # vi /etc/security/limits.conf
```

grid	soft	nproc	2047
grid	hard	nproc	16384
grid	soft	nofile	1024
grid	hard	nofile	65536

```
:wq!
```

```
# cd /etc/security/
# cp limits.conf limits: /etc/security/ limits.conf
? yes
```

- paad: rac dba
- ⑭ To apply the limits for all the shells
- ```
luxoid # vi /etc/profile
```
- ```
if [ $USER = "oracle" ] || [ $USER = "grid" ]; then
    if [ $SHELL = "/bin/ksh" ]; then
        ulimit -p 16384
        ulimit -n 65536
    else
        ulimit -u 16384 -n 65536
    fi
fi
```
- :wq!

```
# cd /etc  
# scp profile.lnx01:/etc/profile  
pwd: rac@ba
```

## (6) Configuring /etc/hosts

Note: If DNS is already configured, not necessary to configure  
hosts file

```
lnx01# vi /etc/hosts  
127.0.0.1 localhost
```

```
192.168.222.224 dns  
# # # public ips # #  
# 192.168.222.40 lnx01 lnx01.oracle.com  
# 192.168.222.41 lnx02 lnx02.oracle.com  
# # # private ips # #  
# 192.168.222.50 lnx01-priv lnx01-priv  
# 192.168.222.51 lnx02-priv  
# # # virtual ips # #  
# 192.168.222.60 lnx01-vip  
# 192.168.222.61 lnx02-vip  
# # # scan ips # #  
# 192.168.223.101 cluster-scan  
# 192.168.223.102 cluster-scan  
# 192.168.223.103 cluster-scan
```

```
[root] # cd /etc  
# cp hosts [new]: /etc/hosts
```

(P) creating the required no.of partitions in the shared storage.

```
[root] # fdisk -l  
# fdisk /dev/sdb
```

: n  
P : 1 : <  
: > : +10G  
: +2G : P  
: n : 2 : <  
e : 1 : <  
: > : P : n  
: n : 2 : <  
: > : +10G

MANOJ ENTERPRISES  
Plot:40, Gasathri Nagar  
Ameerpet, Hyderabad

```
luxo] # part probe
```

Installing AVM related & shared storage related APIs

⑯ [user] # cd /opt

```
#!/usr/bin/rpm -ivh oracleasm* --nodeps --force
```

```
lmax] # cd grid/rph/
```

# U  
# rpm -in cverdisc\* - -nodeps - -force

P#

OPT] # SQL oracle# lnx02: /opt

```
# scp cvgdisk* mn02:/opt
```

⑩ Configuring ACM lib [n.n.n] # Oracle qm configure -i

Default user : grid  
default group : asadmin

四

[unresolved hit]

~~[wzol] # part probe~~

⑨ [w<sub>102</sub>] # cd /opt

三

#

--nodeps --jtree

卷之三

Luxottica Group S.p.A. # CD Pelle (Pellizzetti)

# rph -ish cuedis\*

#

# cd ..

[@2] # oracle on Contour -

РІБІ

:asmarin

۲۰

11

1

```
[myol] # oracleasm init
```

[unro]# Oracle atm hit

⑫ labeling disks for AStI usage

```
luxo] # oracleasm createdisk vol1 /dev/sd[bc  
# oracleasm scandisks  
# oracleasm listdisks
```

```
# oracleasm listdisks  
# oracleasm createdisk vol1 /dev/sd[bc  
# oracleasm deletedisk vol1 /dev/sd[bc  
# dd if=/dev/zero of=/dev/sd[bc bs=8192 Count=2560  
# dd if=/dev/zero of=/dev/sd[bc bs=8192 Count=2560
```

```
luxo] # oracleasm createdisk vol1 /dev/sd[bc  
# oracleasm createdisk vol2 /dev/sd[bc  
# oracleasm createdisk vol3 /dev/sd[bc  
# oracleasm scandisks  
# oracleasm listdisks
```

⑬ # oracleasm scandisks

```
# oracleasm listdisks
```

⑭

## Installing grid infrastructure services

[lnx01] # x host +

# cu - grid

\$ vnc server

pwd: racdba

\$ vnc viewer

vncserver [lnx01.oracle.com:1]

[OK] ↗

:pwd : [racdba] ↘

\$ cd /opt/grid/

\$ ls

\$ ./runInstaller

① skip software update

→ next

② install & configure grid for cluster

→ next

③ Advanced installation

→ next → next

cluster name

[lnx-cluster]

scan name

[cluster-scan]

scan port

[1521]

select  configure DNS

Note:- grid naming service allows virtual IP address to have dynamically assigned by DHCP (dynamic host control protocol) server

click on next → click on [Add]

Host name [lnx01]

→ [OK]

VIP [lnx01-vip]

click on [SSH Connectivity]

or later [grid]

or pwd [grid]

[Setup] → [OK]

click on [test] → [OK] → next

Click on [left] → Choose private  
Click on [virbro] → Choose Do not use → next

① oracle AS11

Disk group Name

AS1-DG-CRS

→ next

② External

ORCL:VOL1 → next

③ use same pwd for these accounts

pwd: recdb2

→ next → yes

④ Do not use IPMLI  
Base: /u01/app/grid

→ next → next

grid path: /u01/product/11.2.0/grid-home

→ next

Inventory /u01/app/oraInventory

→ next

ignore all → next → Install

[u01]# /u01/app/oraInventory/oinstRoot.sh

[u02]# /u01/app/oraInventory/oinstRoot.sh

[u01]# /u01/product/11.2.0/grid-home/root.sh

[u02]# /u01/product/11.2.0/grid-home/root.sh → OK

[u01]# cat /etc/init.d

click on [OK] → yes → close.

[u01]# ps -ef | grep vnc

\$ kill -9 4510.

Installing oracle binaries :-

[u01]# su - oracle

& vncserver

pwd: recdb2

\$ vncviewer

[u01:oracle:com:2]

pwd: recdb2 → OK

