Oracle Multitenant

Simplify Consolidation with Oracle Database 12c

Agenda

- Rethinking Database Architecture for the Cloud
- Multitenant Architecture
- Capabilities Enabled
- Managing Shared Resources
- Upgrading to Multitenant
- Use Cases

Agenda

- **Rethinking Database Architecture for the Cloud**
- Multitenant Architecture
- Capabilities Enabled
- Managing Shared Resources
- Upgrading to Multitenant
- Use Cases

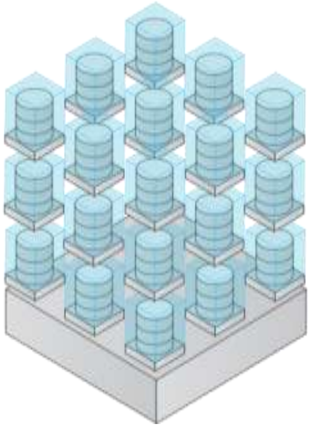
Industry Today

Pressures	Requirements	Trends
Increase Agility	Performance, scalability	Private Cloud
Reduce Costs • Capital Costs • Operating Costs	Reliability, availability	Massively Powerful Servers (Exadata)
Non-stop operations	Security, tenant isolation	Virtualization
DBA challenges: • Patching, upgrades • Provisioning	No application changes	Simplification • Standardized Services • Self-Service

Private Cloud Database Architectures

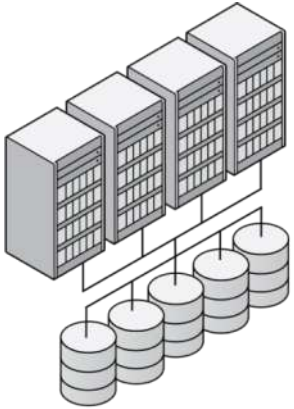
Oracle Database 11g

Virtual Machines



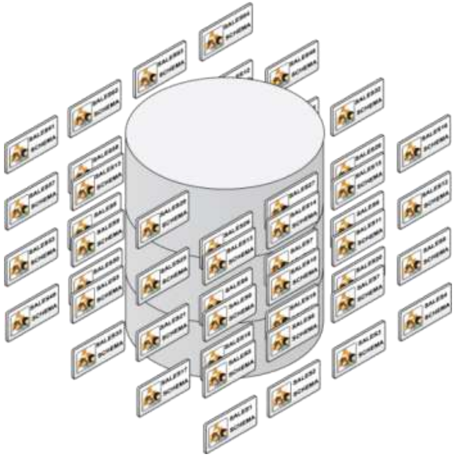
share servers

Dedicated Databases



share servers and OS

Schema Consolidation



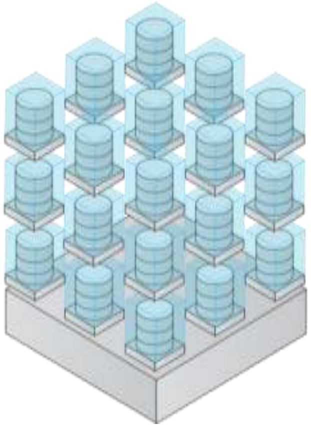
share servers, OS and database

Increasing Consolidation

Private Cloud Database Architectures

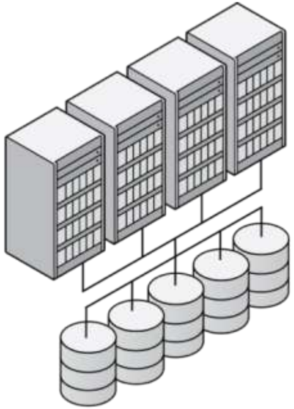
Oracle Database 12*c*

Virtual Machines



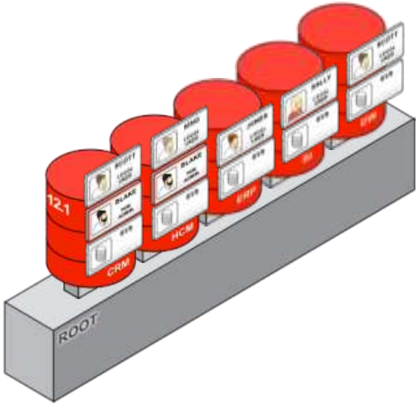
share servers

Dedicated Databases



share servers and OS

Multitenant Database

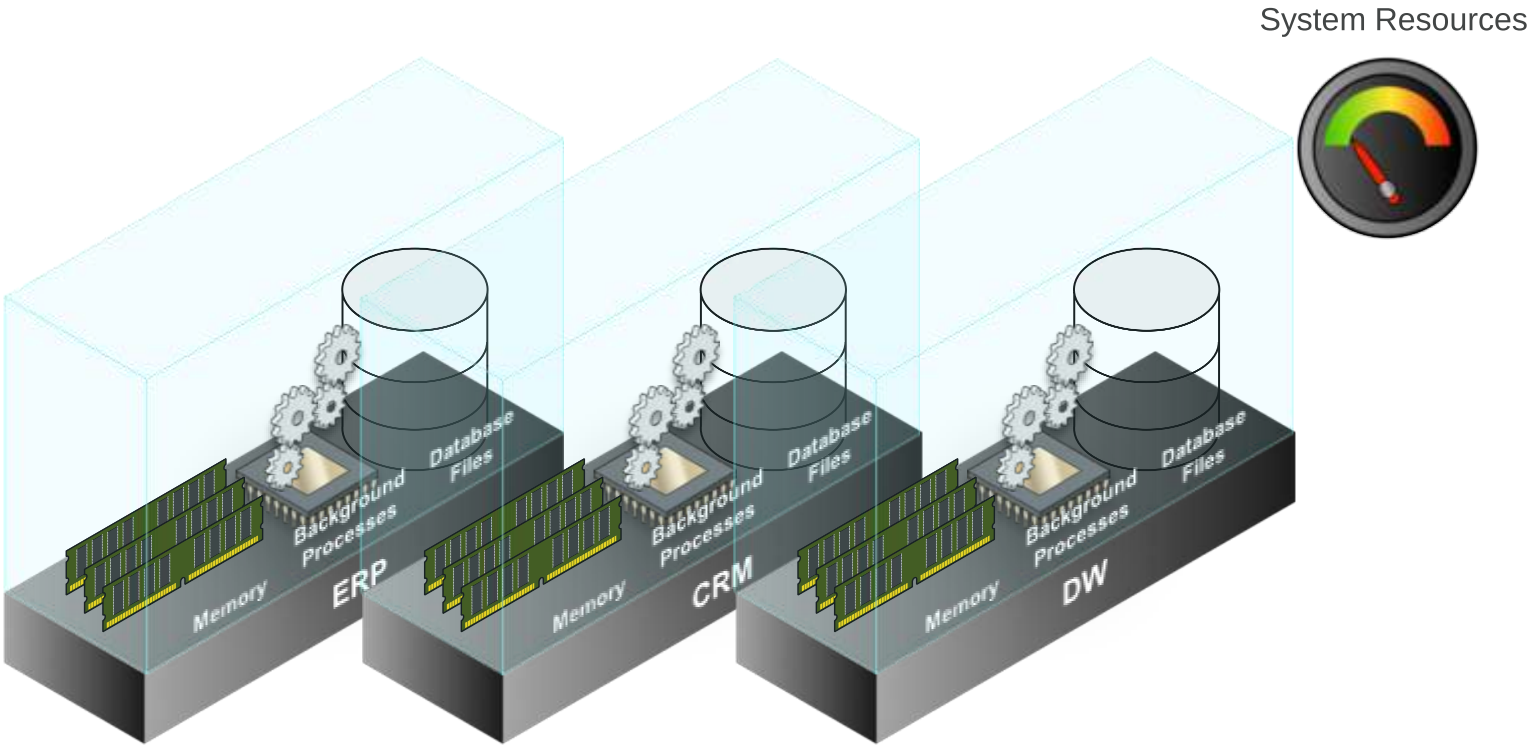


share servers, OS and database

Increasing Consolidation

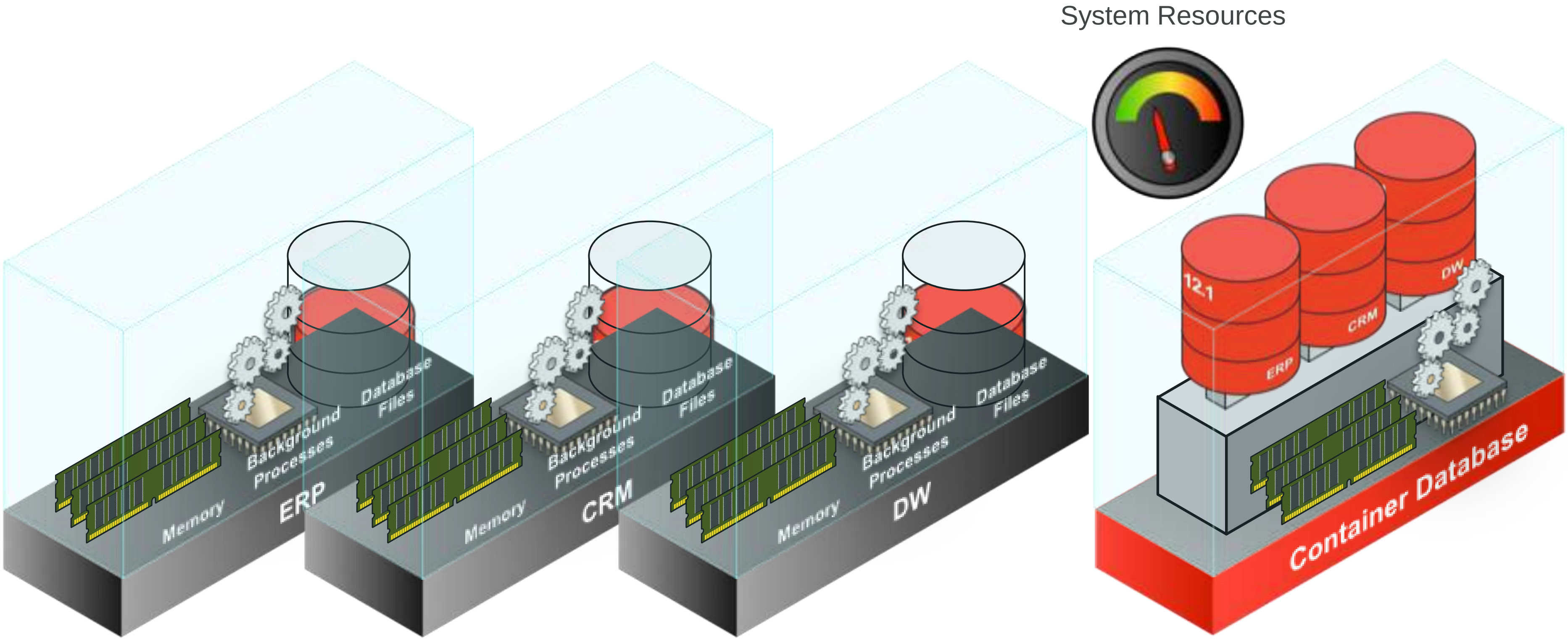
Oracle Database Architecture

Requires memory, processes and database files



New Multitenant Architecture

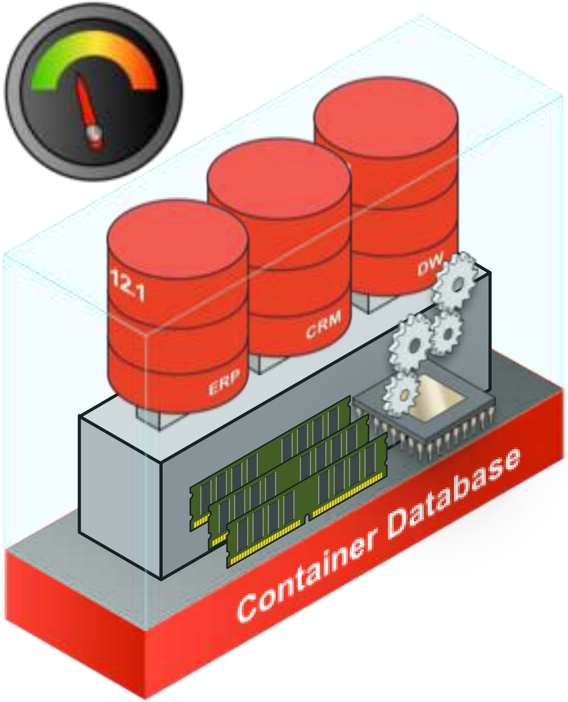
Memory and processes required at multitenant container level only



