

* Restoring SPfile.

Restore SPfile from autobackup recovery area 'FRA' db_name 'orcl';

* Restoring controlfile.

Restore control file from '_____';

* Pwfile users;

V\$PFILE_USERS; [TAKI RECOMMENDS THIS FOR SECURITY]

Encryption:-

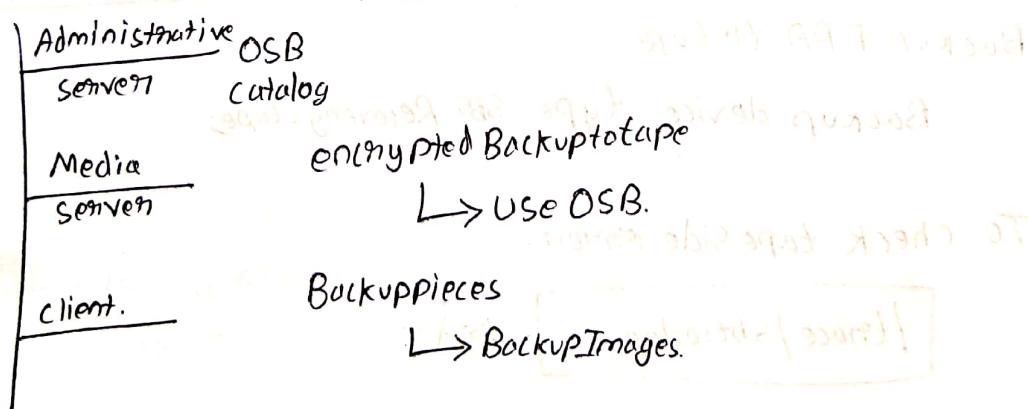
* Create a encryption file in Sqlnet.ora file.

Administrator Key Management create Keystore _____ identified by _____;

Administrator Key Management Keystore open identified by _____;
" " " " identified by _____ with backup using 'test';

ORACLE SECURE BACKUP:-

→ Similar to NSR Backup.



OSB depends on RMAN.

→ Install OSB → NSR Monitor Server.

→ Identify Administrative Server, Media Server, Client

→ update DBID, hostname

→ Retention RMAN

↓ NSR

↓ NSR

- Su-root
- Run the setup.
- Classes similar to those in OS.

Type of Backup: TAPE:

Run

{

allocate channel OEM-SBT-backup type 'SBT-TAPE' format '%U';

}

First tape software checks the DBIO and then allows.

Retention:- delete the obsolete Backups.

Setting Media parameters for RMAN:-

```
run {
  allocate channel c1 device type sbt parms
  ^ENV=(OB-MEDIA-FAMILY=my-m1);
```

Backup FRA to tape

Backup device type SBT Recovery-type;

To check tape side errors:-

/trace/Sbtio.log

OSB → web view

FLASHBACK TECHNOLOGIES:

* Flashback → uses the undo to bring back Data.

↳ Enables to view Data in the past.

↳ Rewind the changes

↳ Minimizing the time to reduce errors.

Forensics analysis:

→ Oracle Flashback Query

→ Oracle Flashback Version Query

→ Oracle Flashback Transaction Query.

Forensic recovery:

→ Oracle Flashback Table

→ Flashback Drop

→ Flashback Database.

3 Parameters

→ Undo_Management = 'Auto'

→ Undo_retention = 900 → amount of time

Giving guarantee undo retention

→ At a tablespace: → guarantee retention;

* Flashback Query

↳ Query all data at specified amount of time

* Flashback Version Query

↳ See all rows between two times;

* Flashback Transaction Query

↳ See all changes by a transaction.

Flashback Query:-

→ Select emp_id as of timestamp
where employee_id = 200.

Flashback Version Query:-

→ Select timestamp
versions between t_1 and t_2

Flashback Table

↳ brings table back in time.
↳ puts lock on the table.

Query:-

Flashback table hr.departments to timestamp

To_timestamp ('YYYY-MM-DD HH24:MI:SS');

Flashback Transaction Queries:-

↳ Transaction Rollout.

Flashback Drop:-

↳ Depends on recycle-bin;

Flashback table employees to Before drop;

DBA_Free_space.

Drop table — purge;

Alter system set recyclebin=off scope=spfile;

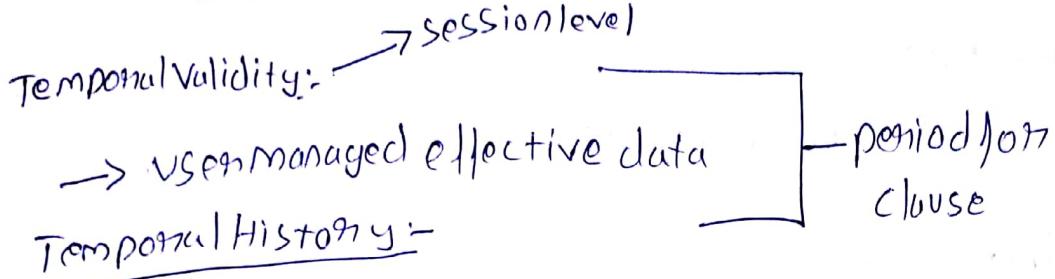
Flashback DataArchives:-

- 1) Create a new tablespace
- 2) Create flashback archive — tablespace —

Optimize Data. Quota 10M retention 1 year;

3) Alter table Flashback Archive →;
we can collect data to the Archive.

→ DDL is supported in Flashback.



DBMS_FLASHBACK_ARCHIVES.

1) Select undo retention

Select Tuned_Undoretention from v\$undostat;

where groupnum=1;

2) alter system set undo_retention = 14400 Scope=both;

3) alter tablespace undotbs1 retention guarantee;

4) alter database datofile 4 autoextend on maxsize unlimited;

5) show recyclebin

Flashback table Bar." → value from recycle bin
" to before drop rename to →;

→ alter table enable row movement;

Flashback table Bar. → timestamp_to_timestamp(' ', ' ')

FLASHBACK DATABASE:

Command → less recovery time.

→ Flashback Database to '2:05 AM' 1440
→ minutes

Simply like Replay Button.

Flashback Database uses the Archive logs and

Redologs. and Flashback logs [NB - Retention = 2880].

Blocks are stored before changing

→ Alter Database Flashback On;

RMAN > Flashback Database to time "TO_DATE('____')";

SCN

To Sequence

To view the FRA

→ v\$Flashback_Database_Log;

Compatible

L> handles how the files are handled.

Guaranteed Restore points

L> will not be deleted, we need to delete it.

TRANSPORTING DATA:-

- 1) Create a Incremental level 0 backup
- 2) Create incremental backup and apply to destination
- 3) Repeat: create and apply incremental Backups
- 4) Perform the final Incremental Backup in read only mode.

Source and destination must use same endian format.

→ get the formatname.

Select * from V\$Transportable_Platform;

Using RMAN CONVERT CLASS:-

Rman > alter tablespace bortsn readonly;

Rman > convert tablespace bortsn readonly to platform '_____';

Format ' /_____ %ov'.

Steps 1 Source destination

RMAN
→ Backup to Transport format 1 .bkf

Datapump Format '/bkf/test.dmp' Tablespace test;

RMAN Destination Source

→ Backup To Platform 'HP Tru64 UNIX'

Format 1. /bkf/test.bkf

Datapump format 1 .dmp test;

→ Restore Foreign tablespace test

Format

✓ → to restore

From Backupset 1

Dump file From Backupset 1 /bkf/test.dmp';

Transporting Database:-

on target

→ Startup mount;

→ Alter database open readonly; ✓

→ Convert Database on target Platform

Convert Script (cnvt.sql) Transport Script

'cnvdb.sql' New database 'newdb' format '/tmp/o/u';

ON SOURCE

→ Convert Database Transportscript (cnvdb.sql)

New database (newdb) to platform _____)

Format '/tmp/o/u';

COPY Binary File, PFILE USE OMF

↳ Oracle Managed Files.

Point In time recovery:-

Recovery Scope:-

- Table point in time recovery
- Tablespace point in time recovery
- Database point in time recovery.

Targetttime:-

↳ SCN number on TARGET.

Recovery Set:

↳ For tablespaces, the datafiles to be recovered.

Auxiliary Instance :- Storage disk location

Remove Relationship on other tables Before performing

point in time recovery.

→ 3 methods

→ Recover tablespace users, example

until time ' _____' Automated Auxiliary

Auxiliary Destination ' _____' Instance

↳ Duplicate Section of Database.

Target Database → Backup data → Auxiliary Instance X → Delete the Auxiliary Instance.

Target Database



until SCN

until Time

until Sequence.

REMAPTABLE

↳ newname

REMAPTABLESPACE

↳ newname

ALWAYS Test the Backups.

Duplicating a Database:-

* Reasons for Duplicating

↳ Disaster Recovery.

↳ Production Testing.

* USE RMAN DUPLICATE COMMAND.

↳ Always needs Auxiliary Instance.

pushmethod

↳ Image copies.

→ Duplicate target Database To oracle

From Active Database Using

↳ Backupsets.

Specify

→ DB_Name

→ Set newname for controlfile.

Specify the location

→ Set newname

→ DB-FILE-NAME-CONVERT

o/o → changing directory structures,

o/o → filenumber

o/o → DBID

BB-FILE-NAME-CONVERT

→ Always use Automatic Auxiliary channel

↳ not set.

Allocate Auxiliary channel

Duplicate skipreadonly

nofilenamecheck

Restricted Session → only users with restricted session will be allowed.

Creating new Diskgroup:

→ su - grid

ASMCA

↳ Create

↳ specify name.