```
$ Cd /opt/oracle
```

```
$ ls
```

```
$.runInstaller
```

I wish to receive security updates → next → yes

skip software update → next

install database software only → next

Oracle RAC database installation

Click on [SSH Connectivity]

as user [oracle] or as [oracle]

↳ [Setup] → [ok] → [Test] → [ok] → next

→ next →  enterprise edition → next

oracle basic [lu01/app/oracle] → next → next

oracle home [lu01/app/oracle/product/11.2.0/db-home] → next → next

ignore all → next → Install  
[root] # /lu01/app/oracle/product/11.2.0/db-home/root.sh

.. → [ok] → close

[root] # "

configuring .bash\_profile for grid user:-

[root] # su - grid

\$ vi .bash\_profile

export ORACLE\_HOME=/lu01/product/11.2.0/grid-home

export PATH=\$ORACLE\_HOME/bin:\$PATH:\$HOME/bin

:wq!

\$ . .bash\_profile

\$ scp .bash\_profile [root]:/home/grid/.bash\_profile

## Configuring .bash-profile for oracle user.

[nxo1]# su - oracle

\$ vi .bash-profile

export ORACLE\_HOME = /u01/app/oracle/product/11.2.0/db-home

export PATH = \$ORACLE\_HOME/bin: \$HOME/bin

:wq!

\$ . .bash-profile

\$ scp .bash-profile [nxo2]:/home/oracle/.bash-profile

## Creating the required no.of ASM diskgroups:-

[nxo1]# su - grid

\$ asmca

Click on **Create**

Disk group name

ASM-DG- DATA

① external

ORCL: VOL2

→ **OK** → **OK**

Click on **Create**

Disk group name

ASM-DG-FRA

① external

ORCL: VOL1

→ **OK** → **OK** → exit → yes

[nxo1]# ps -ef | grep unc

## Creating RAC database:-

[nxo1]# dbca

① Oracle RAC database

→ next

② Create database

→ next → next

Db name

**hrms**

→ **Select all**

→ next → yes

③ Use same pool

pool : **rac dba**

→ next → yes

④ Use common location for all db files → next

ASM pwd : **rac dba** → **OK**

specify FTM

FRA

+ASM-DG-FRA

FTMOSFC

enable Archiving

① ASM-DG-FRA

OK

→ next → next → next → next → finish → OK

[nx01] \$ sructl config database -d hrms

\$ sructl status database -d hrms

\$ sructl status asm -n [nx01]

\$ sructl status asm

\$ sructl status nodeapps -n [nx01]

\$ sructl status nodeapps

\$ sructl status scan

\$ sructl config scan

\$ sructl status scan-listener

\$ sructl config scan-listener

\$ lsnrctl status

\$ cd \$ORACLE\_HOME/network/admin

\$ ls

