Upgrade to Oracle Database 19c and Ensure Performance Stability

Virtual Hands-on Workshop

Photo by Robert Bock on Unsplash

MIKE DIETRICH
Distinguished Product Manager
Database Upgrade and Migrations

mikedietrich

@mikedietrichde

https://mikedietrichde.com



DANIEL OVERBY HANSEN
Senior Principal Product Manager
Cloud Migrations

dohdatabase

@dohdatabase

https://dohdatabase.com



RODRIGO JORGE
Senior Principal Product Manager
Database Patching and Upgrade

rodrigoaraujorge

@rodrigojorgedba

https://dbarj.com.br/en

Welcome Message

Waleed Hosny

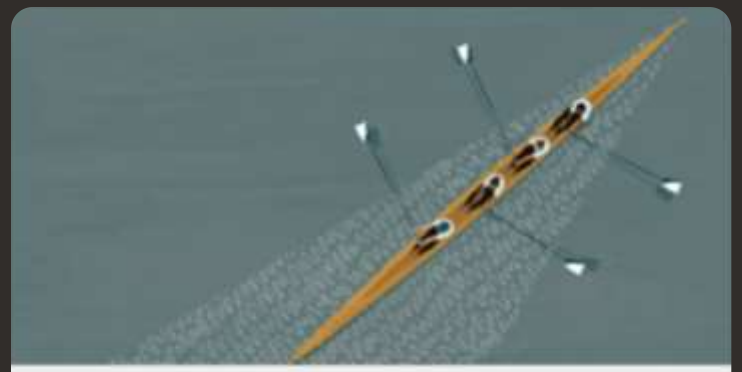
Presales Director

Fallback | Next Webinars

Migrating Very Large Databases

December 9, 2021 | 09:00 GMT / 10:00 CET / 11:00 EET / 13:00 GST

Duration: 120 mins



Data Pump Extreme - Deep Dive with Development

January 27, 2022 | 09:00 GMT / 10:00 CET / 11:00 EET / 13:00 GST

Duration: 120 mins

REGISTER

NEW Episode 1

Release and Patching Strategy

105 minutes – Feb 4, 2021

NEW Episode 2

AutoUpgrade to Oracle Database 19c

115 minutes – Feb 20, 2021

NEW Episode 3

Performance Stability, Tips and Tricks and Underscores

120 minutes – Mar 4, 2021

NEW Episode 4

Migration to Oracle Multitenant

120 minutes – Mar 16, 2021

NEW Seminar 5

Migration Strategies – Insights, Tips and Secrets

120 minutes – Mar 25, 2021

NEW Seminar 6

Move to the Cloud – Not only for techies

115 minutes – Apr 8, 2021

NEW Episode 7

Cool Features – Not only for DBAs

110 minutes – Jan 14, 2021

NEW Episode 8

Database Upgrade Internals – and so much more

Recorded Web Seminars

https://MikeDietrichDE.com/videos/

https://dohdatabase.com/webinars/

https://www.dbarj.com.br/webinars/

Agenda Day 1

- 13:00-13:15h Welcome
- 13:15-13:45h Release and Patching Strategy
- 13:45-14:45h Become an Upgrade/Performance expert in 60 minutes

- 14:45-15:00h Break

- 15:00-15:15h Checkout the environments
- 15:15-17:00h LAB DAY1:
  Generate load, upgrade, performance evaluation

Release and Patching Strategy

Photo by Reid Naaykens on Unsplash

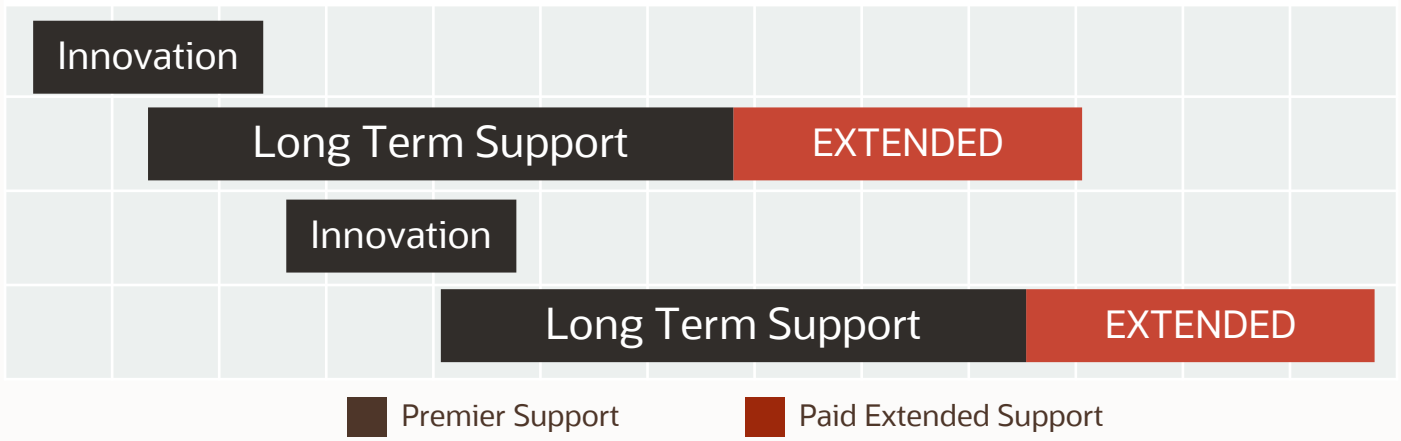
Release Types | Long Term Support vs Innovation Releases

Long Term Support Release

- 5 years of Premier Support followed by 3 years of Extended Support

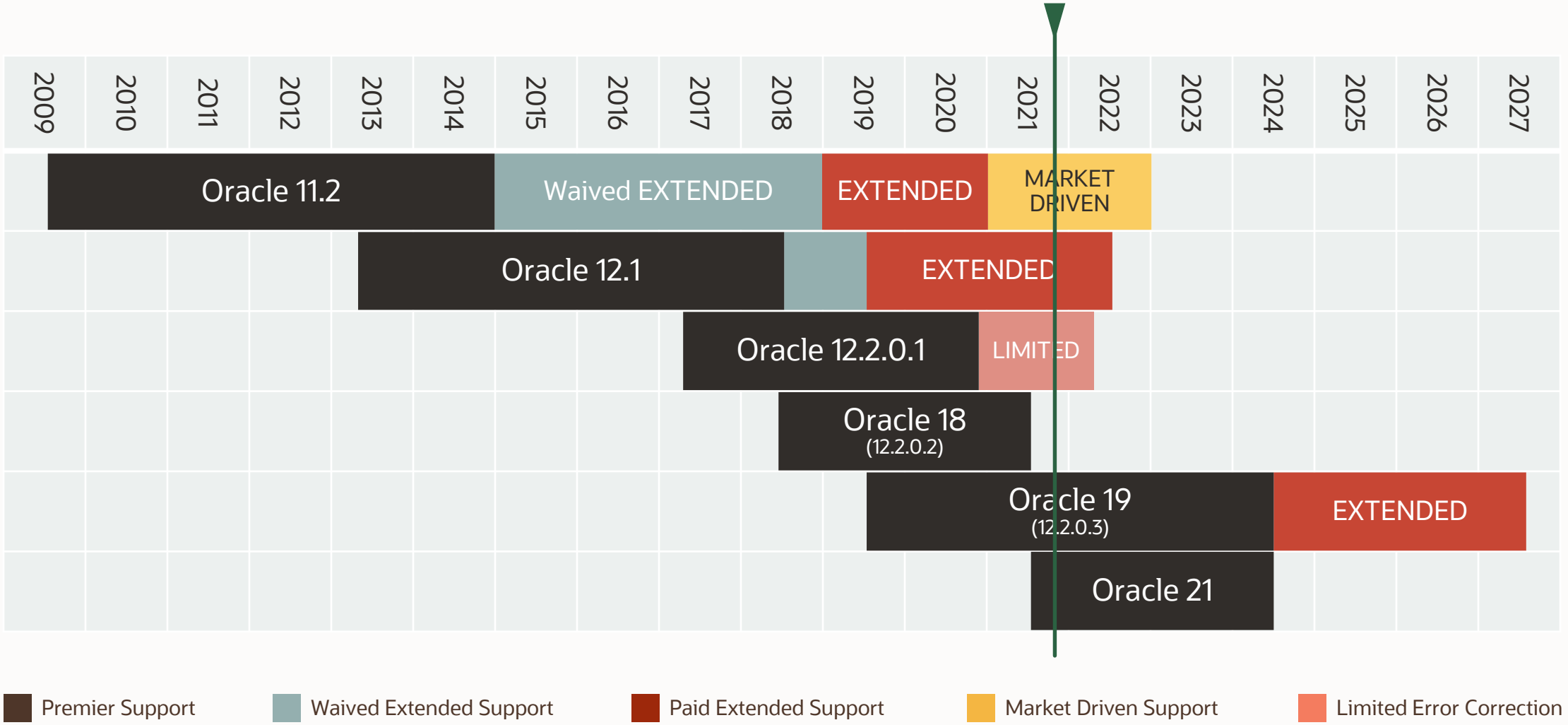
Innovation Release

- 2 years of Premier Support, but there is no Extended Support



Move production databases from one Long Term Support release to the next

Lifetime Support Policy

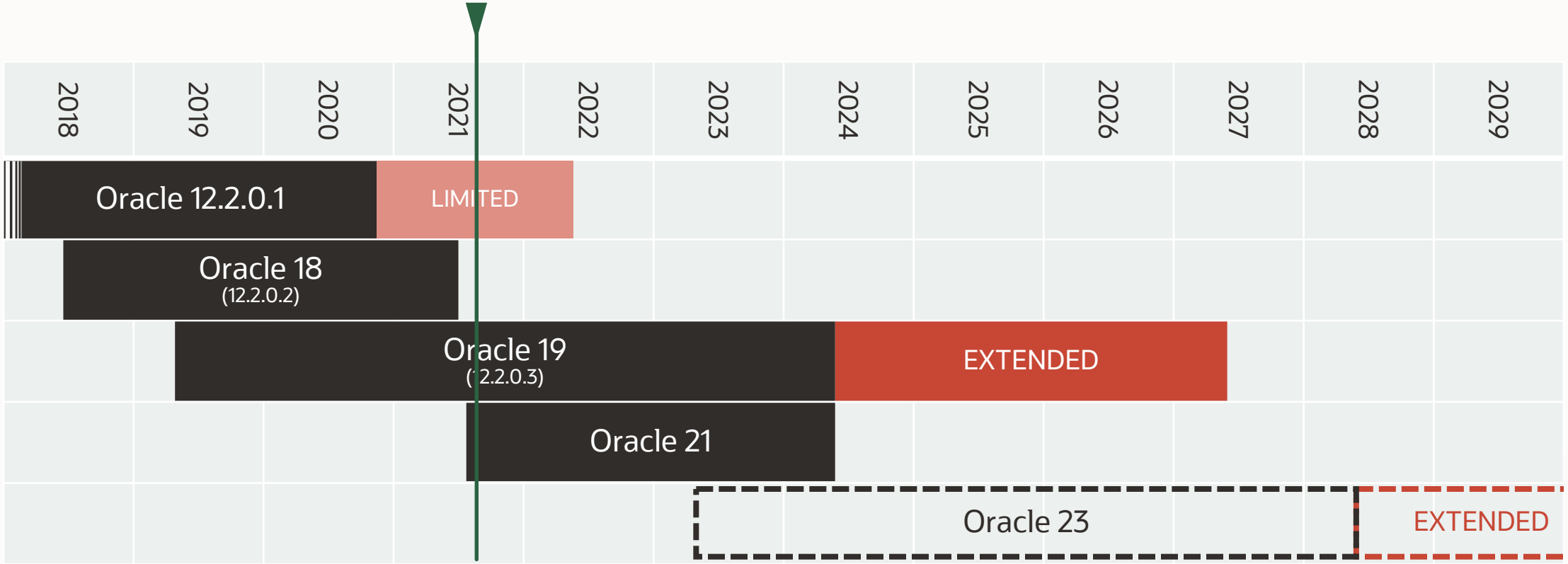


Lifetime Support Policy

Different Support Periods

Premier Support	Bug fixing support regardless of severity
Paid Extended Support	Extra cost extension, 10% / 20% extra cost Included in ULA/PULA contracts
Waived Extended Support	Extended support gets waived to everybody having a valid Support contract for the product
Market Driven Support	Extra cost extension after Extended Support Fixes done only for critical and security issues
Limited Error Correction	Extension for Oracle 12.2.0.1 at no extra cost Only applicable for Sev.1 and security issues
Sustaining Support	Oracle Support assists as long as the customer is using the product – but no new fixes will be delivered

Oracle Database 12.2 and beyond

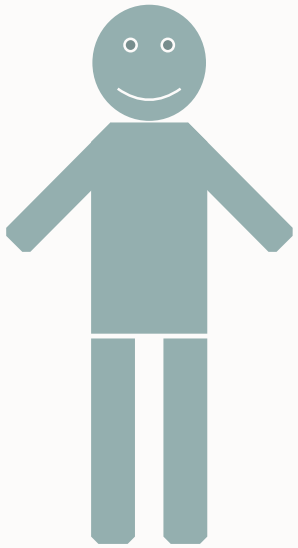


- MOS Note:742060.1 - The Single Source of Truth
- MOS Note:161818.1 - Releases Support Status Summary

Basic Facts | Upgrade vs Migration vs Patching

Database Upgrade

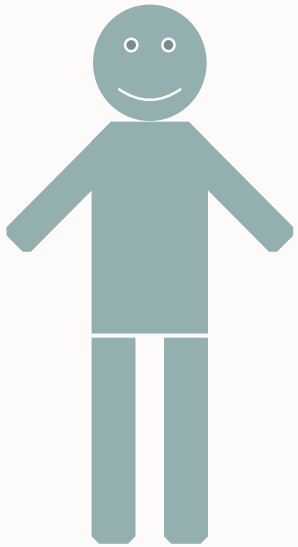
- AutoUpgrade



Oracle 18.6.0 → Oracle 19.12.0

Database Migration

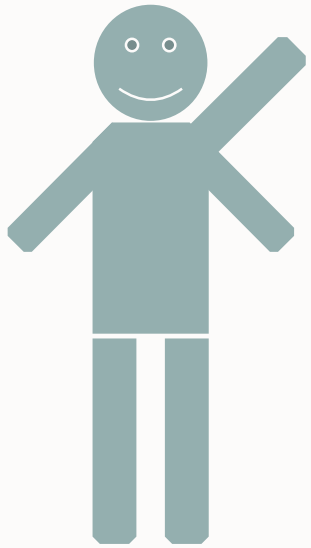
- Data Pump, TTS, FTEX



Oracle 11.2.0.3 → Oracle 19.12.0

Database Patching

- opatch



Oracle 19.3.0 → Oracle 19.12.0

Basic Facts | Patch versus Upgrade

- PATCH

`opatch`

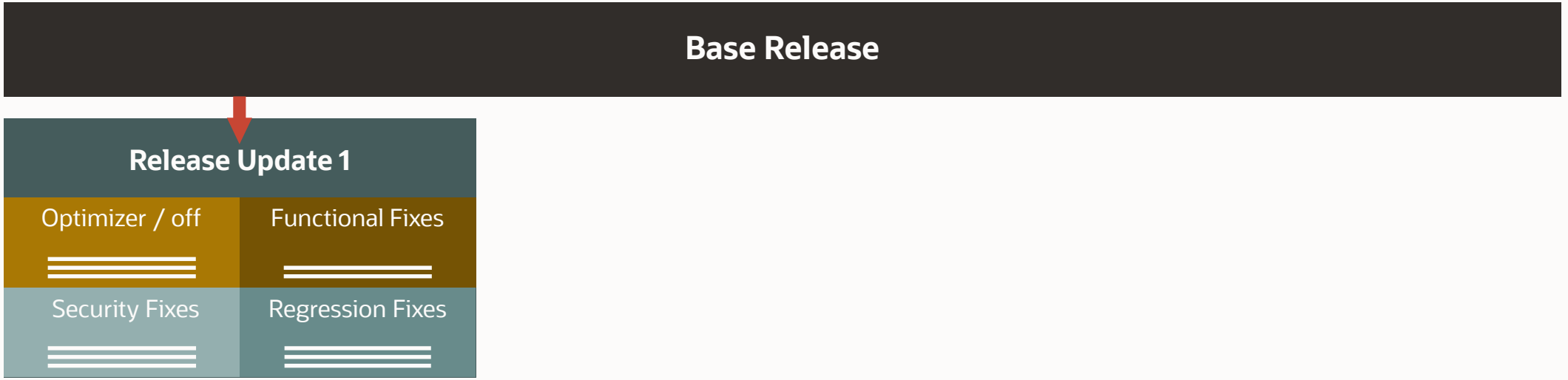
- UPGRADE

`autoupgrade.jar`
or
`dbupgrade`

Patching since Oracle Database 12.2

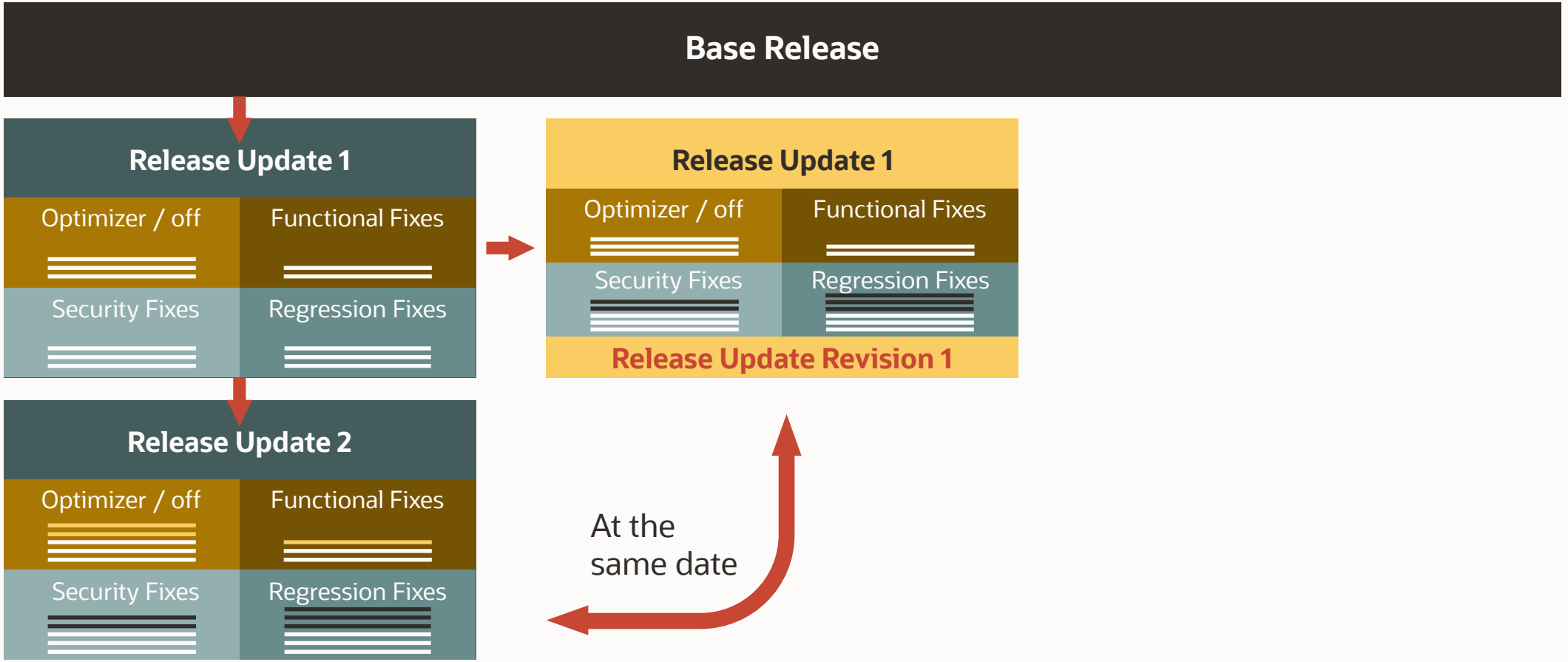
Updates (RU) and Revisions (RUR)

Patching | Release Update 1 (RU)

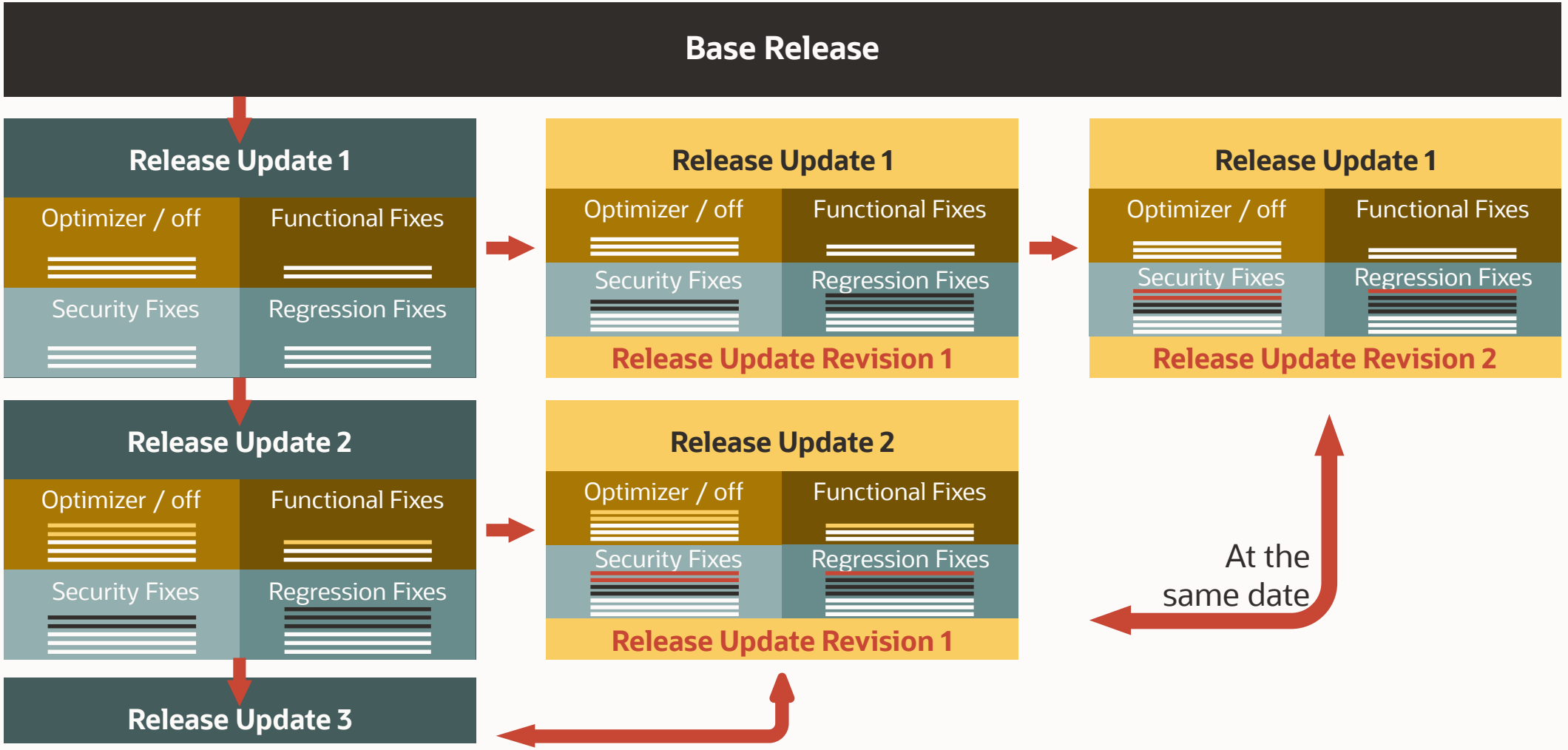


Patching | Release Update 2 (RU)

Patching | Release Update Revision 1 (RUR)



Patching | Release Update Revision 2 (RUR)



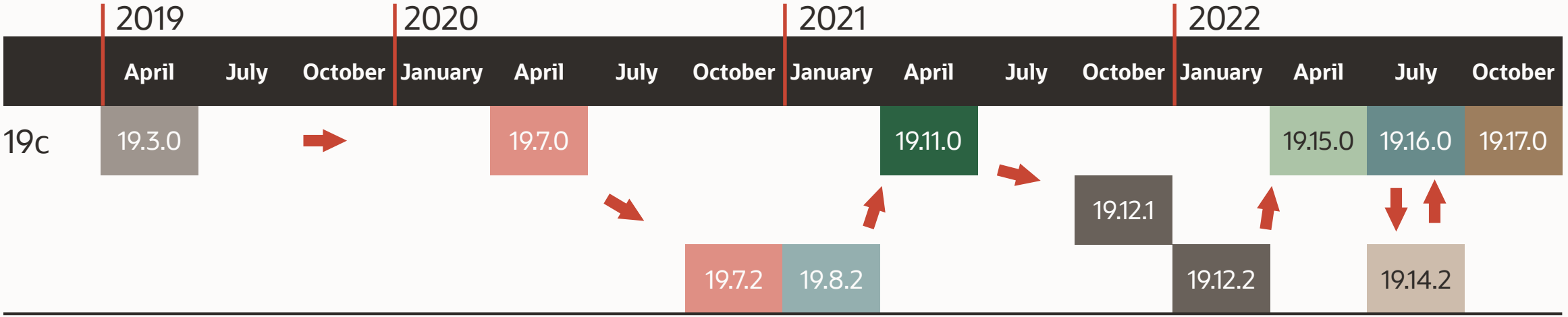
Projection and Recommendation

Yearly Releases, Release Updates and Release Update Revisions

Timeline Example | Future Projection 2022

	2019			2020				2021				2022			
	April	July	October	January	April	July	October	January	April	July	October	January	April	July	October
18c	18.6.0	18.7.0	18.8.0	18.9.0	18.10.0	18.11.0	18.12.0	18.13.0	18.14.0						
	18.5.1	18.6.1	18.7.1	18.8.1	18.9.1	18.10.1	18.11.1	18.12.1	18.13.1	Patching End 18c					
	18.4.2	18.5.2	18.6.2	18.7.2	18.8.2	18.9.2	18.10.2	18.11.2	18.12.2						
19c	19.3.0	19.4.0	19.5.0	19.6.0	19.9.0	19.8.0	19.9.0	19.10.0	19.11.0	19.12.0	19.13.0	19.14.0	19.15.0	19.16.0	19.17.0
		19.3.1	19.4.1	19.5.1	19.6.1	19.7.1	19.8.1	19.9.1	19.10.1	19.11.1	19.12.1	19.13.1	19.14.1	19.15.1	19.16.1
			19.3.2	19.4.2	19.5.2	19.6.2	19.7.2	19.8.2	19.9.1	19.10.2	19.11.2	19.12.2	19.13.3	19.14.2	19.15.2
21c									21.3.0	21.4.0	21.5.0	21.6.0	21.7.0	21.8.0	21.9.0