↳ select diskpath

↳ click create.

duplicate Target Database to dbtest from ActiveDatabase

SPlite parameter_value=convert '+DATA', '+DISK2', 'ORCL', 'DBTEST'.

RMAN Performance Tuning.

↳ output can be found in

→ Alert log

→ Sbtio.log file

→ Oracle Server trace file.

→ VALIDATE BACKUP

→ RESTORE VALIDATE

↳ to check the timing of Backup.

V\$Session_Longops → all process running.

Above 6 secs.

Two Read/write

→ Synchronous I/O

→ Asynchronous I/O

Multiplexing is at
channel level.

Parallelization of Backup sets

→ (Datafile 1,4,5 channel c1)
(Datafile — channel c2)

PERFORMANCE TUNING:-

SLA

↳ Service Level Agreements.

Types of tuning

→ Proactive tuning

→ Bottleneck tuning.

ADDM

↳ Automatic Database Diagnostic Monitor.

Basic Tuning Part 1:

* Cumulative statistics

* Metrics → Secondwise information.

* AWR

↳ Views that capture information about all

* ADDM

↳ every snapshot ADDM runs.

AWR → report provides the top 10 foreground events by Total wait time.



Time Model → AWR report.

V\$SYS_TIME_MODEL

↳ time inside the Database.

V\$SES_TIME_MODEL

Dynamic Performance Views:-

* $V\$$ → views.

$V\$$ Fixed-table → all the $V\$$ views.

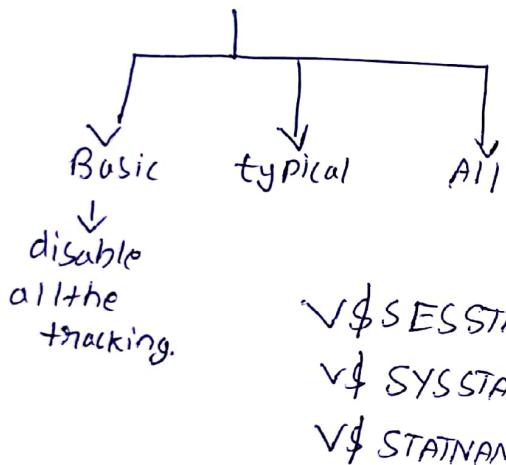
$V\$$ SQLSTATS →

$V\$$ SESSION →

$V\$$ LOCK →

$V\$$ → all are owned by sys.

$V\$$ Statistics_Level → enable or disable



$V\$$ SGAINFO

WAIT EVENTS

→ $V\$$ EVENT_NAME

→ $V\$$ SESSION_WAIT_CLASS

→ $V\$$ SYSTEM_WAIT_CLASS

→ $V\$$ SERVICE_EVENT

Enterprise Manager:-

Express

↳ built inside the Database.



↳ needs Dispatcher
Parameters

DBMS → Underline
views.

CD \$oracle_trace

↳ Inside base.

DDL Logfile

↳ capture DDL commands

→ log.xm

Enable_DDL-Logging.

→ .log.

Debug logfile → used by Oracle Support.

Usertrace files

↳ to track users.

enabling SQLTrace.

Creating an AWR snapshot:-

performance

↳ AWR Administration.

↓
Report

consists of Time model

→ AWR

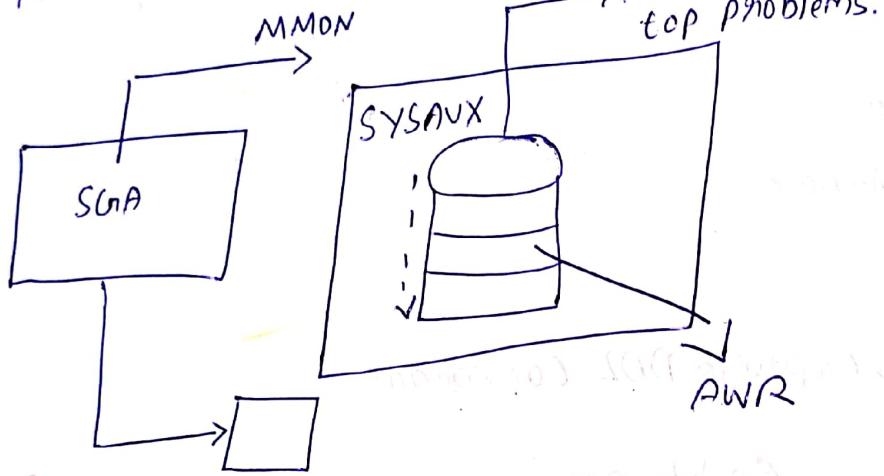
Automatic Workload Respository.

SYSAUX

→ AWR → mmon

snapshots → photographs.

AWR STRUCTURE



OEM

→ performance → AWR → top statements.

→ AWR

→ AWR ADMINISTRATION

→ takes snapshot.

DBMS - Workload - Respository.

Retention → 8 days.

Managing AWR packages

Create Snapshot

Drop - Snapshot - Range

Modify - Snapshot - settings.

executing AWR reports by using SQL plus

→ \$oracle_Home/nlms/admin/awrmp.sql

Select_catalog_role. → must be there

top-sql-statements → AWR report.

Reading the AWR report.

Compare period

↳ compare sessions.

Click on snapshot

↳ to generate report.

Identifying performance Issues:-

→ SQL

→ Instance

→ Operating System.

Steps of tuning

→ Application Design and Development.

→ normalization.

SQL_trace.

→ Deployment.

→ Production.

→ Migration, upgrade and environment changes.

ADDM → gives tuning recommendations.

Metrics and Alerts:-

Metric → rate of change of statistics.

V\$SYSSTAT

Delta tools

↳ to calculate Metric.

V\$SYSMETRIC

ASH

↳ Active Session History.

Analyzing Metrics:-

Oracle Databases.

Collection Schedule

DBA_ALERT_History

↳ to view old Alerts.

Baselines

↳ Used to set the Alerts.

Statistics_Level = typical OR ALL

↳ Baseline Enabled.

Types of Baselines:

→ Moving window

→ static.

Baseline template:

* Create
DBMS_WORKLOAD_Repository

USING AWR TOOLS:-

→ Automatic Management tasks.

↓
Optimizer statistics ↗ Daily run at particular Interval.
Segment Advisor
SQL tuning.

AWR report
↳ Between snapshots.

ADDM

ADDM
↳ Top consuming events.
↳ Analyzes the snapshots.

Statpack
↳ Does not detect much issues.
↳ Oracle Home / Admin/
Admin/
Admin Home

ADDM reports can be compared with each other.

Compare period:

DBMS_ADDM package

AWR → cause

ADDM → effort.

Active session HISTORY

→ history of Database time.

↓ v\$session.

DBA-HIST

EMERGENCY MONITORING

↳ When DB hangs

Before Bounce.

ASH Data (and Hang analysis table).

OEM

HOME

↳ Emergency monitoring. →

Real time ADDM → only GUI

↳ No sessions to kill

↳ DB hangs

Firebreak monitor

firebreak

monitor

Application Monitoring:-

* Database consists of

→ PL/SQL operations

→ In one or more sessions

SQL Monitoring.

→ Single operation

→ Multiple operations.

I) we want to monitor a database operation

→ DBMS_SQL_Monitor.Begin_operation

→ DBMS_SQL_Monitor.End_operation

→

V\$SQL-MONITOR

↳ all SQL-Monitored

Monitoring Applications:-

Services → RAC Important Concept.

↳ multiple instances.

→ 1024 Services per database.

→ 3 Services by default

Creating Service

→ DBMS_Service.Create

exec DBMS_Service.Create_Service('SERV1', 'SERV1.oracle.com')

exec DBMS_Service.Start_Service('SERV1');

Resource Manager

↳ Automatically allocates resources.

Administration

→ consumer group

↳ Resource Manager

↳ used to manage

CPU resources

setting alerts for metrics.

✓ Administration

↳ Metrics and collection settings.

process utility

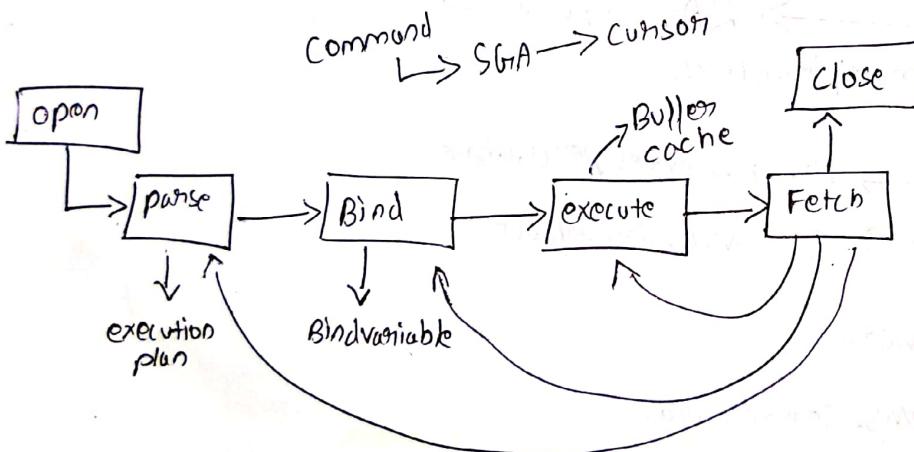
↳ collects all the trace files and combines them into one utility.

tkprof → all trace files by all users.
↳ to view the trace files.

trcexec
↳ will create the trace file

trcexec output = sov1_module.1.trc service = sov1_konut.trc
tkprof sov1_module.1.trc results.trc

SQL STATEMENTS



Parse

- ↳ Syntax
- ↳ Privileges

Types of parse

↳ Soft parse → statements,
↳ In Shared pool

↳ Hard parse

↳ Merges View definitions
and Subqueries, execution plan

SQL statement

↳ Numeric values

↳ Hashvalues

↳ Hashchain

↳ Cursors.