New Multitenant Architecture

Memory and processes required at multitenant container level only

System Resources



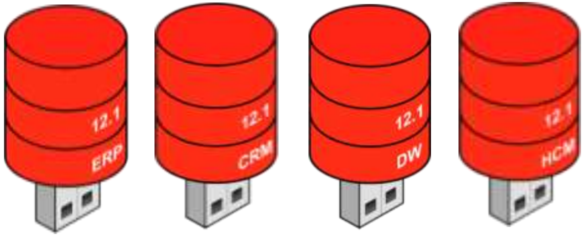
Agenda

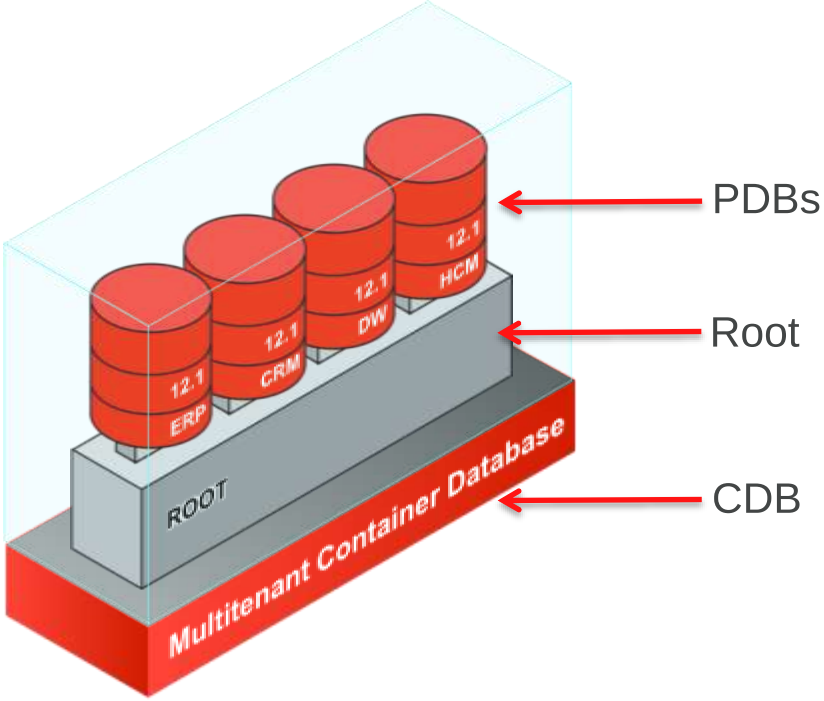
- Rethinking Database Architecture for the Cloud
- **Multitenant Architecture**
- Capabilities Enabled
- Managing Shared Resources
- Upgrading to Multitenant
- Use Cases

Multitenant Architecture

Components of a Multitenant Container Database (CDB)

Pluggable Databases (PDBs)



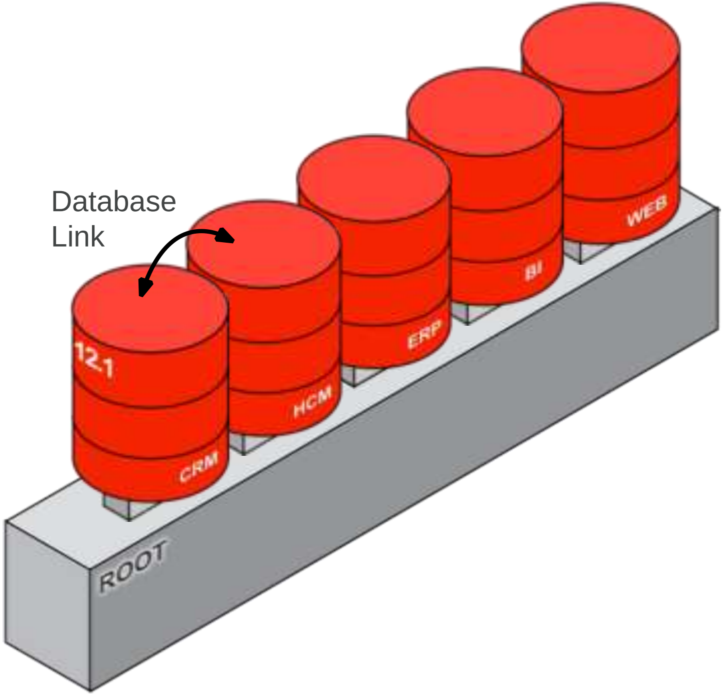


PDBs

Root

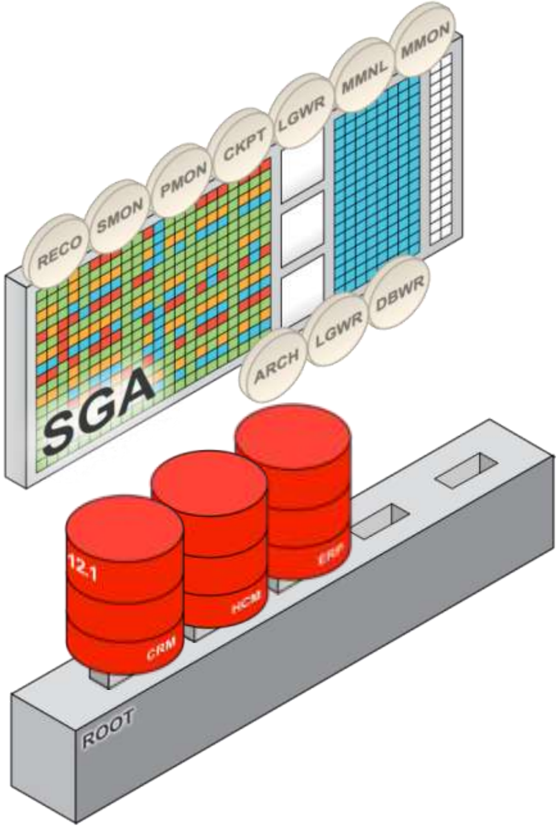
CDB

Multitenant Architecture



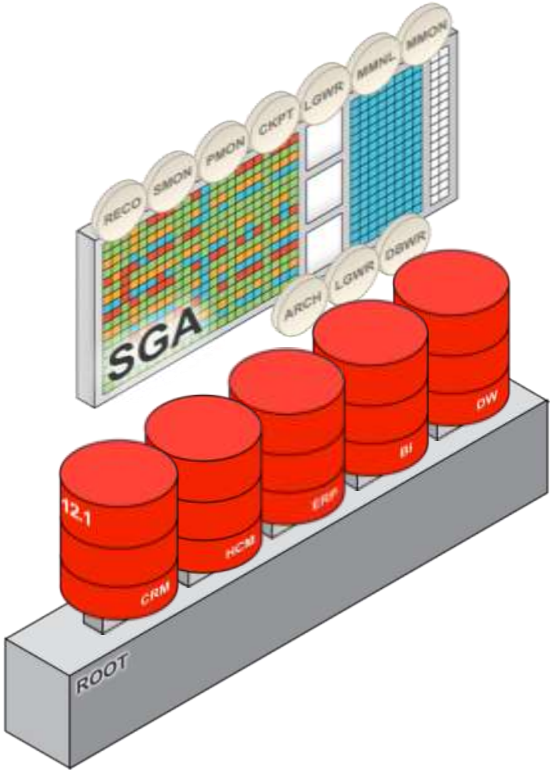
- Multitenant architecture can currently support up to 252 PDBs
- A PDB feels and operates identically to a non-CDB
- You cannot tell, from the viewpoint of a connected client, if you're using a PDB or a non-CDB

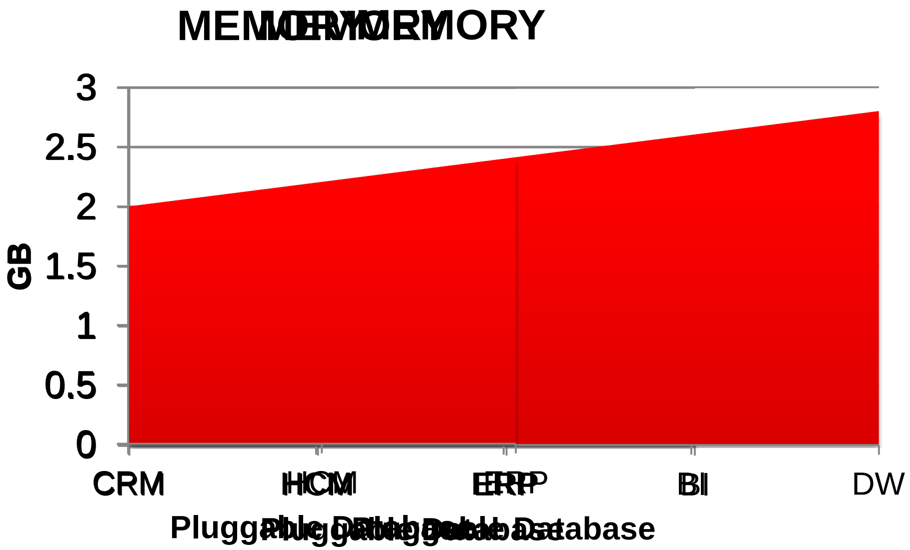
Multitenant Architecture – Dynamics



- PDBs share common SGA and background processes
- Foreground sessions see only the PDB they connect to