Timeline Example | Possibilities

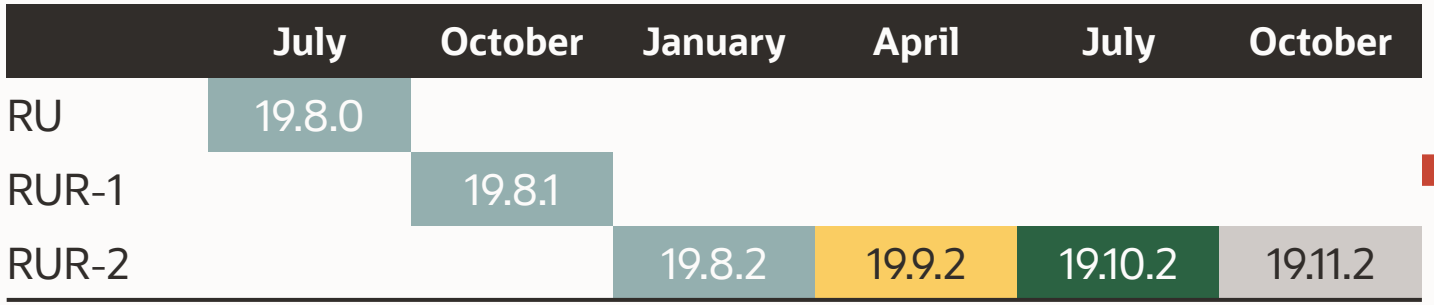


Timeline Example | Much simpler: RUs only

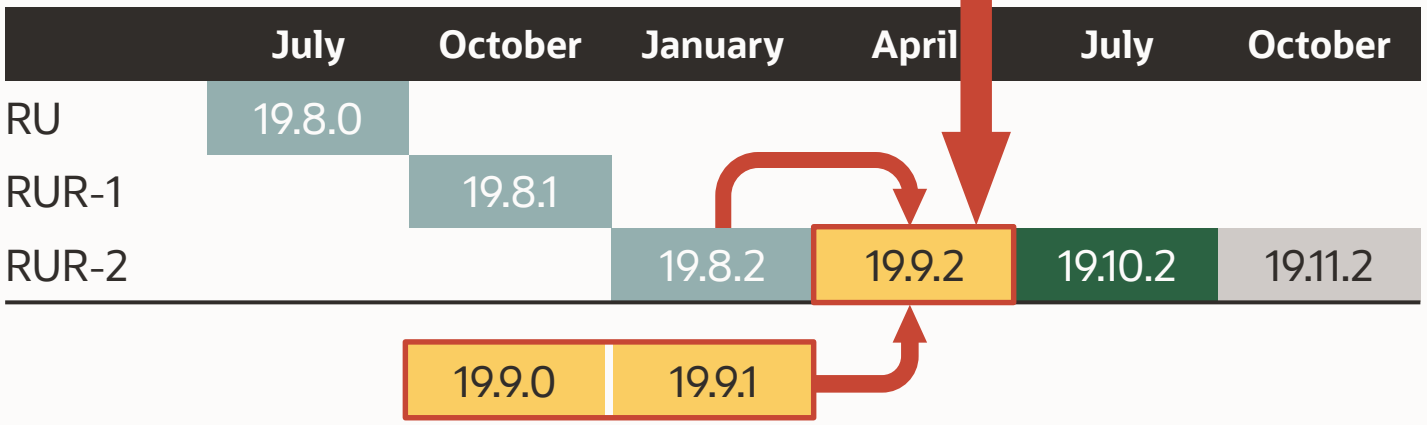
	2019			2020				2021				2022			
	April	July	October	January	April	July	October	January	April	July	October	January	April	July	October
18c	18.6.0	18.7.0	18.8.0	18.9.0	18.10.0	18.11.0	18.12.0	18.13.0	18.14.0						
19c	19.3.0	19.4.0	19.5.0	19.6.0	19.9.0	19.8.0	19.9.0	19.10.0	19.11.0	19.12.0	19.13.0	19.14.0	19.15.0	19.16.0	19.17.0
21c									21.3.0	21.4.0	21.5.0	21.6.0	21.7.0	21.8.0	21.9.0

Attention | The RUR trap!

- You will consume RUs "secretly”



- Stay with RUs!



Attention | The RUR Trap - With Example Numbers

- Assume an RU has 100 new fixes - 25 security only, 75 other fixes
  - Then an RUR1 adds 25 fixes on top, RUR2 adds another 25 on top

	July	October	January	April	July	October
RU	100	200	300	400	500	600
RUR-1		125				
RUR-2			150	?	??	???

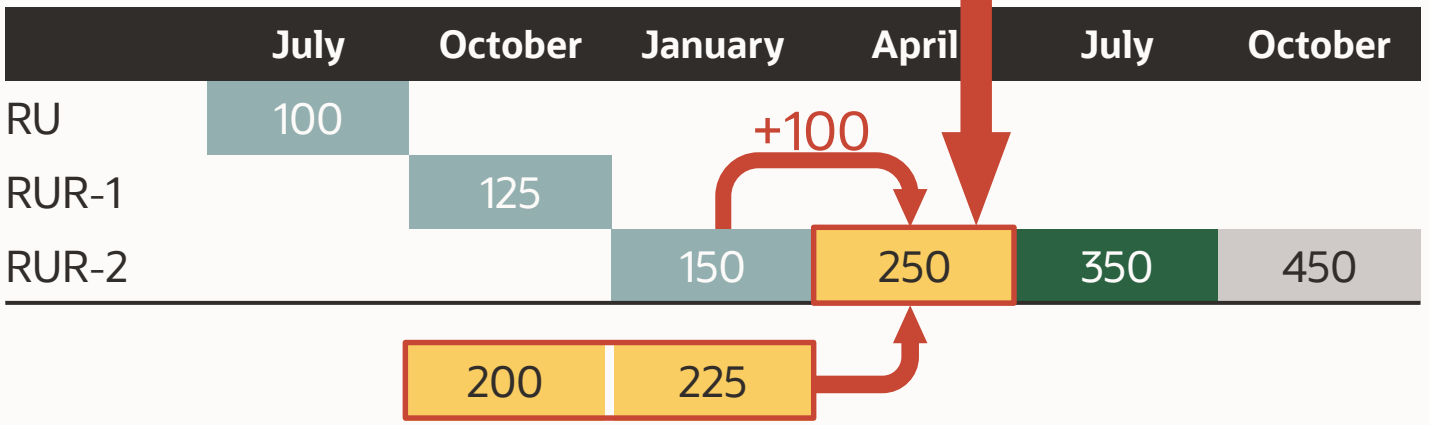


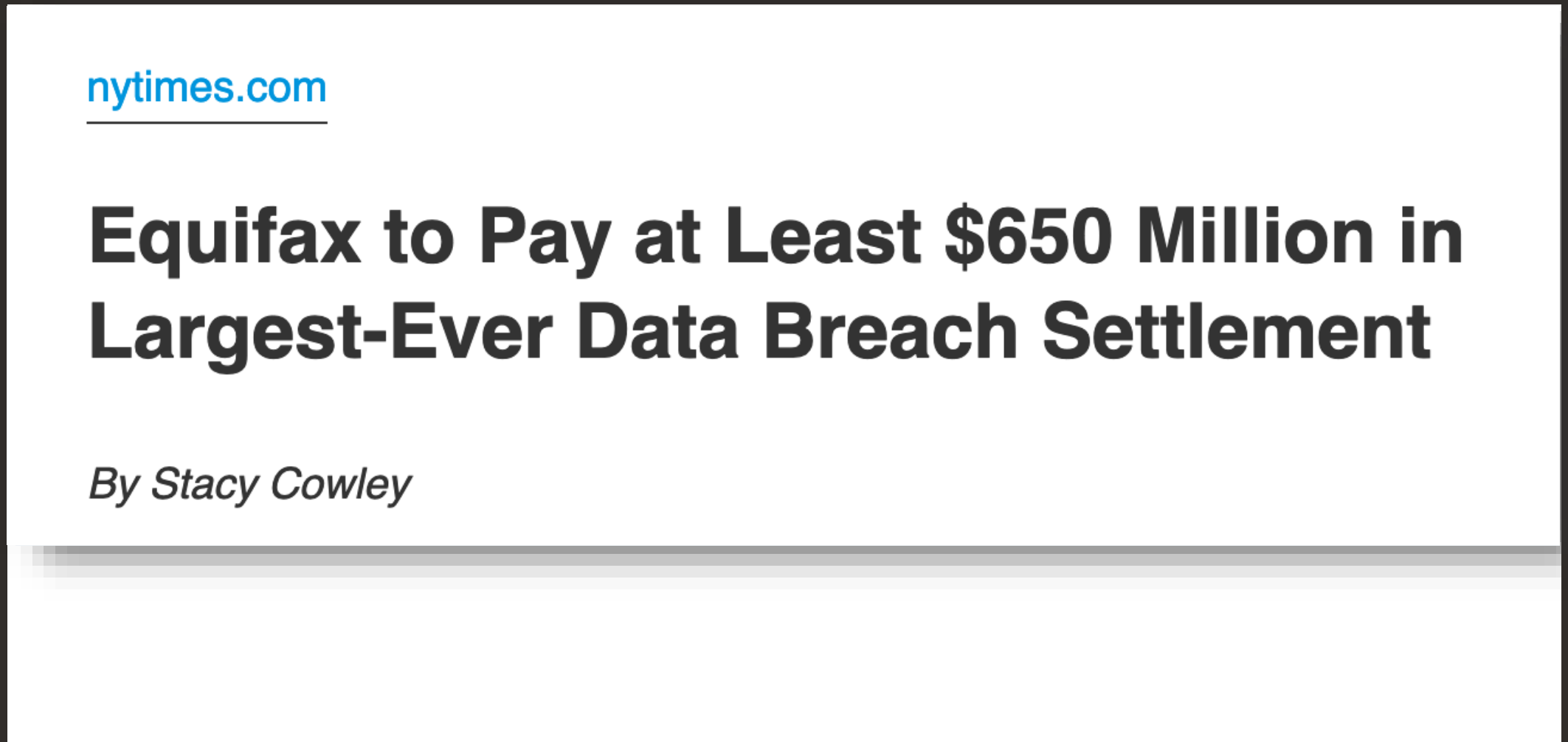
Photo by Jose Fontano on Unsplash

Security

The most important reason to upgrade and patch

Security | Do We Really Need To Say This?

Source: https://www.nytimes.com/2019/07/22/business/equifax-settlement.html?module=inline



"My database is not facing the internet"

Photo by Ben White on Unsplash

92%

of malware gets
delivered via email

Photo by Clint Patterson on Unsplash

Photo by Lubo Minar on Unsplash

Do you need to apply this bundle?

Evaluate the risk

Example: Critical Patch Alert July 2019 | Risk Matrix

CVE#	Component	Package and/or Privilege Required	Protocol	Remote Exploit without Auth.?	CVSS VERSION 3.0 RISK (see Risk Matrix Definitions)									Supported Versions Affected	Notes
					Base Score	Attack Vector	Attack Complex	Privs Req'd	User Interact	Scope	Confid-entiality	Inte-grity	Avail-ability		
CVE-2018-11058	Core RDBMS	None	TCPS/HTTPS	Yes	9.8	Network	Low	None	None	Un-changed	High	High	High	11.2.0.4, 12.1.0.2, 12.2.0.1, 18c, 19c	See Note 1
CVE-2019-2776	Core RDBMS	Create Any Index	OracleNet	No	7.6	[illegible]work	Low	High	None	Changed	High	Low	None	12.1.0.2, 12.2.0.1, 18c, 19c	
CVE-2019-2799	Oracle ODBC Driver	None	Multiple	No	7.5	Network	High	Low	None	Un-changed	High	High	High	11.2.0.4, 12.1.0.2, 12.2.0.1, 18c	See Note 2
CVE-2019-2749	Java VM	Create Session, Create Procedure	Multiple	No	6.8	Network	High	Low	None	Un-changed	None	High	High	11.2.0.4, 12.1.0.2, 12.2.0.1, 18c, 19c	
CVE-2019-2484	Application Express	Valid Account	HTTP	No	5.4	Network	Low	Low	Required	Changed	Low	Low	None	5.1, 18.2	
CVE-2019-2753	Oracle Text	Create Session	OracleNet	No	4.6	Network	Low	Low	Required	Un-changed	Low	None	Low	11.2.0.4, 12.1.0.2, 12.2.0.1, 18c	
CVE-2019-2569	Core RDBMS	Local Logon	Local Logon	No	4.0	Local	High	High	Required	Un-changed	High	None	None	11.2.0.4, 12.1.0.2, 12.2.0.1	
CVE-2016-9572	Spatial	Create Session	OracleNet	No	3.5	Network	Low	Low	Required	Un-changed	None	None	Low	12.2.0.1, 18c	

Photo by Michiel Leunens on Unsplash

Oracle Java Virtual Machine

Do you have OJVM?

Do you need OJVM?

Do you use OJVM?

Oracle Java Virtual Machine | OJVM

OJVM Security Fixes in Quarterly Patches: ⌀ 7.9

Check | OJVM

Is OJVM installed?

```
Select comp_id, comp_name, version from DBA_REGISTRY order by 1;
```

COMP_ID	COMP_NAME	VERSION
CATALOG	Oracle Database Catalog Views	19.9.0.0.0
CATJAVA	Oracle Database Java Packages	19.9.0.0.0
CATPROC	Oracle Database Packages and Types	19.9.0.0.0
JAVAVM	JServer JAVA Virtual Machine	19.9.0.0.0
OLS	Oracle Label Security	19.9.0.0.0
ORDIM	Oracle Multimedia	19.9.0.0.0
OWM	Oracle Workspace Manager	19.9.0.0.0
XDB	Oracle XML Database	19.9.0.0.0
XML	Oracle XDK	19.9.0.0.0

Recommendation | OJVM and STARTUP UPGRADE

Does OJVM require STARTUP UPGRADE?

- No!
- Blog: Do you need STARTUP UPGRADE for OJVM?
  - If you ran it in STARTUP UPGRADE before, use:
    - `./datapatch -verbose -skip_upgrade_check`

New since Jan 2020 | JDK Update in RUs

JDK patches included in RUs

- Always stable JDK from the previous quarter
- Jan 2021 Release Update

```
$ cd $ORACLE_HOME/jdk/bin
$ ./java -version
   java version "1.8.0_271"
   Java(TM) SE Runtime Environment (build 1.8.0_271-b09)
   Java HotSpot(TM) 64-Bit Server VM (build 25.271-b09, mixed mode)
```

  - Always [current version -1]
  - https://www.oracle.com/java/technologies/javase/8u-relnotes.html

Pro Tip:
MS Windows port requires separate JDK patching unfortunately

Agenda Day 1

- 13:00-13:15h Welcome
- 13:15-13:45h Release and Patching Strategy
- 13:45-14:45h Become an Upgrade/Performance expert in 60 minutes

- 14:45-15:00h Break

- 15:00-15:15h Checkout the environments
- 15:15-17:00h LAB DAY1:
  Generate load, upgrade, performance evaluation

Upgrade to Oracle Database 19c

Photo by SpaceX on Unsplash

your key to

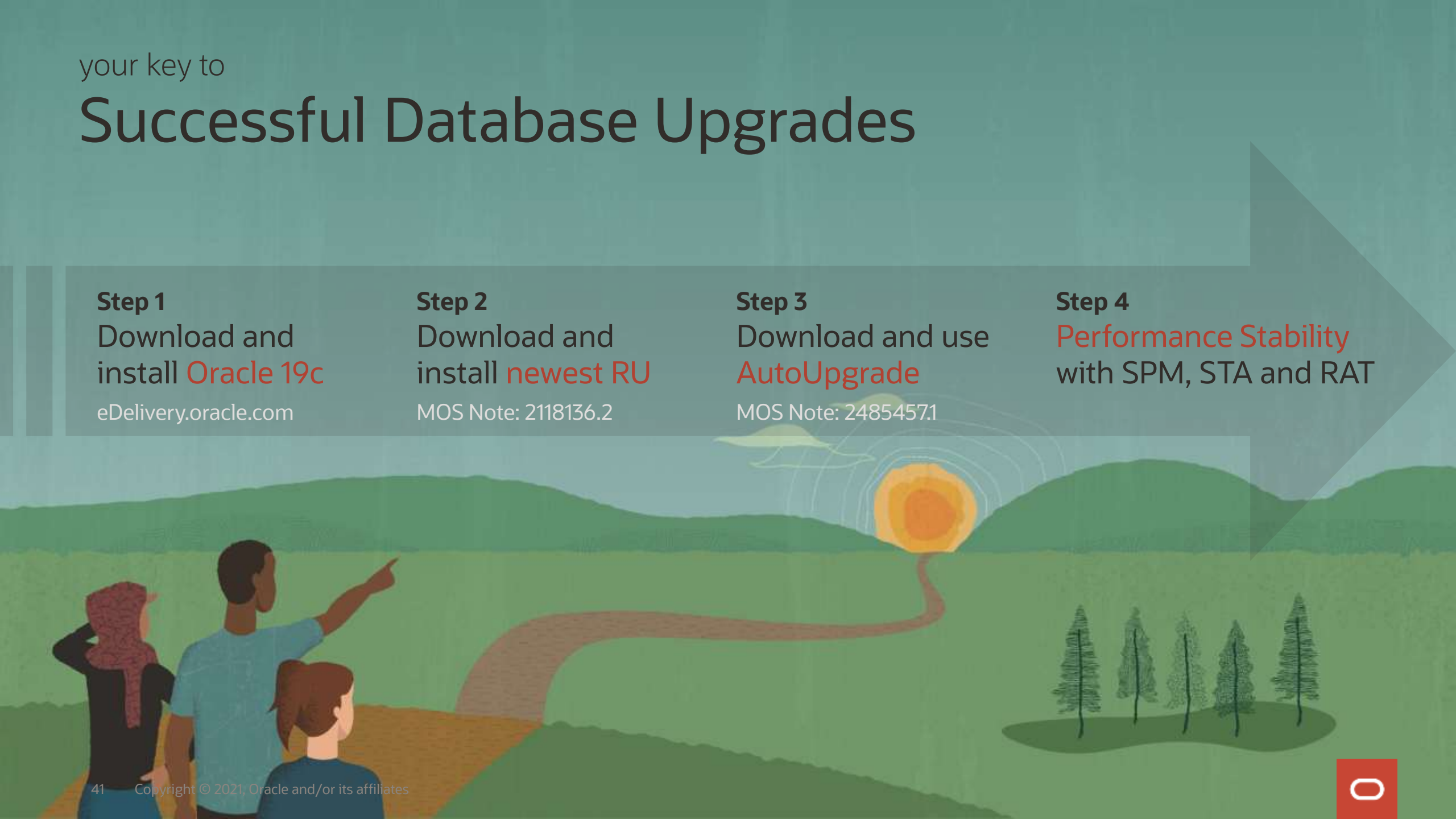
Successful Database Upgrades

Step 1
Download and install Oracle 19c
eDelivery.oracle.com

Step 2
Download and install newest RU
MOS Note: 2118136.2

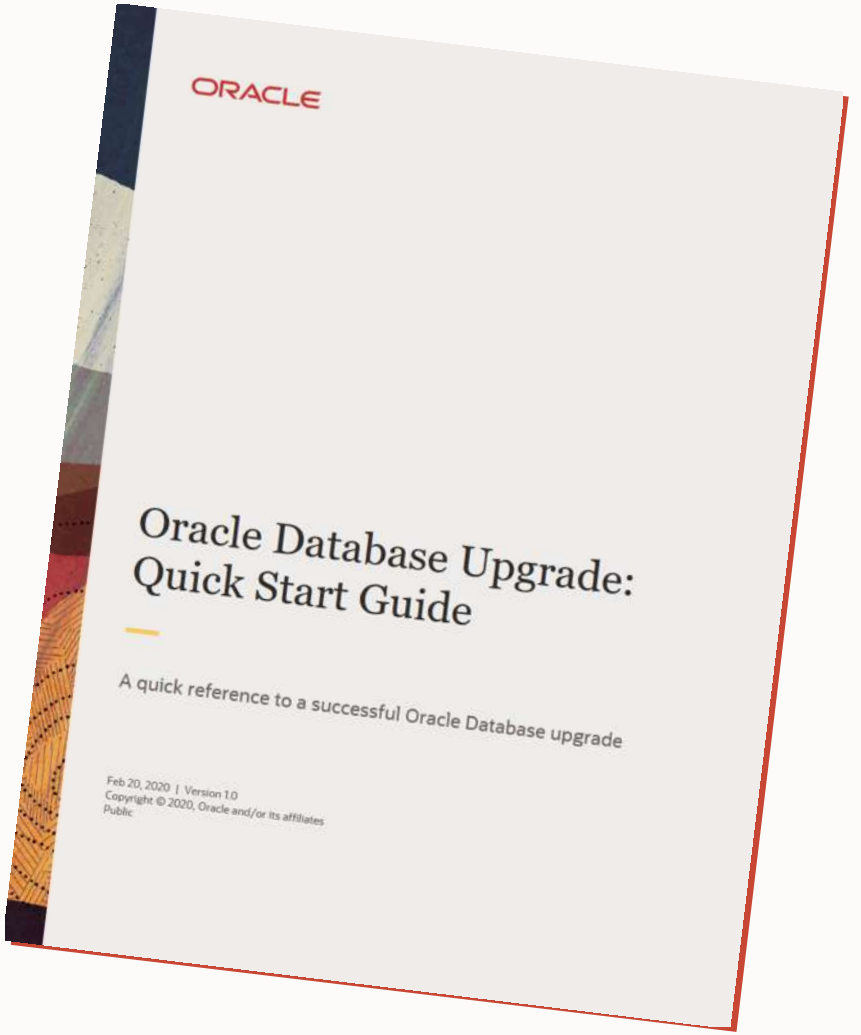
Step 3
Download and use AutoUpgrade
MOS Note: 24854571

Step 4
Performance Stability with SPM, STA and RAT

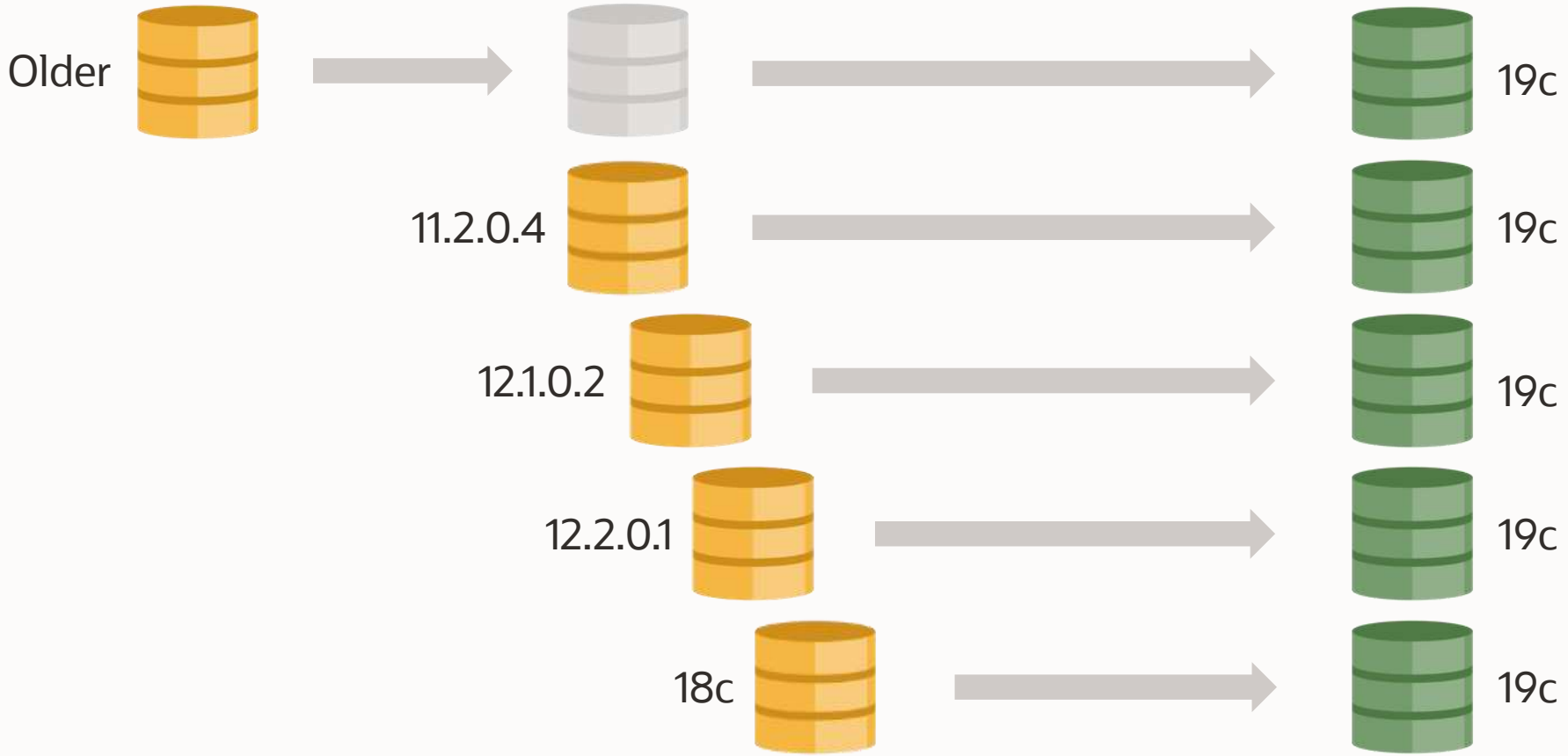


Get started | Quick Start Guide

Simple overview
Read it, try it
Download from oracle.com



Database Upgrade | Supported Releases



Database Upgrade | Supported Releases

Database / Oracle / Oracle Database / Release 19

Database Upgrade Guide

Oracle Database Releases That Support Direct Upgrade

Review the supported options for direct upgrades to the latest Oracle Database release.