UP

SP



Optimizer is the brain of the SQL statements

↳ how to jointables

↳ uses object and system

Bad SQL's:-

→ more resources.

Poor

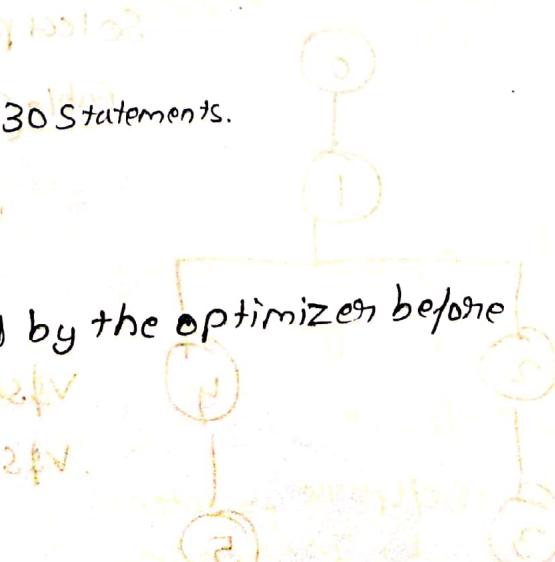
Identify SQL:-

→ AWR report. → 30 statements.

Execution plan

↳ A set of steps executed by the optimizer before

performing a SQL Statement.



To view execution plan

↳ `V$SQL-PLAN`

↳ `EXPLAIN PLAN`

↳ `AUTOTRACE`

↳ `DBMS-XPLAN`

To view execution plan

→ `DISPLAY`

→ `DISPLAY-AWR`

→ `DISPLAY-CURSOR`

`EXPLAIN PLAN`

↳ Shows how it will be executed

↳ Stores In PLAN table.

`V$SQL-PLAN`

↳ Actually executed.

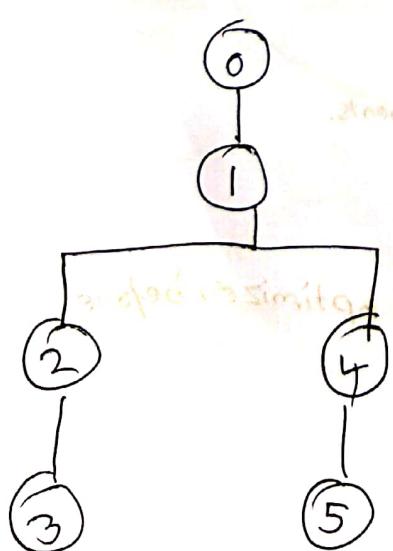
↳ `explain plan for select * from v$database;`

↳ `select count(*) from plan_table;`

`Select plan_table_output from`

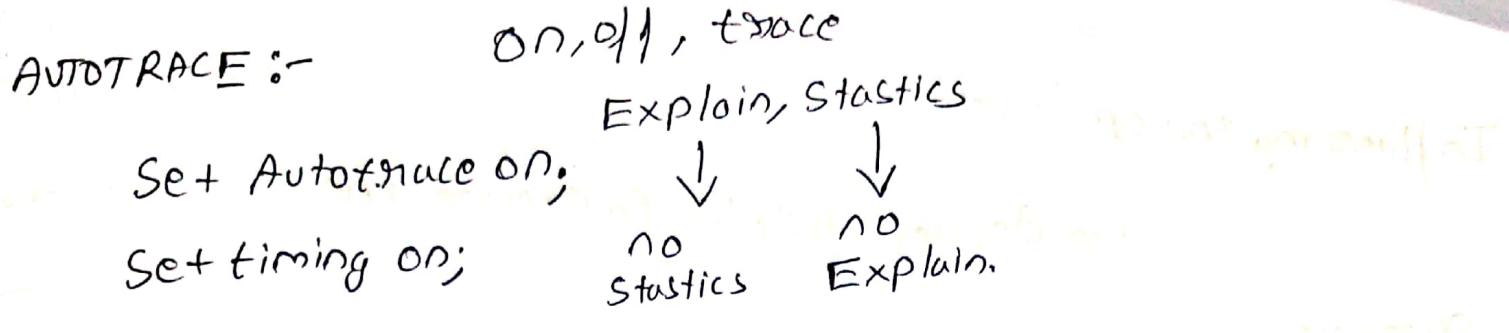
`Table(DBMS_XPLAN.Display_Cursor('_____'));`

We can even query the AWR.



`V$SQL_PLAN_STATISTICS`

`V$SQL_PLAN_STATISTICS_ALL`



SQltrace:-

→ traces all the SQL's;

alter session set SQL Trace=on;

Ekprtof : orcl_ora_12253@rc_joe.txt.txt

↳ change alertlog to txt.

process

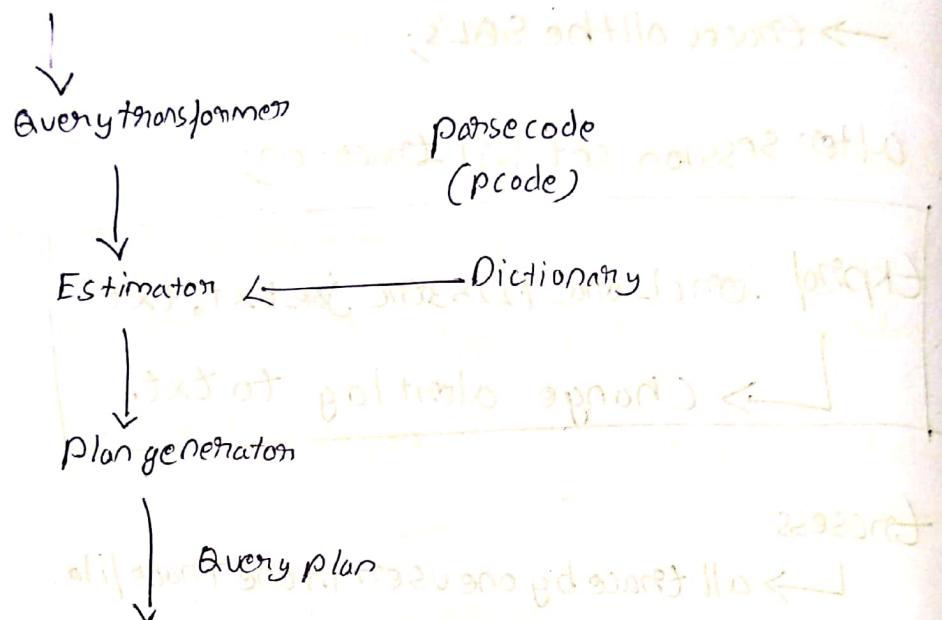
↳ all trace by one user in one tracefile.

exec dbms_monitor.database_trace_enable();

StatisticsLevel

Basic Typical ALL

Influencing the optimizer:
 ↳ decides what plan's to execute the statement.



plan methods:

- ✓ Selectivity → Select Statement
- ✓ Cardinality → Index or fulltable scan → 0.0 to 1.0
- ✓ Cost → resources

Optimizer works on Statistics:-

- Object statistics
- Dictionary statistics

V\$SQL
 ↳ soft parse.

Different statistics for different columns and indexes.

V\$optimizer_ENV

V\$SES_optimizer_ENV

optimizer parameters:-

- optimizer_index_caching
- cursor_sharing → Increases soft parses.
- PGA_Aggregate_Target → RAM usage.
↳ Join operation in memory
- optimizer_mode
- optimizer_features_enable = '11.2.0.1'

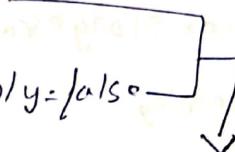
Adaptive execution plan

↳ A query plan changes during execution because runtime conditions indicate that optimizer estimates are inaccurate

- Dynamic plans → changes as the plan runs.
- Re-optimization → cursor_sharing, changes the plan next time when it runs.

optimizer_features_enable = '12.0'

optimizer_Adaptive_Reportonly = false



Adaptive execution plan activated.

Select /*+ gather_plan_statistics */ /product_name
From Order_items O, product_information P
Where O.unit_price = 0;

optimizer setting

* Fast response

* Best throughput.

Hints

↳ Show the path to the statement.

Access paths

- ✓ Full table scan
- ✓ RowID scan
- ✓ Index scan
- ✓ Sample table scan
- ✓ Cluster scan

way to access Data.

Full table scans

- Lack of index → Multiblock I/O calls
- Large amount of data → All blocks below high-water mark
- Small table

Row ID scans

- explain plan for select * from emp where rowid = 'every Datafile has Row and Rowid.'