Multitenant Scalability

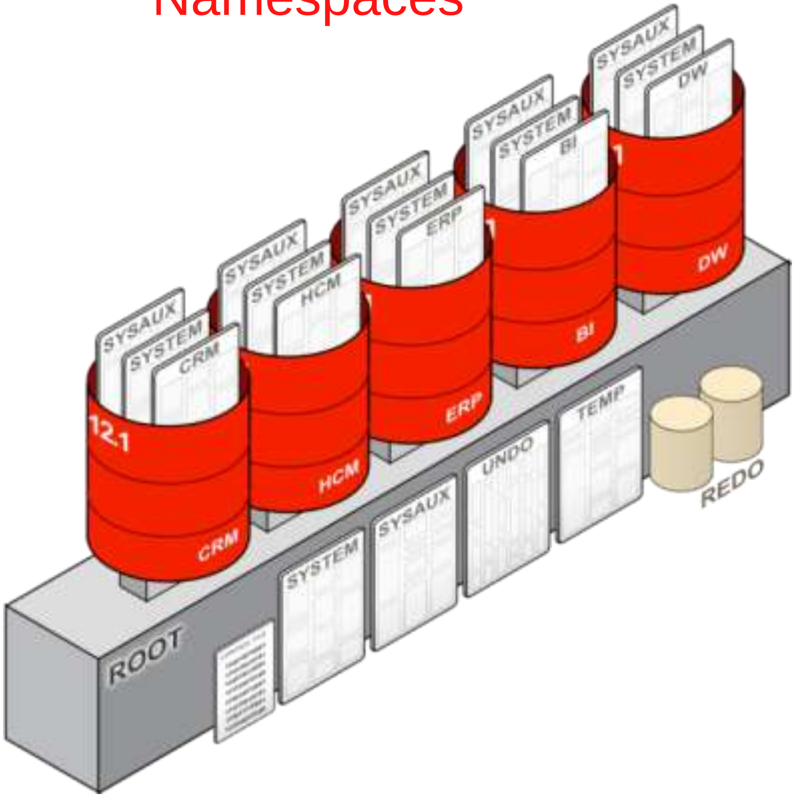




- Only small increments in memory as additional PDBs are added

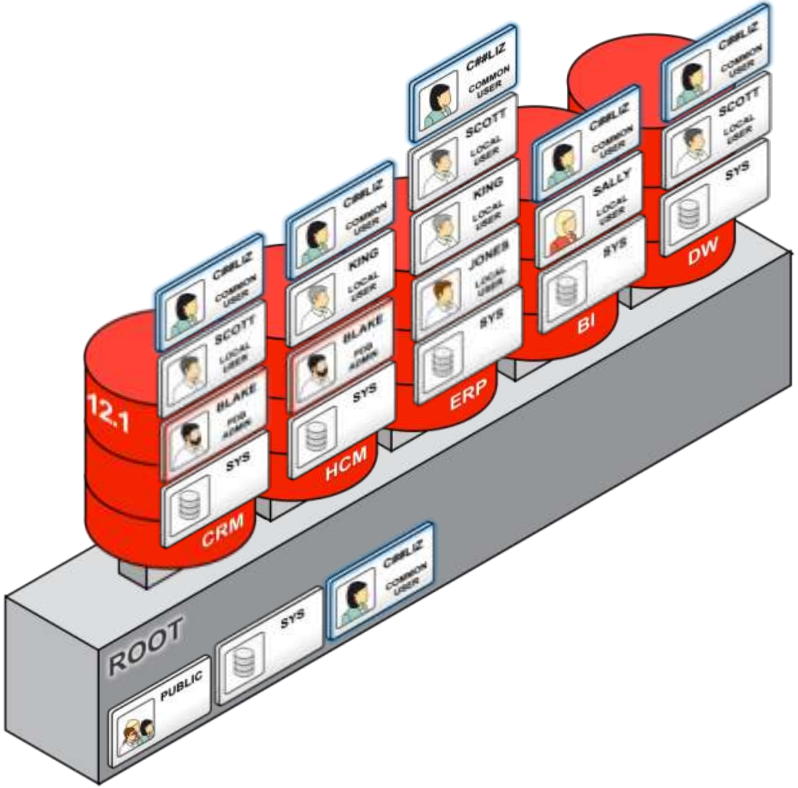
Files in the CDB

Namespaces



- Each PDB has its own set of tablespaces including SYSTEM and SYSAUX
- PDBs share UNDO, REDO and control files, (s)pfile
- By default the CDB has a single TEMP tablespace but PDBs may create their own

Users



- Local users are the successors for customer-created users in a non-CDB
- A local user is defined only in a PDB
- A local user can administer a PDB
- A common user is defined in the root and is represented in every PDB
- A common user can log into any PDB where it has “Create Session” and can therefore administer a PDB
- The Oracle system is owned by common users

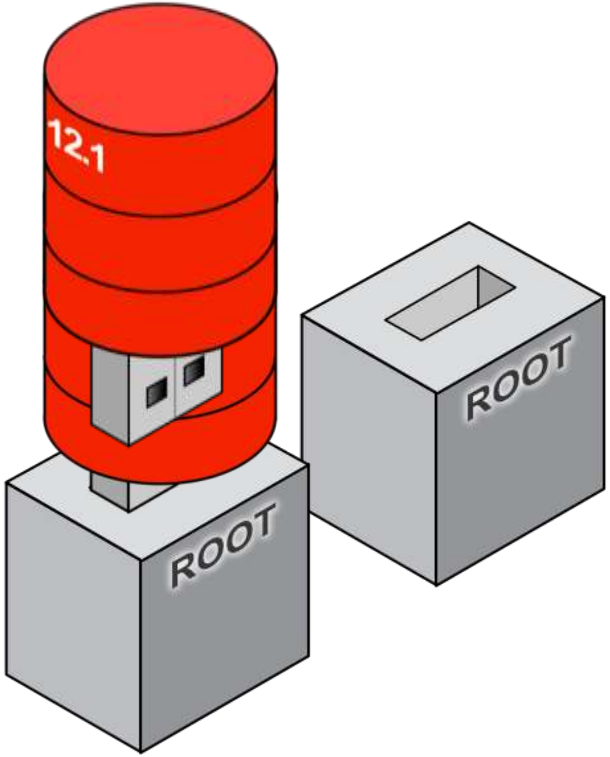
Common Users and Privileges

Authorization is checked in the same way as as pre-12.1

- A common user can be granted privileges locally in a PDB (or root) and therefore differently in each container
- A common user can, alternatively, be granted a system privilege *commonly* – the grant is made in root and every PDB, present and future
- You can create a common role
- A common role can be granted to a common user commonly
- Authorization is checked in the container where the SQL is attempted considering *only* the privileges that the user has in that container

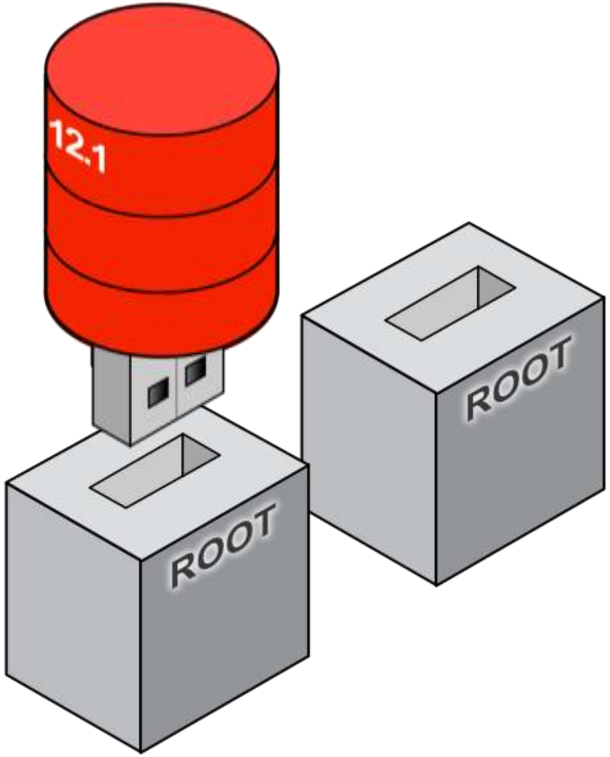
Unplug / plug

Simply unplug from the old CDB…



Unplug / plug

…and plug in to the new CDB…



- Moving between CDBs is a simple case of moving a PDB’s metadata
- Upgrading and patching become much simpler
- An unplugged PDB carries with it lineage, opatch, encryption key info etc

Unplug / plug

Example

Unplug

```
alter pluggable database HCM
unplug into '/u01/app/oracle/oradata/…/hcm.xml'
```

Plug

```
create pluggable database My_PDB
using '/u01/app/oracle/oradata/…/hcm.xml'
```

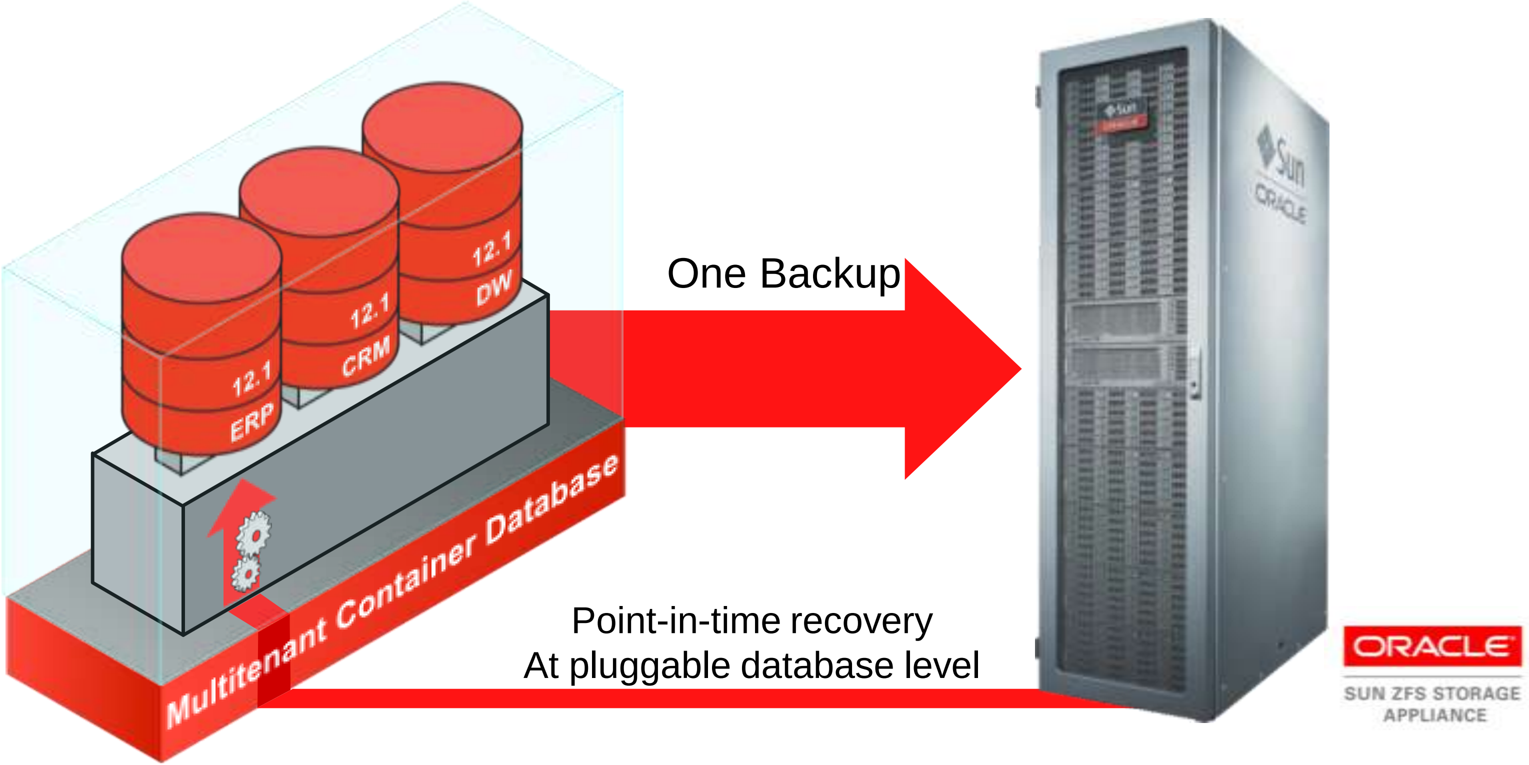
Agenda

- Rethinking Database Architecture for the Cloud
- Multitenant Architecture
- **Capabilities Enabled**
- Managing Shared Resources
- Upgrading to Multitenant
- Use Cases