You can perform a direct upgrade to the new release from the following releases:

- 11.2.0.4
- 12.1.0.2
- 12.2.0.1
- 18

Database Upgrade Guide

Photo by Joshua Sortino on Unsplash

AutoUpgrade

The ONLY recommended way to upgrade databases

Upgrade | AutoUpgrade - As Easy As 1-2-3

Supported source releases
- 11.2.0.4
- 12.1.0.2
- 12.2.0.1
- 18
- 19

All architectures (CDB and non-CDB)
All supported operating systems
All editions (SE2, EE)
All types (single instance and RAC)

Upgrade | AutoUpgrade - As Easy As 1-2-3

START	1. DOWNLOAD	2. CONFIG	3. DEPLOY	SUCCESS

Download from My Oracle Support ID 2485457.1

Upgrade | AutoUpgrade - As Easy As 1-2-3

Simple text file

```
upg1.source_home=/u01/app/oracle/product/12.2.0.1
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=CDB1
```

Upgrade | AutoUpgrade - As Easy As 1-2-3

One command

```
$ java -jar autoupgrade.jar -config cdb1.cfg -mode deploy
```

Advanced monitoring and logging

Upgrade | AutoUpgrade - As Easy As 1-2-3

Supported target releases

- 12.2.0.1
- 18
- 19
- 21
- Any future release

AutoUpgrade | Need And Don't Need

JAVA	JAR FILE	AGENTS	ENTERPRISE MANAGER	DBUA	EXTRA LICENSE

- Java 8 required
  - Part of Oracle Home since 12.1.0.2
- 3 MB jar file

AutoUpgrade | Need And Don't Need

JAVA	JAR FILE	AGENTS	ENTERPRISE MANAGER	DBUA	EXTRA LICENSE

- No agents to install
- Enterprise Manager not needed
- AutoUpgrade offers superior functionality
- No extra license

AUTOUPGRADE
Essentials

Photo by Daniel Cheung on Unsplash

AutoUpgrade | Essentials

Download

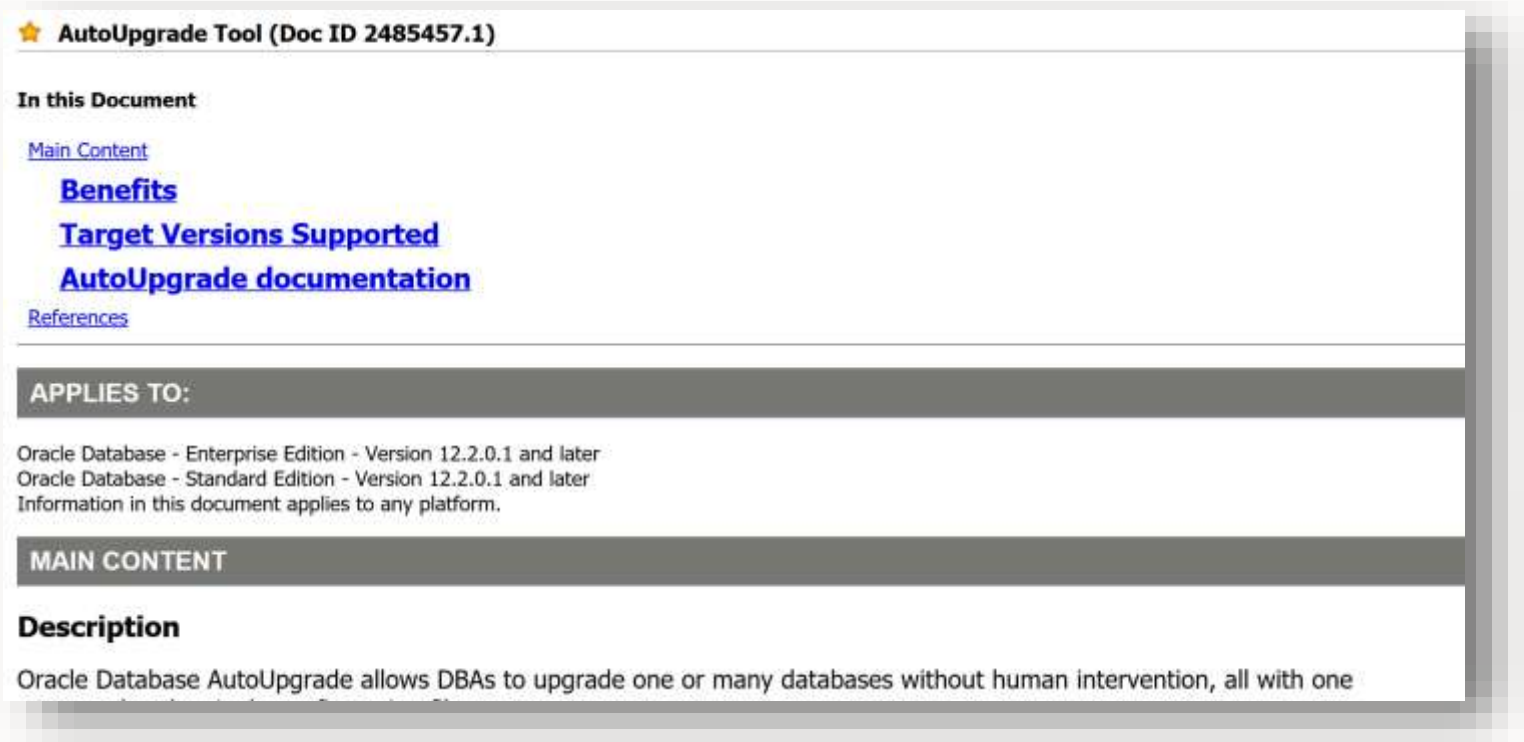
Configure

Analyze

Check

Upgrade

Always download latest version from MOS



AutoUpgrade Tool (Doc ID 2485457.1)

In this Document

Main Content

Benefits

Target Versions Supported

AutoUpgrade documentation

References

APPLIES TO:

Oracle Database - Enterprise Edition - Version 12.2.0.1 and later
Oracle Database - Standard Edition - Version 12.2.0.1 and later
Information in this document applies to any platform.

MAIN CONTENT

Description

Oracle Database AutoUpgrade allows DBAs to upgrade one or many databases without human intervention, all with one

AutoUpgrade | Essentials

Download
Configure
Analyze
Check
Upgrade

Check your version

```
$ java -jar autoupgrade.jar -version

build.version 21.3.211115
build.hash 081e3f7
build.date 2021/11/15 11:57:54
build.max_target_version 21
build.supported_target_versions 12.2,18,19,21
build.type production
```

Compare to latest version on MOS

Download

The most recent version of AutoUpgrade can be downloaded via this link: version 20211115.

AutoUpgrade | Essentials

Download

Configure

Analyze

Check

Upgrade

AutoUpgrade handles older releases as well

```
$ java -jar autoupgrade.jar -version

build.version 21.3.211115
build.hash 081e3f7
build.date 2021/11/15 11:57:54
build.max_target_version 21
build.supported_target_versions 12.2,18,19,21
```

AutoUpgrade | Essentials

- Download
- **Configure**
- Analyze
- Check
- Upgrade

Shortest possible config file version

```
upg1.source_home=/u01/app/oracle/product/12.2.0.1
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=CDB1
```

Or, generate a sample config file

```
$ java -jar autoupgrade.jar -create_sample_file config

Created sample configuration file /home/oracle/sample_config.cfg
```

Pro tip: *upg1* is a prefix that you decide. Use it to define multiple databases

AutoUpgrade | Essentials

- Download
- Configure
- **Analyze**
- Check
- Upgrade

Analyze your database

```
$ java -jar autoupgrade.jar -config CDB1.cfg -mode analyze

...

upg> Job 100 completed

Please check the summary report at:
/u01/app/oracle/cfgtoollogs/autoupgrade/cfgtoollogs/upgrade/auto/status/status.html
/u01/app/oracle/cfgtoollogs/autoupgrade/cfgtoollogs/upgrade/auto/status/status.log
```

Pro tip: Analyze is similar to running `preupgrade.jar`

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- **Check**
- Upgrade

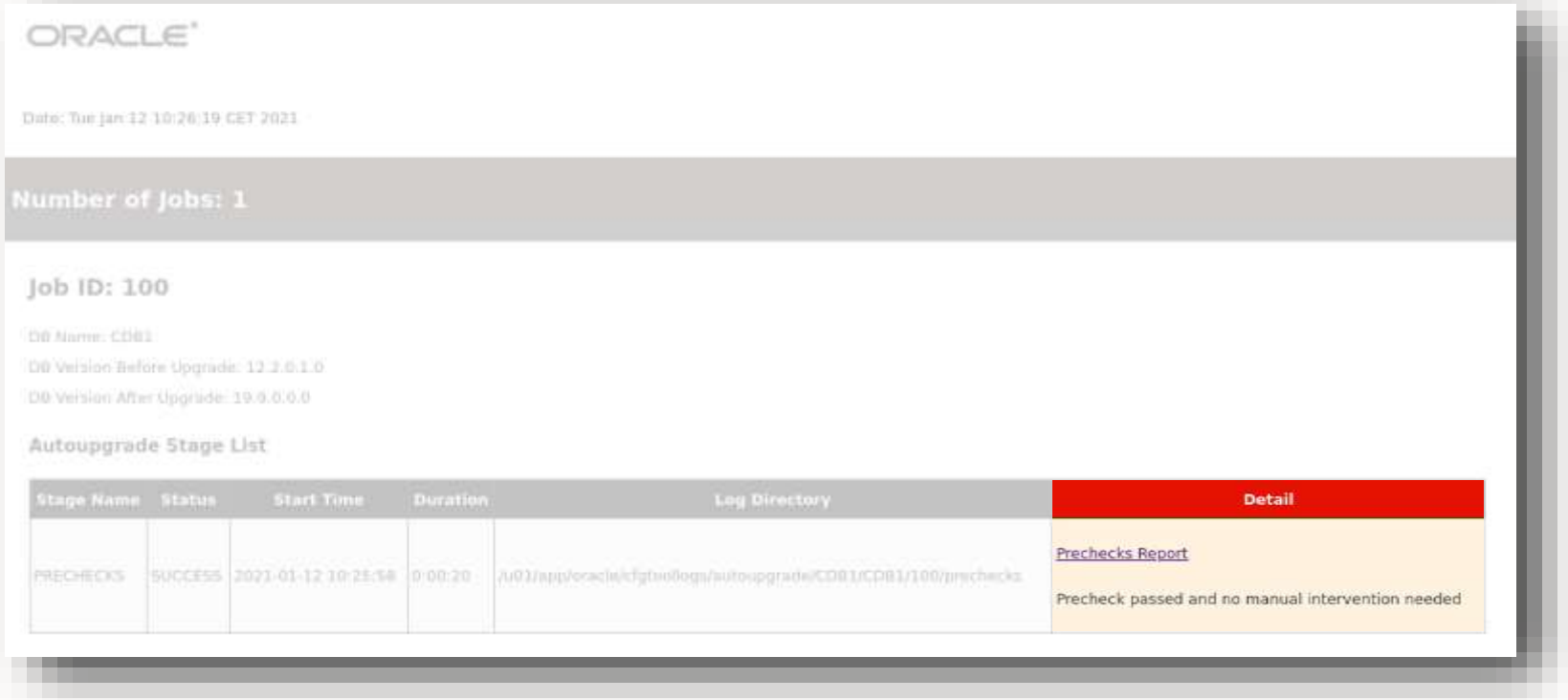
Summary report - text

```
=============================================
          Autoupgrade Summary Report
=============================================
[Date]           Tue Jan 12 10:26:19 CET 2021
[Number of Jobs] 1
=============================================
[Job ID] 100
=============================================
[DB Name]                   CDB1
[Version Before Upgrade]  12.2.0.1.0
[Version After Upgrade]   19.9.0.0.0
---------------------------------------------
[Stage Name]     PRECHECKS
[Status]         SUCCESS
[Start Time]     2021-01-12 10:25:58
[Duration]       0:00:20
[Log Directory] /u01/app/oracle/upg/CDB1/100/prechecks
[Detail]         /u01/app/oracle/upg/CDB1/100/prechecks/cdb1_preupgrade.log
                 Precheck passed and no manual intervention needed
---------------------------------------------
```

AutoUpgrade | Essentials

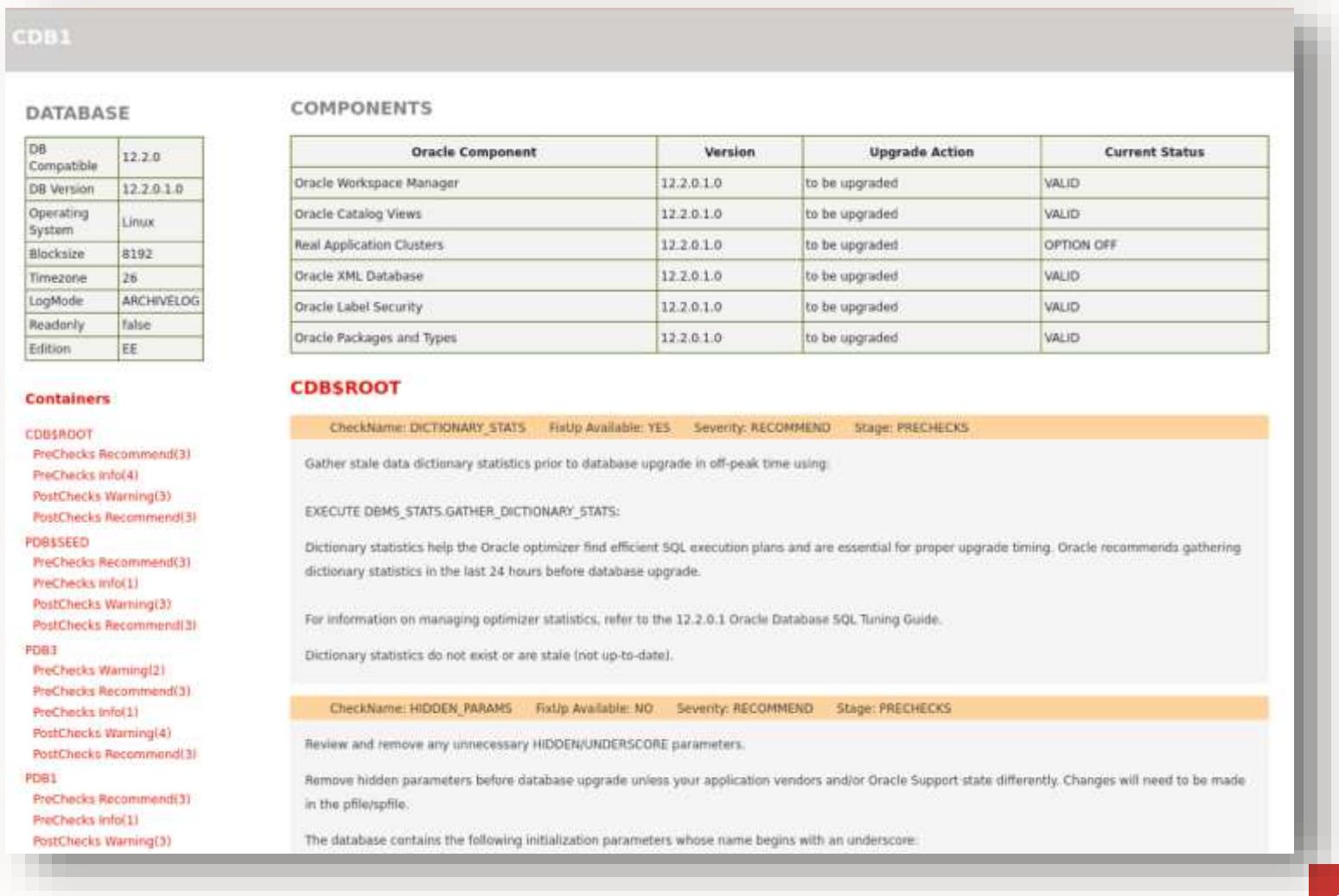
- Download
- Configure
- Analyze
- **Check**
- Upgrade

Summary report - HTML



AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- **Check**
- Upgrade



CDB1

DATABASE

DB Compatible	12.2.0
DB Version	12.2.0.1.0
Operating System	Linux
Blocksize	8192
Timezone	26
LogMode	ARCHIVELOG
Readonly	false
Edition	EE

Containers

CDB$ROOT
PreChecks Recommend(3)
PreChecks Info(4)
PostChecks Warning(3)
PostChecks Recommend(3)
PDB$SEED
PreChecks Recommend(3)
PreChecks Info(1)
PostChecks Warning(3)
PostChecks Recommend(3)
PDB3
PreChecks Warning(2)
PreChecks Recommend(3)
PreChecks Info(1)
PostChecks Warning(4)
PostChecks Recommend(3)
PDB1
PreChecks Recommend(3)
PreChecks Info(1)
PostChecks Warning(3)

COMPONENTS

Oracle Component	Version	Upgrade Action	Current Status
Oracle Workspace Manager	12.2.0.1.0	to be upgraded	VALID
Oracle Catalog Views	12.2.0.1.0	to be upgraded	VALID
Real Application Clusters	12.2.0.1.0	to be upgraded	OPTION OFF
Oracle XML Database	12.2.0.1.0	to be upgraded	VALID
Oracle Label Security	12.2.0.1.0	to be upgraded	VALID
Oracle Packages and Types	12.2.0.1.0	to be upgraded	VALID

CDB$ROOT

CheckName: DICTIONARY_STATS FixUp Available: YES Severity: RECOMMEND Stage: PRECHECKS

Gather stale data dictionary statistics prior to database upgrade in off-peak time using:

EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

Dictionary statistics help the Oracle optimizer find efficient SQL execution plans and are essential for proper upgrade timing. Oracle recommends gathering dictionary statistics in the last 24 hours before database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

Dictionary statistics do not exist or are stale (not up-to-date).

CheckName: HIDDEN_PARAMS FixUp Available: NO Severity: RECOMMEND Stage: PRECHECKS

Review and remove any unnecessary HIDDEN/UNDERSCORE parameters.

Remove hidden parameters before database upgrade unless your application vendors and/or Oracle Support state differently. Changes will need to be made in the pfile/spfile.

The database contains the following initialization parameters whose name begins with an underscore:

AutoUpgrade | Essentials

Download
Configure
Analyze
Check
Upgrade

Preupgrade report comes in:

- HTML
- Text
- JSON

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- Check
- **Upgrade**

Upgrade

```
$ java -jar autoupgrade.jar -config CDB1.cfg -mode deploy
```

Have a cup of coffee and wait, or ...

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- Check
- **Upgrade**

Monitor

```
upg> lsj

+----+-------+---------+---------+-------+--------------+--------+---------------+
|Job#|DB_NAME|    STAGE|OPERATION| STATUS|    START_TIME| UPDATED|        MESSAGE|
+----+-------+---------+---------+-------+--------------+--------+---------------+
| 101|   CDB1|PREFIXUPS|EXECUTING|RUNNING|20/11/24 13:38|13:39:26|Remaining 12/13|
+----+-------+---------+---------+-------+--------------+--------+---------------+
```

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- Check
- **Upgrade**

All the details

```
upg> status -job 101

Progress
-----------------------------------
Start time:      20/11/24 13:38
Elapsed (min):   13
Last update:     2020-11-24T13:48:52.139
Stage:           DBUPGRADE
Operation:       EXECUTING
Status:          RUNNING
Stage summary:
    SETUP             <1 min
    GRP               <1 min
    PREUPGRADE        <1 min
    PRECHECKS         <1 min
    PREFIXUPS         8 min
    DRAIN             <1 min
    DBUPGRADE         3 min (IN PROGRESS)

Job Logs Locations
-----------------------------------
Logs Base:    /home/oracle/autoupg_default/CDB1/CDB1
Job logs:     /home/oracle/autoupg_default/CDB1/CDB1/101
Stage logs:   /home/oracle/autoupg_default/CDB1/CDB1/101/dbupgrade
TimeZone:     /home/oracle/autoupg_default/CDB1/CDB1/temp
```

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- Check
- **Upgrade**

All the details - continued

```
...

Additional information
----------------------------------
Details:
[Upgrading] is [0%] completed for [cdb1-cdb$root]
                 +---------+---------------+
                 |CONTAINER|     PERCENTAGE|
                 +---------+---------------+
                 | CDB$ROOT|   UPGRADE[12%]|
                 | PDB$SEED|UPGRADE PENDING|
                 |     PDB3|UPGRADE PENDING|
                 +---------+---------------+

Error Details:
None
```

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- Check
- **Upgrade**

Success

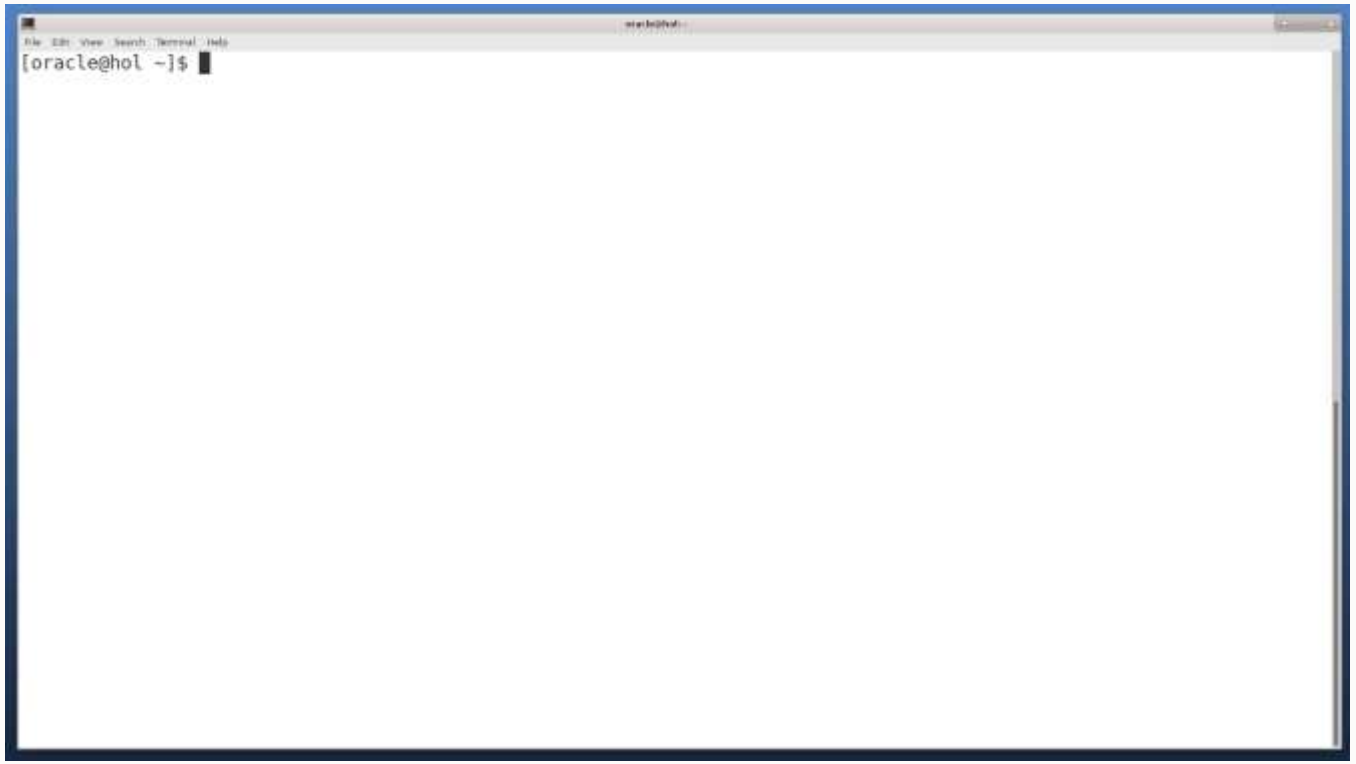
```
upg> Job 101 completed
------------------- Final Summary --------------------
Number of databases            [ 1 ]

Jobs finished successfully     [1]
Jobs failed                    [0]
Jobs pending                   [0]
------------- JOBS FINISHED SUCCESSFULLY -------------
Job 101 for CDB1

---- Drop GRP at your convenience once you consider it is no longer needed ----
Drop GRP from CDB1: drop restore point AUTOUPGRADE_9212_CDB1122010
```

And it includes:

- Recompilation (utlrp.sql)
- Time zone file upgrade
- Postupgrade fixups
- … and so much more

AutoUpgrade | Essentials

- Download
- Configure
- Analyze
- Check
- **Upgrade**



Watch on YouTube

AUTOUPGRADE
Details

Photo by Ciprian Boiciuc on Unsplash

AutoUpgrade | Monitoring

Watch on YouTube

AutoUpgrade | Log File Structure

- Logs written in TEXT and JSON format
  - /cfgtoollogs
    - ./upgrade/auto ◀ Status Log
  - /database_1
    - ./job_number
      - ./prechecks ◀ HTML Report
      - ./preupgrade
      - ./prefixups
      - ./drain
      - ./dbupgrade ◀ Upgrade Logs
      - ./postupgrade
    - ./temp
  - /database_2
    - …

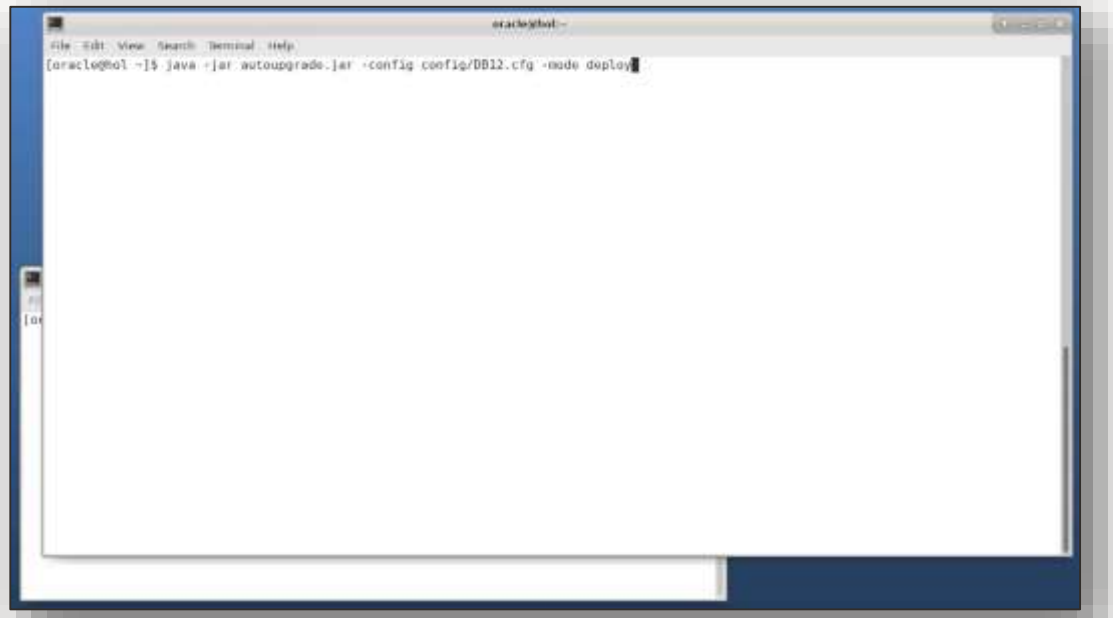
Photo by Chris Briggs on Unsplash

What if ...