\$ cd

\$ cd /u01/product/11.2.0/grid-home/network/admin/

\$ ls

\$ cat listener.ora

\$ exit

# su - grid

\$ crctl check crs

\$ crctl check cluster -all

\$ crctl check has

Steps to bringdown the RAC environment:

[nxo1]# su - oracle

① \$ export ORACLE\_SID=hrns  
\$ emctl stop dbconsole  
\$ export ORACLE\_UNQNAME=hrns  
\$ emctl stop dbconsole

→ EM Console  
→ Services  
→ d6  
→ ASM  
→ nodeapp  
→ cluster

11.9  
→ EM Console  
→ Services  
→ d6  
→ cluster

- ② \$ sructl stop service -d hrns  
③ \$ sructl stop database -d hrns

#### ④ stopping cluster:-

[nxo1] # cd /u01/product/11.2.0/grid-home/bin/  
# ./crsctl stop cluster -all

- ⑤ stopping ha:- (high availability services)  
node1 & node2:-

bin] # ./crsctl stop ha

#### ⑥ Steps to bringup the RAC environment:-

- ① starting cluster:- node1 & node2:-  
[nxo1 bin] # ./crsctl start cluster -all

bin] # ./crsctl start ha.

[nxo1] # ps -ef | grep smon

[nxo1 bin] # ./crsctl check cluster -all  
# ps -ef | grep lomon

#### ② starting database:-

[nxo1]# sructl start database -d hrns.  
\$ sructl status database -d hrns.  
\$ ps -ef | grep smon.  
\$ kill -9 15073 (hrns1)

## Backup & Restore of Clusterware files.

### OCR File:-

- By default oracle takes the backup of OCR for every 4 hours, at the end of the day & at the end of the week.
- By default master node takes the backup.
- To know the master node and the default backup location

\$ocrconfig -showbackup

lxx01	\$ORA-CRS-HOME/cdata/lxx01/ocroot.ocf
"	" " " /day.ocf
"	" " " /week.ocf

- To take the backup of OCR Manually
- Note:- whenever we make some structural changes to the cluster (adding a node, deleting a node, creating services, database etc..), recommended to take the backup of OCR manually. From 11gR2, by default oracle does this.

\$ocrconfig -export /opt/ocr.bkp

- To restore OCR from the auto backup.

→ Stop cluster

→ #ocrconfig -restore \$ORA-CRS-HOME/cdata/lxx01/ocroot.ocf

- To restore OCR from the manual backup

(i) Stop cluster

(ii) #ocrconfig import /opt/bcr.bkp

### Voting File:-

- In 10g, by default oracle will not take the backup of voting file. (In 11gR2 it takes the backup into OCR).

- If the voting file is in clusterfile system, use cp command to take the backup.

