

Performance

Objectives

After completing this lesson, you should be able to:

- Monitor performance in a CDB and PDBs
- Manage SGA and PGA limits at the PDB level
- Manage AWR snapshots at the CDB and PDB levels
- Run ADDM tasks for CDB and PDB recommendations
- Manage application shared object statistics
- Control query DOP involving the containers() construct
- Manage Heat Map and ADO policy declaration in a PDB
- Manage a CDB fleet
- Describe use cases for Consolidated Database Replay



Tuning a CDB

Basic rules:

- The PDB appears to applications exactly the same as a non-CDB.
- Some initialization parameters can be set at the PDB level.
- SQL statements are tuned on a per PDB basis.
 - Common objects statistics are gathered in the application root of the common object.
 - Local objects statistics are gathered in the PDB of the local object.
- AWR tools run at the CDB and PDB level: ASH / ADDM
- AWR snapshots can be taken at CDB or PDB level.
- Instance-wide information is kept in the CDB root container.

3



Sizing the CDB

Areas of concern:

- Memory (SGA and PGA)
 - Buffer Cache
 - Shared Pool
 - Program Global Area
- CPU over allocation
- SQL Tuning

4



Testing the Estimates

Consolidated Database Replay tests:

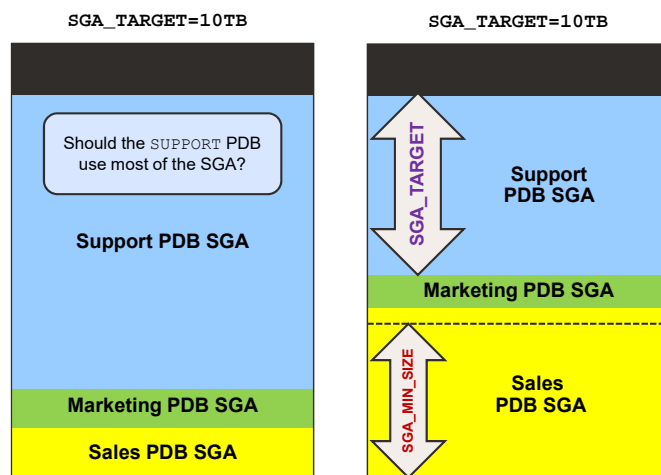
- Consolidation of servers
- Scale-up
- Peak load capacity

5



Managing SGA for PDBs

CDB Instance

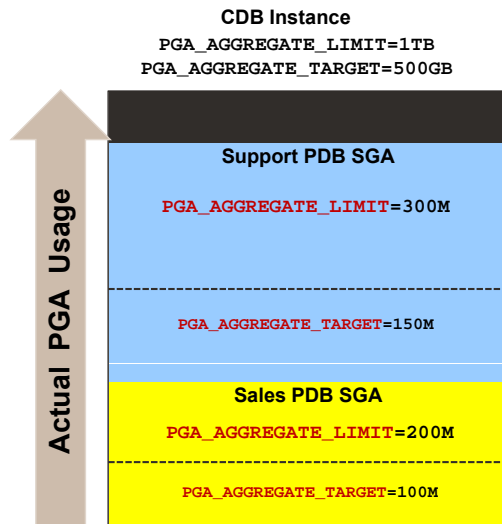


6

- **SGA_TARGET** set at the PDB level enforces a hard limit for the PDB's SGA.
- **SGA_TARGET** at the PDB level provides more SGA for other containers.
- **SGA_MIN_SIZE** set for a PDB guarantees SGA space for the PDB.
- Parameters at PDB level are:
 - DB_CACHE_SIZE
 - SHARED_POOL_SIZE
- (PDB minimums) cannot be > 50 percent of memory.



Managing PGA for PDBs



Instance PGAAggregateLimit

- No more PGA can be allocated.
- Call or session of the largest PGA users is terminated.