Tips and Tricks and Workarounds

AutoUpgrade | What if … your session is lost

- AutoUpgrade is fully resumable
- Restart using the same command line
- Previous work is preserved
  - upgrade restarts from where it left



Watch on YouTube

AutoUpgrade | What if … AutoUpgrade fails

1. Create zip file

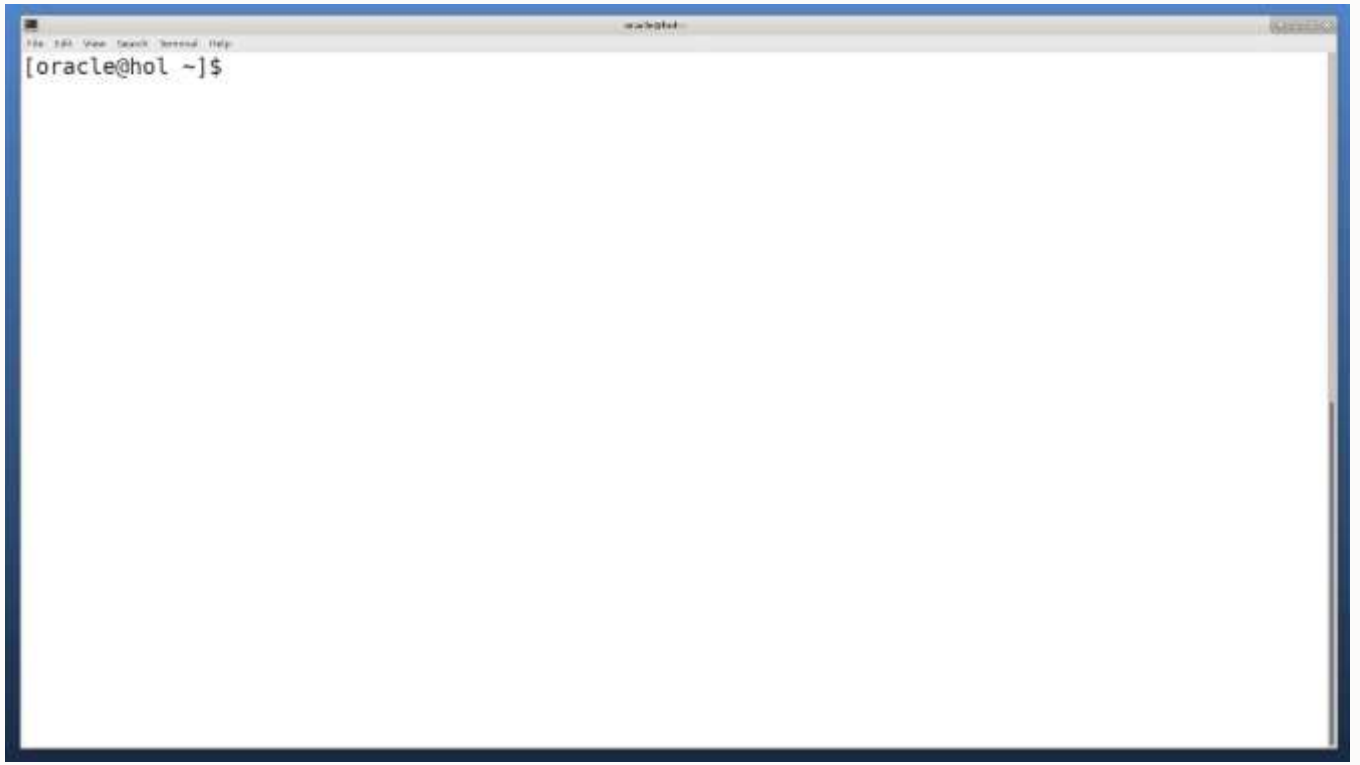
```
$ java -jar autoupgrade.jar -config config.cfg -zip
```

2. Optionally, add `opatch lsinventory`

```
$ $ORACLE_HOME/OPatch/opatch lsinventory > opatch.txt
$ zip -r AUPG_210419_0735_461.zip opatch.txt
```

3. Upload it to My Oracle Support

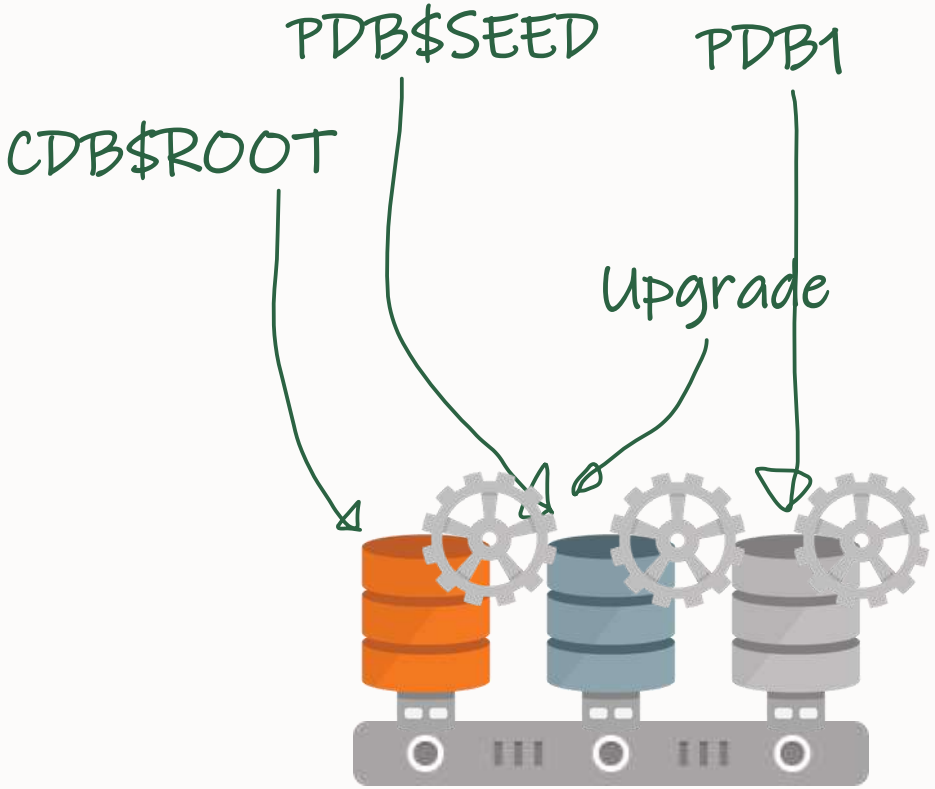
AutoUpgrade | What if ... AutoUpgrade fails



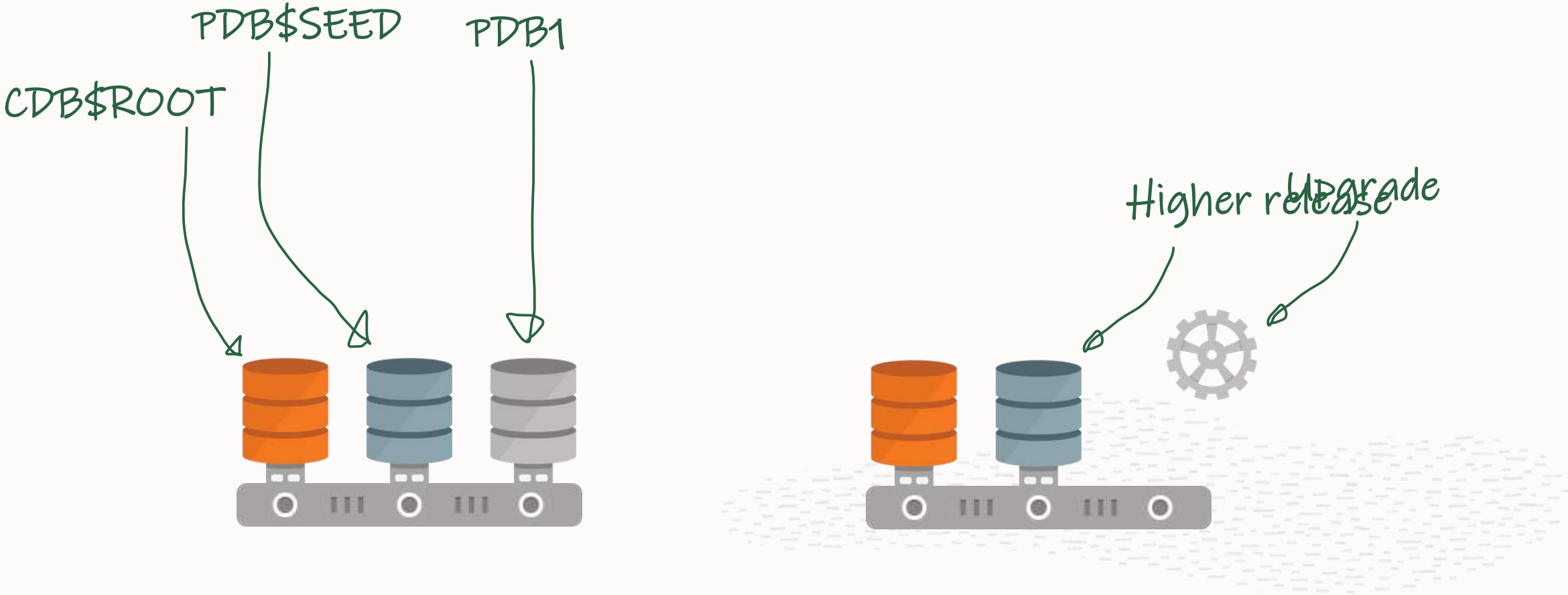
Watch on YouTube

Unplug / Plug / Upgrade

Container Database Upgrade | Concept



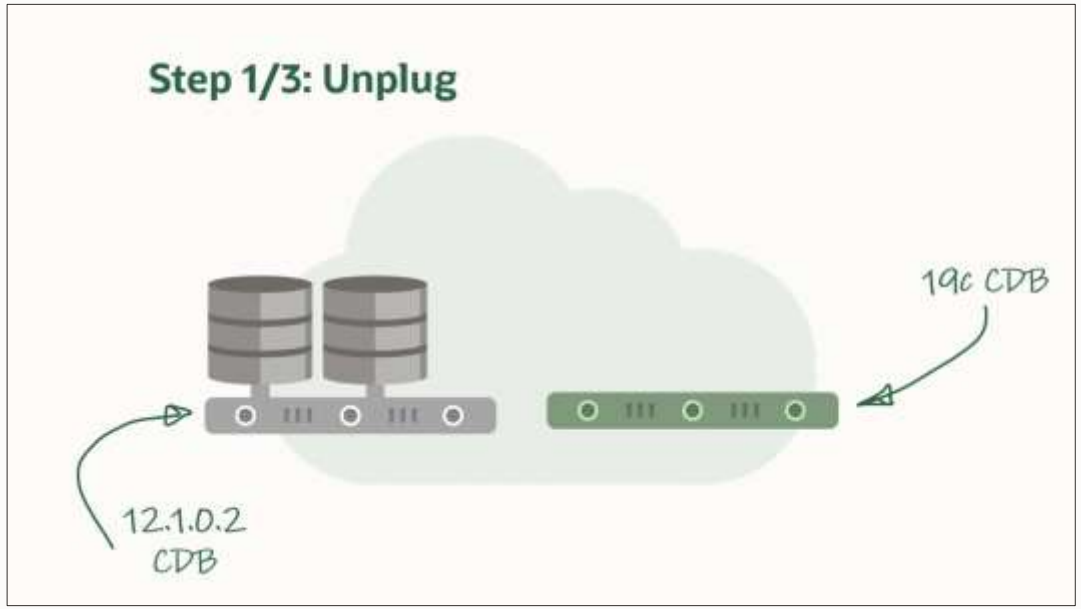
Unplug-plug Upgrade | Concept



AutoUpgrade | Unplug-plug Upgrade

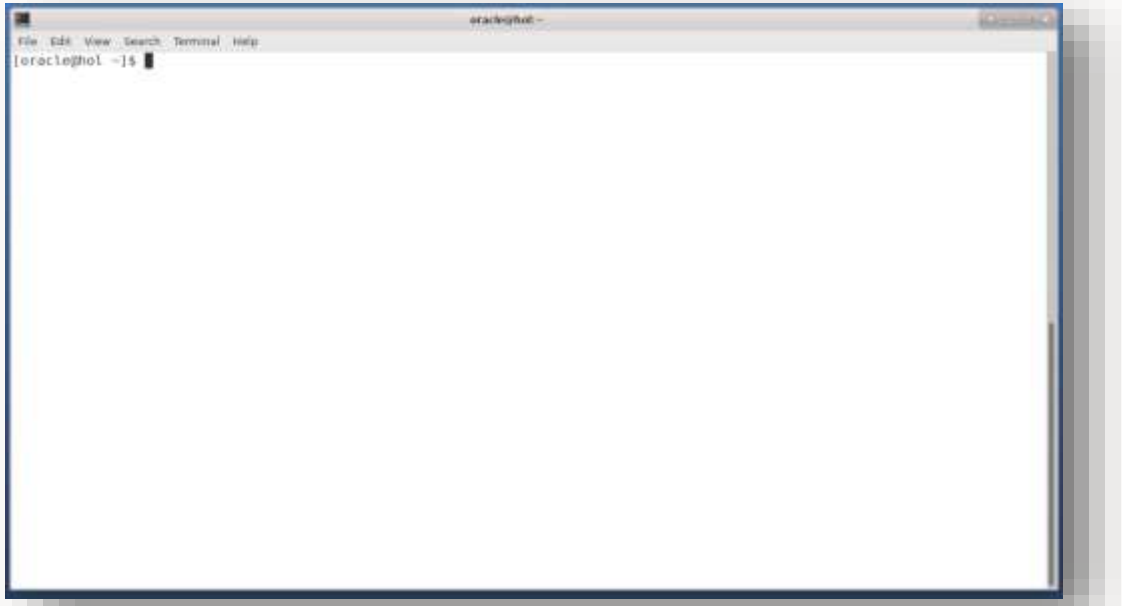
Upgrade a single PDB

- Faster
- More flexible
- Requires compatible target CDB
- Not compatible with Flashback Database
  - Consider using Refreshable PDBs
  - Copy data files (`target_pdb_copy_option`)



AutoUpgrade | Unplug-plug Upgrade

```
upg1.sid=CDB12102
upg1.target_cdb=CDB19
upg1.pdbs=pdb1
upg1.source_home=/u01/app/oracle/product/12102
upg1.target_home=/u01/app/oracle/product/19
```



Watch on YouTube

AutoUpgrade | Unplug-plug Upgrade

Upgrade several PDBs

```
upg1.pdbs=pdb1,pdb2,pdb3
```

Rename a PDB

```
upg1.pdbs=pdb1
upg1.target_pdb_name.pdb1=sales
```

Copy data files on plug-in

```
upg1.pdbs=pdb1
upg1.target_pdb_copy_option.pdb1=file_name_convert=('pdb1','sales')
```


Performance Stability

Photo by Loic Leray on Unsplash

your key to

Successful Database Upgrades

Step 1
Download and install Oracle 19c

eDelivery.oracle.com

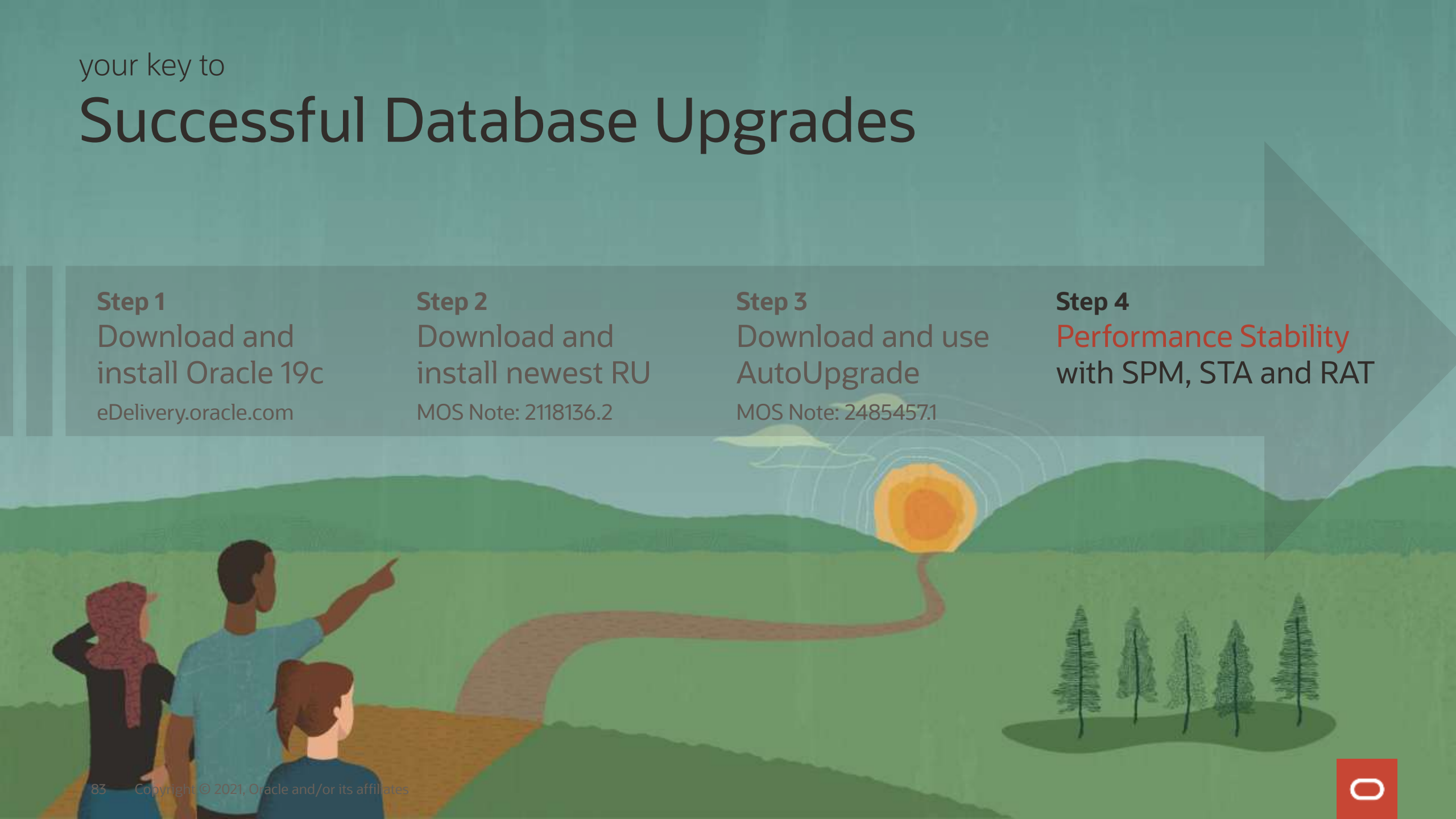
Step 2
Download and install newest RU

MOS Note: 2118136.2

Step 3
Download and use AutoUpgrade

MOS Note: 2485457.1

Step 4
Performance Stability with SPM, STA and RAT



Performance Stability Prescription

SQL Tuning Set | Definition

An SQL Tuning Set (STS) enables you to group SQL statements and related metadata in a single database object, which you can use to meet your tuning goals.

Specifically, SQL tuning sets achieve the following goals:

- *Providing input to the performance tuning advisors*
- *Transporting SQL between databases*

Database 19c SQL Tuning Guide, chapter 23

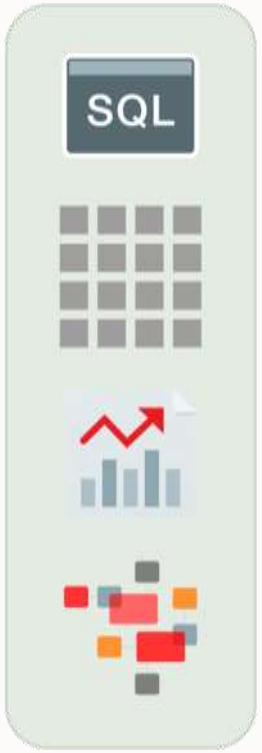
SQL Tuning Set | Definition

SQL statement

Context

Statistics

Plans



SQL Tuning Set | Create

First, create a SQL Tuning Set

```
SQL> BEGIN
  DBMS_SQLSET.CREATE_SQLSET (
    sqlset_name  => 'UPG_STS_1',
    description  => 'For upgrade - from source'
  );
END;
/
```

Pro tip: You can also use DBMS_SQLTUNE to create a SQL Tuning Set

SQL Tuning Set | Capture

Next, capture statements from AWR

```
SQL> DECLARE
   begin_id number;
   end_id number;
   cur sys_refcursor;
BEGIN
   SELECT min(snap_id), max(snap_id) INTO begin_id, end_id
   FROM dba_hist_snapshot;

open cur for
  select value(p) from table(dbms_sqltune.select_workload_repository(
       begin_snap       => begin_id,
       end_snap         => end_id,
       basic_filter     => 'parsing_schema_name not in (''SYS'')',
       ranking_measure1 => 'elapsed_time',
       result_limit     => 5000,
       attribute_list   => 'ALL')) p;

  dbms_sqltune.load_sqlset('UPG_STS_1', cur);

close cur;

END;
/
```

Pro tip: Consider excluding other internal schemas like *DBSNMP*, *ORACLE_OCM*, *LBACSYS*, *WMSYS*, *XDB*, *SYSTEM*

SQL Tuning Set | Capture

Optionally, capture statements from cursor cache

```
SQL> BEGIN
   DBMS_SQLSET.CAPTURE_CURSOR_CACHE_SQLSET(
      sqlset_name     => 'UPG_STS_1',
      time_limit      => 900,
      repeat_interval => 60,
      capture_option  => 'MERGE',
      capture_mode    => DBMS_SQLTUNE.MODE_ACCUMULATE_STATS,
      basic_filter    => 'parsing_schema_name not in (''SYS'')',
      sqlset_owner    => NULL,
      recursive_sql   => 'HAS_RECURSIVE_SQL');
   END;
/
```

Careful - puts load on your system

Pro tip: SQL Tuning Guide shows how to load all statements from a given schema

SQL Tuning Set | License

SQL Tuning Sets can also be accessed by way of database server APIs and command-line interfaces. Usage of any subprograms in the DBMS_SQLSET package to manage SQL Tuning Sets is part of the EE and EE-ES offerings.