ORACLE

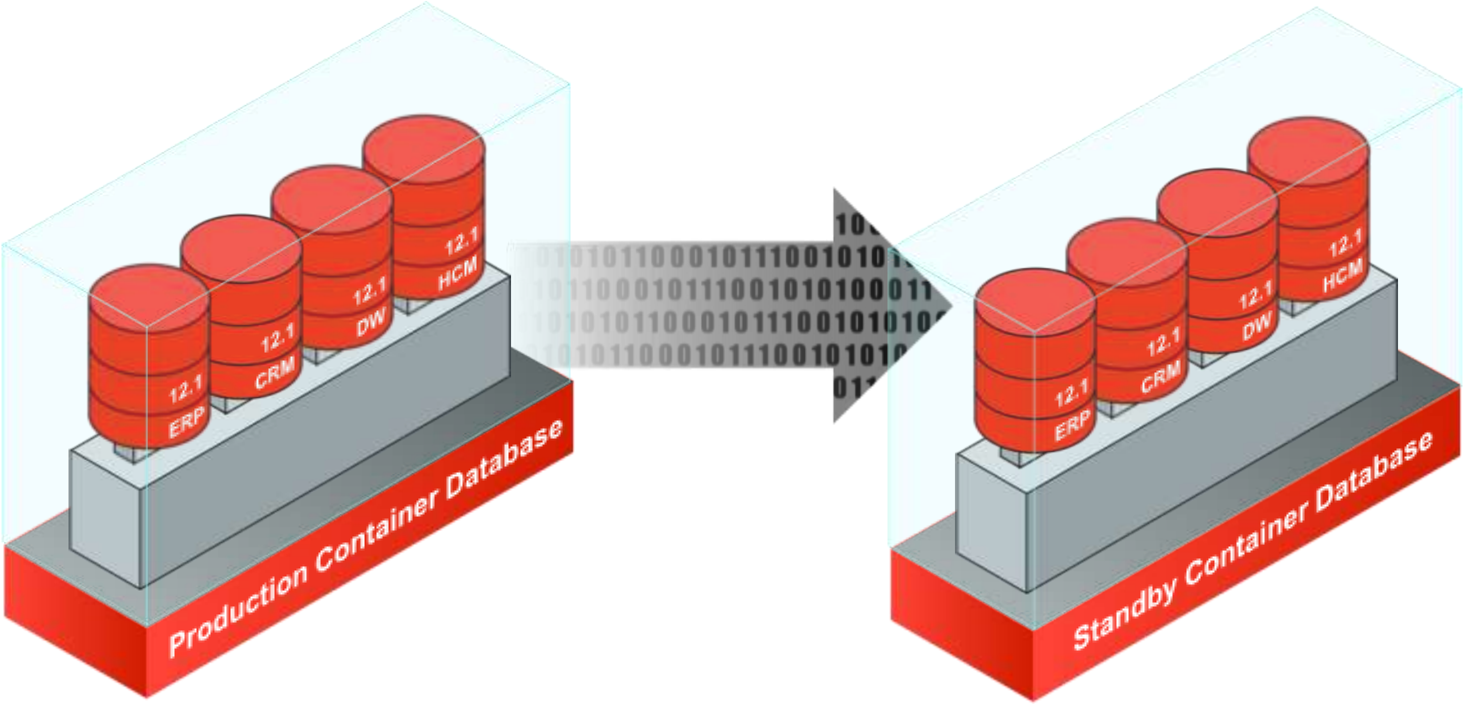
Manage Many as One with Multitenant

Backup databases as one; recover at pluggable database level


12.1
ERP
12.1
CRM
12.1
DW
Multitenant Container Database
One Backup
Point-in-time recovery
At pluggable database level
ORACLE
SUN ZFS STORAGE APPLIANCE

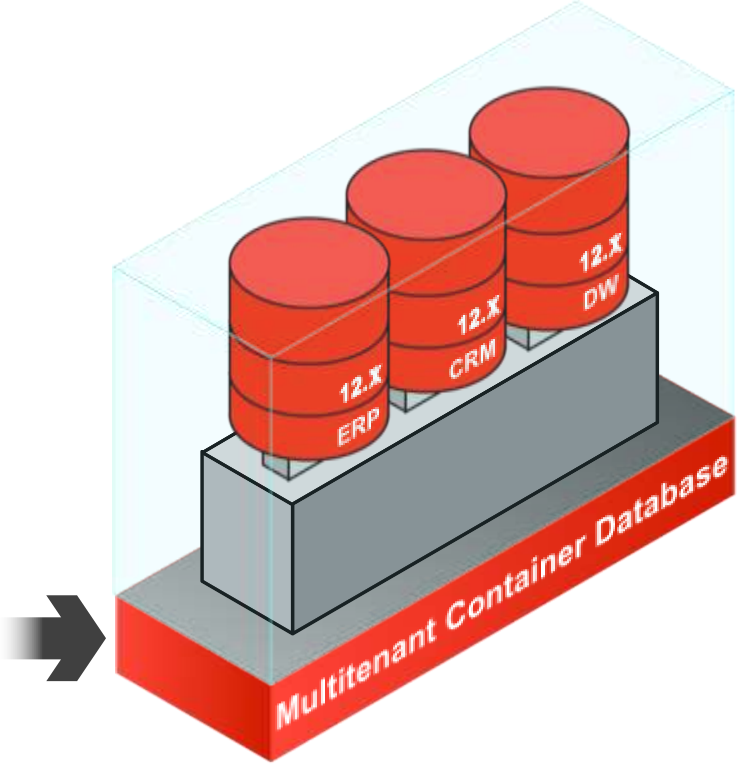
Manage Many as One with Multitenant

One standby database covers all pluggable databases



Multitenant for Simplified Patching

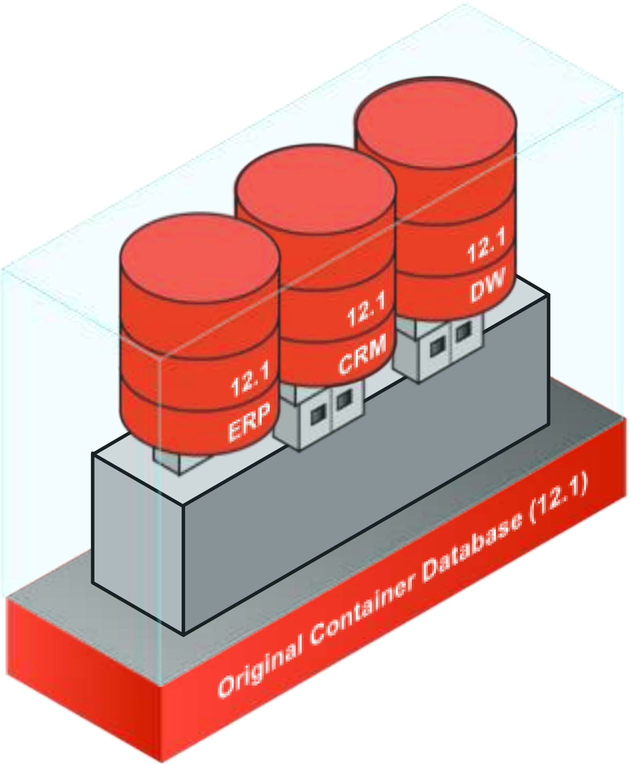
Apply changes once, all pluggable databases updated