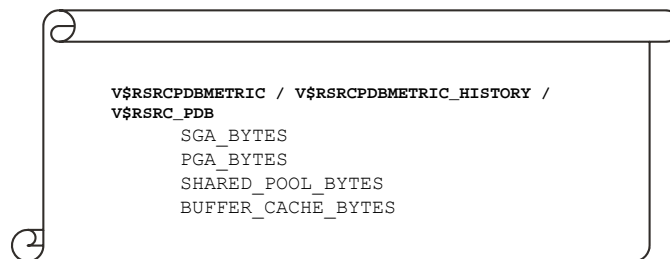
Instance PGAAggregateTarget

- All sessions must use TEMP rather than PGA.
- PDB PGAAggregateLimit
- PDB PGAAggregateTarget
- These parameters set the same behavior at the PDB level.

7

Monitoring PDB Memory Usage

- Monitor memory usage before and after configuring PDB memory parameters.



- Monitor per PDB history statistics.



8

AWR and ADDM Behavior

- AWR snapshots are created to collect statistics at CDB or PDB levels:

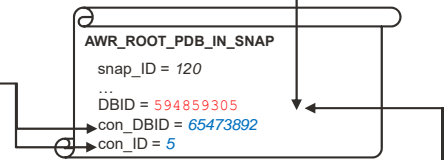
```
SQL> CONNECT / as sysdba
SQL> exec DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT (FLUSH_LEVEL => 'TYPICAL', DBID => 594859305)
```

- Collects statistics at the PDB level
- Collects statistics for each PDB opened

- ADDM runs at the CDB or PDB levels :

- Recommendations at the CDB level and PDB level

```
SQL> CONNECT / AS SYSDBA
SQL> var task_name VARCHAR2(60)
SQL> DECLARE
    taskid NUMBER;
BEGIN
    dbms_advisor.create_task('ADDM',taskid,:task_name);
    dbms_advisor.set_task_parameter(:task_name, 'START_SNAPSHOT', 97);
    dbms_advisor.set_task_parameter(:task_name, 'END_SNAPSHOT', 119);
    dbms_advisor.set_task_parameter(:task_name, 'DB_ID', 594859305);
    dbms_advisor.execute_task(:task_name);
END;
```



9

Configuring Automatic ADDM Analysis at the PDB Level

- Analyze a PDB's AWR data stored inside the PDB.
- Analyze AWR data imported into a PDB.
 - Better PDB-specific ADDM analysis and recommendations than ADDM analysis on the CDB root

1. Enable PDB AWR snapshot creation on the CDB root and on each PDB:

```
SQL> ALTER SYSTEM SET awr_pdb_autoflush_enabled = TRUE;
```

2. Set the AWR snapshot interval to greater than 0 at the PDB level:

```
SQL> CONNECT sys@PDB1 AS SYSDBA
SQL> EXEC dbms_workload_repository.modify_snapshot_settings(interval => 60)
```

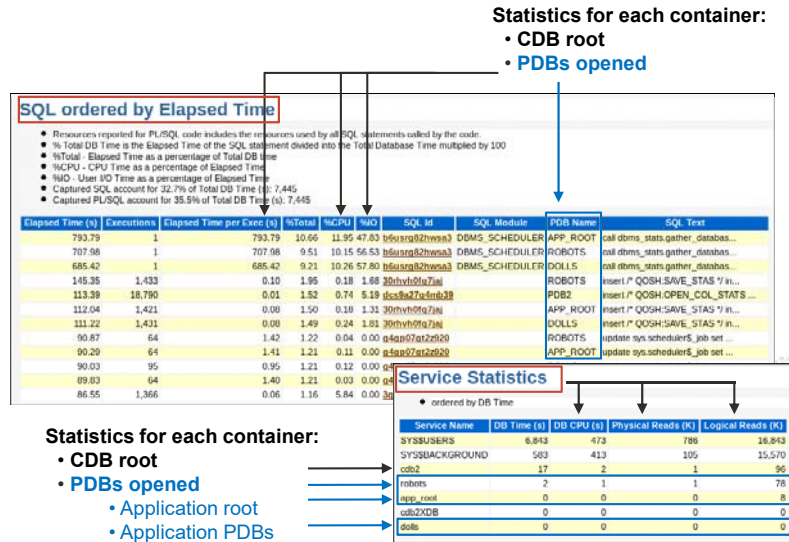
3. Execute the ADDM task (manually when required):

```
SQL> CONNECT sys@PDB1 AS SYSDBA
SQL> EXEC DBMS_ADDM.ANALYZE_DB(:tname, begin_snapshot =>1, end_snapshot =>2)
```