In addition, the following subprograms, part of the DBMS_SQLTUNE package, provide an older interface to manage SQL Tuning Sets and are also part of the EE and EE-ES offerings:

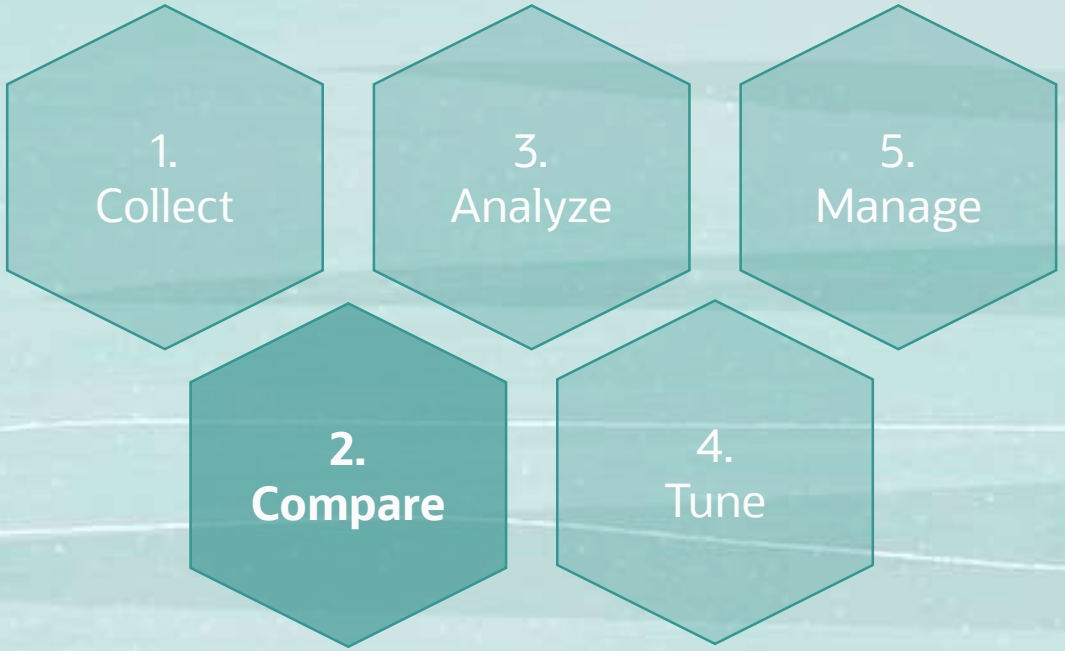
ADD_SQLSET_REFERENCE
CREATE_STGTAB_SQLSET
LOAD_SQLSET
SELECT_CURSOR_CACHE
UNPACK_STGTAB_SQLSET
CAPTURE_CURSOR_CACHE_SQLSET
DELETE_SQLSET
PACK_STGTAB_SQLSET
SELECT_SQLSET
UPDATE_SQLSET
CREATE_SQLSET
DROP_SQLSET
REMOVE_SQLSET_REFERENCE
SELECT_WORKLOAD_REPOSITORY

Database 19c Database Licensing Information User Manual

SQL Tuning Set | Recommendation

Always capture workload data into SQL Tuning Sets

Performance Stability Prescription



AWR | Diff Report

Compare AWR report
from two different periods

1. AWR snapshot
2. Execute workload
3. AWR snapshot

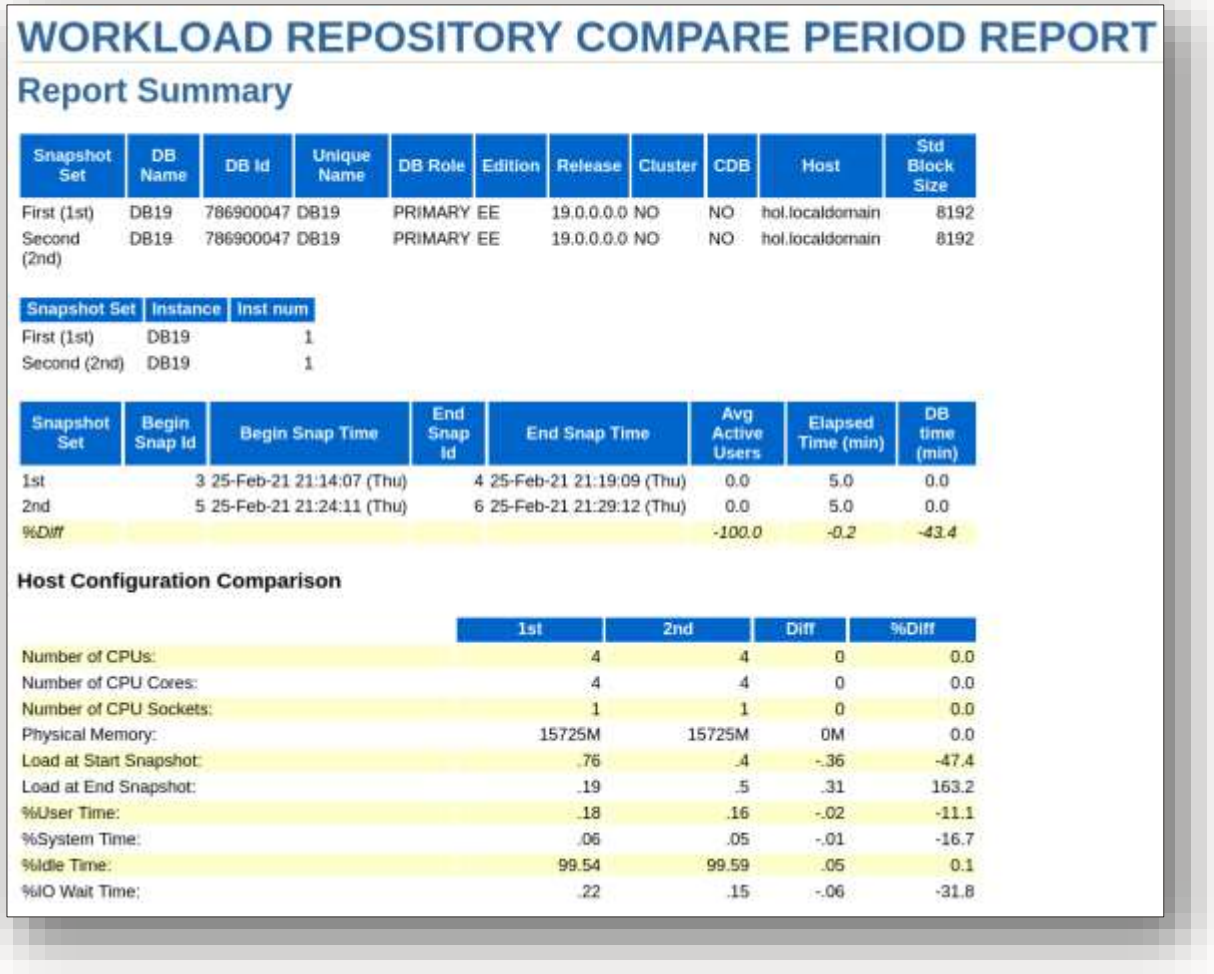
4. Upgrade

5. AWR snapshot
6. Execute workload
7. AWR snapshot

8. Compare

AWR | Diff Report

Use script `awrddrpt.sql`



WORKLOAD REPOSITORY COMPARE PERIOD REPORT

Report Summary

Snapshot Set	DB Name	DB Id	Unique Name	DB Role	Edition	Release	Cluster	CDB	Host	Std Block Size
First (1st)	DB19	786900047	DB19	PRIMARY	EE	19.0.0.0.0	NO	NO	hol.localdomain	8192
Second (2nd)	DB19	786900047	DB19	PRIMARY	EE	19.0.0.0.0	NO	NO	hol.localdomain	8192

Snapshot Set	Instance	Inst num
First (1st)	DB19	1
Second (2nd)	DB19	1

Snapshot Set	Begin Snap Id	Begin Snap Time	End Snap Id	End Snap Time	Avg Active Users	Elapsed Time (min)	DB time (min)
1st	3	25-Feb-21 21:14:07 (Thu)	4	25-Feb-21 21:19:09 (Thu)	0.0	5.0	0.0
2nd	5	25-Feb-21 21:24:11 (Thu)	6	25-Feb-21 21:29:12 (Thu)	0.0	5.0	0.0
%Diff					*-100.0*	*-0.2*	*-43.4*

Host Configuration Comparison

	1st	2nd	Diff	%Diff
Number of CPUs:	4	4	0	0.0
Number of CPU Cores:	4	4	0	0.0
Number of CPU Sockets:	1	1	0	0.0
Physical Memory:	15725M	15725M	0M	0.0
Load at Start Snapshot:	.76	.4	-.36	-47.4
Load at End Snapshot:	.19	.5	.31	163.2
%User Time:	.18	.16	-.02	-11.1
%System Time:	.06	.05	-.01	-16.7
%Idle Time:	99.54	99.59	.05	0.1
%IO Wait Time:	.22	.15	-.06	-31.8

AWR | Diff Report

Use script `awrddrpt.sql`

Top Timed Events

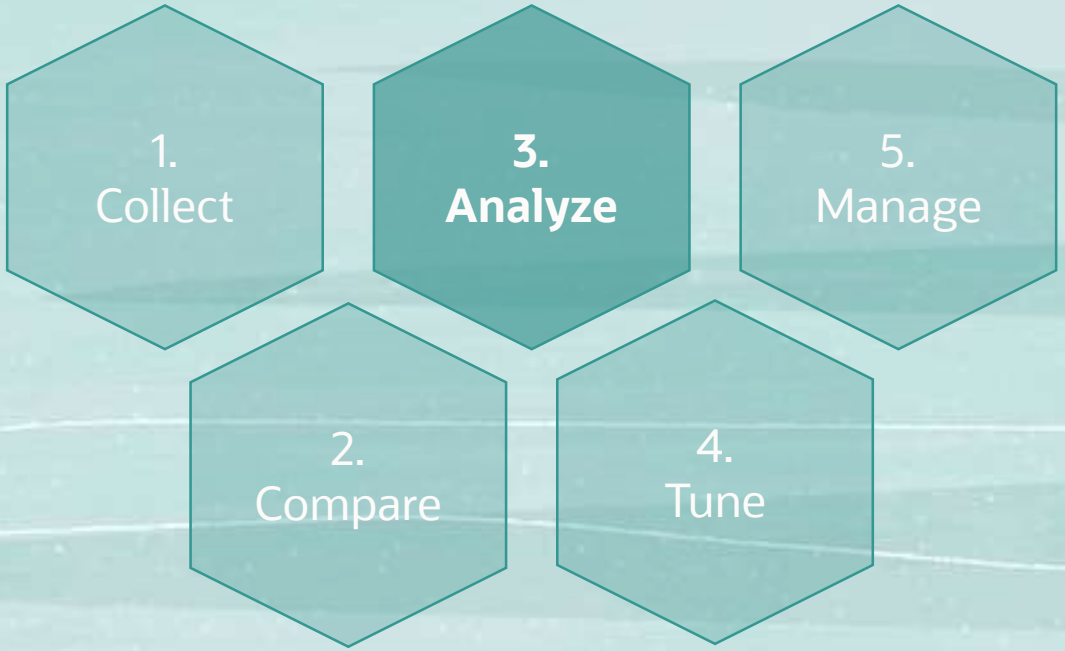
- Events with a "-" did not make the Top list in this set of snapshots, but are displayed for comparison purposes

1st						2nd					
Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time	Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time
CPU time			68,289.05		43.73	db file sequential read	User I/O	22,193,998	114,919.21	5.18	23.17
db file sequential read	User I/O	6,686,953	37,737.81	5.64	24.17	enq: SS - contention	Configuration	3,913	98,997.90	25,299.74	19.96
gc buffer busy	Cluster	12,508,244	23,886.55	1.91	15.30	CPU time			73,786.55		14.88
TCP Socket (KGAS)	Network	680,629	12,514.65	18.39	8.01	row cache lock	Concurrency	73,940	48,472.30	655.56	9.77
db file scattered read	User I/O	1,572,296	4,271.68	2.72	2.74	reliable message	Other	41,148	47,600.87	1,156.82	9.60

Requires Enterprise Edition + Diagnostic pack

Pro tip: For migrations, you can transport AWR data

Performance Stability Prescription



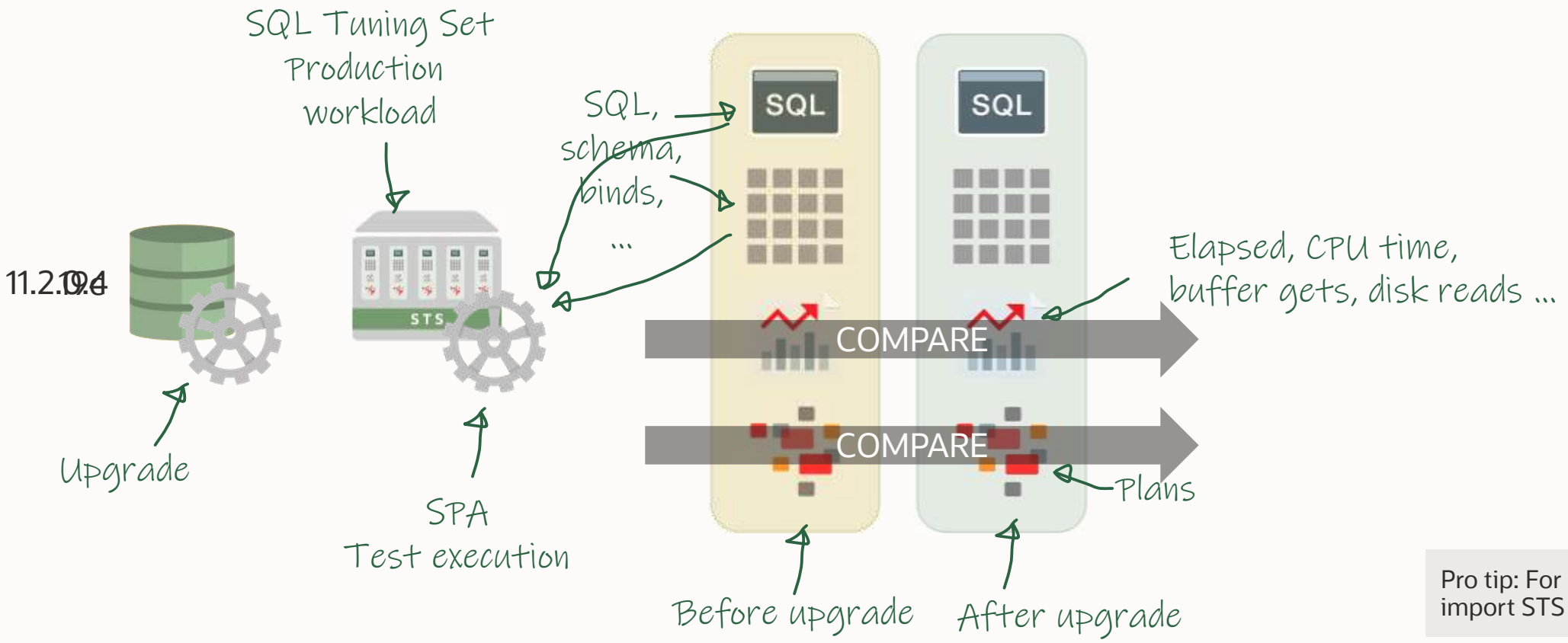
SQL Performance Analyzer | SPA

> *SPA provides fine-grained assessment of environment changes on SQL execution plans and statistics by running the SQL statements both in isolation and serially manner in before-change and after-change environments.*
>
> *SPA functionality is well integrated with existing SQL Tuning Set (STS), SQL Tuning Advisor, and SQL Plan Management functionality.*

Oracle Database Real Application Testing Data Sheet

Requires Enterprise Edition + Real Application Testing

SPA | Concept



Pro tip: For migrations,
import STS into target database

SPA | Regressed Report

Regressed SQL Statements

	SQL ID	Net Impact on Workload (%)	Buffer Gets		Net Impact on SQL (%)	New Plan
			SQL Trial 1	SQL Trial 2		
⇩	3fv28gfu9y0aq	-0.050	26,504	29,573	-11.580	Y
⇩	czzzubf8fjz96	-0.030	1,410	1,981	-40.500	Y

From production workload

From test execution

SPA | Regressed Report

Regressed SQL Statements

	SQL ID	Net Impact on Workload (%)	Buffer Gets		Net Impact on SQL (%)	New Plan
			SQL Trial 1	SQL Trial 2		
⇩	3fv28gfu9y0aq	-0.050	26,504	29,573	-11.580	Y
⇩	czzzubf8fjz96	-0.030	1,410	1,981	-40.500	Y

SPA | Regressed Report

Regressed SQL Statements

	SQL ID	Net Impact on Workload (%)	Buffer Gets		Net Impact on SQL (%)	New Plan
			SQL Trial 1	SQL Trial 2		
⇩	3fv28gfu9y0aq	-0.050	26,504	29,573	-11.580	Y
⇩	czzzubf8fjz96	-0.030	1,410	1,981	-40.500	Y

SQL Details: czzzubf8fjz96

Parsing Schema APPS Execution Frequency 3

▷ SQL Text

```
SELECT /* my_query_21 */ /*+ ORDERED INDEX(t1) USE_HASH(t1) */ 'B' || t2.take_02 take_02, 'B' || t2.take_15
take_15, 'B' || t2.take_08 take_08, 'r' || t3.record_nr price_eur_id,...
```

Single Execution Statistics

	Execution Statistic Name	Net Impact on Workload (%)	Execution Statistic Collected		Net Impact on SQL (%)
			SQL Trial 1	SQL Trial 2	
⇩	Elapsed Time (sec)	-0.240	0.112	0.164	-46.170
⇧	Parse Time (sec)	0.220	0.001	0.001	14.490
⇩	CPU Time (sec)	-0.030	0.108	0.114	-5.040
⇨	User I/O Time (sec)	0.000	0.000	0.000	0.000
⇩	Buffer Gets	-0.030	1,410	1,981	-40.500

SPA | Regressed Report

Regressed SQL Statements

	SQL ID	Net Impact on Workload (%)	Buffer Gets		Net Impact on SQL (%)	New Plan
			SQL Trial 1	SQL Trial 2		
⇩	3fv28gfu9y0aq	-0.050	26,504	29,573	-11.580	Y
⇩	czzzubf8fjz96	-0.030	1,410	1,981	-40.500	Y

Plan Comparison

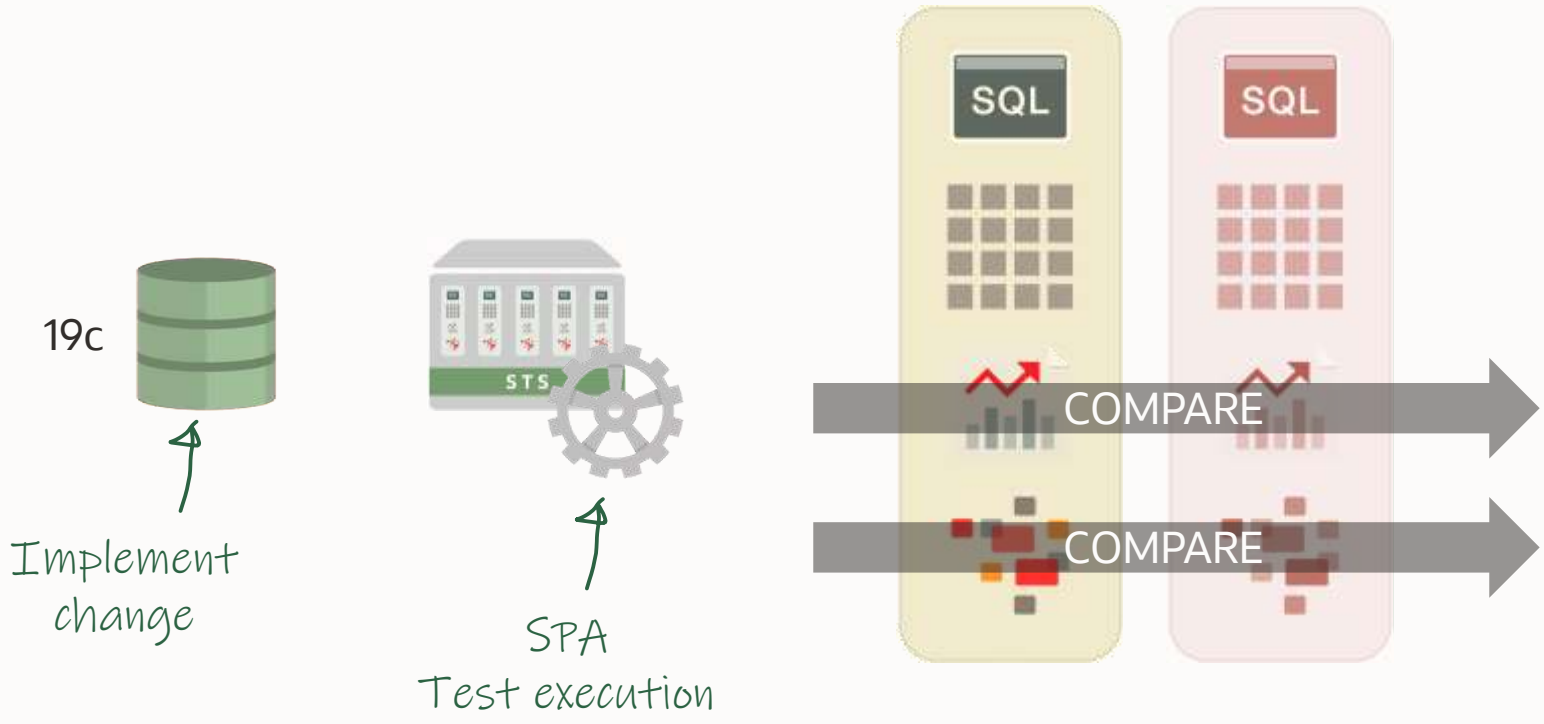
SQL_TRIAL_1353942463446

Plan Hash Value 1165613724

Expand All | Collapse All

Operation	Line ID	Object	Rows	Cost	Predicate
▽ SELECT STATEMENT	0		1	9,830	
▽ HASH GROUP BY	1		1	9,830	
▽ MERGE JOIN	2		1	9,829	
▽ SORT JOIN	3		8	9,795	
▽ HASH JOIN	4		8	9,794	"T1"."PERIOD_CODE"="T4"."FLYER...
INDEX RANGE SCAN	5	APPS.IDX$$_080F0004	1	2	"T4"."EXPORT_LIC_NR"=14659
▽ HASH JOIN	6		14,210	9,792	"T1"."SKU_NR"="T2"."SKU_NR" AN...

SPA | Continuous Improvement



SPA | Regressed Report

Regressed SQL Statements

	SQL ID	Net Impact on Workload (%)	Buffer Gets		Net Impact on SQL (%)	New Plan
			SQL Trial 1	SQL Trial 2		
⇩	3fv28gfu9y0aq	-0.050	26,504	29,573	-11.580	Y
⇩	czzzubf8fjz96	-0.030	1,410	1,981	-40.500	Y

Performance Stability Prescription

1. Collect
2. Compare
3. Analyze
4. **Tune**
5. Manage

SQL Tuning Advisor

> *SQL Tuning Advisor is SQL diagnostic software in the Oracle Database Tuning Pack.*
>
> *...*
>
> *SQL Tuning Advisor is a mechanism for resolving problems related to suboptimally performing SQL statements.*

Database 19c SQL Tuning Guide, chapter 24

SQL Tuning Advisor | Findings

Types of findings:

1. Collection of object statistics
2. Creation of indexes
3. Rewriting SQL statements
4. Creation of SQL profiles

.... and more

Pro tip: SQL Developer has a good interface to SQL Tuning Advisor

Performance Stability Prescription

1. Collect
2. Compare
3. Analyze
4. Tune
5. **Manage**

SQL Plan Management | SPM

> *SQL plan management uses a mechanism called a **SQL plan baseline**, which is a set of accepted plans that the optimizer is allowed to use for a SQL statement.*
>
> *...*
>
> *SQL plan management prevents performance regressions caused by plan changes.*

Database 19c SQL Tuning Guide, chapter 27

SPM | Concept

Repeatable SQL

Filtering:

- Schema
- SQL text
- Module
- Action

Something changed

- New statistics
- New parameters
- Upgrade

```
OPTIMIZER_USE_SQL_PLAN_BASELINES=TRUE
```

No plans in baseline

Plan B

Filter

Plan history

Dedicated part of SQL plan baseline

Plan A is used

Plan A
Accepted

Plan B
Not accepted

Plan History

SPM | Plans

The plans in a SQL plan baseline can be:

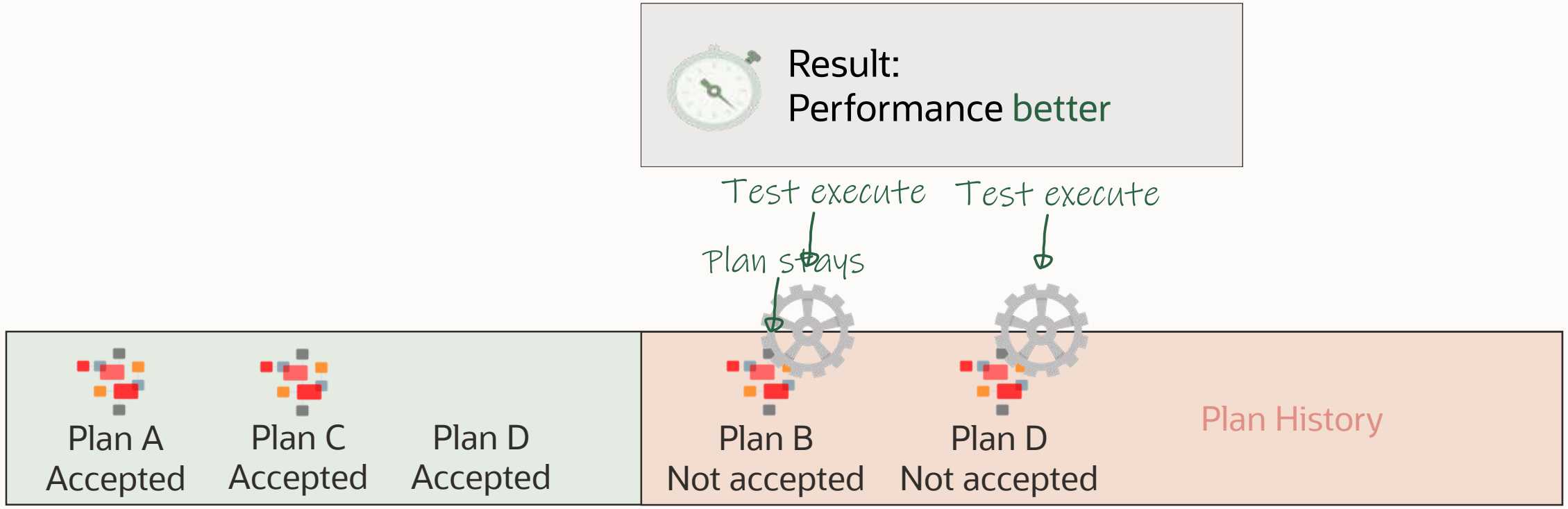
- Enabled
- Accepted
- Fixed

To change status use `DBMS_SPM.ALTER_SQL_PLAN_BASELINE`

You can also prevent plans from getting purged by setting the `autopurge` property.