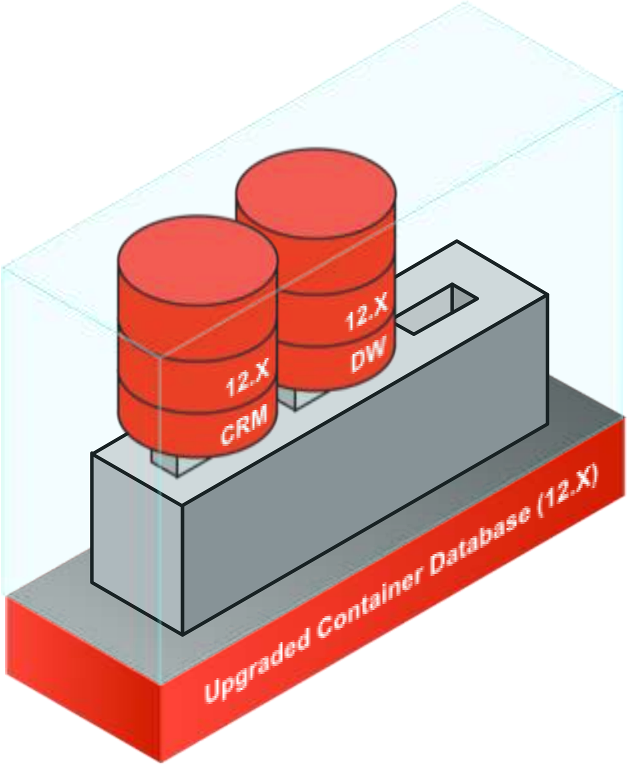


Upgrade in-place

Multitenant for Upgrades

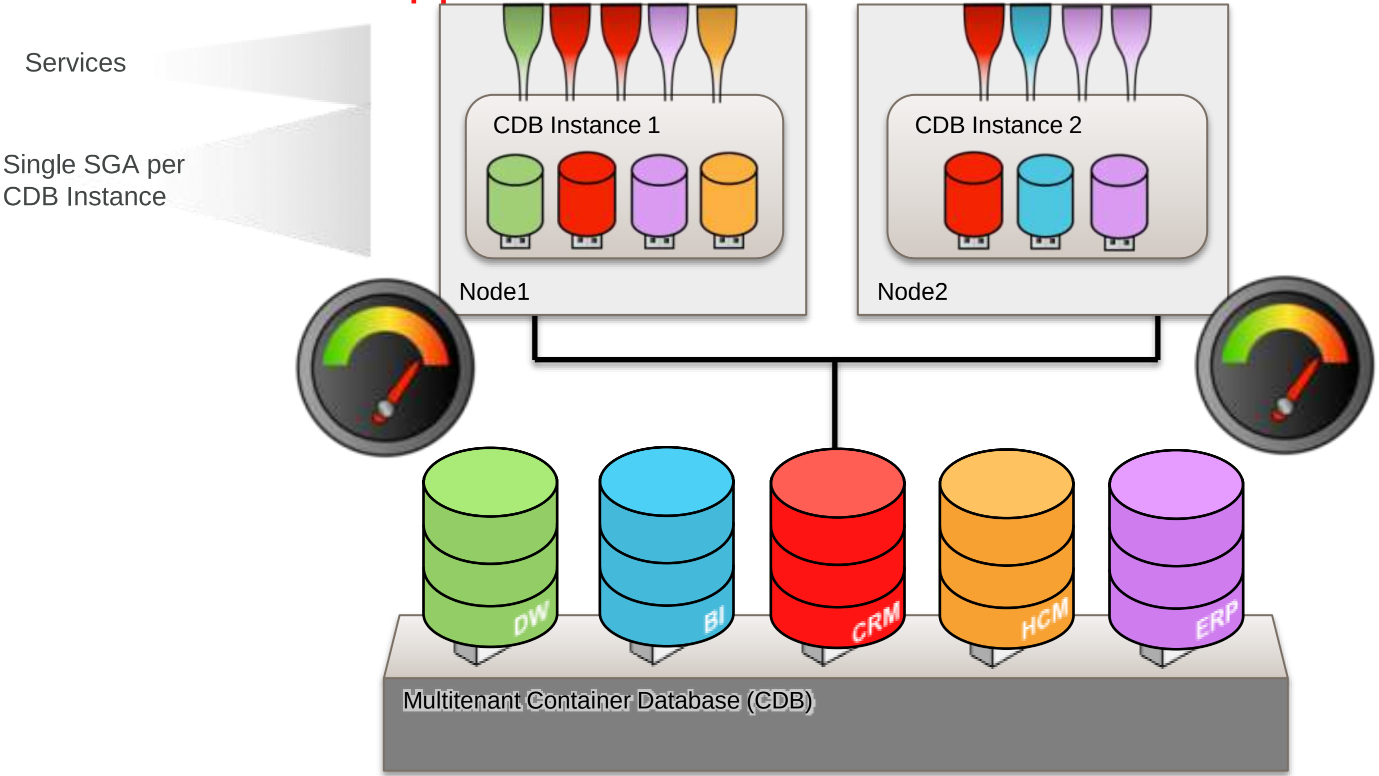
Flexible choice when patching & upgrading databases





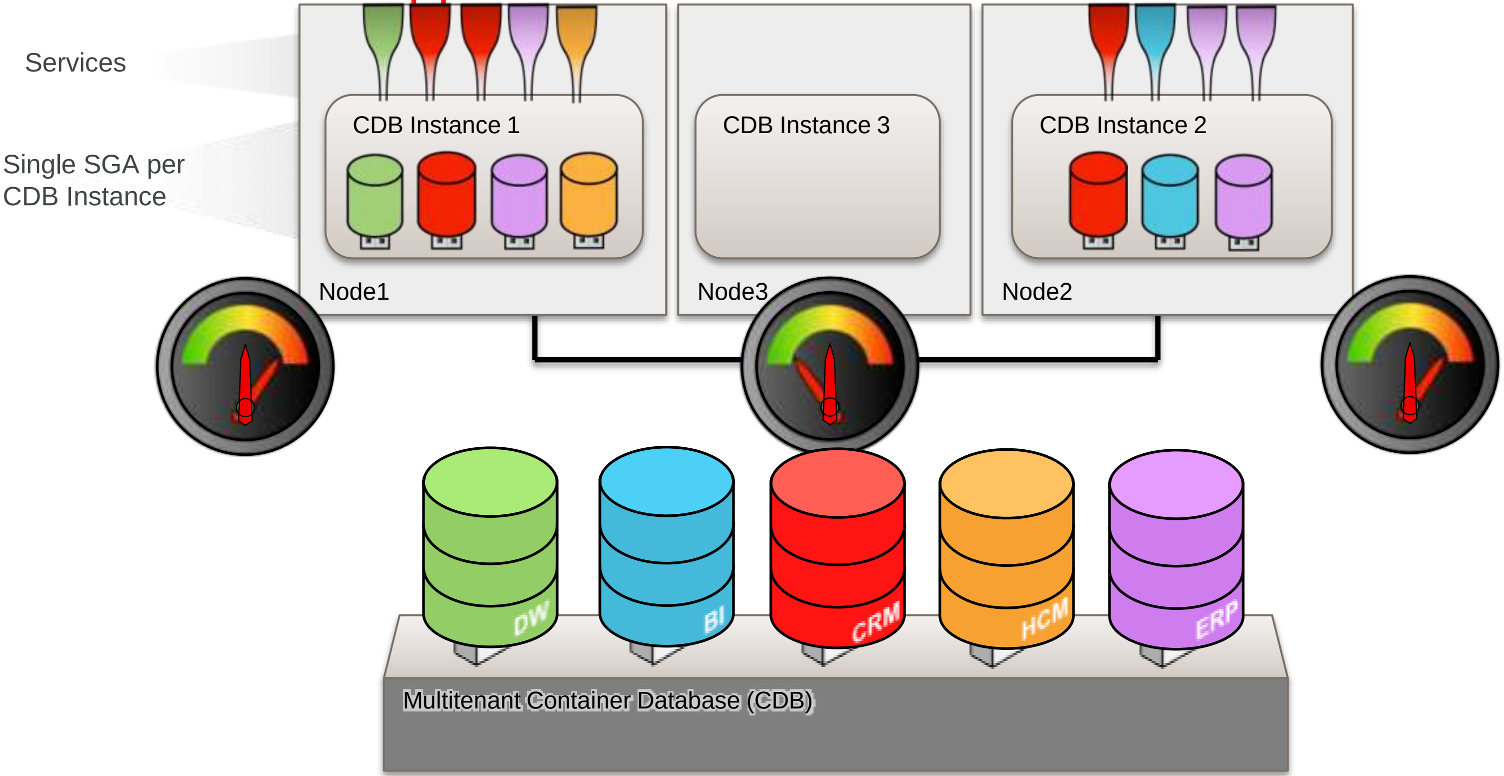
Improved Agility With Changing Workloads

Expand Cluster to Support Flexible Consolidation Model


Services
Single SGA per CDB Instance
CDB Instance 1
CDB Instance 2
Node1
Node2
DW
BI
CRM
HCM
ERP
Multitenant Container Database (CDB)

Improved Agility With Changing Workloads

Expand Cluster to Support Flexible Consolidation Model



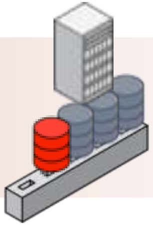
Unprecedented Agility with Pluggable Portability

PDB migrates through SLAs as it becomes more mission critical

GOLD RAC, Data Guard, Daily Incrementals

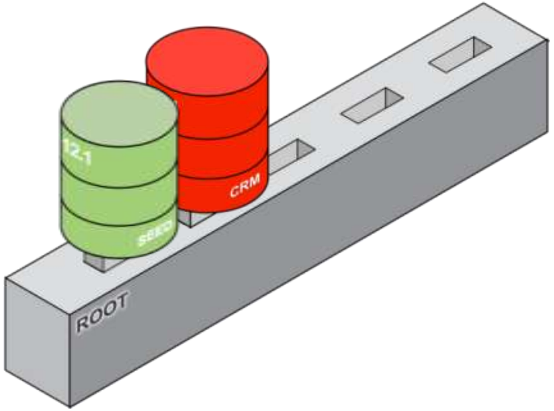
SILVER Data Guard, Daily Incrementals

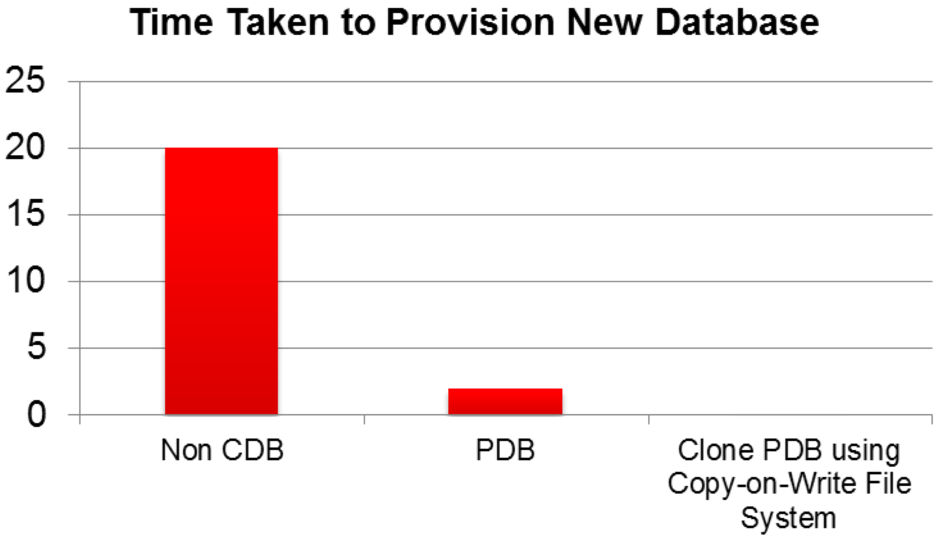
BRONZE Weekly Full Backups



Multitenant for Fast Provisioning

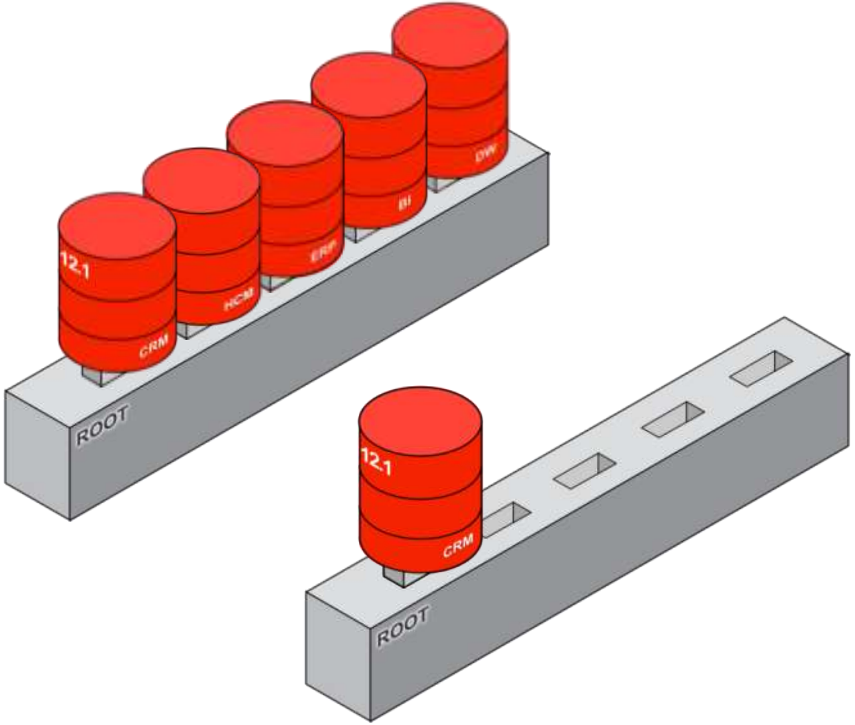
Pluggable databases can be quickly provisioned from seed





Multitenant for Provisioning

Fast cloning of PDBs



- PDBs can be cloned from within the same CDB
- PDBs can be cloned from remote CDBs

Cloning a PDB

Example

Local

```
create pluggable database HCMBI from HCM
```

Remote (DB Link)

```
create pluggable database HCMBI from HCM@us.acme.db1
```

Per PDB *vs* per CDB

Common operations on CDB with granular control where appropriate

Per CDB

- Single Oracle Software Version
- Data Guard
- Scheduled RMAN Backups
- Some parameters/properties e.g. homogeneous character set
- Redo and Undo