Index scans

→ Index uniqueness

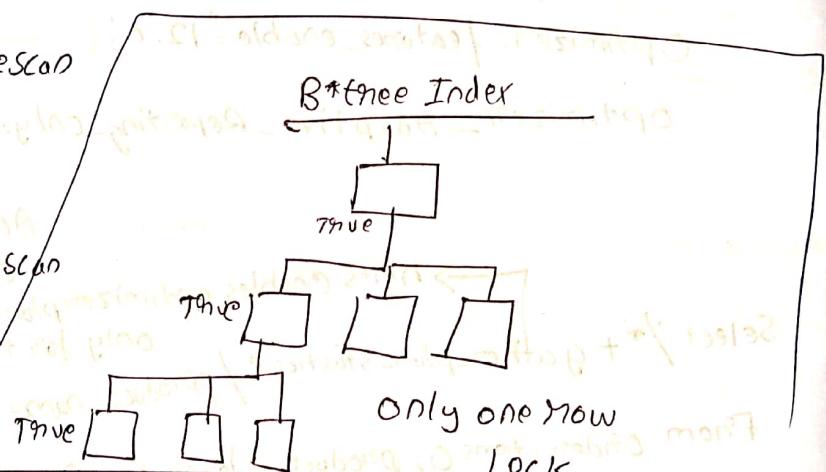
→ Index rangescan

→ Index rangescan
descending

→ Full scan

→ Fast-index full scan

→ Index Join



Bitmap Index

BB	1				
DE	1	0	0	0	
FR	0	0	1	1	1
IL	0	0	0	0	1

all the rows lock.

true
returns
the Row_id.

Bitmaps can be combined.

Join operations

→ Joins two row sources.

Optimizer decides the best join plan

→ order to join the table

→ Best join to jointables

→ Access path.

Joins

→ Nested loop join

→ Sort-merge join

→ Hash join → Fastest.

Nested loop join

↳ Small table

→ Outer table ↳ Indexes on

→ Inner table. ↳ Join columns.

Hash join

↳ Large tables

→ many partitions ↳ Large portion of table is selected for the join

Sort-Merge join

↳ when index is present.

→ Non-equi join

→ Row sources are already sorted

↳ what join is used

↳ join type

↳ cardinality

↳ distribution

↳ partitioned

↳ number of rows

↳ access path

Exadata

↳ what query

↳ everything changes.

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

↳

<p

Reducing the cost of SQL operations:-

→ Cost means I/O cost

Maintaining Indexes

- Some
 - Rebuilding → Index gets broken so we have to rebuild.
 - Coalescing → puts block empty together.
 - Creating and dropping.

Maintaining Tables

→ Controlling Space Management.

Analyze_Index name Validate structure;

Bitmap Index

- OLTP X
- OLAP ✓

We can monitor the Index usage

Alter index < > monitoring usage;

Select index_name, used, monitoring

From v\$object_usage

Text impact of index.

Alter index < > invisible;

→ hides from optimizer.

SQL Access Advisor

Composite Index

B*tree Indexes

Compression

Reverse key indexes

Composite key indexes

Bitmaps-non

Table Maintenance for performance:-

- Empty/partial blocks below the high-water affect full table scans.
- Migrated Inert affect lookup.

Reorganization

Methods

- DBMS REDEFINITION
- Shrink Segments , → Alter move → Create a table or select and then generate

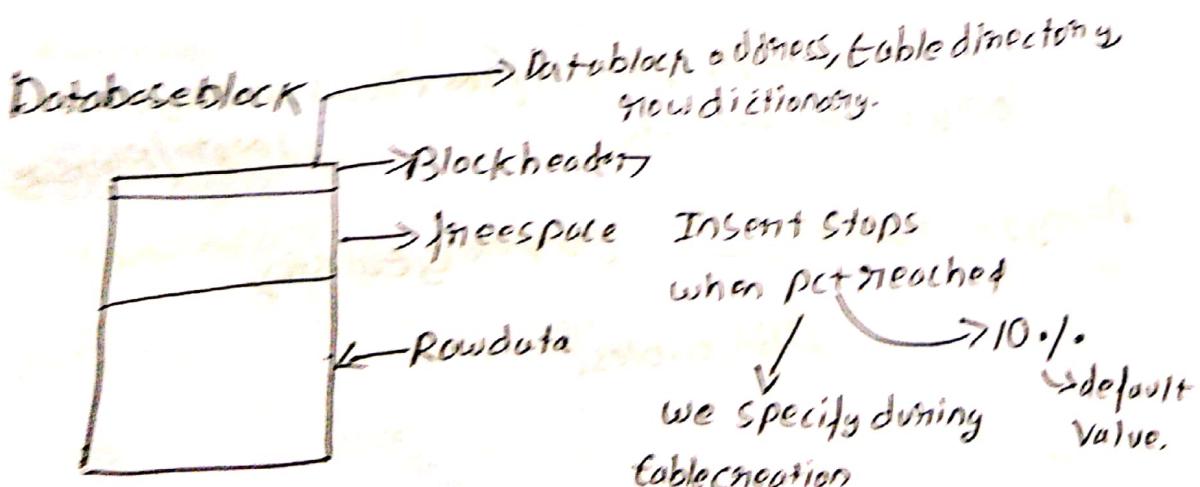
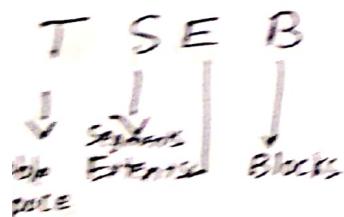
Extent Management

→ Dictionary managed

→ Locally managed

Create tablespace Datafile' size 100 Extent management

Uniform Size

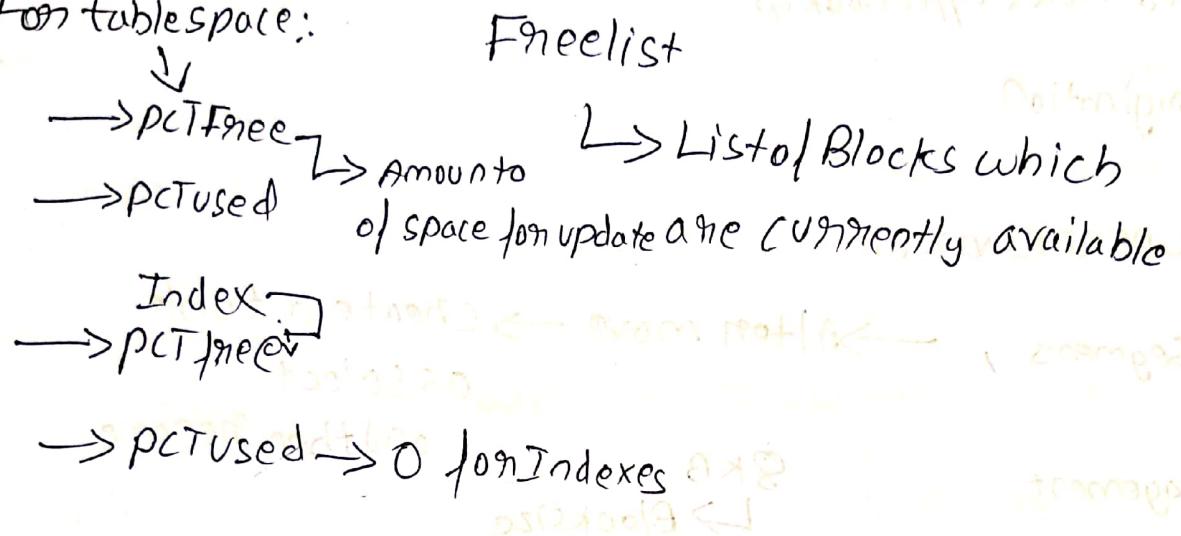


when pctfree reached → moves watermark to next row.

Automatic Segment Storage Management

↳ ASSM ↳ space managed with Bitmaps

For tablespace:



Migration and Chaining:-

Row Migration can be avoided by correct pct_free;

PCTFree

→ Default: 10

$$\Rightarrow \text{PCTFree} = 100 \times \text{upd} / (\text{average row length})$$

→ PCTUsed%

Default: 40

$$= 100 - \text{PCTFree} - (100 \times \text{rows} \times \text{average row length} / \text{Blocksize})$$

Only with freelist

Analyze Table Compute Statistics;

DBA_Tables

Administration:-

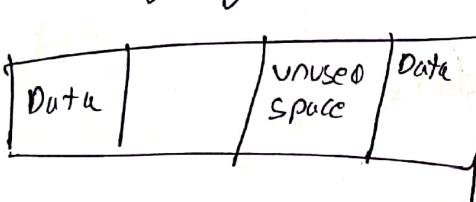
Segment Advisor.

eliminating Migrated Row

✓ Export the table

✓ Drop or truncate the table

✓ Import the table.

Shrinking Segments -  HWM → only ASM

Enabled
Tablespaces.

Alter table employees enable row movement;

Alter table employees shrink space cascade;

Index
also
shinks.

Data Compression

* Basic

* Advanced

Oracle Hybrid Columnar Compression

* Query

can be used

* Archive

portable level also.

Enable compression:-

Create table t1 Rowstore compress advanced;

Alter table t2 Rowstore compress advanced;

Alter Index Coalesce;

Alter table pct-free 10;

SQL performance Analyzer:-

RAT

↳ Real application testing

license.

↳ Just checking

will be performed

SPA process:-

1) Capture SQL Workload

by SPA.

2) Transfer the Data to test

3) make record before changes captured data

4) Make changes

5) after-change data

6)

Capturing a SQL Workload

→ Create a Staging Table Export it to the test database

Unpack it to the test database.

OEM → SQL performance Analyzer

SQL Tuning Advisor provides the profile.

Creating SQL-tuning set

↳ PL/SQL package

SQL Tuning

→ Optimizer Statistics.

↳ Set the correct optimizer statistics for good execution plan.

Stats gathering → provides optimizer the input.

↳ helps to identify the new execution plans.

Automated Maintenance Tasks

* Gathering optimizer Tasks

* Automatic Segment Advisor

* Automatic SQL Advisor.

Oracle Schedules

↳ Automated maintenance Tasks.

exec dbms_stats.gather_table

→ Automatic SQL tuning:-

→ Normal Mode

→ Automatic tuning Mode.