10

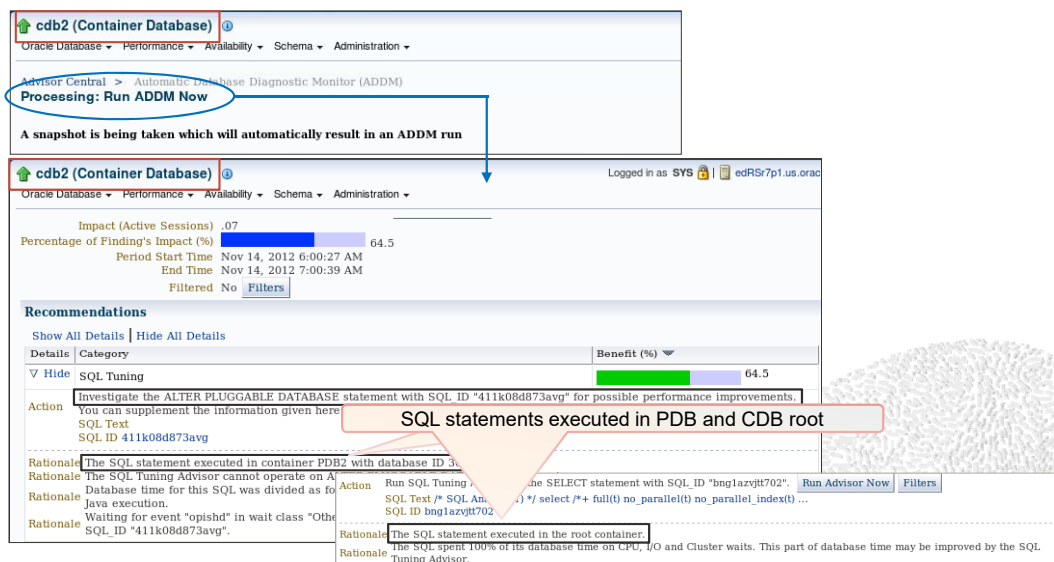
```
DBA_ADDM_TASKS
CDB_TYPE_DETECTED
```

AWR Report



11

ADDM Tasks: At the CDB or PDB Levels



12

Enabling ADDM in a Pluggable Database

To enable ADDM in a PDB:

- Set the AWR_PDB_AUTOFLUSH_ENABLED initialization parameter to TRUE in the PDB using the following command:

```
SQL> ALTER SYSTEM SET AWR_PDB_AUTOFLUSH_ENABLED=TRUE;
```

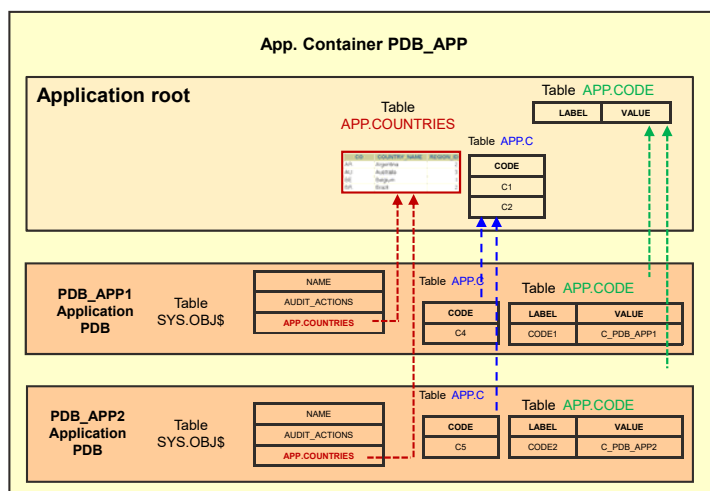
- Set the AWR snapshot interval greater than 0 in the PDB using the command as shown in the following example:

```
SQL> EXEC  
dbms_workload_repository.modify_snapshot_settings(interval=>60);
```