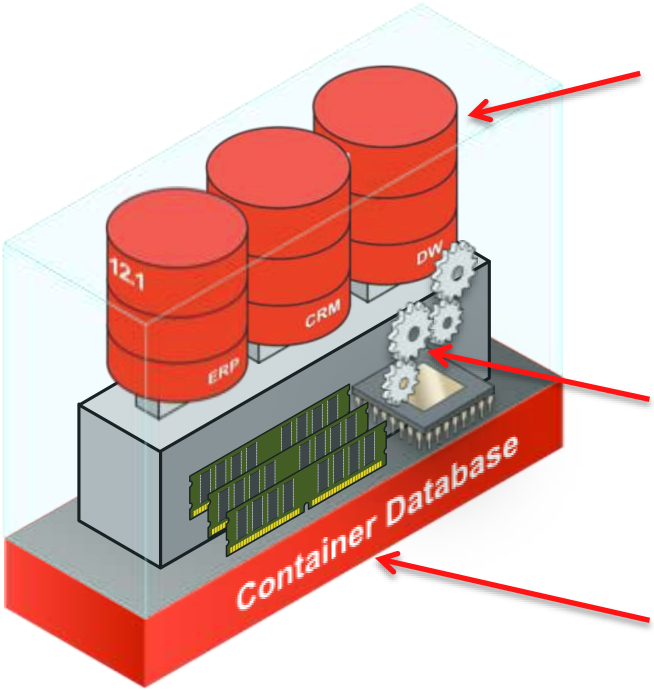
Per PDB

- RMAN point-in-time recovery
- Ad hoc RMAN backups
- Flush shared pool
- Parameters where

  `IsPDB_Modifiable = 'TRUE'`

Advantages of Multitenant Architecture

Reduced CapEx & OpEx, Increased Agility, Easy Adoption



Self-contained PDB for each application

- Applications run unchanged
- Rapid provisioning (via clones)
- Portability (via pluggability)

Shared memory and background processes

- More applications per server

Common operations performed at CDB level

- Manage many as one (upgrade, HA, backup)
- Granular control when appropriate

Key Benefits

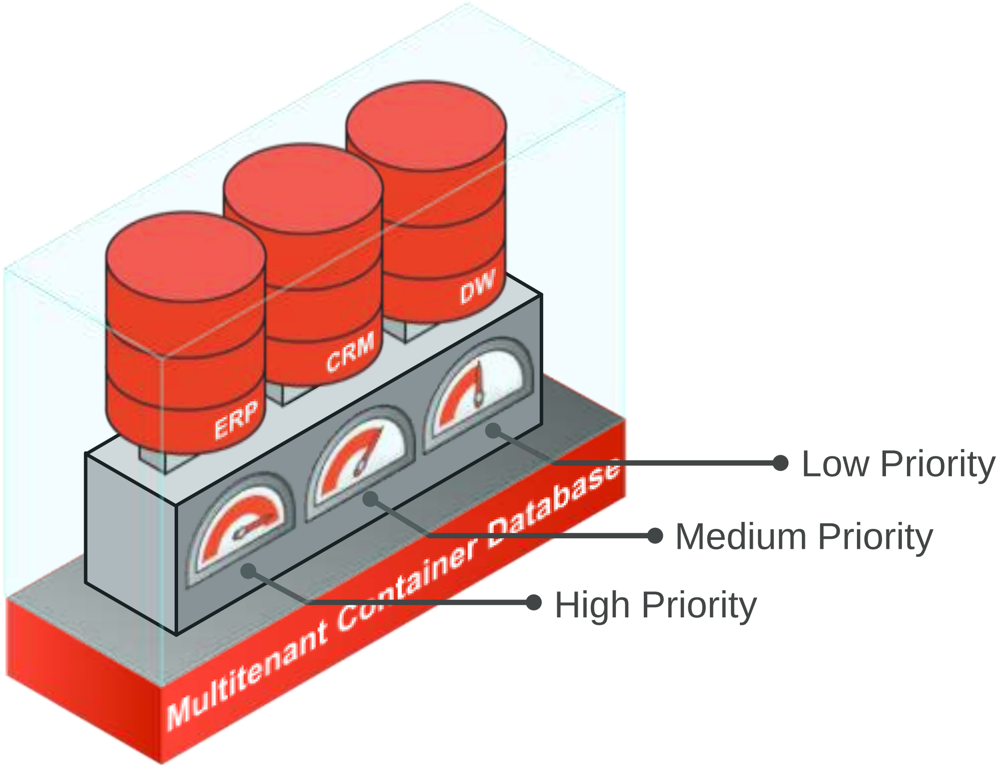
Benefit	Capability Enabled
Minimize CapEx	• More applications per server
Minimize OpEx	• Manage many as one • Standardized procedures & service levels • Rapid provisioning
Maximize Agility	• Cloning for development / testing • Portability through “pluggability” • Scalability with RAC
Ease of Adoption	• Applications run unchanged

Agenda

- Rethinking Database Architecture for the Cloud
- Multitenant Architecture
- Capabilities Enabled
- **Managing Shared Resources**
- Upgrading to Multitenant
- Use Cases

Managing Shared Resources

Resource management in multitenant environment



Low Priority

Medium Priority

High Priority

Managing Resources between PDBs

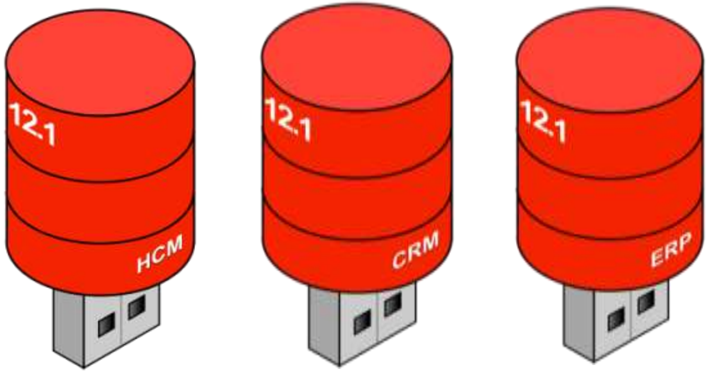
- PDBs vie for shared resources
- Using Resource Manager, you can control
  - CPU
  - Exadata I/O
  - Sessions
  - Parallel execution servers
- Configure a policy that controls how resources are utilized
  - Default configuration that works, even as PDBs are added or removed
  - Hard limits, for “get what you pay for”

Managing Resources between PDBs

- The model is “industry standard” based on two notions:
  - A number of shares is allocated to each PDB
  - A “cap” (a.k.a. maximum utilization limit) may be applied to each PDB

Manage CPU

A CDB Resource Plan uses *shares* to specify how CPU is distributed between PDBs



2 Shares 1 Share 1 Share

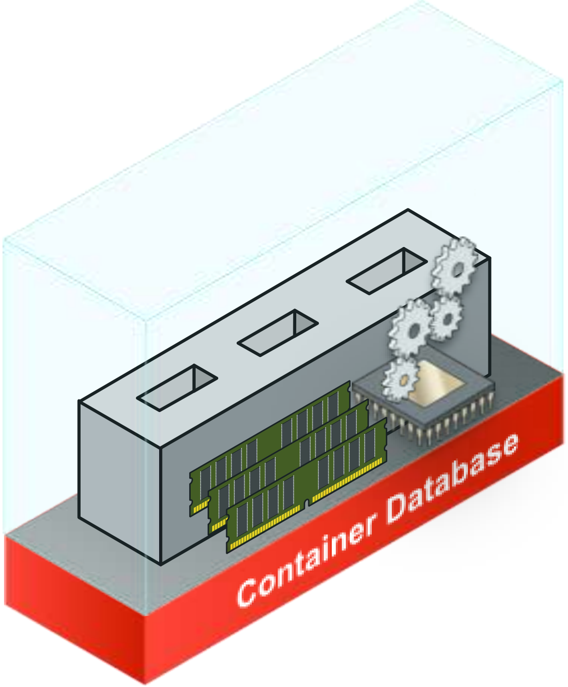
Pluggable Database	Shares	Guaranteed CPU	Maximum CPU
HCM	2	2/4 = 50%	100%
CRM	1	1/4 = 25%	100%
ERP	1	1/4 = 25%	100%

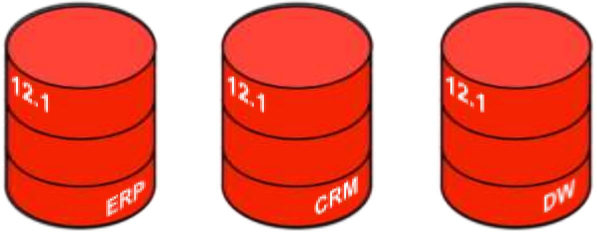
Agenda

- Rethinking Database Architecture for the Cloud
- Multitenant Architecture
- Capabilities Enabled
- Managing Shared Resources
- **Upgrading to Multitenant**
- Use Cases