- If the voting file is in Raw partition, use dd command to take the bkp. (dd = disk dump)

- To restore voting file

(i) Stop cluster

(ii) #dd if=/opt/voting.bkp of=/dev/sdb5

## OLR (Oracle Local Repository).

- For the first time, by default oracle takes the backup of OLR (`$grid-home/cdata/node-name(ORACLE_HOME)`)
- To take the backup of OLR manually  
`#ocrconfig -local -manualbackup` [`a GRID-home/cdata/node-name`]
- To restore OLR
  - (i) stop cluster
  - (ii) `#ocrconfig -local -restore <path/file name>`.

## Oracle RAC Architecture & Internals:-

### Cache Fusion:-

- In version 9i, oracle has made lots of architectural changes to its high availability clustering technology and renamed the technology from OPS (Oracle parallel server) to RAC.
- "cache fusion is nothing but movement of data (or) transfer of data across the instances through cluster interconnect (or) private interconnect is called Cache fusion".

### Cache Coherence:-

- Maintaining consistency of data across all the instances in RAC system is called cache coherence.

### Resource Affinity:-

- Dynamically changing the ownership of resources in a RAC system is called resource affinity.

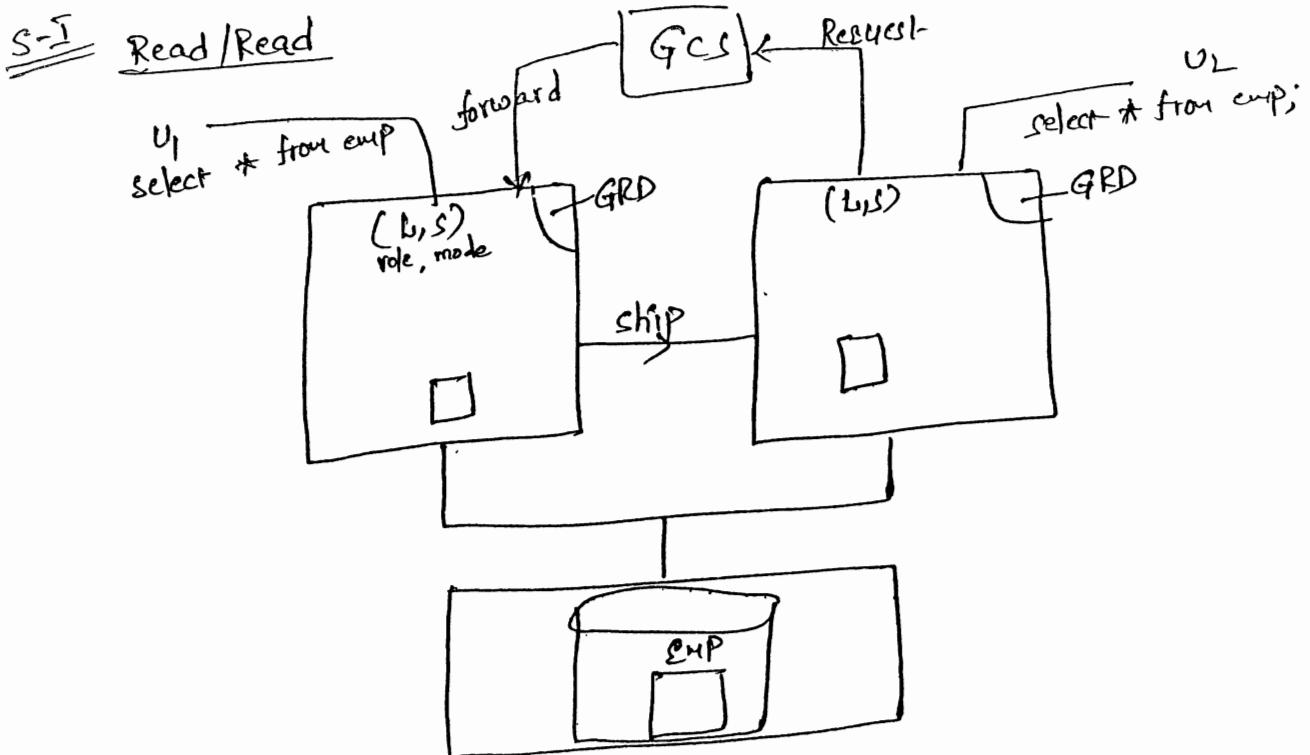
### GRD (Global Resource Directory):-

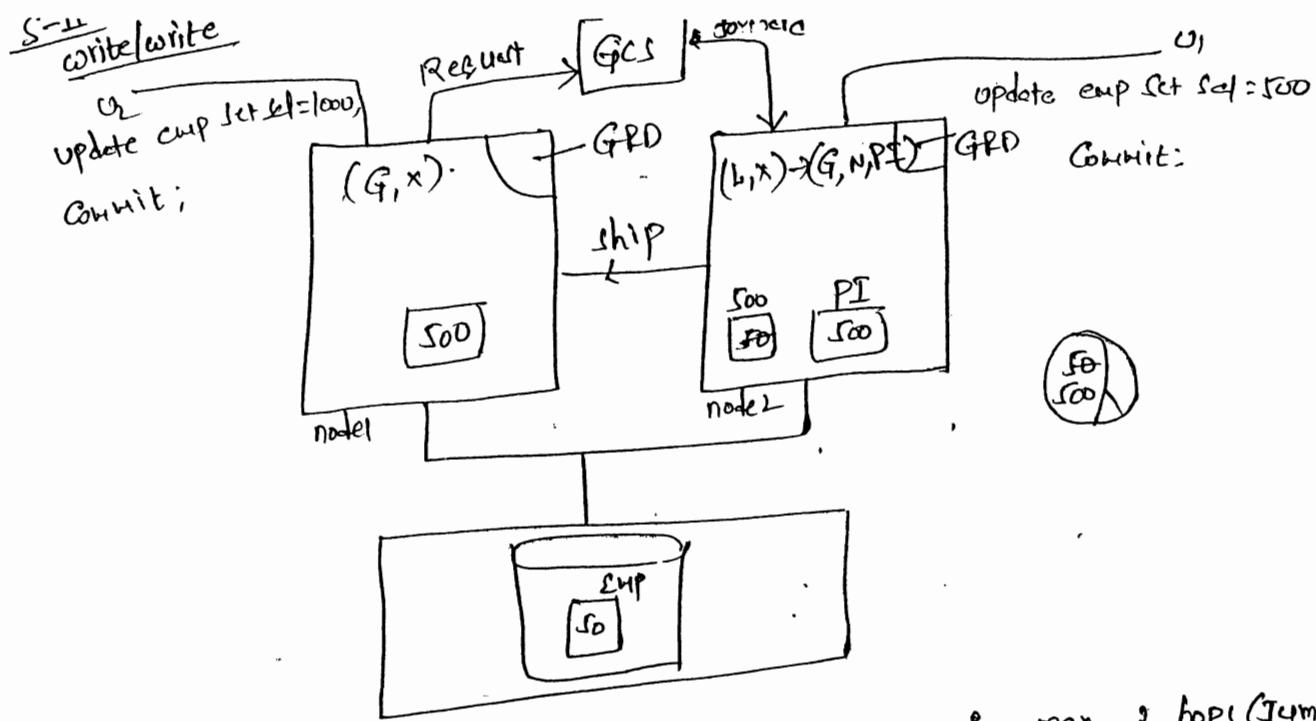
- Every instance that is participating in cluster maintains GRD.
- GRD is maintained by two services
  - ① Global Cache Service (GCS)
  - ② Global Enqueue Service (GES)
- GRD contains
  - ① data block address
  - ② Role of the resource

- (1) Mode of the resource      (2) Location of latest version w.r.t. the resource
- Some portion of memory from shared pool has been allocated to GRD.
  - (2) Role of the resource can be either Local (or) Global
  - In standalone system, role is always local to that particular instance
  - In RAC system, role can be either local (or) global.
  - (3) Mode of the resource can be either null (or) shared (or) exclusive.
- Null:- Null represents a place holder on the resource without much significance.

Shared:- Generally resources are held in shared mode in case of select

Exclusive:- Generally resources are held in exclusive mode in case of update