13



Basic Rules: Statistics for Common Objects



- Statistics for common data-linked objects are gathered in the application root.
- Statistics for common metadata-linked objects are gathered in the application PDB.
- Statistics for common extended data-linked objects are gathered in both the application root and the application PDB.
- Statistics for local objects are gathered in the application PDB.

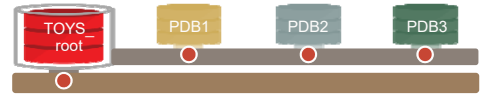
14



Controlling the Degree of Parallelism of Queries

```
SQL> CONNECT toys_app@toys_root
```

```
SQL> SELECT sum(revenue), year  
FROM CONTAINERS(sales_data)  
WHERE year = 2014 GROUP BY year;
```



- Queries using the `containers()` construct execute in parallel by default.
- The query DOP used is 4: sum (app. root + opened application PDBs).

```
SQL> ALTER SESSION SET containers_parallel_degree = 12;
```

```
SQL> SELECT sum(revenue), year FROM CONTAINERS(sales_data)  
WHERE year = 2014 GROUP BY year;
```

- The query DOP that is used for each statement by using the `containers()` construct is now 12.

```
SQL> SELECT /*+ CONTAINERS(DEFAULT_PDB_HINT=PARALLEL 8*/ sum(revenue), year  
FROM CONTAINERS(sales_data)  
WHERE year = 2014 GROUP BY year;
```

- The query DOP that is used for the statement is now 8.

15



Heat Map and ADO Support

Oracle Database CDBs support ADO and Heat Map statistics.

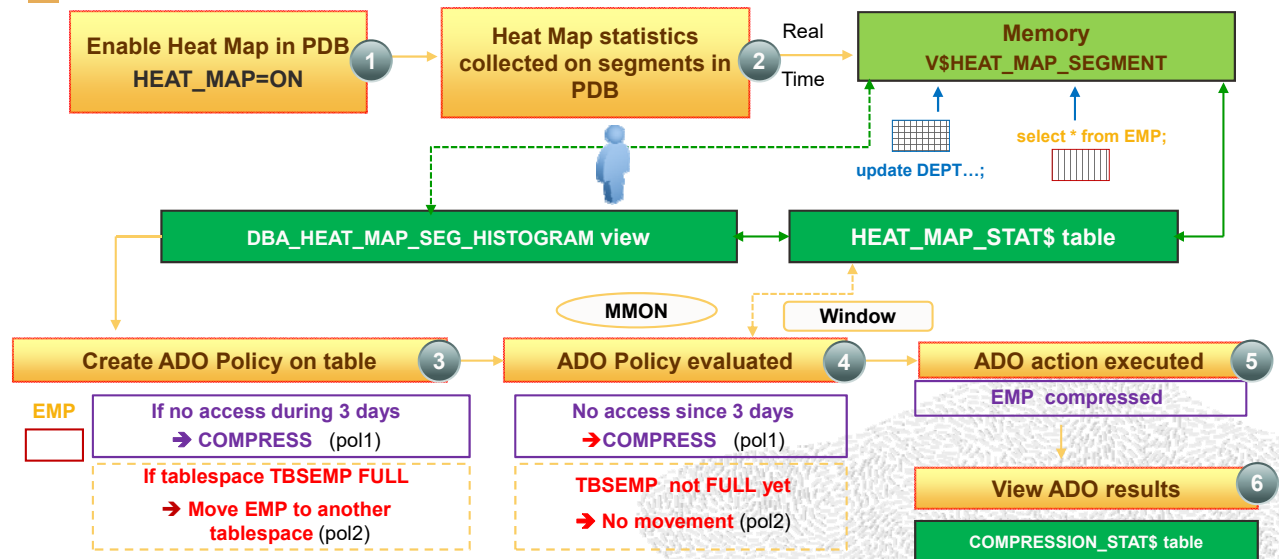
- ADO policies automatically compress data in objects in PDBs.
- ADO policies automatically move segments in PDBs to other tablespaces in the same PDB when necessary.
- ADO is dependent on Heat Map statistics collection and does not work unless Heat Map is enabled.

```
HEAT_MAP = ON
```