Pro tip: The *Accepted* attribute can only be set by a test execution

SPM | Evolve



SPM | Evolve

Evolving happens in maintenance task SYS_AUTO_SPM_EVOLVE_TASK

- Part of Automatic SQL Tuning Task

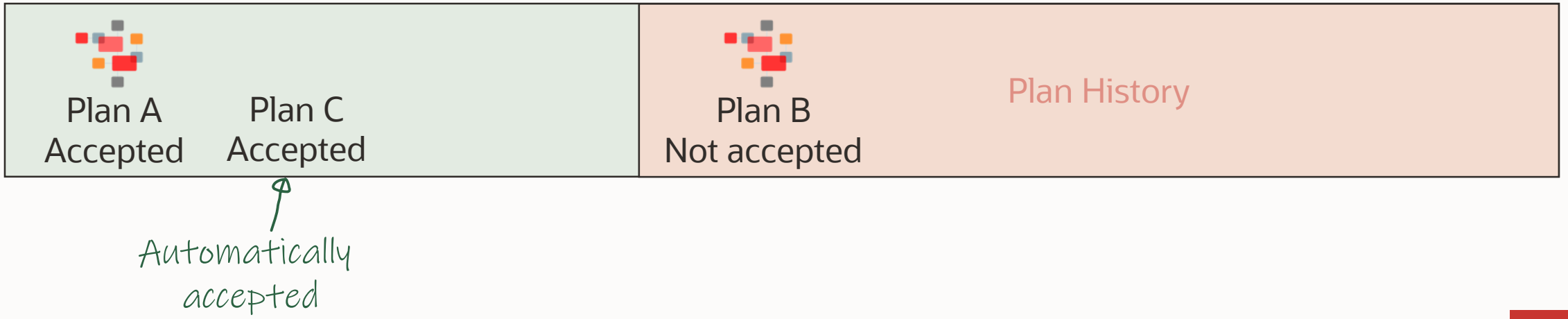
You decide whether recommendations are implemented automatically

```
SQL> BEGIN
   DBMS_SPM.SET_EVOLVE_TASK_PARAMETER(
      parameter => 'accept_plans',
      value     => 'true');
   END;
/
```

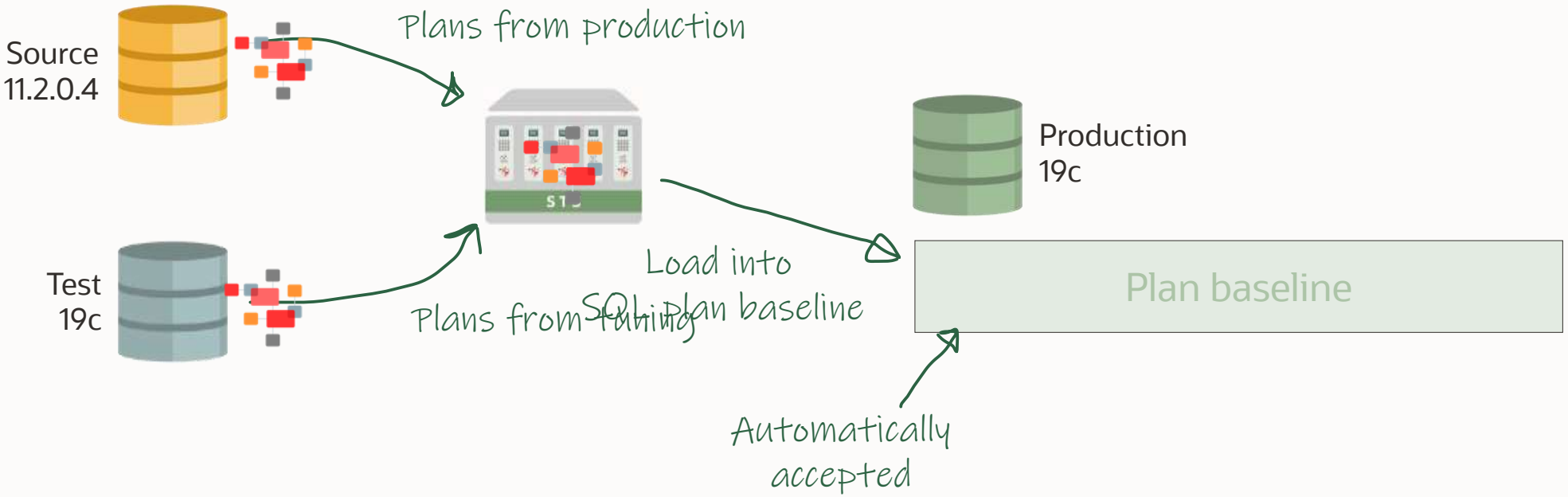
You can evolve plans manually

SPM | Load from STS


```
SQL> DECLARE
       cnt number;
   BEGIN
       cnt := DBMS_SPM.LOAD_PLANS_FROM_SQLSET('UPG_STS_1');
   END;
/
```



SPM | Use Case



SPM | Use Case

Plan baseline

```
SQL> DECLARE
  plans_loaded NUMBER;
  filter VARCHAR2(255);
BEGIN
  filter := 'sql_id=''czzzubf8fjz96'' AND plan_hash_value=''1165613724''';

  plans_loaded := DBMS_SPM.LOAD_PLANS_FROM_SQLSET (
    sqlset_name  => 'UPG_STS_1',
    basic_filter => filter
  );
END;
/
```

Pro tip: The function `LOAD_PLANS_FROM_SQLSET` can also *fix* the plans

Performance Stability Prescription

1. Collect
2. Compare
3. Analyze
4. Tune
5. Manage

Agenda Day 1

- 13:00-13:15h Welcome
- 13:15-13:45h Release and Patching Strategy
- 13:45-14:45h Become an Upgrade expert in 60 minutes

- 14:45-15:00h Quickly provision your environments before BREAK

- 15:00-15:15h Checkout the environments
- 15:15-17:00h LAB DAY1:
  Generate load, upgrade, performance evaluation

Accessing the lab

1. You need your Oracle account email to logon

   If you don't have one yet, please create one:
   https://profile.oracle.com/myprofile/account/create-account.jspx

Accessing the lab

2. Go to the LiveLabs page we created for this workshop

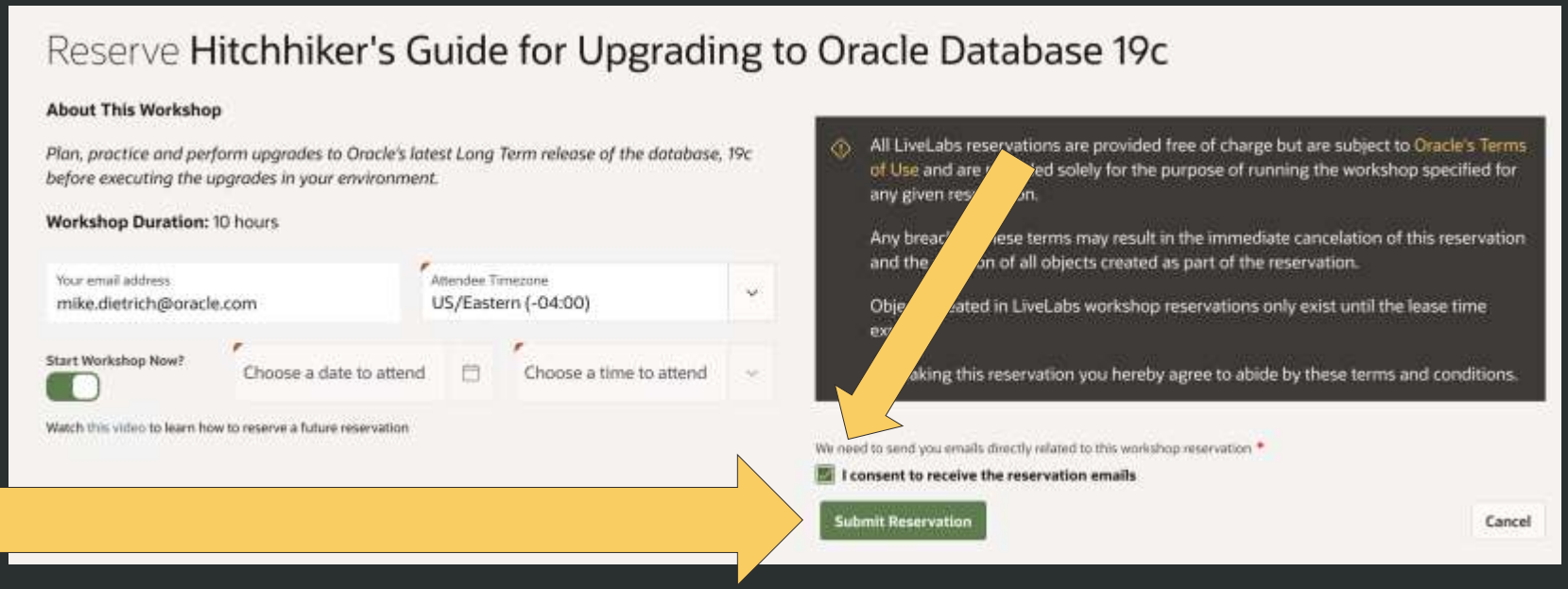
https://bit.ly/3xoIURX

Accessing the lab

3. Click on RESERVE WORKSHOP ON LIVELABS

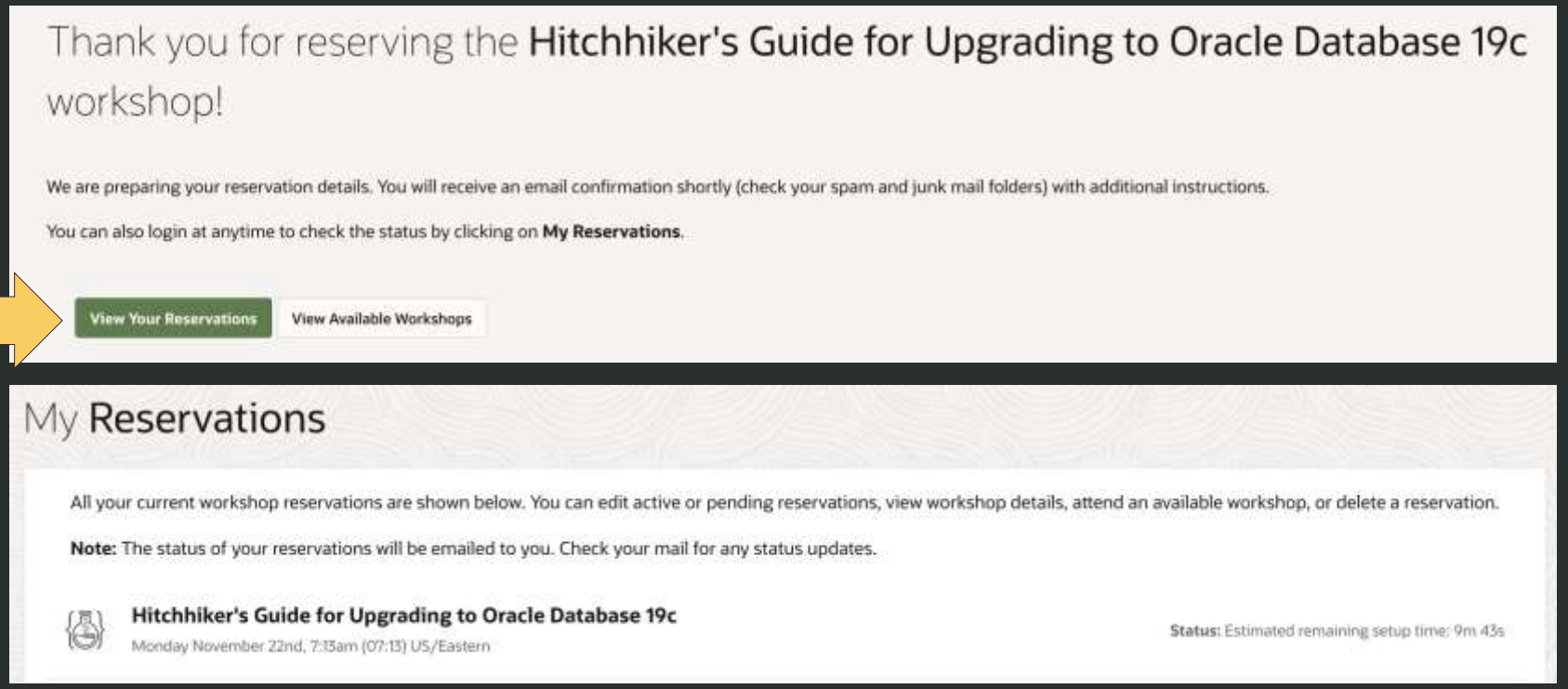
Accessing the lab

4. Leave the defaults, click the CONSENT and then click SUBMIT RESERVATION



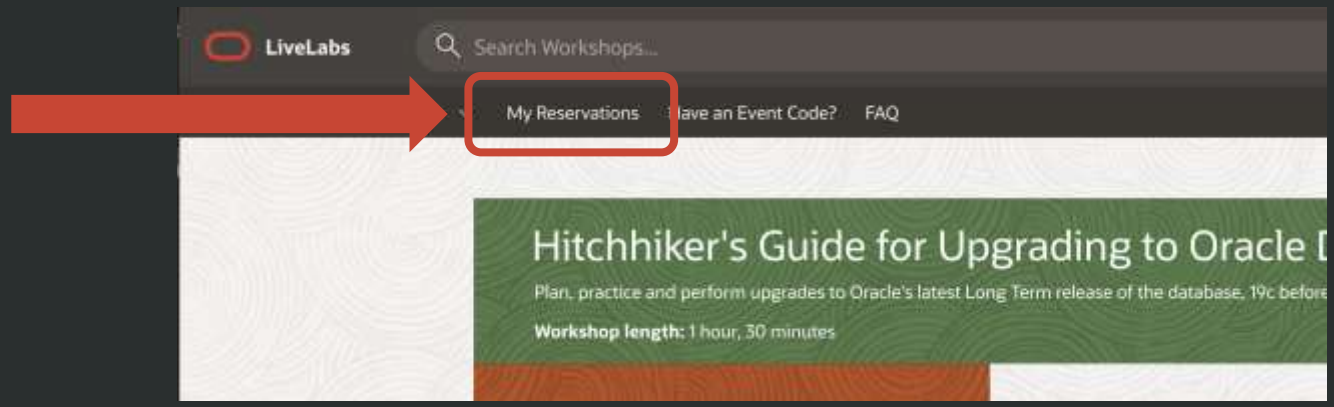
Accessing the lab

5. Now we need to wait 5-10 minutes before we can start



Accessing the lab

6. Click on "My Reservations"



https://bit.ly/3xoIURX

Agenda Day 1

- 13:00-13:15h Welcome
- 13:15-13:45h Release and Patching Strategy
- 13:45-14:45h Become an Upgrade expert in 60 minutes

- 14:45-15:00h Break

- 15:00-15:15h Checkout the environments
- 15:15-17:00h LAB DAY1:
  Generate load, upgrade, performance evaluation

Agenda Day 1

- 13:00-13:15h Welcome
- 13:15-13:45h Release and Patching Strategy
- 13:45-14:45h Become an Upgrade expert in 60 minutes

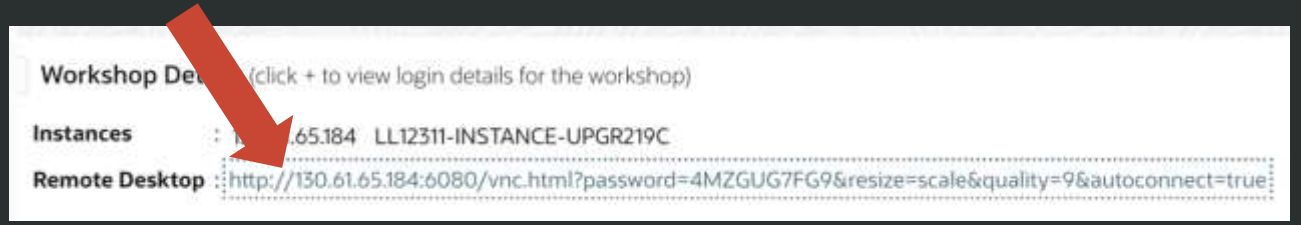
- 14:45-15:00h Break

- 15:00-15:15h Checkout the environments
- 15:15-17:00h LAB DAY1:
  Generate load, upgrade, performance evaluation

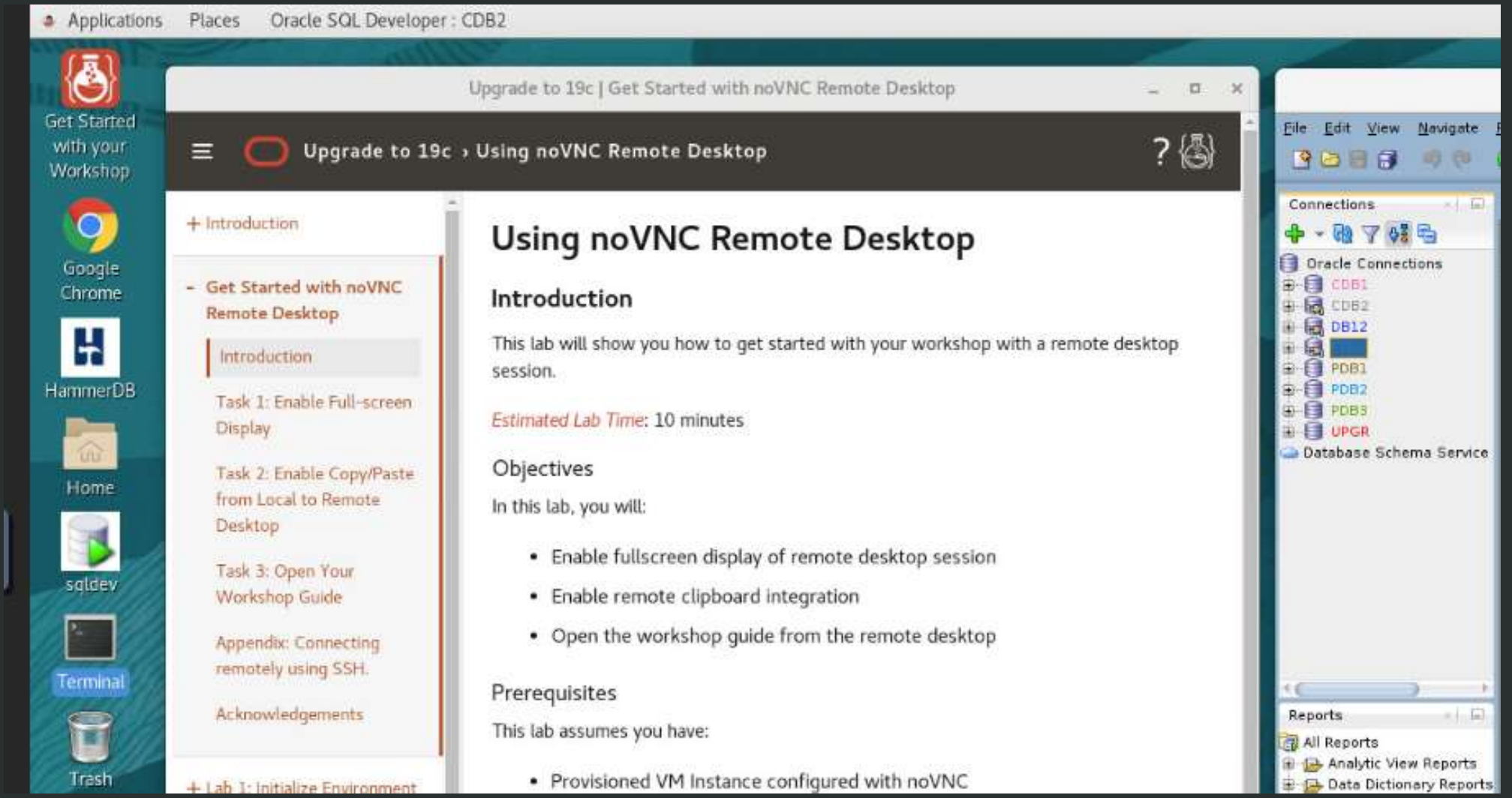
Accessing the lab

7. Launch the workshop

8. Click on the link and it opens a noVNC session

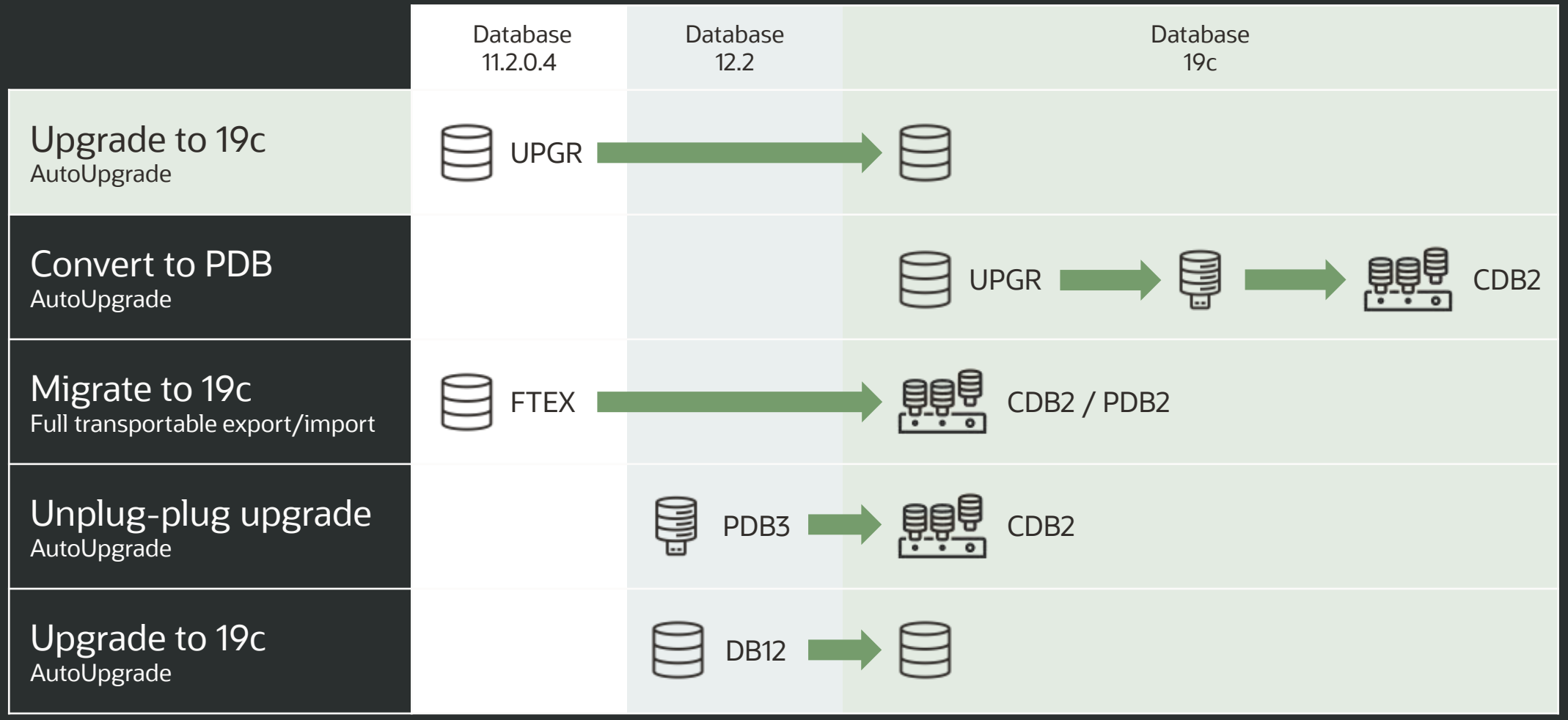


In the Workshop



Introduction to the Lab

Overview – Start with LAB 2



Overview – Setting the Environments

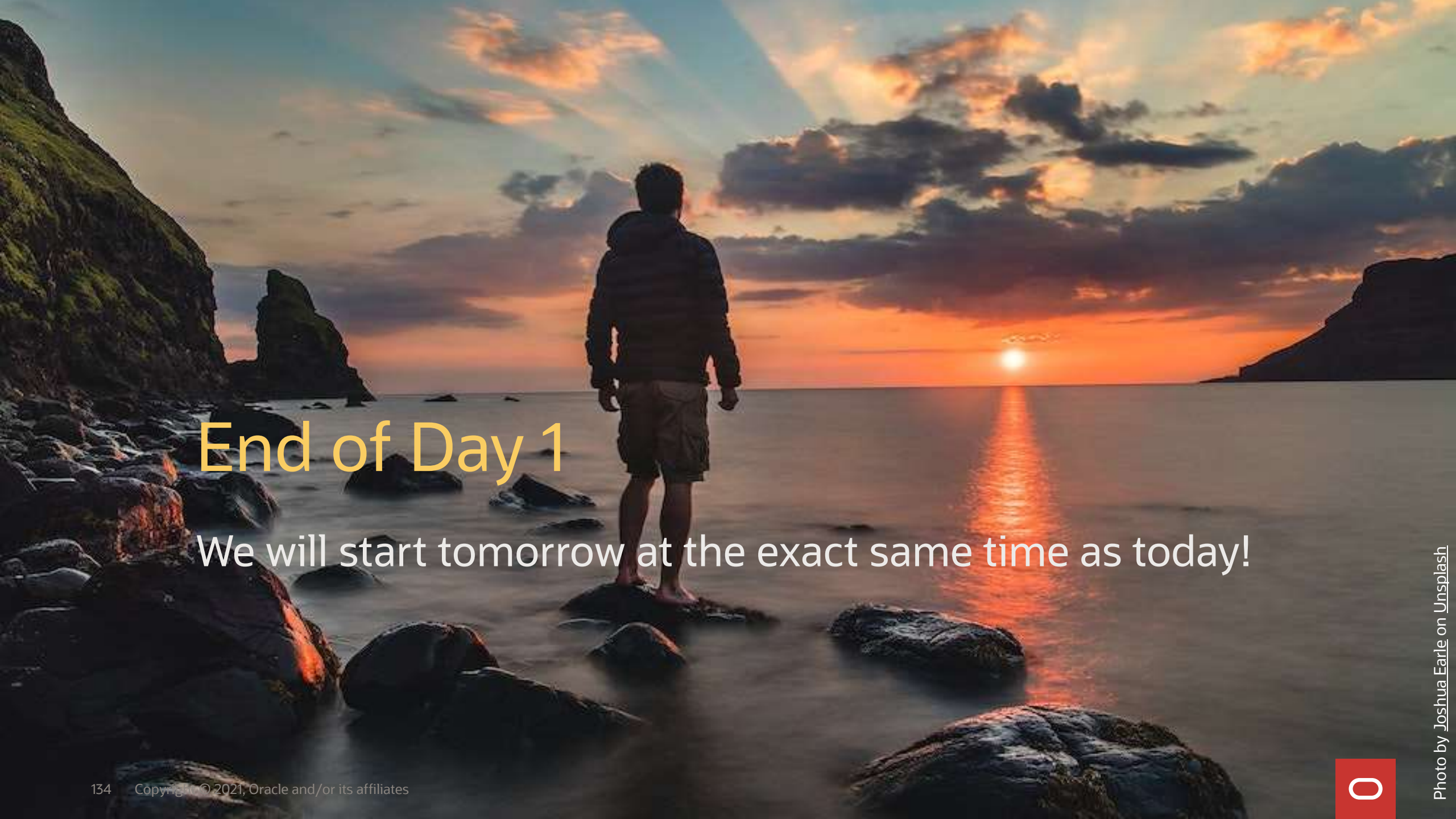
	Database	PDBs	Set environment
Oracle Database 11.2.0.4 /u01/app/oracle/product/11.2.0.4	UPGR		$> . upgr
	FTEX		$> . ftex
Oracle Database 12.2.0.1 /u01/app/oracle/product/12.2.0.1	DB12		$> . db12
	CDB1	PDB3	$> . cdb1
Oracle Database 19 /u01/app/oracle/product/19	CDB2	*(none)*	$> . cdb2