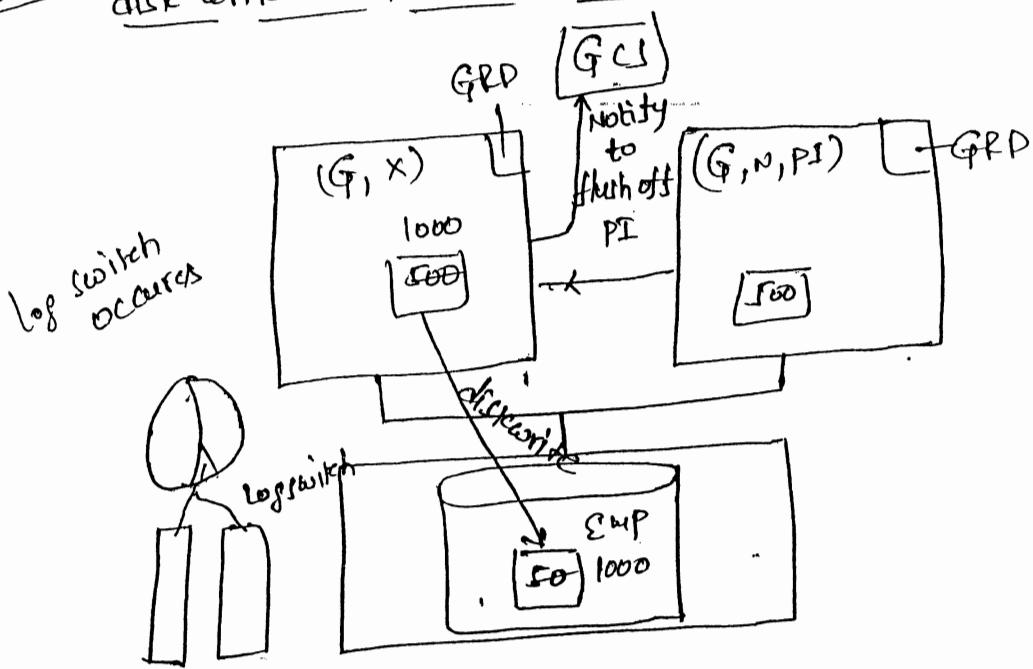


- In a 2 node cluster, an instance gets a resource in max. 2 hops (Jumps)
- \*→ In more than 2 node cluster, an instance gets a resource in max. 3 hops.

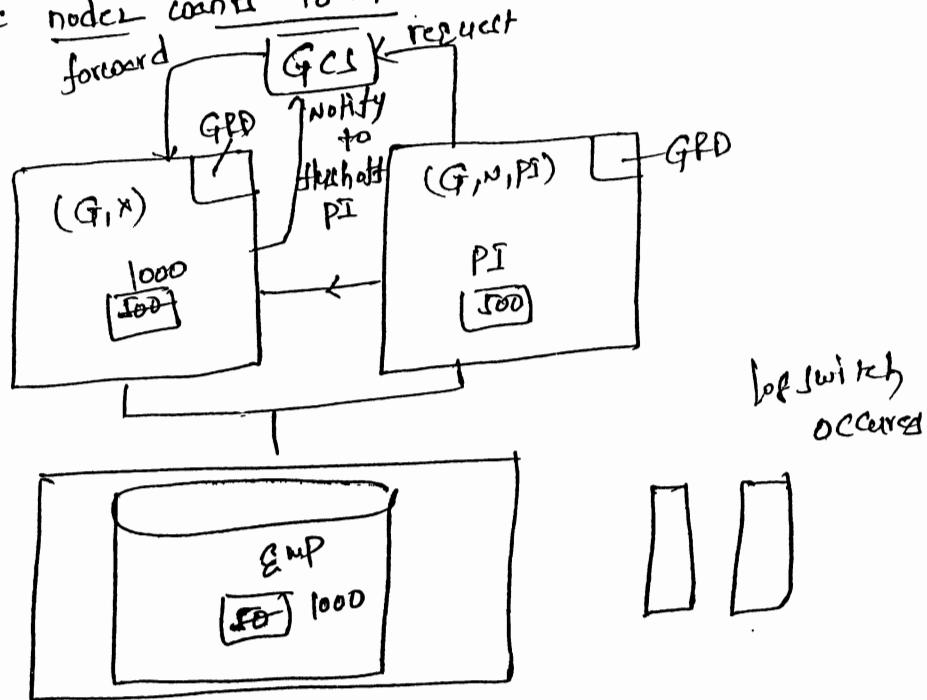
#### Concept of past image:-

- past image concept was introduced in 9i version of oracle.
- when an instance requests a resource which is already available in some other instance (updated version), the owning instance retains a copy of the resource before shipping into the requested instance.
- This copy of the resource is called past image.
- Past images are generally held in null mode and they are used to reconstruct the version of the resource in the event of requested instance failures.

S-III disk write node wants to write



S-IV disk write node wants to write



GCR :-

→ GCR plays an important role in Cache fusion. It mainly keeps track of the role, mode & location of the latest version of the resource.

GCS :-

→ GCS mainly takes care of intra instance locking mechanism related to library cache locks, Data dictionary cache locks etc..

→ In oracle, at high level there are 2 types of locking mechanisms

① Locks (held at object level)

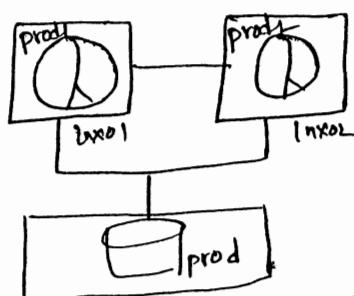
② Latches (held at oracle memory structures level)

③ enqueue (It is one type of low level locking mechanism which serializes the access to the resources).

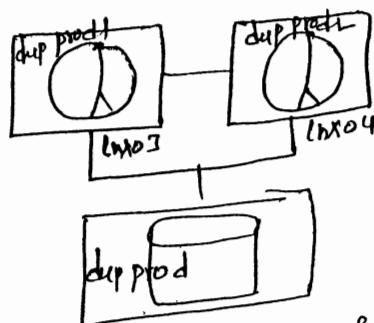
### cloning RAC to RAC:-

→ straight away we don't have the concept of cloning RAC to RAC. First we need to clone it as Standalone database and then we need to convert it into RAC either manually (or) using rconfig utility.

prodenv



Testenv



- ① Create pfile from spfile & comment all RAC specific parameters
- ② Change db\_name to dupprod
- ③ → If file system naming convention is different  
Set db\_file\_name\_convert = ('source path', 'dest path')
- ④ log\_file\_name\_convert = ('source path', 'dest path')
- ⑤ set\_no\_recovery\_through\_resetlogs = True
- ⑥ to remove internal bug 415552
- ⑦ Take rman backup along with archive logs
- ⑧ scp pfile, rman backup to test server
- ⑨ Configure tnsnames.ora (to-dupprod)
- ⑩ \$ rman target /