16



Managing Heat Map and ADO Policies in PDB

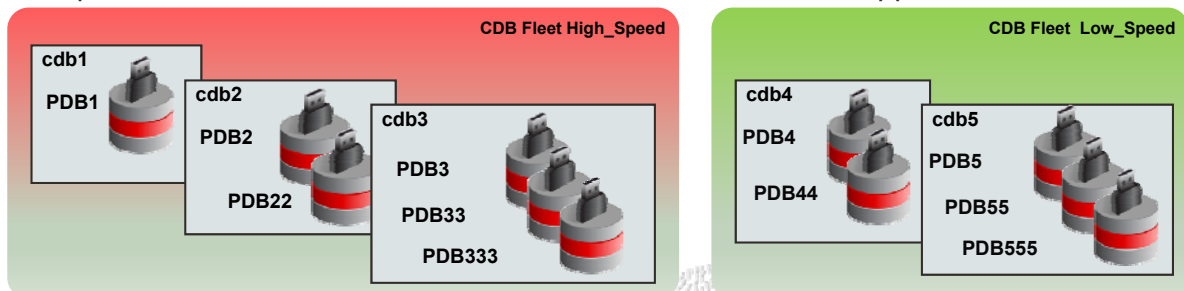


17

CDB Fleet

A CDB fleet is a collection of different CDBs that can be managed as one logical CDB:

- To provide the underlying infrastructure for massive scalability and centralized management of many CDBs
- To provision more than the maximum number of PDBs for an application

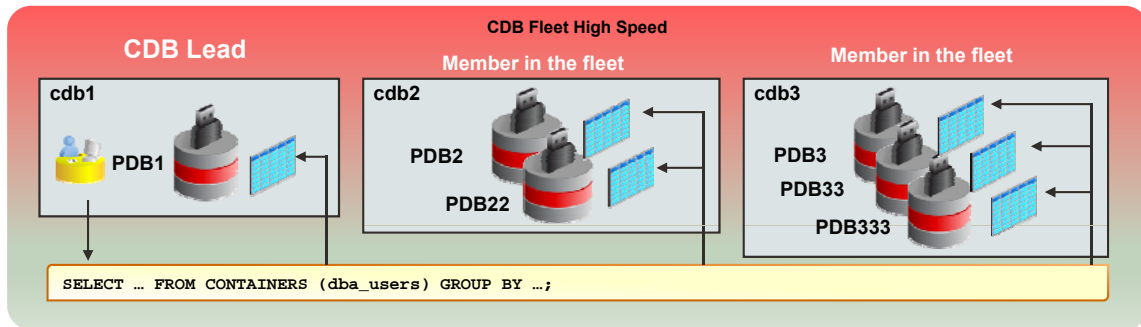


- To manage appropriate server resources for PDBs, such as CPU, memory, I/O rate, and storage systems

18

Use Cases

- Monitoring and collecting diagnostic information across CDBs from the lead CDB
- Querying Oracle-supplied objects, such as DBA views, in different PDBs across the CDB fleet