Agenda Day 1

- 13:00-13:15h Welcome
- 13:15-13:45h Release and Patching Strategy
- 13:45-14:45h Become an Upgrade expert in 60 minutes

- 14:45-15:00h Break

- 15:00-15:15h Checkout the environments
- 15:15-17:00h LAB DAY1:
  Generate load, upgrade, performance evaluation



End of Day 1

We will start tomorrow at the exact same time as today!

Photo by Joshua Earle on Unsplash

Upgrade to Oracle Database 19c and Ensure Performance Stability

Virtual Hands-on Workshop

Photo by Robert Bock on Unsplash

Agenda Day 2

- 13:00-13:15h Review of Day 1 and Results
- 13:15-14:45h Many ways to plugin to Multitenant
- 14:45-15:00h Break
- 15:15-16:45h LAB DAY2:
  Plugin. Unplug/Plug/Upgrade. Migration
- 16:45-17:00h Wrap Up, Review and Goodbye

Agenda Day 2

- 13:00-13:15h Review of Day 1 and Results
- 13:15-14:45h Many ways to plugin to Multitenant
- 14:45-15:00h Break
- 15:15-16:45h LAB DAY2:
  Plugin. Unplug/Plug/Upgrade. Migration
- 16:45-17:00h Wrap Up, Review and Goodbye

Multitenant Migration

Photo by Beki Dönmez on Unsplash

Oracle Multitenant

Architecture

Multitenant | Support

Desupport of Non-CDB Oracle Databases

Starting with Oracle Database 21c, installation of non-CDB Oracle Database architecture is no longer supported.

The non-CDB architecture was deprecated in Oracle Database 12c. It is desupported in Oracle Database 21c.

Database 21c, Upgrade Guide, chapter 10

Multitenant | Support

What does this mean?

1. Oracle Database 19c is the last release to support non-CDB architecture

2. After upgrade to Oracle Database 21c or beyond, you must convert to the mulititenant architecture

Pro tip: For further details see Release Schedule of Current Database Releases (Doc ID 742060.1)

Multitenant | License

Included in all offerings (SE2/EE)

12.1.0.2 12.2.0.1 18c	Single tenant Max. 1 PDB
19c 21c	Multitenant Max. 3 PDB

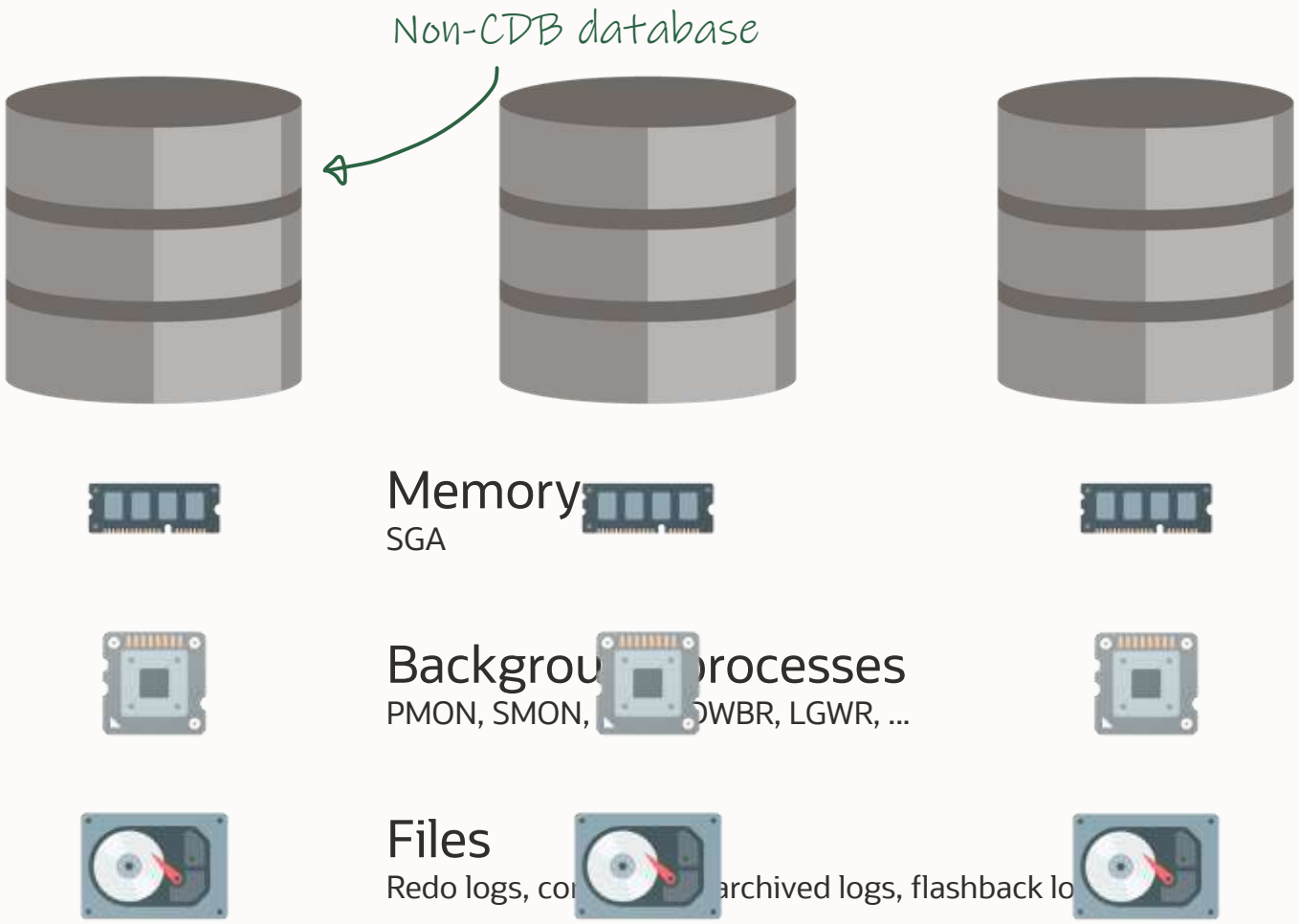
```
SQL> alter system set max_pdbs=3;
```

Multitenant | License

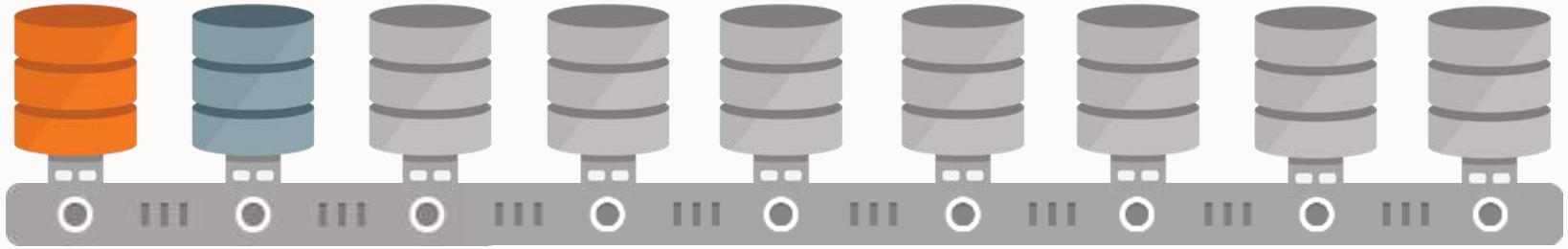
Included in Multitenant Option (EE only)

On-prem Enterprise Edition	252 PDBs Extra cost
Engineered Systems Exadata / ODA	4096 PDBs Extra cost
ExaCS DBCS EE-HP DBCS EE-EP	4096 PDBs Included

Multitenant | Architecture



Multitenant | Architecture



Container database

Memory

SGA

Background processes

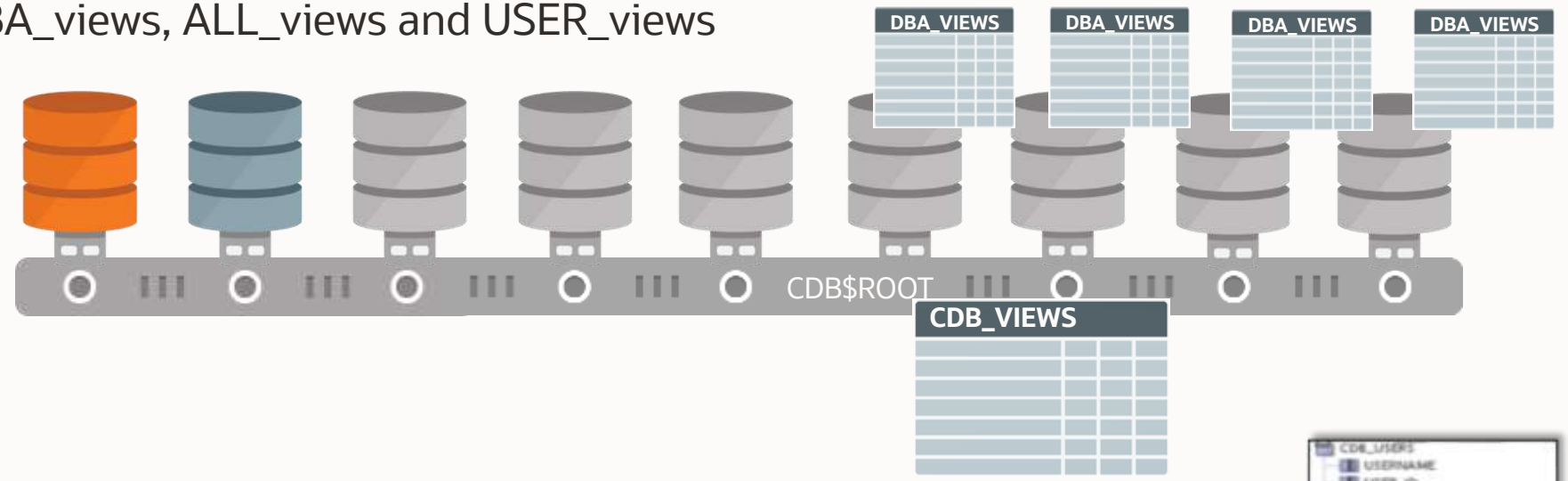
PMON, SMON, CKPT, DWBR, LGWR, ...

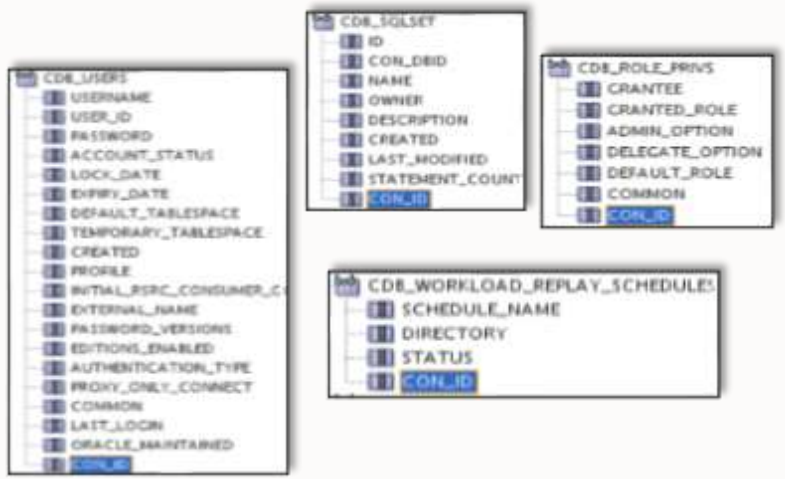
Files

Redo logs, control files, archived logs, flashback logs, ...

Multitenant | Dictionary

CDB_views and
DBA_views, ALL_views and USER_views



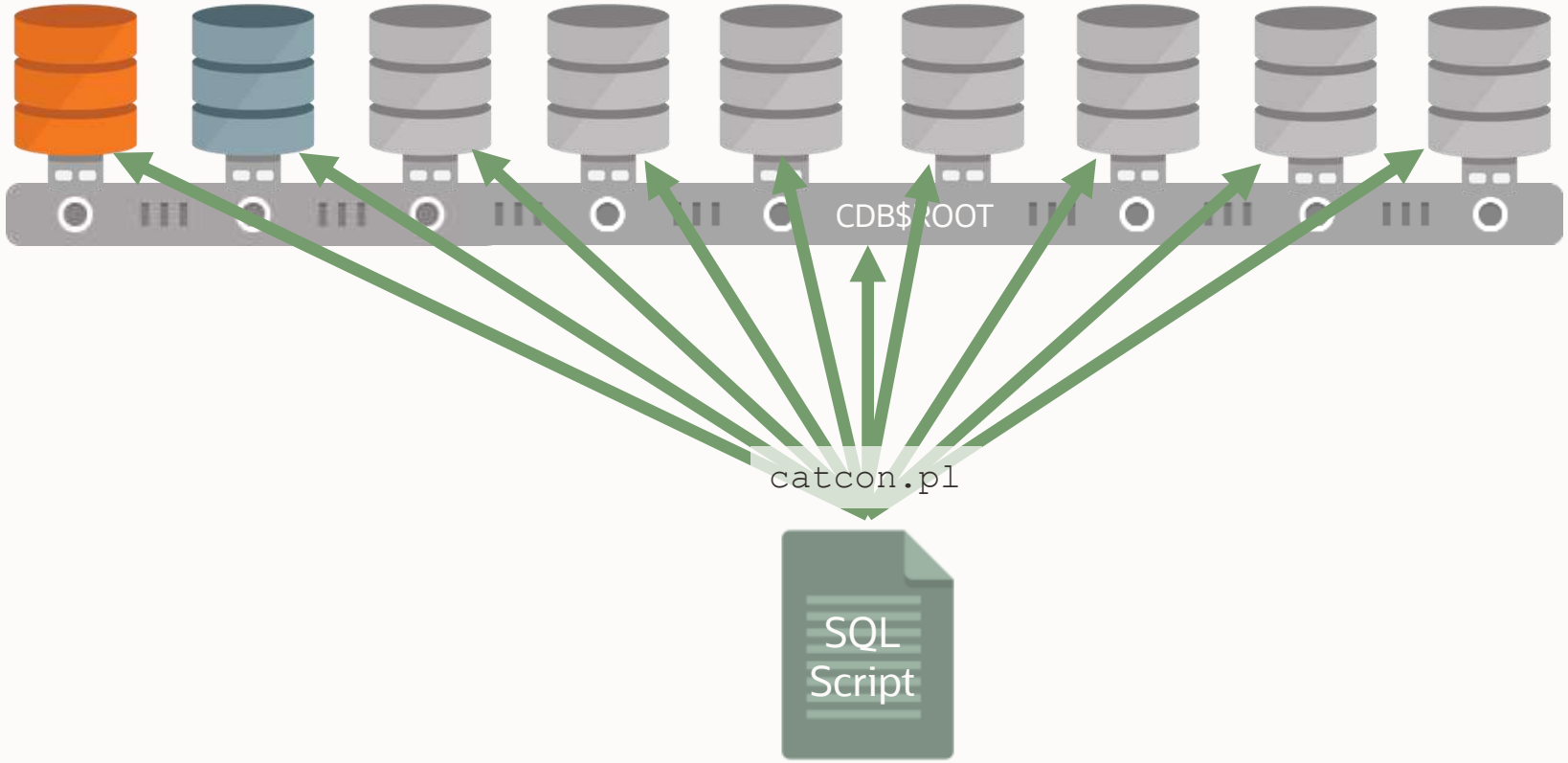


- CON_ID defines ownership / visibility
  - CON_ID: 0 non-CDB
  - CON_ID: 1 CDB$ROOT
  - CON_ID: 2 PDB$SEED
  - CON_ID: 3..n PDB

Multitenant | Script Execution

catcon.pl

- MOS Note: 1932340.1 - How to execute sql scripts in Multitenant environment (catcon.pl)



How to

MIGRATE

to multitenant architecture

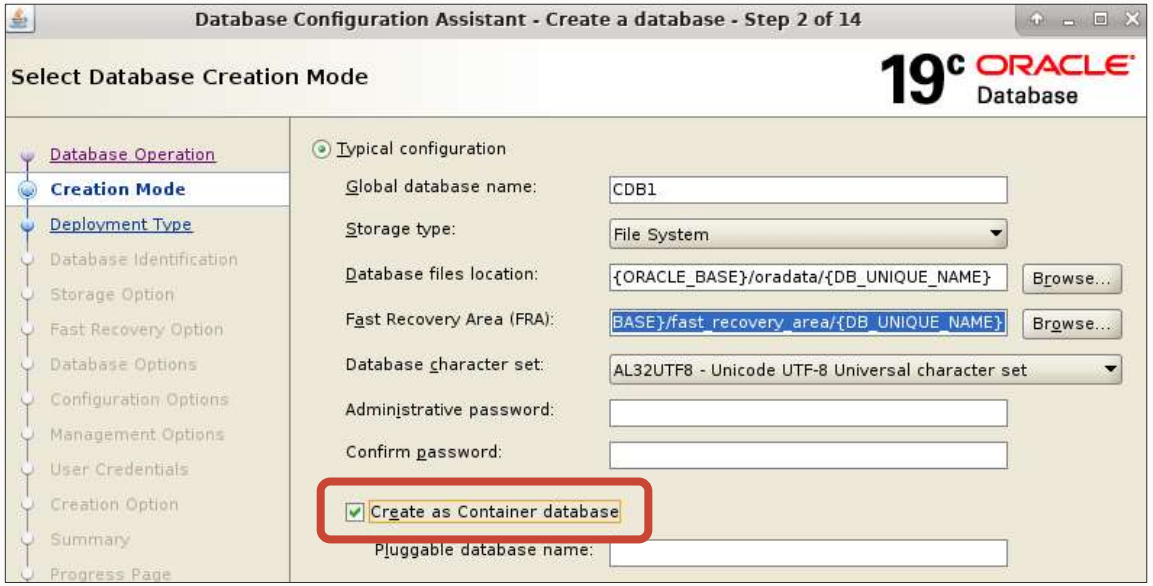
1. **Create**
2. Upgrade
3. Plug In
4. Convert

CDB | Creation

CREATE DATABASE statement

```
SQL> create database "CDB1"
     ...
     enable pluggable database
     ...
```

DBCA



CDB | Character Sets

Recommendation

- Use AL32UTF8
  National character set AL16UTF6

- Allows use of different character sets in PDB
  As of Oracle Database 12.2

CDB | Undo Mode

Recommendation

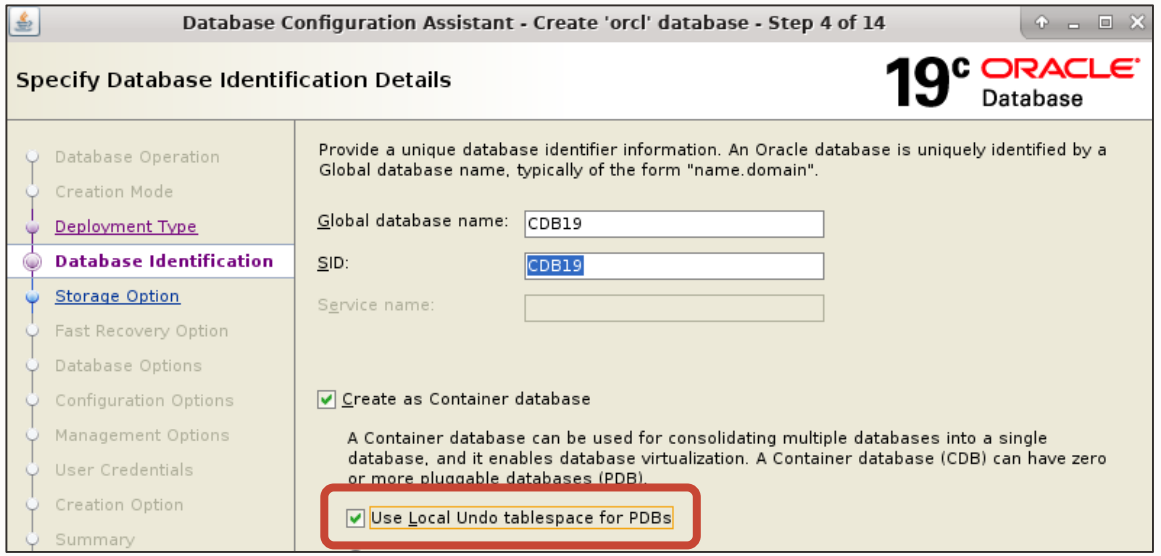
- Use Local Undo mode
- Each PDB has its own UNDO tablespace

```
SQL> startup upgrade
SQL> alter database local undo on;
```