Upgrading to Multitenant

Step 1: Upgrade databases in-place

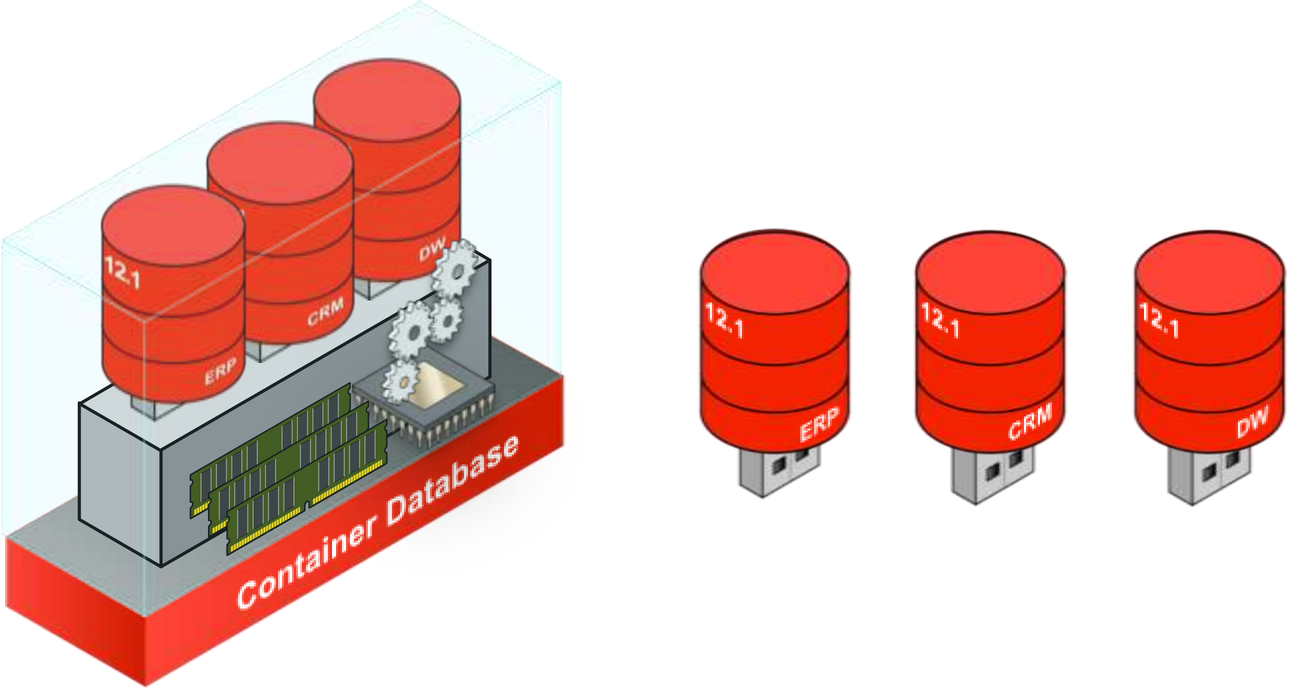




Upgrade in Place

Upgrading to Multitenant

Step 2: Plug-in upgraded databases



Upgrading to Multitenant

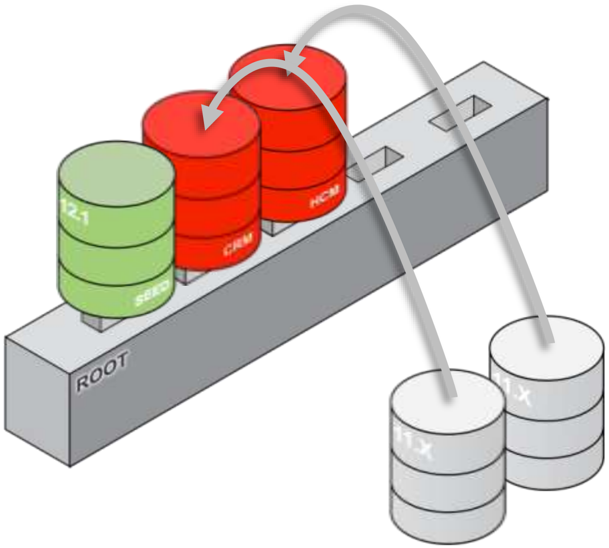
Step 3. Change applications to work with Multitenant

Upgrading to Multitenant

~~Step 3. Change applications to work with Multitenant~~

- No application changes required.

Migrate using Replication



① Provision new PDB from Seed

② Replicate using technologies such as Oracle GoldenGate or Data Pump

New in 12.1, you ask that full database export and full database import make maximum use of transportable tablespaces in the single *expdb* and *impdb* commands.

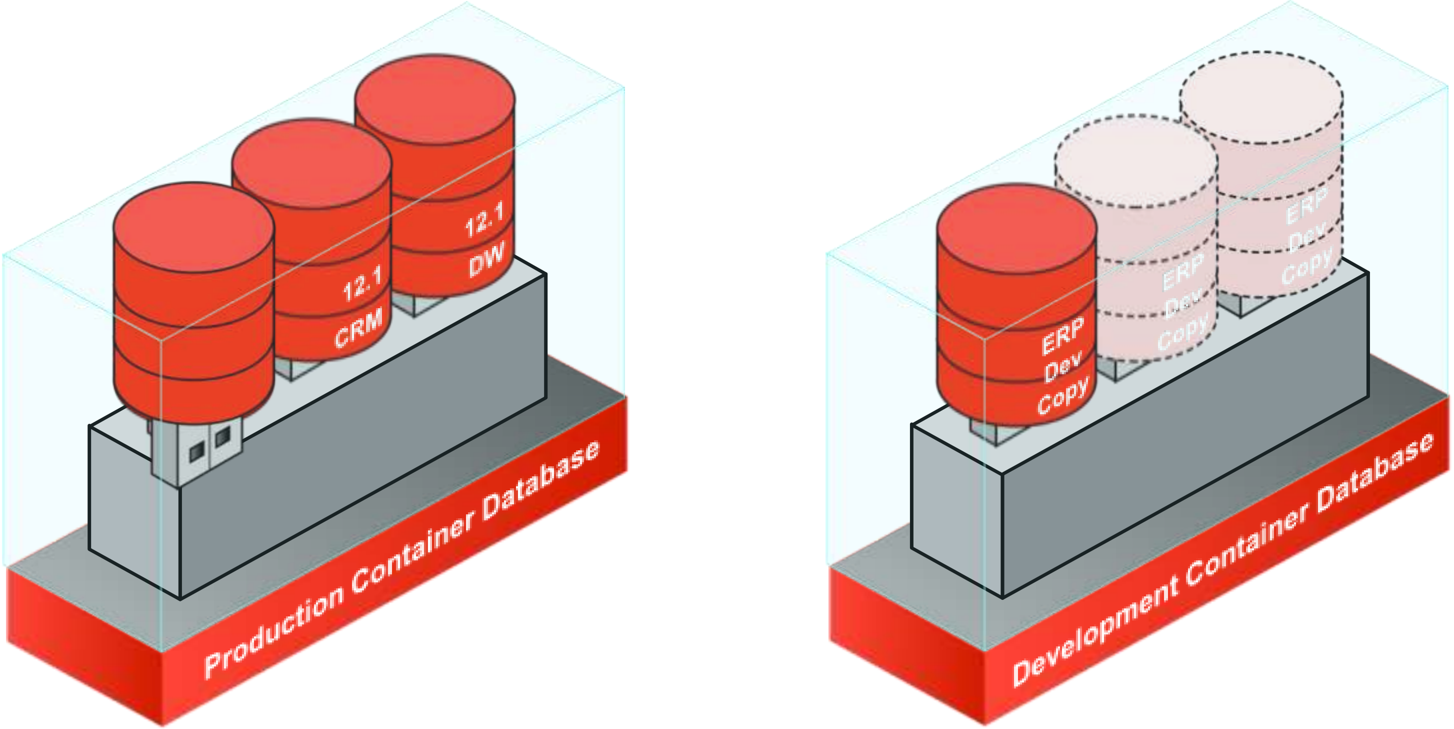
(Backported to 11.2.0.3.)

Agenda

- Rethinking Database Architecture for the Cloud
- Multitenant Architecture
- Capabilities Enabled
- Managing Shared Resources
- Upgrading to Multitenant
- **Use Cases**