- ⑪ log → rman> duplicate target database to 'dupprod' nologfile namecheck;
- ⑫ log → rman> duplicate target database to 'dupprod' from activatedatabase nologfile namecheck;

- ⑤ check whether listener is up & running
- ⑥ copy initprod.ora to initdupprod.ora & initdupprodz.ora & place it in db directory
- ⑦ create password file using orapwd utility
- ⑧ \$ export ORACLE\_SID = dupprod
- ⑨ SQL> startup nomount
- ⑩ SQL> alter database add log file thread 2 group 1 <path> <size>;
- SQL> alter database add log file thread 2 group 4 <path> <size>;
- SQL> alter database enable public thread 2;
- ⑪ uncomment all RAC specific parameters, create spfile from pfile & place it in shared storage
- ⑫ shutdown immediate
- ⑬ \$cructl add database -d dupprod -o \$ORACLE\_HOME
- \$cructl add instance -i dupprod1 -d dupprod -> lnx03
- \$cructl add instance -i dupprod2 -d dupprod -> lnx04
- \$cructl start database -d dupprod
- \$cructl status database -d dupprod

Steps to create OCR file & VPD file in the event of loss & no backup:-

- 11gR1
- ① Execute rootdeinstall.sh & root delete.sh from \$ORACLE-CRS-HOME/install
  - ② Execute root.sh from \$ORACLE-CRS-HOME.
  - ③ \$cructl add database -d hrms -o \$ORACLE-HOME  
\$cructl add instance -i hrms1 -d hrms -n lnx01  
\$cructl add instance -i hrms2 -d hrms -n lnx02  
\$cructl add asm -i +ASM1 -n lnx01  
\$cructl add asm -i +ASM2 -n lnx02  
\$cructl add service -s test -d hrms -r hrms1 -a hrms2

[ops(P)] -P basic

steps in 11gR2 :-

on all nodes

- ① # cd \$Grid-home/crs/install  
# perl rootors.pl -deconfig -force

- on last node
- ② # perl rootors.pl -deconfig -force -lastnode

- ③ execute root.sh from \$GRID-HOME

- ④ \$cructl add database -d hrms -o \$ORACLE-HOME  
\$cructl add instance -i hrms1 -d hrms -n lnx01  
\$cructl add instance -i hrms2 -d hrms -n lnx02  
\$cructl add asm -i +ASM1 -n lnx01  
\$cructl add asm -i +ASM2 -n lnx02  
\$cructl add service -s test -d hrms -r hrms1 -a hrms2

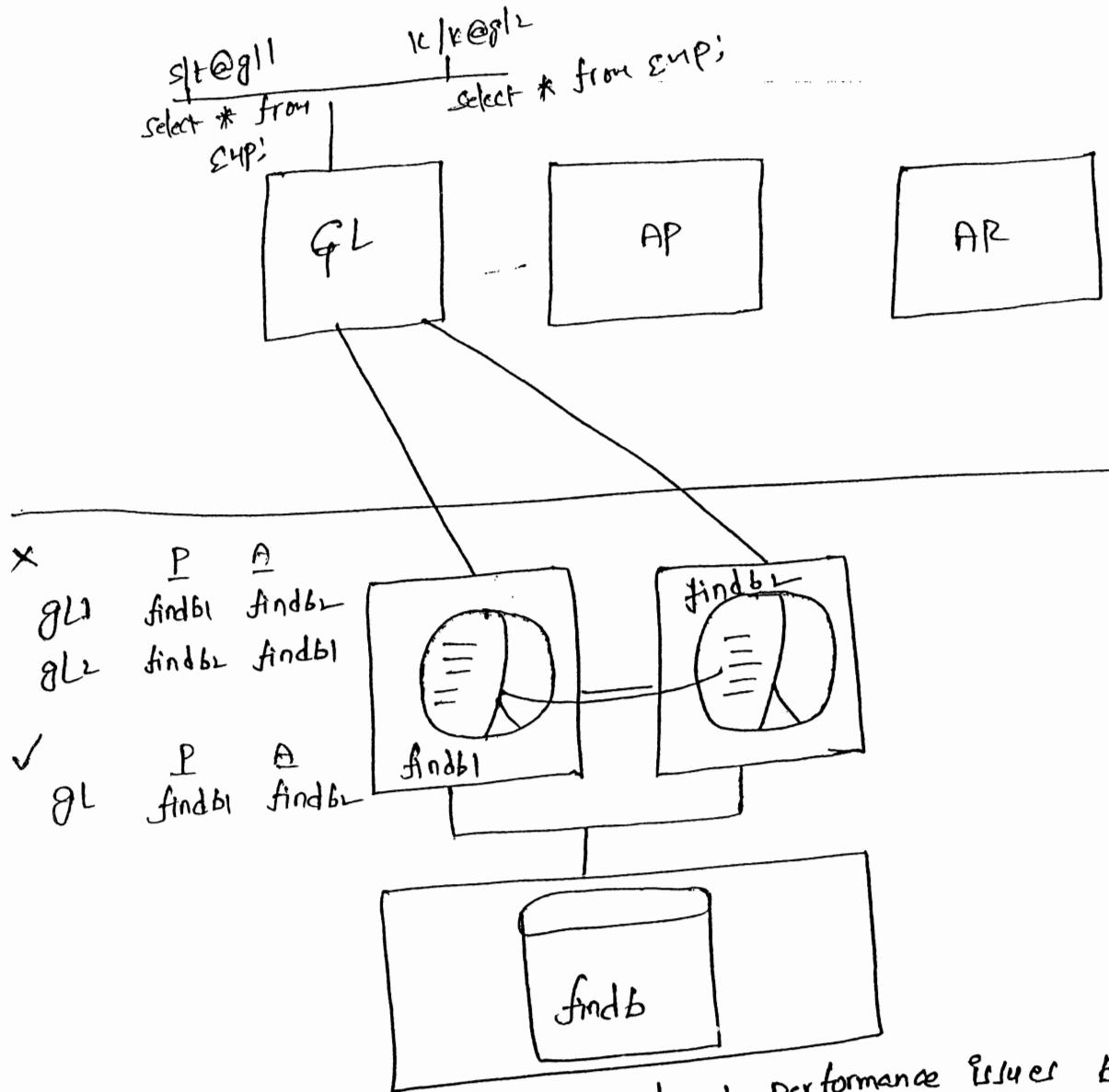
-P basic.

## Adding a node:-

- ① Configure hardware and operating system.
- ② Propagate clusterware to the new node by executing `addnode.sh` from `$ORACLE_HOME/oui/bin` [In 11gR2, GUI is not supported]
- ③ Reconfigure virtual IP's by invoking `$vipca` [11gR2 not required]