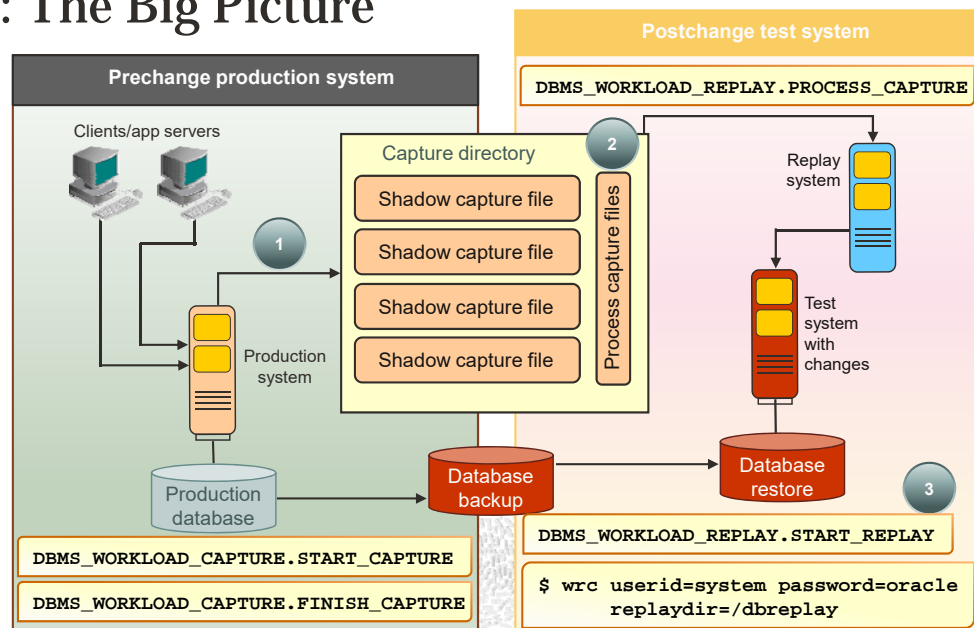
- Serving as a central location where you can view information about and the status of all the PDBs across multiple CDBs

19

DB Replay: The Big Picture



Capture at the CDB or PDB levels



20

Capturing and Replaying in a CDB and PDBs

DB Replay procedures at the CDB level:

```
SQL> CONNECT / AS SYSDBA
SQL> exec DBMS_WORKLOAD_CAPTURE.START_CAPTURE( NAME => 'OLTP_peak', -
                                                DIR => 'OLTP' )
```

DB Replay procedures at the PDB level:

```
SQL> CONNECT sys@PDB1 AS SYSDBA
SQL> exec DBMS_WORKLOAD_CAPTURE.START_CAPTURE( NAME => 'OLTP_peak_PDB1', -
                                                DIR => 'OLTP_PDB1' )
```

21



Consolidated Database Replay Use Cases

Perform realistic load testing for scenarios including:

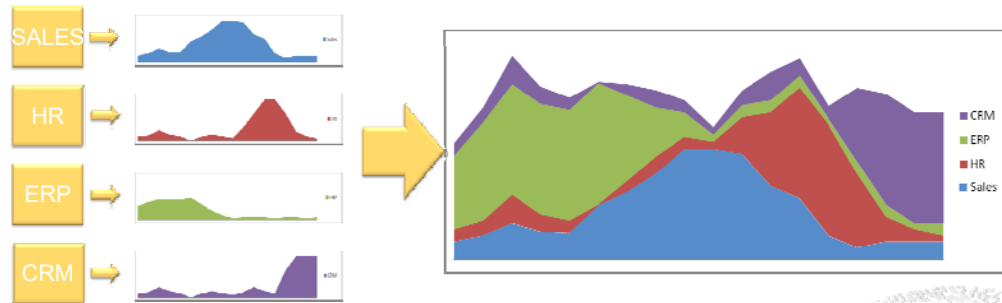
- Consolidation of servers
- Workload scale-up
- Test peak load capacity

22



Use Cases: Source Workloads

Workload: Each application has distinct peaks at different times of the workday.