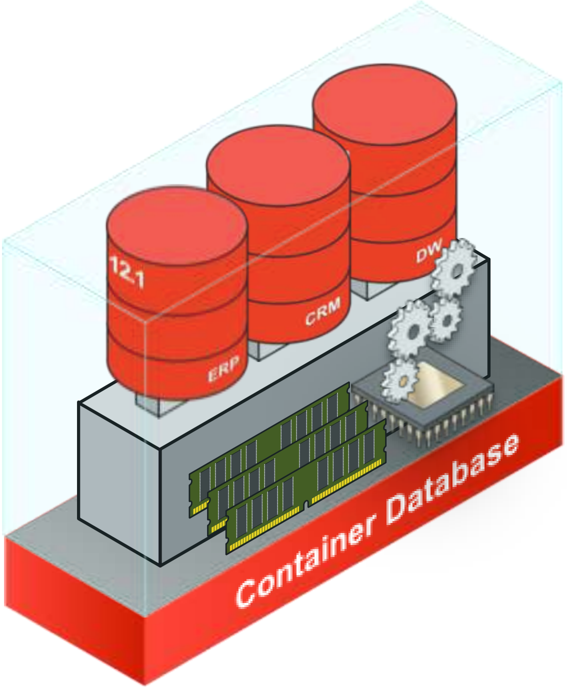
1. Multitenant for Test and Development

Fast, flexible copy and snapshot of pluggable databases



2. Consolidation of Disparate Applications

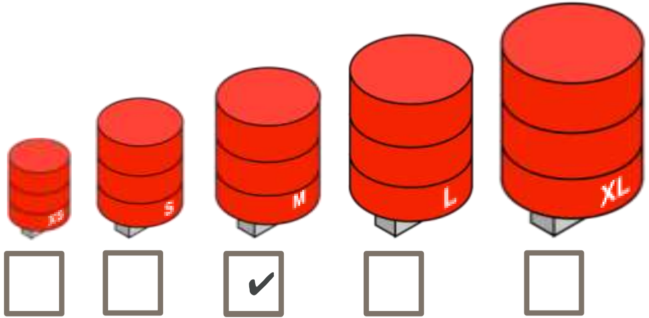
Shared overhead of memory and processes



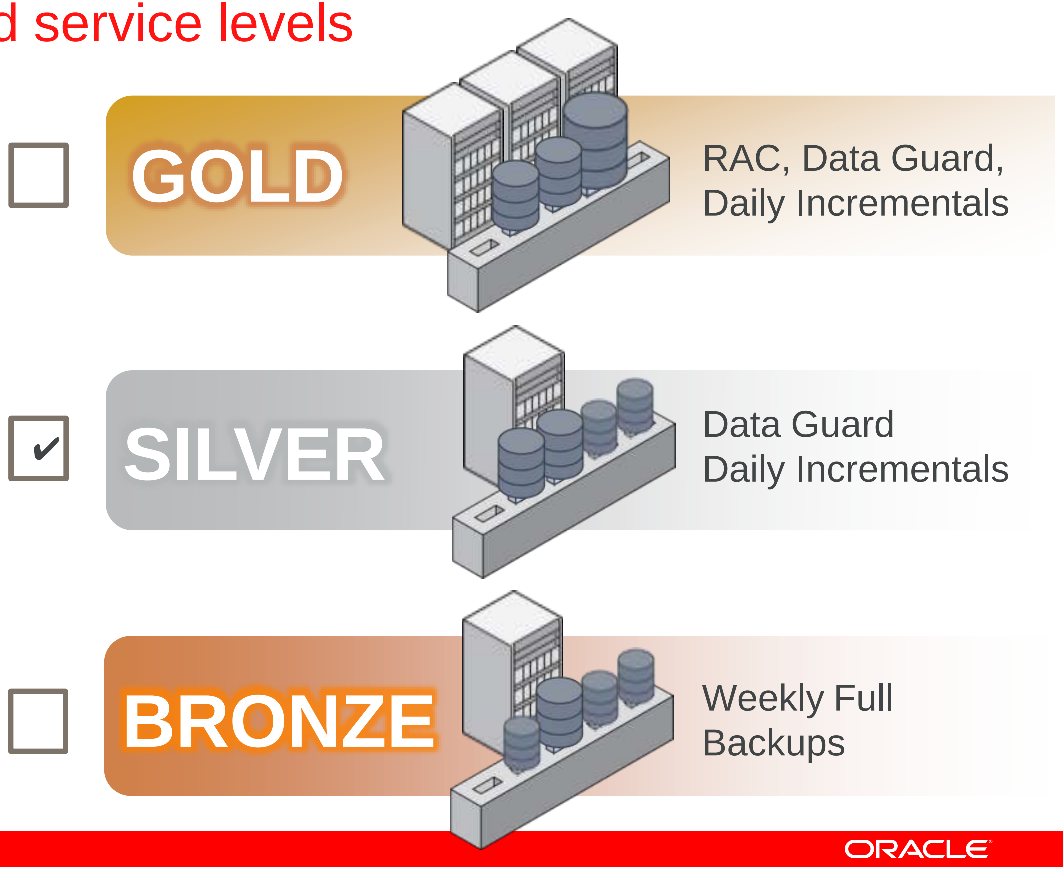
System Resources

3. Self-Service Database as a Service (DBaaS)

Pick from standard sizes and service levels



□ □ ☑ □ □



□ GOLD

RAC, Data Guard,
Daily Incrementals

☑ SILVER

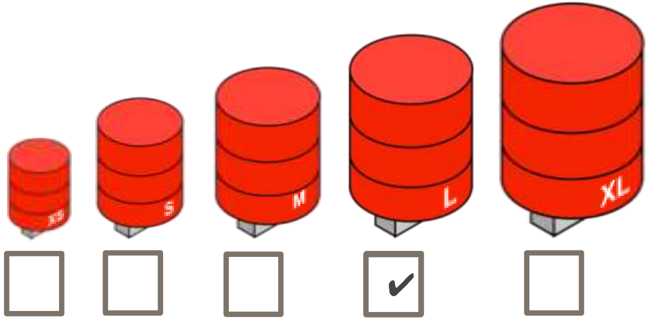
Data Guard
Daily Incrementals

□ BRONZE

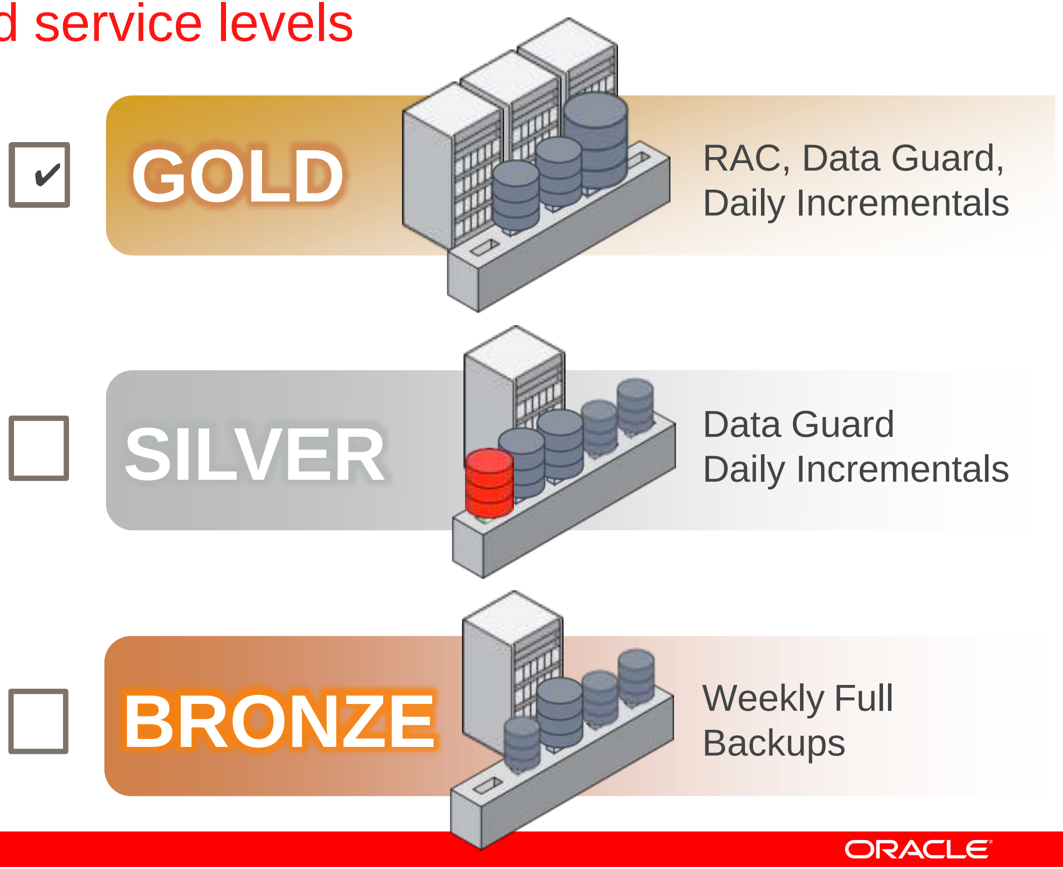
Weekly Full
Backups

3. Self-Service Database as a Service (DBaaS)

Pick from standard sizes and service levels



□ □ □ ☑ □



☑ GOLD

RAC, Data Guard,
Daily Incrementals

□ SILVER

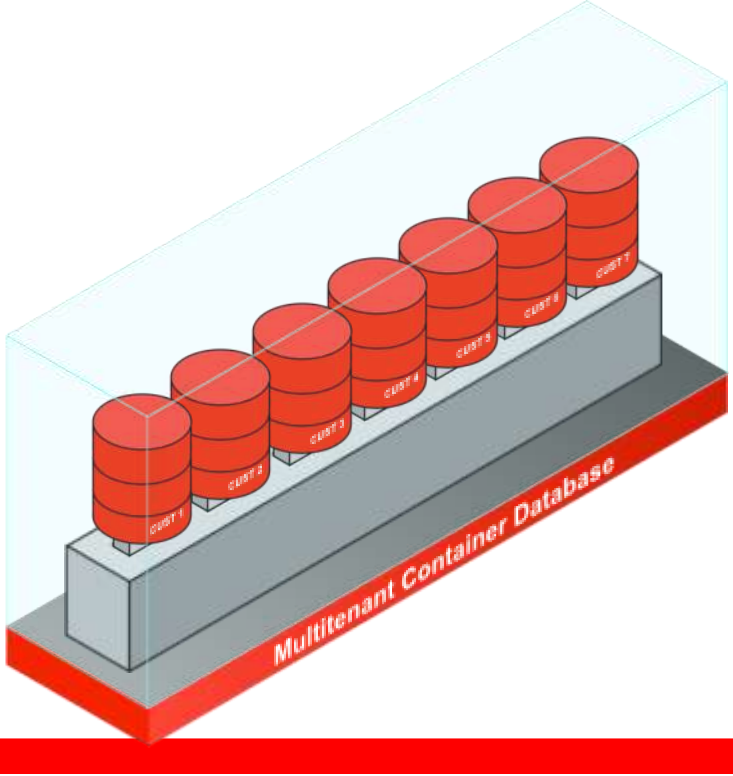
Data Guard
Daily Incrementals

□ BRONZE

Weekly Full
Backups

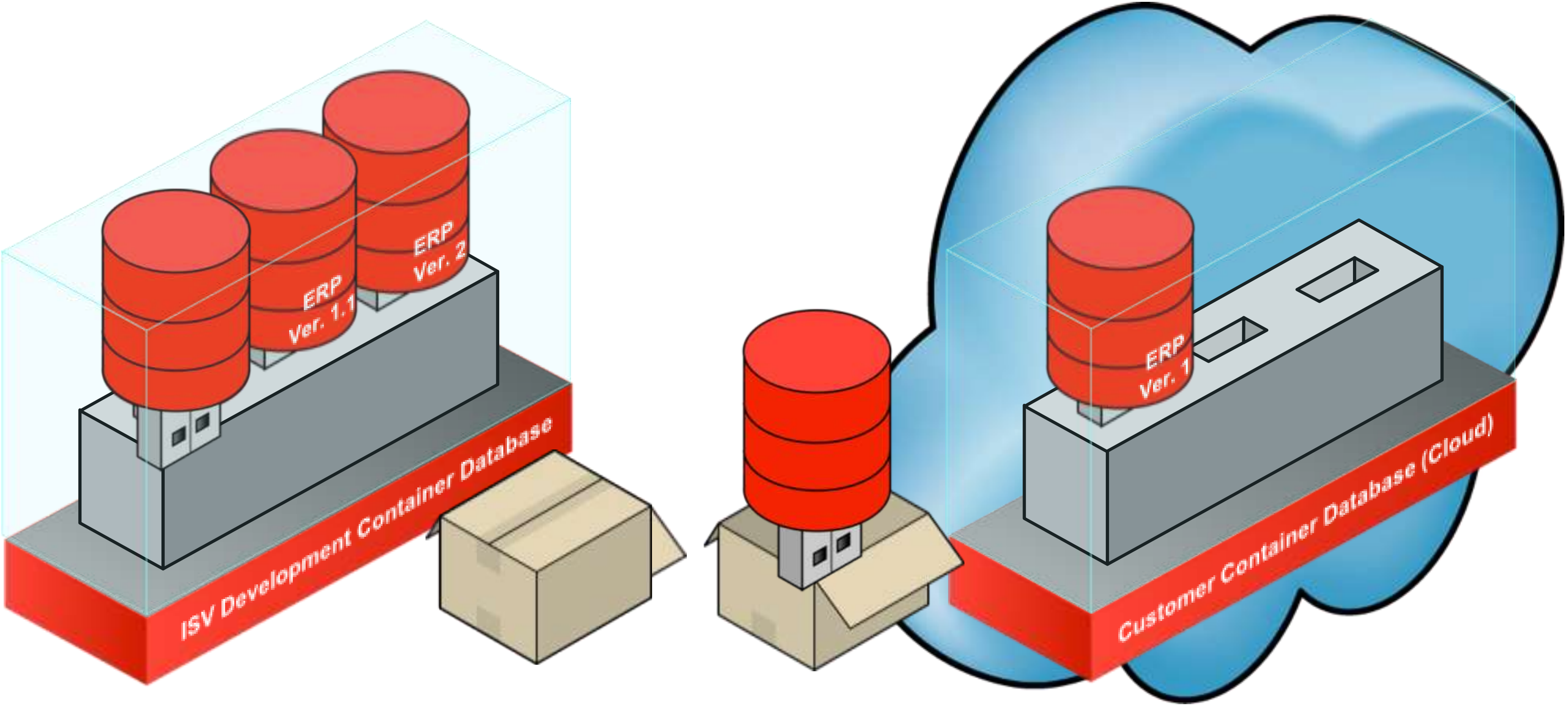
4. Multitenant. Perfect for SaaS.

Multitenancy implemented by the Database, not the Application



5. Multitenant. Perfect for ISVs.

Packaged apps and reference data are easily distributed


ERP
Ver. 1.1
ERP
Ver. 2
ISV Development Container Database
ERP
Ver. 1
Customer Container Database (Cloud)

Use Cases

1. Development / Testing – *start here!*
2. Consolidation of Disparate Applications
3. Database as a Service (DBaaS) on Private Cloud
4. Software as a Service (SaaS)
5. ISVs: Distribution of Packaged Apps and Data

Oracle Multitenant

What customers are saying…

“Oracle Multitenant is a step forward in that it gives us more control over our schemas and applications; to be able to isolate them, give definitive statements on how performance is working and be able to manage more databases better.” **Carfax**

“With Oracle Database 12*c*, we can now copy an entire database from one instance to another using full transportable export/import. This minimizes the downtime when migrating clients’ databases.” **Accenture**

“Oracle Multitenant allows us to consolidate hundreds of databases onto a RAC environment that guarantees the separation that drove us to put them on separate servers previously.” **Logical Technology**

“Undoubtedly the number one most compelling feature of Oracle Database 12*c* is the support for consolidation. Oracle Multitenant can share memory resources, and make management easier because it is still a single database instance.” **Pythian**

Key Benefits

Benefit	Capability Enabled
Minimize CapEx	• More applications per server
Minimize OpEx	• Manage many as one • Standardized procedures & service levels • Rapid provisioning
Maximize Agility	• Cloning for development / testing • Portability through “pluggability” • Scalability with RAC
Ease of Adoption	• Applications run unchanged

Hardware and Software

ORACLE

Engineered to Work Together

ORACLE®