- ④ Propagate Oracle binaries to the new node by executing `addnode.sh` from `$ORACLE_HOME/oui/bin` [In 11gR2, GUI not supported]
- ⑤ Reconfigure listener by invoking `$netca` [11gR2 not required]
- ⑥ Add instance by invoking `dbca`  
`dbca` → Instance management → Add instance.

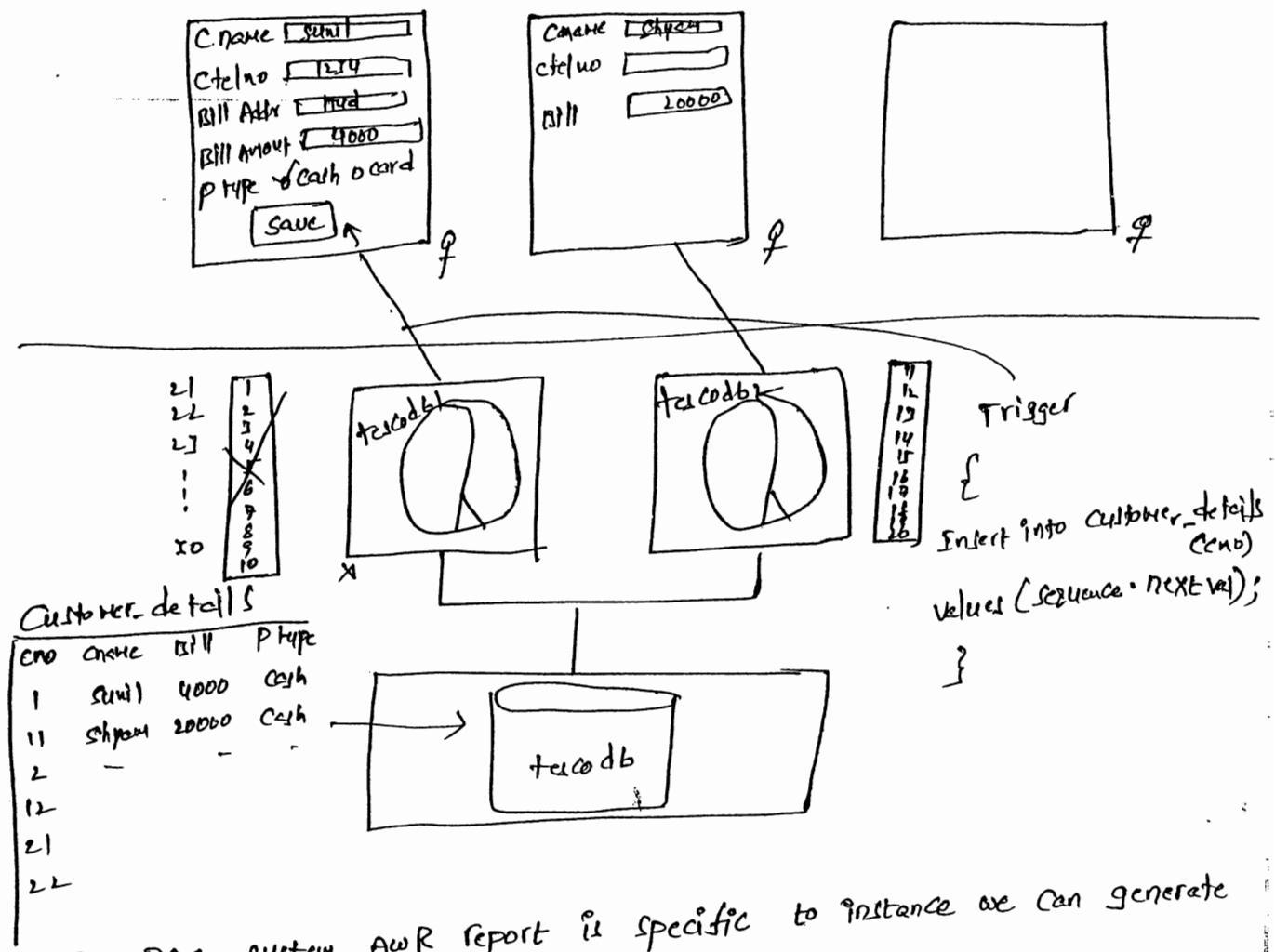
## Performance Monitoring & Tuning in RAC System:-

- In RAC system, performance issues mainly arises either because of CPU or cluster interconnect.
- If CPU utilization is very high (or) if the cluster interconnect doesn't perform well, entire RAC system suffers.
- If the latency in shipping the data is more than 15 ms, there could be a problem with the cluster interconnect.
- We can identify the latency, global cache blocks, global cache blocks.
- Corrupt by issuing a query on `gv$sysstat`
- In RAC system, cache fusion degrades the performance.
- In RAC system, cache fusion, RAC system gives better performance.
- If u avoid cache fusion, RAC system gives better performance.
- In an enterprise application environment, we can totally avoid cache fusion by creating high availability services if such a way all specific application users will hit a specific instance.



- In RAC system, sometimes we also get performance issues because of sequences
- Sequence is a database object which generates sequence of integer values.

SQL> create sequence customer\_seq\_cno  
 start with 1  
 increment by 1  
 cache 10;



- In RAC system, AWR report is specific to instance we can generate the report by executing the following script.  
`SQL> @$ORACLE-HOME/rdbms/admin/awrrpti.sql`
- From 11gR2, we can generate a consolidated AWR report by executing the following script  
`SQL> @ $ORACLE-HOME/rdbms/admin/awrgrpt.sql`
- In RAC system, apart from normal wait events [db file sequential read, db file scattered read, log file sync, free buffer etc..], we also come across RAC specific wait events like gc current block busy, gc cr block busy, gc current multi block request, gc cr multi block request etc.,
- After preliminary investigation and trouble shooting, if the occurrence of this wait events is more, raise a service request with oracle support team.