Automatic tuning Advisor

↳ generates a SQL profile automatically.

By default all SQL profiles

↳ configure through OEM. are enabled by default.

No profile uses → Nested Loops.

Access Advisor

↳ Is used for Indexes.

Using SQL Access Advisor: Suggest whether to
utility. → Build Indexes, Materializedviews.

SQL PLAN MANAGEMENT

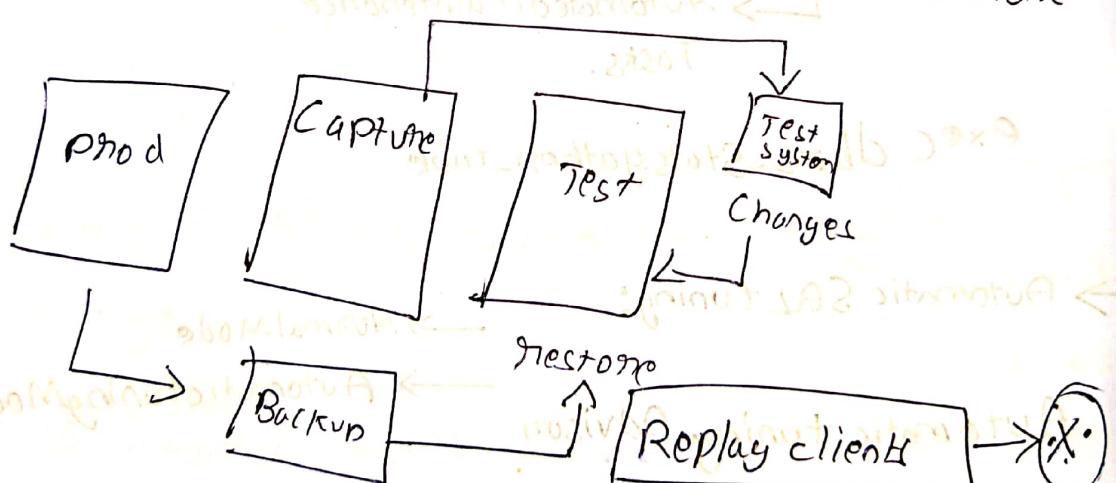
→ Baseline plan whether inHistory or not.

Database Replay:

Capture the Loads in the old database.

Clone and create a new database. → Test Database

→ Check the changes. ↴ In the clone



↳ which makes the changes in the clients

OEM

↳ performance

↳ Database Replay.

Database Replay Packages

- * DBMS_WORKLAD_Capture
- * DBMS_WORKLAD_Replay

WNC
↳ Replayer

PL/SQL commands are there to execute the database replay.

WNC is the client which can be calibrated over the time.

Practice:-

- * OEM → Database Replay

↳ Create through OEM

Tuning the Shared Pool:-

Components of Shared Pool

- * X\$ in the Shared Pool
- * Library Cache → PL/SQL/SQL stores cursors.

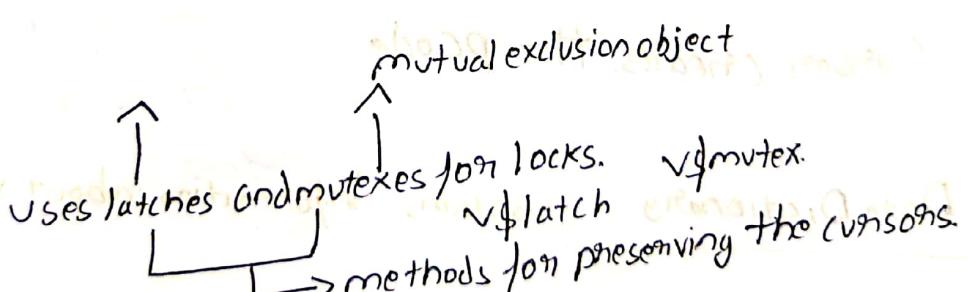
* Data Dictionary

* Server Result Cache

* User Global Area.

uses LRU

LARGEPOOL
↳ UGA.



V\$LIBRARYCACHE → X\$

parameters affecting:

Shared_Pool_Size, Open_Cursors, Session_Cached_Cursors,

Cursor_Sharing, Shared_Pool_Reserved_Size.

DBtime should be always high.

V\$SYS_TIME_MODEL;

In SQL AREA pct).miss should be less than 10%.

Sharing - Cursors:-

Exact (default) v\$SQL

Force.

↳ CursorView.

Sizing the shared pool:-

* Memory Advisors

↳ SharedPool Advisors

Avoid fragmentation

Shared - pool - reserved - size.

Keep the large objects in shared pool

Cursor creates the pcode

Data Dictionary contains information about v\$views

ResultCache_mode

↳ Cache the frequently used query.

modes: Manual

Force

Practice:-

* Sizing the Shared pool:-

* Through OEM Memory Advisor

* Reducing the Hard parse

* Change the Cursor sharing to force.

* Keeping large objects in the temporary block
use the PL/SQL procedure.

Tuning the Buffer Cache:-

→ Default buffer cache → Working sets up to 100%
↳ less wait events.

→ keep buffer cache → always have some tables in keep cache.

→ Recycle buffer cache

→ nk buffer cache. (needed memory going out of memory)

Cache fusion → RAC

Solutions to Buffer Cache Issue.

→ Properly size the Buffer Cache

→ Use the keep and Recycle pools

→ Adjust the

→ DB_BLOCK_SIZE

→ DB_CACHE_SIZE

→ DB_CACHE_ADVICE → on

↳ Advisor → off
↳ Advisor → normal

Alter Index custname_idx Storage(Buffer_Pool keep);

We can use Multiple Block sizes

DB_NK_Cache =

Database Flush cache RAM

↳ New RAM

↳ which extends the Buffer cache.

Create table mycache

Tablespace tbs

Storage (Flash_cache_Keep),

Flushing the Buffer cache.

Alter system flush Buffer-cache;

Tuning PGA and temporary memory:-

PGA is the place where HashJoin, pcode conversion takes place.

Factors

→ single pass

→ multi pass → affects performance.

→ optimal pass

AUTOMATIC PGA MANAGEMENT
PGA_AGGREGATE_TARGET

PGA_AGGREGATE_LIMIT

↳ not ending
If will
produce alert

SQL Memory Management

X views

V\$SQL - WorkArea_Active

V\$SQL - WorkArea

V\$PGASTAT

Temporary Tablespace Management.

↳ can be shared by many people.

Alter database open readonly;

No need log for Temp tablespaces

V\$Temp_usage ^{SEG} → X

↳ to view temporary tablespaces.

Temporary tablespace group

↳ Some user can use

↳ OEM

Create temporary tablespace .temp tempfile
size 600M reuse Autoextend on Maxsize

↳ uniform size 1m; unlimited. Extent Management Local

Alter tablespace temp shrink space [keep 200M]

Add tablespace temp shrink tempfile 1 .dbf;

Temporary table can be created in the temporary
tablespace.

Using Automatic memory management :-

Automatic Shared memory management.

Dynamic SGA:-

SGA-MAX SIZE

Adjusts the size of

DBCache, Large pool according to

the load.

changes size SGA size dynamically.

V\$SGAINFO

SGA is made up of granules.

SGA-target

Advisors

V\$DB_CACHE_Advice

↳ Implementations

V\$Shared_Pool_Advice

V\$JAVA_POOL → JVM

V\$Streams_Pool_Advice

MMon → is the Important for Memory management.

SGA-MAX_SIZE → high 1GB → Maximum

SGA-target → low 1MB → how much it can use.

Automatic Memory Management

↳ Adjusts between SGA

PGAs

After System Set Memory-target = 300M;

GLOBAL DATA SERVICES

→ Maintain multiple copies across many places

GDS → enables failovers

Logical components

→ GDS Configuration

→ GDS Pools → Set of Databases Default name

→ GDS Regions

Oracle DB pool

Physical

GDS

→ GDS CTI

→ GISM → command line interface.

minimum 3 GISM's.

listeners

set of databases

which provide global services

Only 12C

→ GDS Catalog

↳ Stores information about the pools.

→ Uses gsm.ora.

Cardinality

Replication lag

↳ no. of services

↳ time taken to reproduce the change in secondary

Server side load balancing

↳ Chooses the best database.

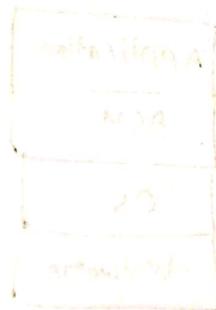
Region affinity

Local region affinity

Any region affinity

Standby's

- primary
- physical-standby
- Logical-Standby
- Snapshot-Standby



GSM

↳ Global services Manager.

Download the software.

→ use ./runInstaller.

→ GDS must always have a catalog.

→ Create user gsm-admin identified by _____.

→ Grant gsmadmin_role to gsm-admin;

gosc1> create catalog /database

add gsm-gsm _____

listener 1523)

- Catalog

To provide global services, a database must be added to the Global data services pool.



Workflow

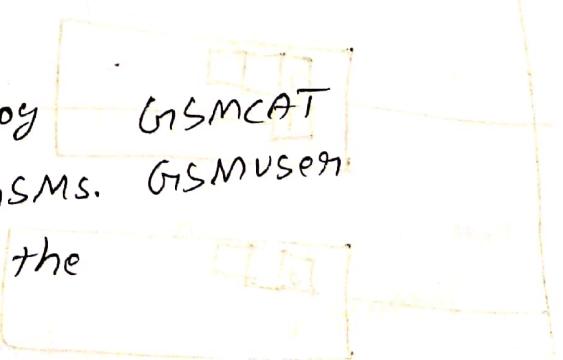
→ make changes in the catalog

GSMCAT

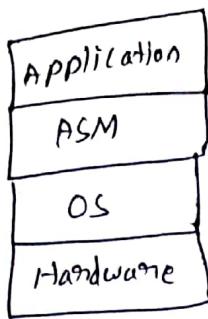
GSMUSER

→ catalog notifies all the GSMS.