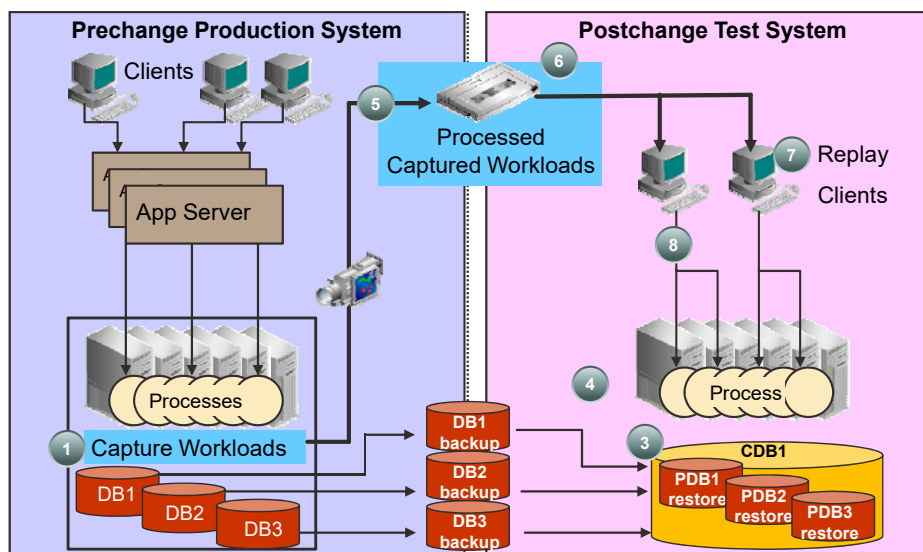


- Each workload captured on different databases and being consolidated is independent of the other.
- Allows multiple workloads captured from different non-CDBs or PDBs to be replayed concurrently in a single CDB

23



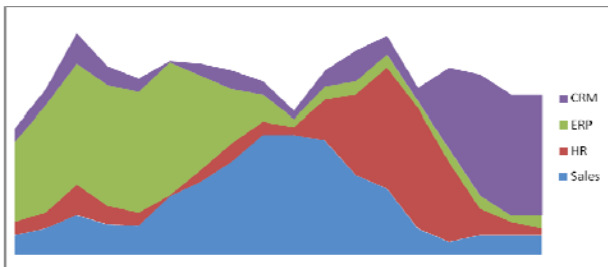
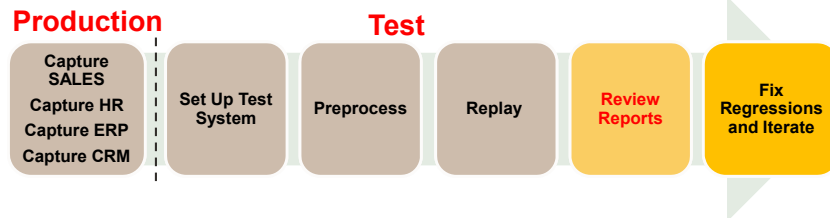
The Big Picture



24



Step By Step



- Verify:
 - Quality of service metrics as perceived by application user per workload
 - Overall capacity
- Only replay can provide definitive answers.

25



Summary

In this lesson, you should have learned how to:

- Monitor performance in a CDB and PDBs
- Manage SGA and PGA limits at the PDB level
- Manage AWR snapshots at the CDB and PDB levels
- Run ADDM tasks for CDB and PDB recommendations
- Manage application shared object statistics
- Control query DOP involving the containers() construct
- Manage Heat Map and ADO policy declaration in a PDB
- Manage a CDB fleet
- Describe use cases for Consolidated Database Replay



26



Practice 10: Overview

- 10-1: Monitoring performance at the CDB and PDB levels
- 10-2: Getting performance ADDM recommendations at CDB and PDB levels
- 10-3: Monitoring and tuning SQL executions at the PDB level
- 10-4: Configuring a CDB fleet with its CDB lead and CDB members