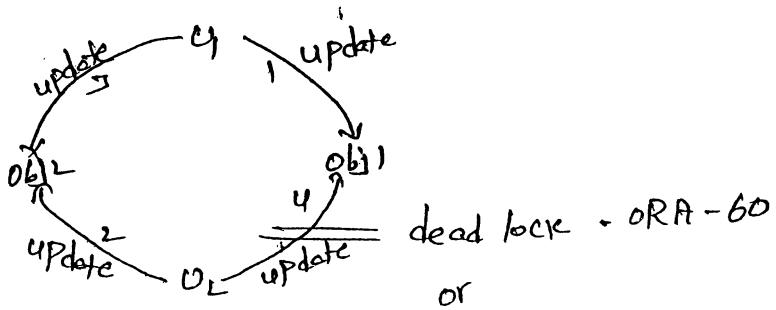
## RAC SPECIFIC (Background process)

### ① LMON (Lock monitor / Global Enqueue Service Monitor):-

- This daemon is responsible for re-configuration of locked resources whenever an instance joins (or) leaves the cluster.
- This daemon is responsible for dynamic lock re-mastering.

### ② LMD (Lock Monitor daemon / Global Enqueue Service daemon):-

- This daemon handles deadlock detections & remote enqueue requests.  
[Error: ORA - 60 (deadlock error)]



### ③ LHSN (LOCK Manager server process):-

- These are the daemons for Global Cache service.
- These daemons are responsible for shipping of blocks between the instances whenever cache fusion requests arises.
- These daemons manage cross instance call operations for shared resources.
- No. of daemons depends on the parameter gcs\_server\_processes.

### ④ LCK (Lock monitor process):-

- This daemon handles requests that are related to library cache and data dictionary cache.

### ⑤ Diag (Diagnostic Monitor):-

- Monitors the health of the instance and captures diagnostic information for failures.

Note:- During node eviction issue, we also need to monitor diag related trace files to understand the problem.

New background processes that are introduced from 11.2.0.2 :-

(1) ACMS (Automatic Controlfile to memory slave):-

- This process ensures that all operations are executed on every instance in oracle RAC despite failures.
- Co-ordinates consistent updates to a controlfile resource with its SGA Counter part on all instances in RAC.

(2) LHHS (Global Cache service heart beat Monitor):-

- Monitors the heart beat of LION, LHD & LHSN processes

(3) GCRDM (Global Config resolution slave process):-

- These are transient slaves that are started & stopped as required by "LHHS"

(4) GTXN (Global transaction process):-

- Helps to perform two phase commit for global transactions in the cluster.

(5) Ping :- (InterConnect measurement latency process):-

- Int every few seconds the process in one instance sends a message in each instance. These messages will be received by ping.

1. Mike Fult & Madhu tamra
2. GDpal krishnan
3. steve Shaw
4. [www.oracleracsig.org](http://www.oracleracsig.org).