→ GSM makes changes in the databases



ASM

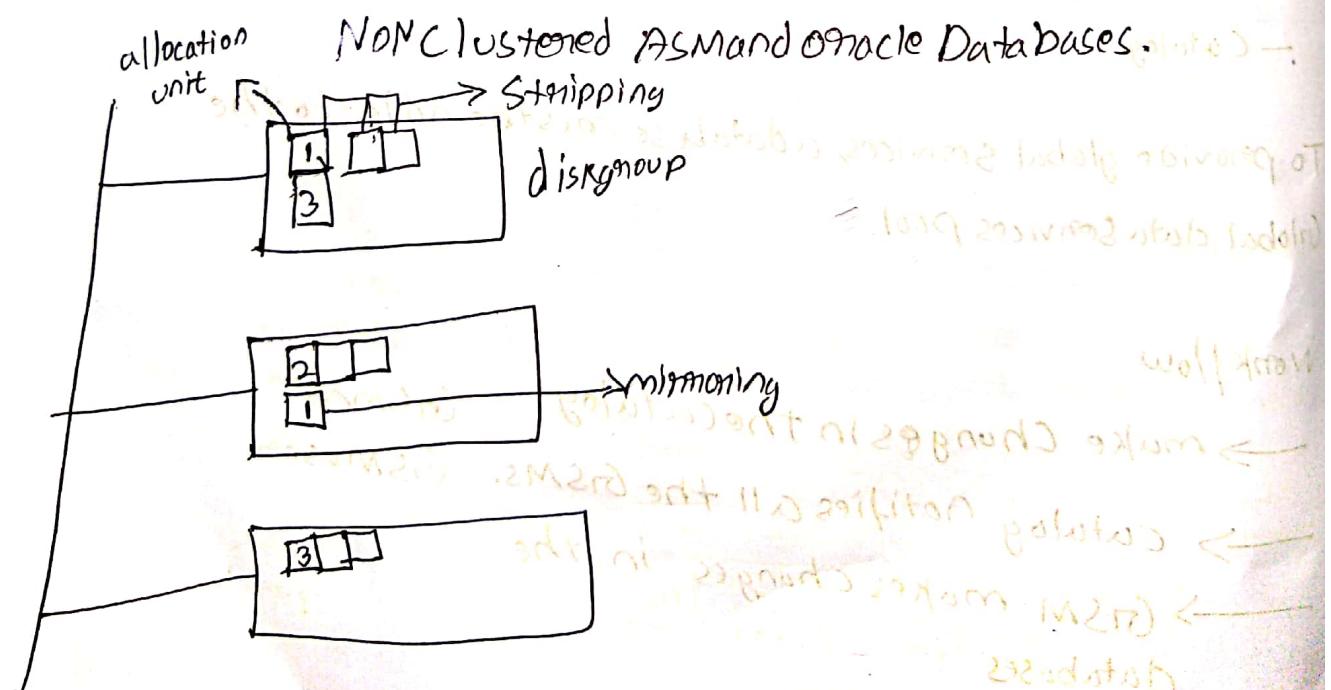


Oracle Flex ASM

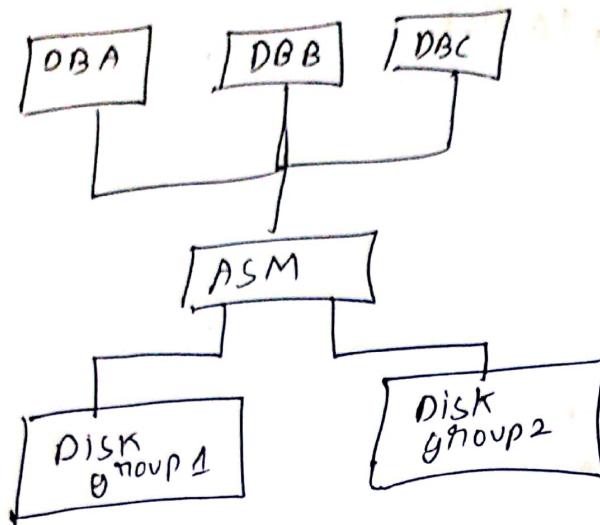
→ run a separate server apart from the database server.

ASM Features

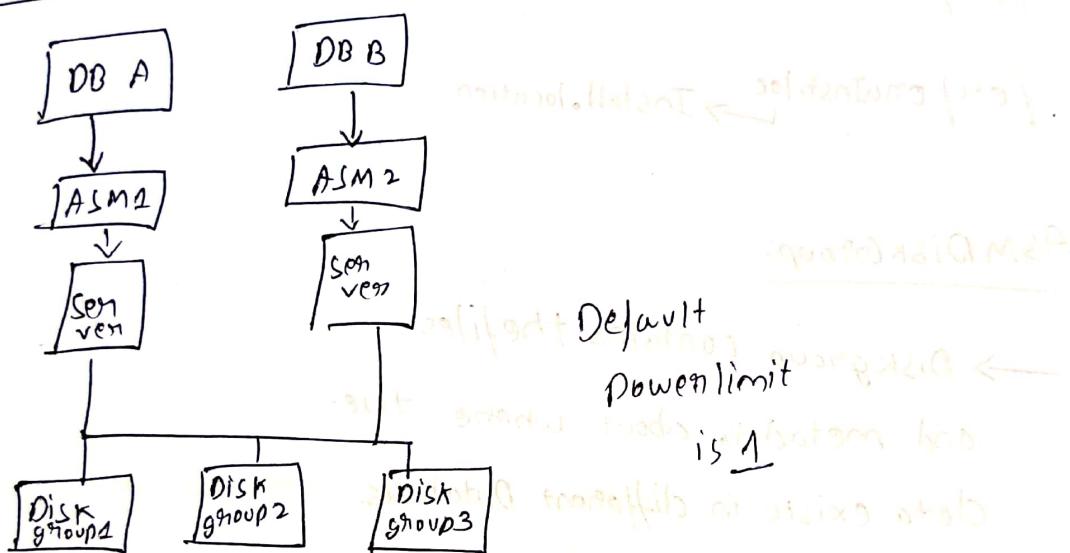
- ✓ Stripes files
- ✓ Redundancy on a file basis
- ✓ Dynamic rebalancing
- ✓ Provides adjustable rebalancing speed.



ASM Nonclustered ASM and Oracle DBS



clustered ASM

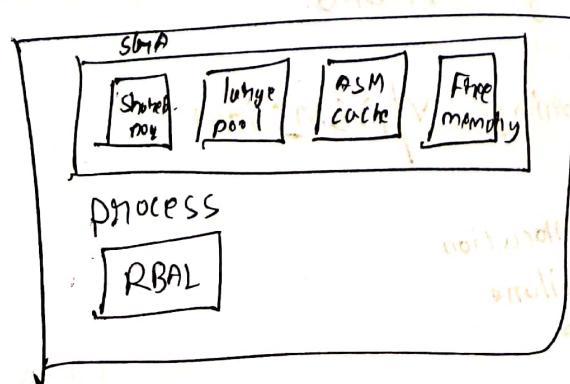


Installation:-

oracle - Base

oracle - Home

ASM Components



RBAL

↳ When new disk group added it moves allocation unit to another disk

ARBn

↳ Agent that performs Rebalancing.

ASM provides just important information.

Database writes directly to disk.

ASM also has listeners.

Files for ASM

→ pfile

→ spfile

→ oraPw+ASM

listener.ora
sqlnet.ora

/etc/oratab

/etc/oraInst.loc → Install.location

ASM Disk Group:-

→ Disk group contains the files and metadata about where the data exists in different databases.

Failure group contains voting files

↳ needed for startup.

Types of striping

→ Coarse grain striping

→ fine grain striping. → 128KB.

Redundancy

External

Normal

High → 3 way

Migration
↓
Storage Allocation
Unit in failure group

Vf ASM client.

TOOLS

- ASMCA
- ASMCMD
- server control utility Srvctl.

SYSASM

SYSDBA

SYSOPER

Sqlplus / as SYSASM

asmcmd -- privilege sysdba.

maximum 511 disk groups.

10,000 disks.

DB Maxsize

→ 128TB.

Administering ASM instances:-

ASMCMD > Shutdown -- normal
-- abort
-- immediate.

Startup -- restrict
-- normal

Sqlctl start asm -n host01

Sqlctl status asm -n host02

Sqlctl stop asm → all instances.

Sqlplus / as SYSASM

Startup.

Cnslctl Stop Cns.

* Mandatory parameters

Instance-type = ASM

ASM parameters start with

ASM -

ASM_DISKGROUPS

- ↳ list of all Diskgroups
- ↳ Default Value NULL

ASM_Power_limit

- ↳ Used for Re Balancing
- ↳ Default value is 1

Memory_target.

Storing password file in diskgroup.

\$ oraPwdFile = ' + _____ ' dbuniqueName = 'orcl'

V\$ASM_FILE

V\$ASM_DISKGROUP

V\$ASM_CLIENT

V\$ASM_OPERATION

V\$ASM_TEMPLATE

V\$ASM_CLIENT

To find database connected
osmcmd lsctdate
osmcmd lsctfma

FLEX ASM

↳ ASM

runs on another SERVER

If ASM fails clients connect to
another ASM

FLEXASM

↳ is different

USES FLEXASM

↳ USE ASM networks.

Each ASM has 3 listeners

↳ which is multiplexing.

Strict relocate ASM - currentnode — targetnode —

↳ currentnode

↳ targetnode

↳ Oracle uses interconnection network for client connection

ADMINISTERING ASM DISK GROUPS

We can use XML file to create disk group

`mkdg data.xml`

disk string → mount points
to be added
to the disk group

Syntex

Create Diskgroup 'FRA' Normal Redundancy.

Disk ' /dev/mapper/mra1 : /NAME 'FRA-DISK1' SIZE 977M,

Now we can specify

the content type.

Filter Diskgroup Dataset Attribute 'content.type='data'
Recovery.

Rename Diskgroups:-

→ `renamedg dgname=/mra1 newdgname=/mra2`

`asm-diskstring = '/devices/disk*'`

`Verbose = true.`

Viewing Attributes:

↳ `ASM_Attribute`

`asmcmd lsattr -lm -G diskgroups`

Sectordisks

Emulation native mode mode

Creating a diskgroup

Create DISK group. — Normal Redundancy

Fullgroup — DISK

' / — | — ;'

Attribute 'compatible.asm'='11.2'

'sector-size'= '4096';

viewing v\$ASM-DISKgroup

Asmcmd 1sdg

In database

v\$ASM-DISK

Adding Disks

AlterDiskgroup FRA ADDdisk

' — ' Name ' — ' FRA Drop;

Asmcmd:-

chdg' / chdgName="FRA" power="3">> ladd>

<fgName="f84">

<diskString=" — " ></fg></add></chdg>

V\$ASM_OPERATION

Asmcmd> Isop -G1 Data

We can add and drop in the same command.

Replacing Disks in Diskgroup

SQL> Alter Diskgroup data2 Replace Disk diskC7 with

'____|___' power3;

Alter Diskgroup All Dismount;

Alter Diskgroup Data2, Data3 Mount;

Alter Diskgroup add Diskgroup Rebalance Power0;

while renaming a Diskgroup
remove the old diskgroup.

partner status Table

↳ Information about all the failure group, Disknumber, Diskstatus.

partner Disks

↳ X\$KFO PARTNER DISKS

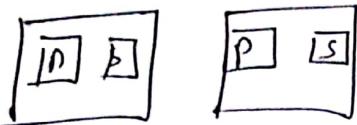
Keeps copy of Allocation units

SCRUB

↳ checks all the data is good.

Alter Diskgroup Data scrub repair;

ASM Fast Mirroring

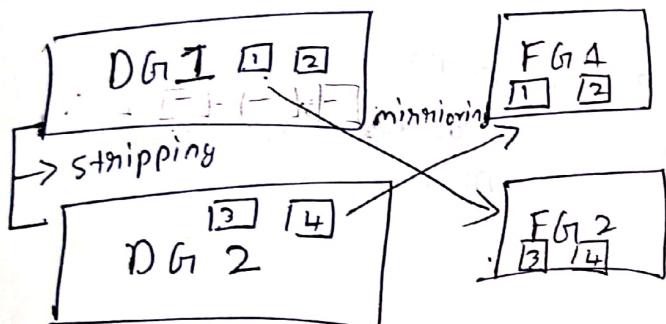


→ Do not perform Rebalancing

when disk is down

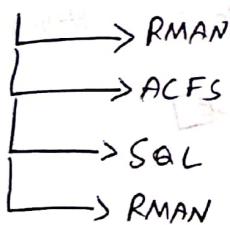
performs only when greater than Disk-repair-time;

DISK GROUP

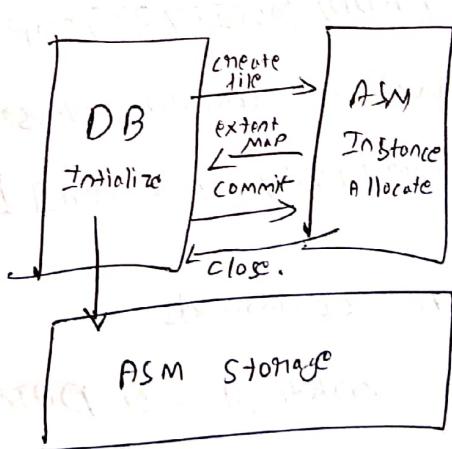


ADMINISTERING ASM FILES:

ASM clients



Interaction between ASM



Migration to ASM

can be done
using rman.

DBMS_FILE_TRANSFER

```
destination  
Create directory <-----  
Create directory <-----  
Begin  
  DBMS_FILE_TRANSFER, COPY-FILE  
End;  C'-----'
```

Fully Qualified Domain Name:

* Alter Diskgroup — add directory 'dir_name'

* Alter Diskgroup Data add files '+DATA/mydir/System.dbf'

For '+DATA/incarnation';

Templates are used to set the striping, redundancy, region attributes

Striping methods:

* FINE: 128KB Strip

* Coarse: 1AU Strip

V\$ASM TEMPLATE

Alter Diskgroup diskgroup_name

{ { ADD | MODIFY } } Template

Qualified-template-clause ::= template_name Attributes

[Emirror / High] unprotect [FINE / COARSE]

Use the commands

MKtmp1 -G DATA --redundancy unprotected

Chtmp1 -G Data --striping fine Datafile

Attributes

intensity

disk protect

Intelligent Data Placement :-

↳ place frequently accessed disk in the Hot tracks

↳ by specifying it in the template.

Select * from dbfile;

High availability

ASM filter Driven

↳ Like Datavault.

DATAGAURD:-

Physical Standby

↳ Active Datagaurd

↳ directly applies
redo logs

Logical Standby

↳ transfers as SQL statements
and applies it.

Snapchat Database:-

Created from physical standby.

Use Logical standby rolling upgrade.

Create a physical Standby Database

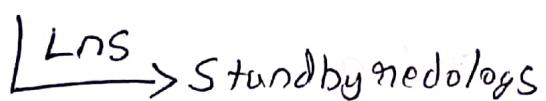
Duplicate target Database from stand by from

Active Database.

Steps at primary Database

- enable archive log mode
- create standby redo logs.
- enable force logging.

Logwriter



Online redo logs

Alter Database add

+1 always 1 extra.
is standby redo log
Standby Logfile
Size 50M.

→ Archived

and MRP applies
to Standby Database.

Online and standby should be

same. If not standby
redo log must be large.

Parameters

Log_ArchiveConfig = 'DG-Config=(Primary,Standby)'
physical

Log_Archive_Dest_N
Standby

Log_Archive_Dest_Status_N
databases
has DB-name

Archive_Lag_Target
But different

Log_Archive_Trace.
DB_Unique_Name.

→ In primary side

Log_Archive_Dest_2 =

→ DB names

↳ 'Service=PC_01Sby1' ASYNC

Valid_for=(Online_Logfiles, Primary_Role)

DB_Unique_Name='PC_01Sby1'

Log_Archive_Dest_Status_2=enable

In standby side

→ Primary

Sync or Async

↓
default

Affirm or NonAffirm

↳ Synchronous
or Asynchronous

when Standby has different directory structures

DB-file-name-convert

Log-file-name-convert

Standby-management = Manual/Auto

↳ automatic

datafiles should
be added
to the
Standby

datafiles
will be
automatically
added.

Add standby tns in primary.
tnsnames.ora

copy the
password file.

Grid
Combination
of
Clusterware
or
ASM

False server → primary
False client → detects gap automatically.
↳ which database it is located.
↳ For Standby

Use

Duplicate Database
Command
in RMAN

TO Start MRP

Altior Database Recover Managed Standby Database

Using current logfile disconnect from session;

Datagaurd broker:-

→ DGMGRL

alter system set dg_broker_start=false

DG Broker configuration files.

DGMGRL > help

Show configuration Verbose.

Install DG Broker through OEM

Configuring:-

DG Broker

↳ DB_Unique_Name
= -DGMGRL.com

When DGMGRL is installed

• dat file will be in Oracle_Home/ dbs

Create configuration 'DGconfig1' as primary Database is
Connect identifier is

Add Database pco1sby4 as connect identifier

pco1sby4;

↳ tnsnames
↳ listenername

Enable configuration

Show configuration

DBM ODL

> edit database properties

Set State = 'Transportoff'

Logxpert

↳ Sync

↳ Async

Logshipping

↳ Transposition

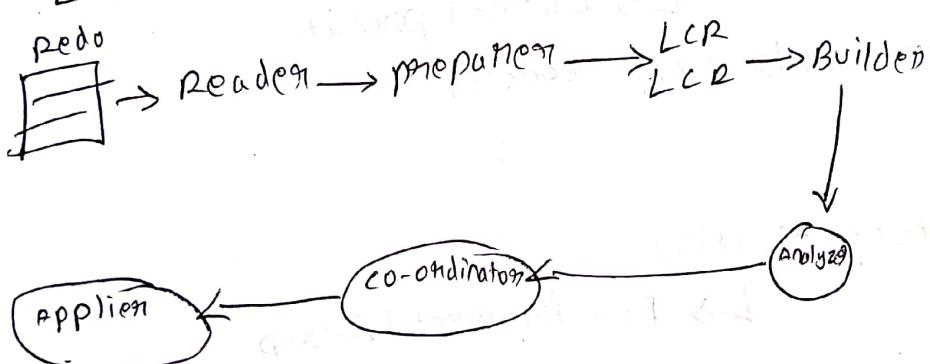
Physical Standby Configuration:

Logical Standby uses

↳ can be done through OEM

- ↳ LOGMINER.
- ↳ always should be open.

LOGMINER Process



LogArchive-Dest-2=

Location =

valid_lots=(Standby-Logfiles,
Standby-Nodes)

DB_Unique_name = _____

Steps

→ Create a physical standby

→ Stop prelog Apply → STOP MRP

→ prepare primary to support
logical standby

→ create a logminerr table →

→ open logical standby →

set
Undo_retention = 3600

execute DBMS_LOGSTDBY.BUILD;

→ Alter Database Recover

to LogicalStandby db-name;

→ Alter Database Open ResetLogs;

→ Alter Database Start LogicalStandby
Apply Immediate

Creating a physical Snapshot Standby.

↳ Testing purposes.

↳ They receive logs but do not apply them.

DBMSRL> Convert Database To Snapshot Standby;

Oracle Active Datagaurd.

Alter Database openreadonly
↳ Licensed Product.

Block change tracking

↳ For Incremental Backup.

Alter database enable block change tracking.

Configuring Data protection modes:-

performance

> max performance.

Max protection → primary shuts when not able to write to

Max availability → not shutdown

Max performance. → keep on writing.

Switchover:-

Failover to 'pcosby1' [IMMEDIATE]



complete

↳ Apply

→ Reinstate DataBase



Recreate the

Database

and attach

to Dymgr1

→ enable Database.

Flashback:-

Same as old methods.

Fast start failover

Same as ~~cheap~~ failover

but automatic includes

fast start failover.

Loss of connectivity

> fast-start

failover threshold.

observer

→ OCI

Important Component.

- Specify standby
- Fast start failover threshold
- Set additional db properties
- Set fast-start failover
- Enable fast-start failover

Edit Database Set FaststartFailoverTarget = PC01Sby1;

Edit Database Set FaststartFailover

Configuration threshold = _____;

→ modes

→ FaststartFailoverlaglimit.

→ Faststart FailoverPriorityShutdown

→ Set property observerConnectIdentification = ,

enable faststart/crossover;

enable observer;

configuring services for client connectivity.

use DBMS_Service.Create_Service(

Backup and Recovery configurations:

→ primary and standby should use the same recovery catalog.

Specify deletion of Archivelog

→ configure Archivelog retention policy to shipped to all standby;

→ " " " applied to all standby;

→ " " " applied to backup 1 times

to device type Disk;

Restoring file from Standby to primary

→ Rman connect

→ take Backup

→ Catalog

→ Set newname

Switch Datafile

Block Recovery

Recover Datafile 6 Block 3;

Excluded standby

1. Upgrade types and SQL Apply:-

→ Disable the DG broker
Before doing upgrade

→ Rolling upgrade

uses Logical standby database.

→ Stop the SQL Apply

→ first upgrade the

oracle logical standby.

→ Start the SQL Apply

→ Switchover to the

Logical standby.

Last Step
→ Upgrade primary

and switchback

→ Apply the SQL Apply

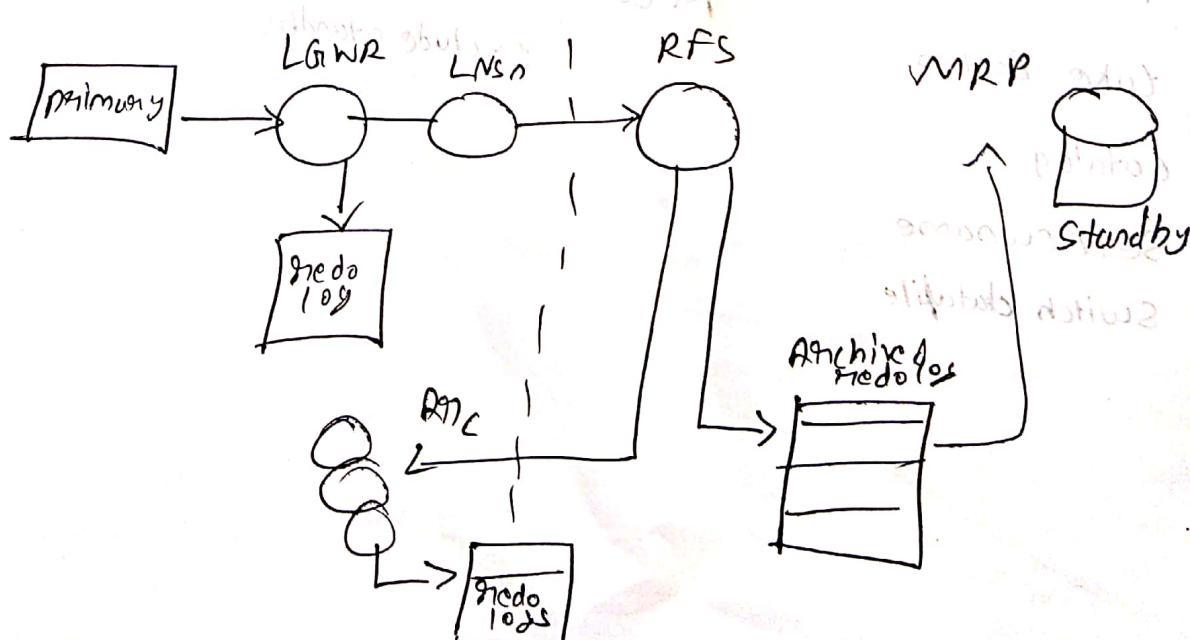
in reverse
fashion

parameters

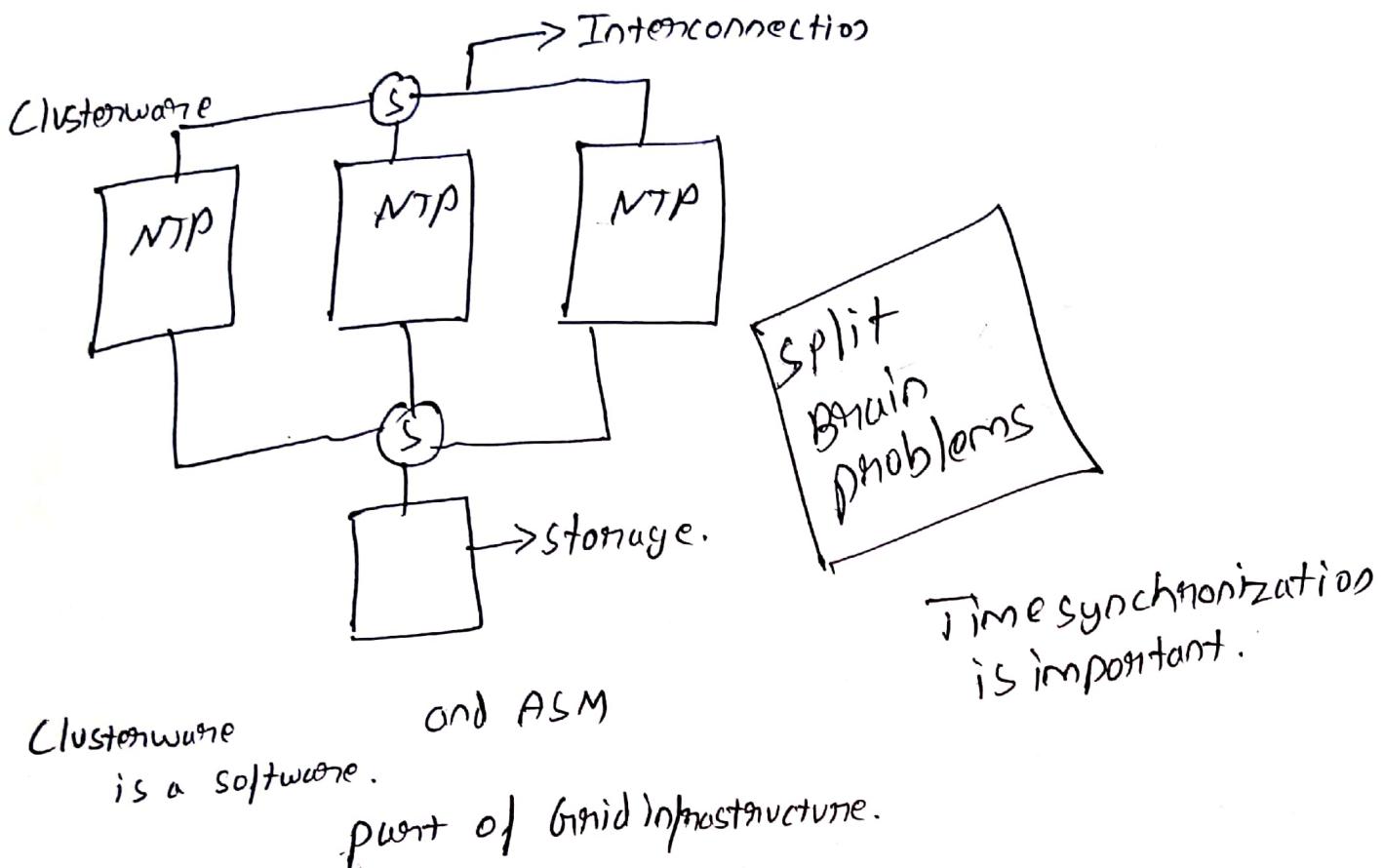
set property / NetTimeOut=20; MaxConnection.

Delay apply

Dataguard process



CLUSTERWARE AND RAC



ACFS

↳ ASM filesystem
to store
non database files.

Features

- node monitoring
- FAN
- ↳ Fast Application
Notification.

voting files

↳ Something
fails
updates
in voting
files.

HaIP