Allows use of

- Flashback Pluggable Database
- Hot Cloning

... and more



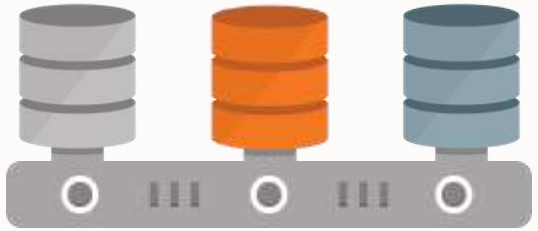
CDB | Components

CDB$ROOT must be a superset of all PDBs

CATALOG
CATPROC
XDB
OWM

CATALOG
CATPROC
XDB
OLS

CATALOG
CATPROC
XDB
SPATIAL



CATALOG
CATPROC
XDB
OWM
OLS
SPATIAL

CDB | Components

Recommendation

1. Install as many components as required

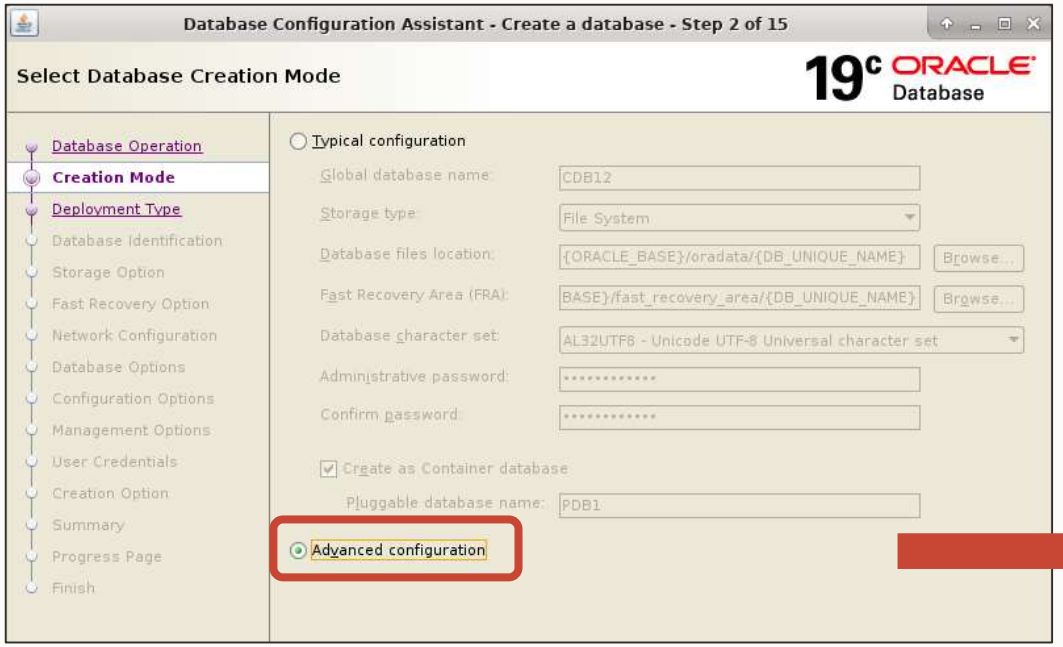
2. But no more than that

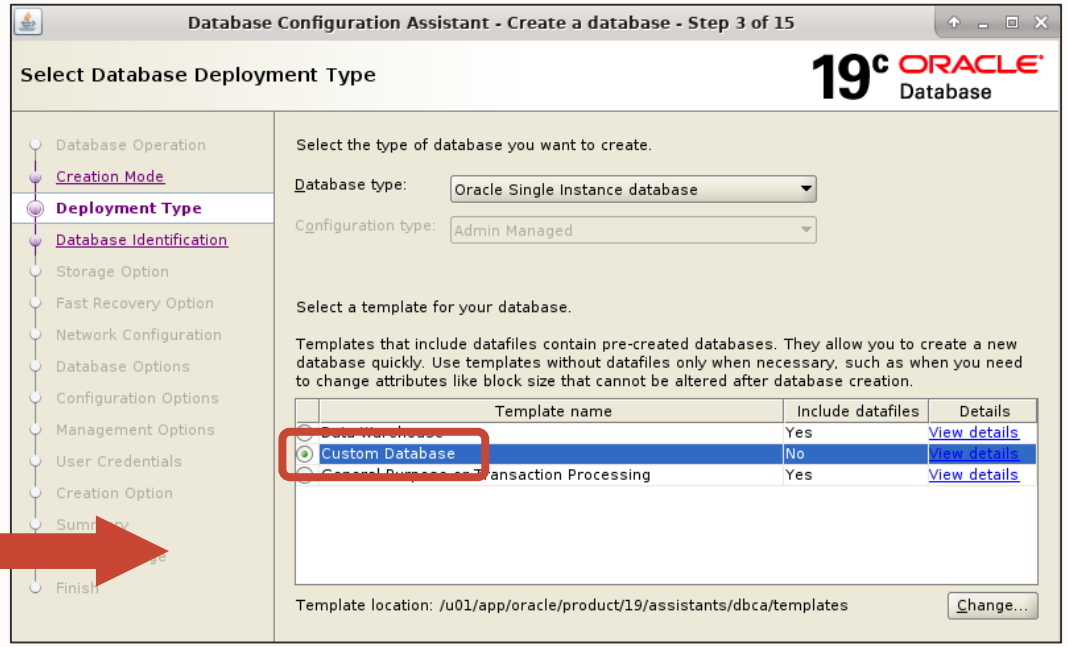
Number of components have big effect on upgrade duration

Components (e.g., JAVAVM) may require patch regular activity

CDB | Components

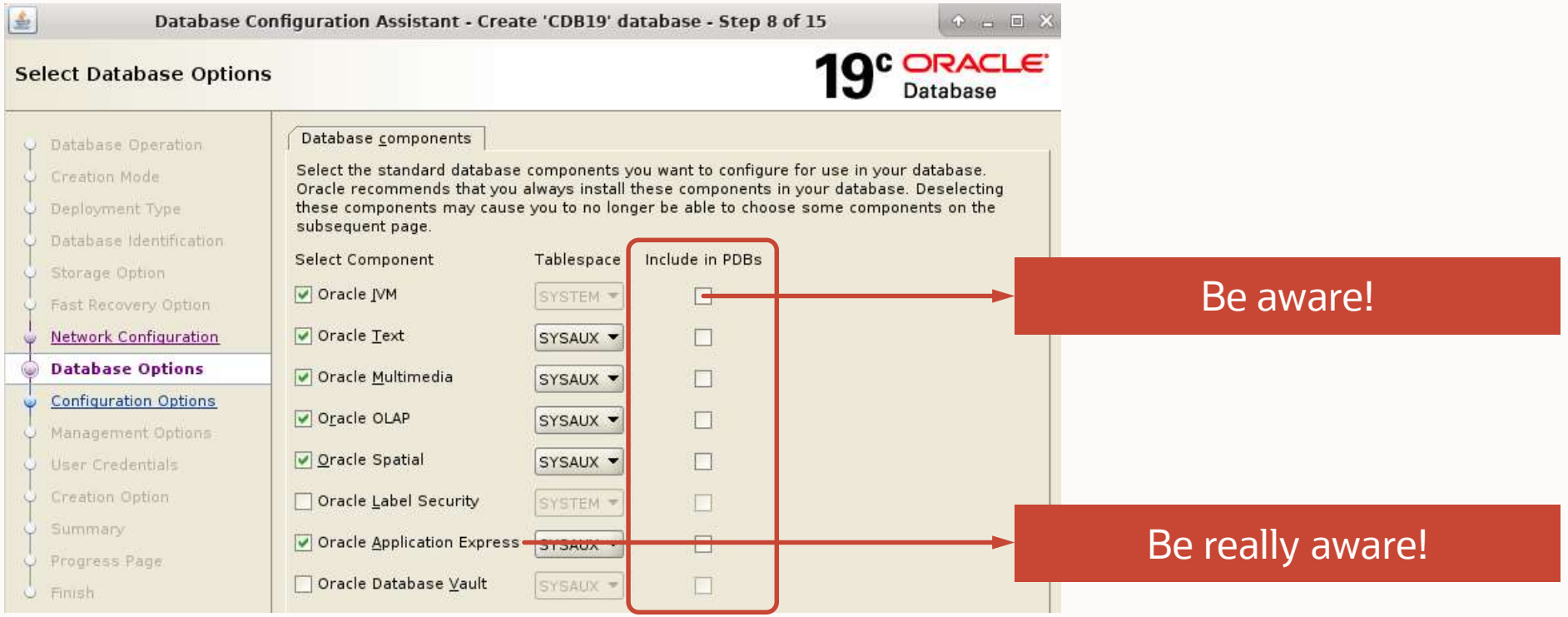
- Create CDBs with fewer components
- DBCA and custom mode mode creation





CDB | Components

Only *Advanced Configuration* > *Custom Database* gives you



CDB | Components

Add or remove components

- Blog post

Component Clean Up Series

- Remove and Clean Up Components from Oracle 11.2 – 12.2 – General Guidelines and Precautions
- APEX – Oracle Application Express Clean Up
- OWM – Oracle Workspace Manager Clean Up
- DV – Oracle Database Vault Clean Up
- OLS – Oracle Label Security Clean Up
- SDO – Oracle Spatial Data Option Clean Up
- CONTEXT – Oracle Text Clean Up
- ORDIM – Oracle Multi Media Clean Up
- XOQ – Oracle OLAP API Clean Up
- APS – Oracle OLAP Analytical Workspace Clean Up
- AMD – Oracle OLAP Catalog Clean Up
- OWB – Oracle Warehouse Builder Clean Up
- EXF/RUL – Oracle Expression Filters and Rules Manager Clean Up
- EM – Enterprise Manager Database Control Clean Up
- JAVAVM/XML – Oracle Java Virtual Machine and XDK Clean Up
- XDB – Oracle XML Database Clean Up

Photo by Markus Spiske on Unsplash

CDB | Compatible

Recommendation

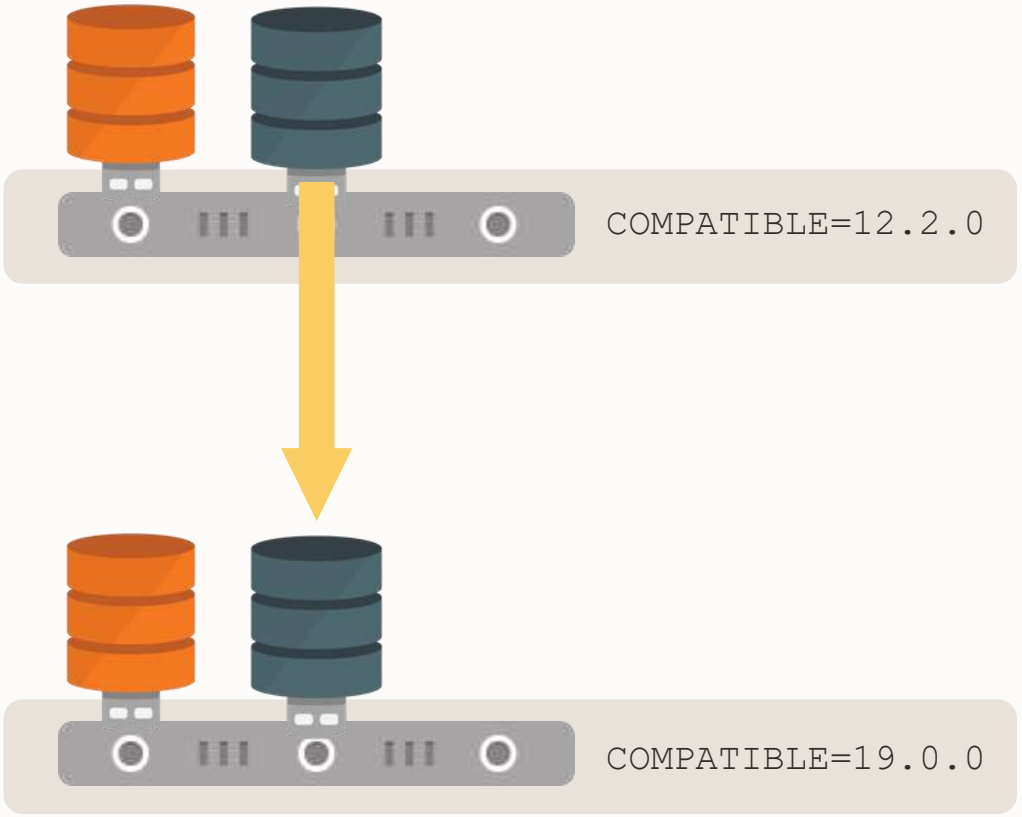
- Always use default of a given version
  - Example `19.0.0`
  - Always use three digits only

- Should you change `COMPATIBLE` after applying a Release Update?
  - Example `19.10.0`
  - Never

CDB | Silent Compatible Change

Be aware

- When you unplug and plugin a PDB, it will adopt `COMPATIBLE` of the CDB automatically
- Changing `COMPATIBLE` will prevent re-plug and downgrade



CDB | Additional Information

Blog posts:

- https://mikedietrichde.com/2018/08/08/creating-cdbs-non-cdbs-with-less-options/
- https://mikedietrichde.com/2017/07/11/always-create-custom-database/
- https://mikedietrichde.com/2017/07/26/remove-clean-components-oracle-11-2-12-2/

How to

MIGRATE

to multitenant architecture

1. Create
2. **Upgrade**
3. Plug In
4. Convert

Migrate | Upgrade First

Recommendation

1. Upgrade (if needed)

2. Convert to PDB

Fallback: Upgrade can be reverted

Convert is irreversible, re-runnable from 12.2

Pro tip: Plugging in a PDB makes changes to the file headers of the datafiles that cannot be undone - even by flashback database

How to

MIGRATE

to multitenant architecture

1. Create
2. Upgrade
3. **Plug In**
4. Convert

Plug In | Compatibility Check

Is my database compatible with this CDB?

1. In source, generate manifest file

```
SQL> exec dbms_pdb.describe('/tmp/DB19.xml');
```

2. In CDB, check compatibility

```
set serveroutput on

BEGIN
   IF dbms_pdb.check_plug_compatibility('/tmp/DB19.xml') THEN
      dbms_output.put_line('PDB compatible? ==> Yes');
   ELSE
      dbms_output.put_line('PDB compatible? ==> No');
   END IF;
END;
/
```

Pro tip: You can generate a manifest file of a remote database via a database link

Plug In | Compatibility Check

3. Always check the details

```
SQL> select type, message
     from   pdb_plug_in_violations
     where  name='DB19' and status<>'RESOLVED';


TYPE       MESSAGE
_____________________________________________________________________________________________________
ERROR      '19.9.0.0.0 Release_Update' is installed in the CDB but no release updates are installed in the PDB
ERROR      DBRU bundle patch 201020: Not installed in the CDB but installed in the PDB
ERROR      PDB's version does not match CDB's version: PDB's version 12.2.0.1.0. CDB's version 19.0.0.0.0.
WARNING    CDB parameter compatible mismatch: Previous '12.2.0' Current '19.0.0'
WARNING    PDB plugged in is a non-CDB, requires noncdb_to_pdb.sql be run.
```

Pro tip: Errors will prevent you from opening the PDB in unrestricted mode

Plug In | Compatibility Check

What is a manifest file

- Data files
- Components
- Parameters
- Services
- Patch level
- Time zone

  ... and more

```
<?xml version="1.0" encoding="UTF-8"?>
<PDB>
  <xmlversion>1</xmlversion>
  <pdbname>DB12</pdbname>
  <cid>0</cid>
  <byteorder>1</byteorder>
  <vsn>203424000</vsn>
  <vsns>
    <vsnnum>12.2.0.1.0</vsnnum>
    <cdbcompt>12.2.0.0.0</cdbcompt>
    <pdbcompt>12.2.0.0.0</pdbcompt>
    <vsnlibnum>0.0.0.0.24</vsnlibnum>
    <vsnsql>24</vsnsql>
    <vsnbsv>8.0.0.0.0</vsnbsv>
  </vsns>
  <dbid>1852833295</dbid>
  <ncdb2pdb>1</ncdb2pdb>
  <cdbid>1852833295</cdbid>
  <guid>86D5DC2587337002E0532AB2A8C0A57C</guid>
  <uscnbas>4437941</uscnbas>
  <uscnwrp>0</uscnwrp>
  <undoscn>8</undoscn>
  <rdba>4194824</rdba>
  <tablespace>
    <name>SYSTEM</name>
    <type>0</type>
    <tsn>0</tsn>
    <status>1</status>
    <issft>0</issft>
    <isnft>0</isnft>
    <encts>0</encts>
    <flags>0</flags>
    <bmunitsize>8</bmunitsize>
    <file>
      <path>/u02/oradata/DB12/system01.dbf</path>
      <afn>1</afn>
      <rfn>1</rfn>
```

Plug In | Compatibility Check

- Optional
- Possible when source database is 12.1 or newer
- Enables you to see which plug-in violations will occur
- If you plan on renaming the PDB, check the new name

```
dbms_pdb.check_plug_compatibility('/tmp/DB19.xml', 'SALES')
```

Pro tip: Default PDB name in manifest file is `DB_NAME`

Plug In | Create PDB

1. Restart database in read-only mode

```
SQL> shutdown immediate
SQL> startup mount
SQL> alter database open read only;
```

2. Generate manifest file and shut down

```
SQL> exec dbms_pdb.describe('/tmp/DB19.xml');
SQL> shutdown immediate;
```

3. In CDB, create PDB from manifest file

```
SQL> create pluggable database DB19
     using '/tmp/DB19.xml' nocopy tempfile reuse;
```

Pro tip: `CREATE PLUGGABLE DATABASE` has many options

How to

MIGRATE

to multitenant architecture

1. Create
2. Upgrade
3. Plug In
4. **Convert**

Convert | Create PDB

1. Open PDB

```
SQL> alter pluggable database DB19 open;
SQL> alter session set container=DB19;
```

2. Convert and restart

```
SQL> @?/rdbms/admin/noncdb_to_pdb.sql
```

3. Restart PDB

```
SQL> alter pluggable database DB19 close;
SQL> alter pluggable database DB19 open;
```

Pro tip: As of 12.2 the script `noncdb_to_pdb.sql` is re-runnable

Convert | Create PDB

4. Check plug-in violations

```
SQL> select type, message
     from   pdb_plug_in_violations
     where  name='DB19' and status<>'RESOLVED';
```

5. Ensure PDB is open `READ WRITE` and unrestricted

```
SQL> select open_mode, restricted from v$pdbs;
```

6. Configure PDB to auto-start

```
SQL> alter pluggable database DB19 save state;
```

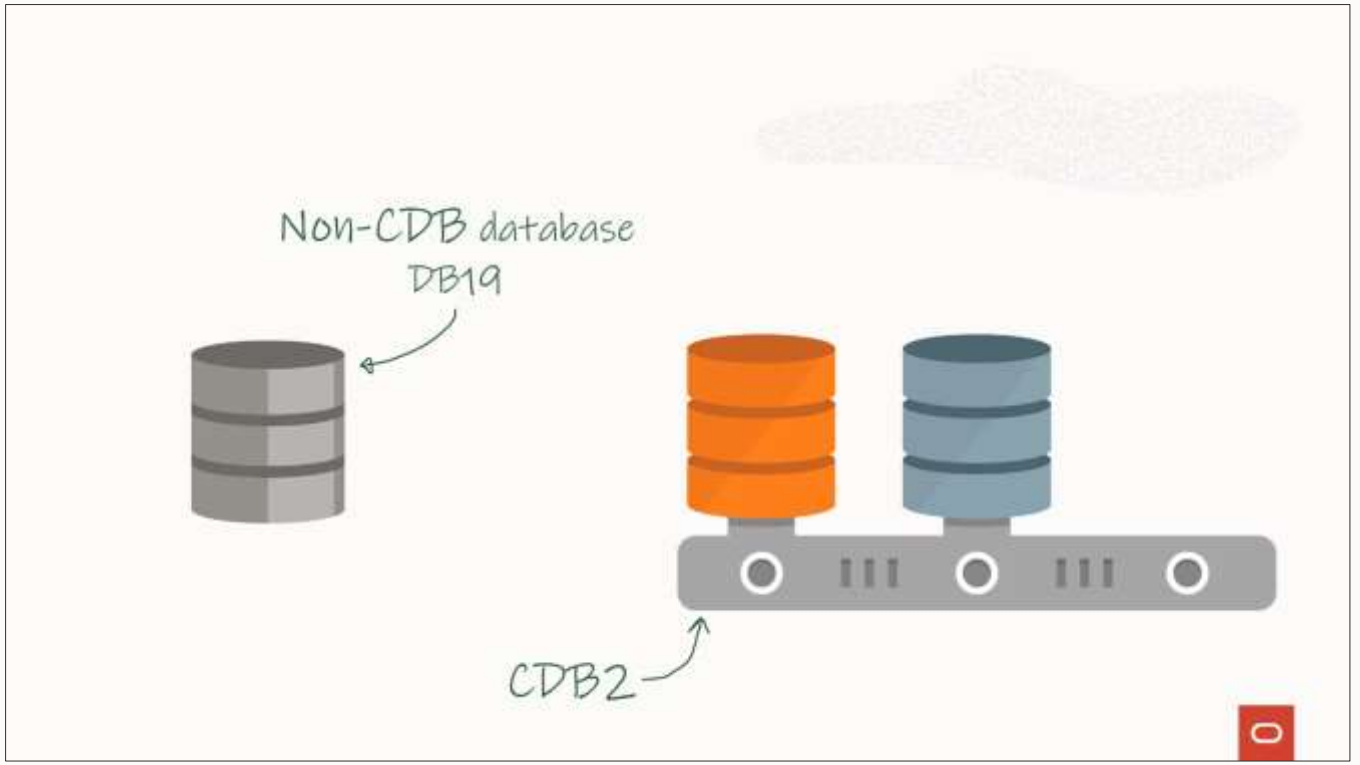
Pro tip: For RAC databases control PDB startup through Clusterware

Convert | noncdb_to_pdb.sql

- Requires downtime
- Runtime varies - typically 10-30 min
  - Fix for Bug 25809128 is included since 19.9.0 and adds a significant improvement
- It depends on number of objects - not the physical size
- Runs only once in the life of a database
- Irreversible
- Re-runnable from 12.2

Pro tip: If you want more details refer to blog post

Migrate | Demo



Watch on YouTube

other

MIGRATION

options

GoldenGate

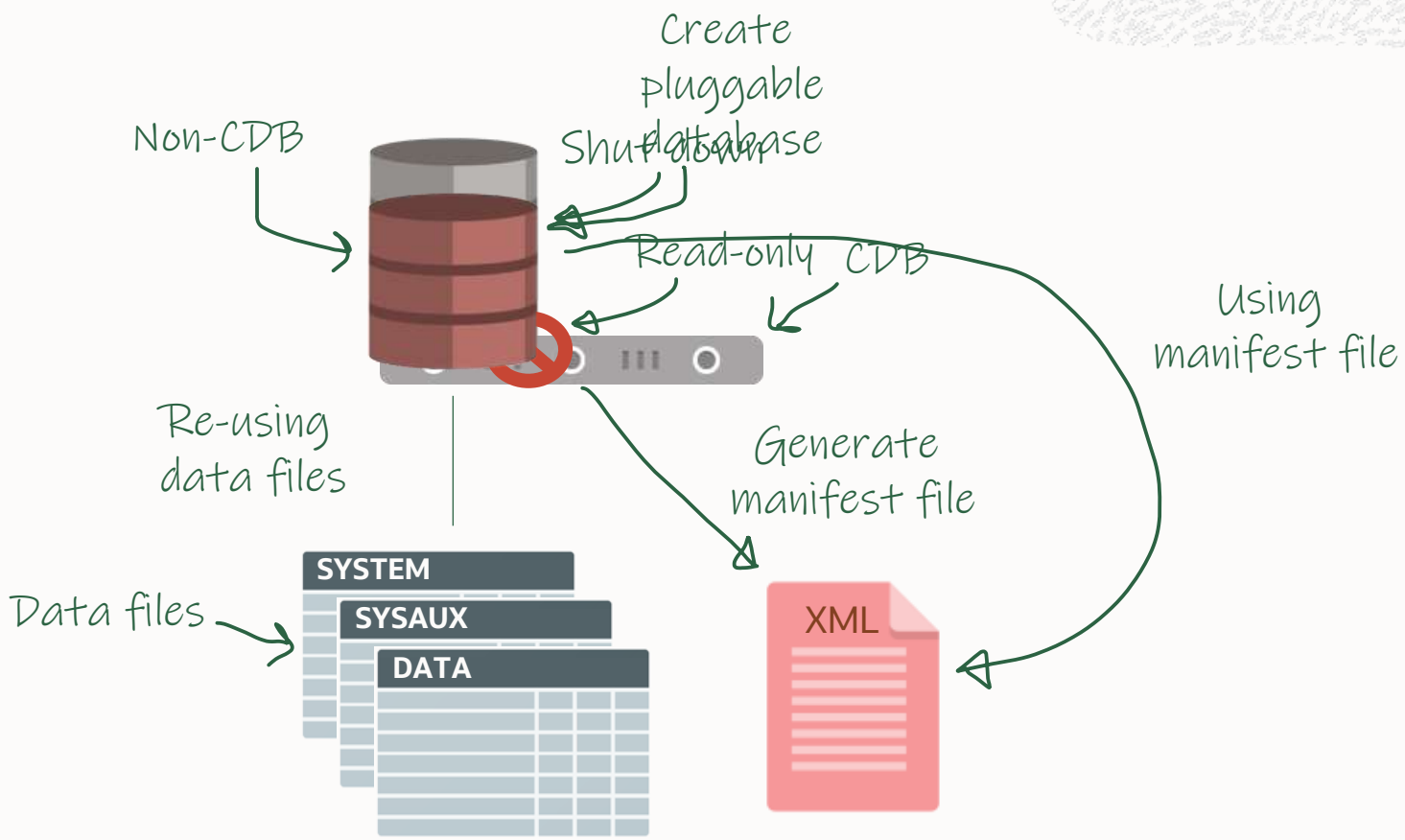
Transportable

Data Pump

Plug-in Copy

Plug-in NoCopy

Plug-in NoCopy | Concept



Plug-in NoCopy | Create

Re-use existing data files

```
SQL> CREATE PLUGGABLE DATABASE DB19 ... NOCOPY ... ;
```

Move data files

```
SQL> CREATE PLUGGABLE DATABASE DB19 ... MOVE ... ;
```


Plug-in NoCopy | AutoUpgrade

Fully automated plug-in

```
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=DB19
upg1.target_cdb=CDB2
```

Command

```
java -jar autoupgrade.jar -config DB19.cfg -mode deploy
```

Pro tip: Always get latest version of AutoUpgrade from MOS 2485457.1

Plug-in NoCopy | AutoUpgrade

Upgrade - and plug in

```
upg1.source_home=/u01/app/oracle/product/12.2.0.1
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=DB12
upg1.target_cdb=CDB2
#Optionally, rename PDB
#upg1.target_pdb_name=SALES
```

Command

```
java -jar autoupgrade.jar -config DB19.cfg -mode deploy
```

Blog post: Oracle AutoUpgrade between two servers – and Plugin?

Pro tip: You can also plug in manually and upgrade PDB with `dbupgrade -c DB19`

Plug-in NoCopy | Nice to know

No fallback

- Data files are re-used

Fast option

Cross-platform

- Potentially roll off patches before unplug
- But can't go across Endian format

other

MIGRATION

options

GoldenGate

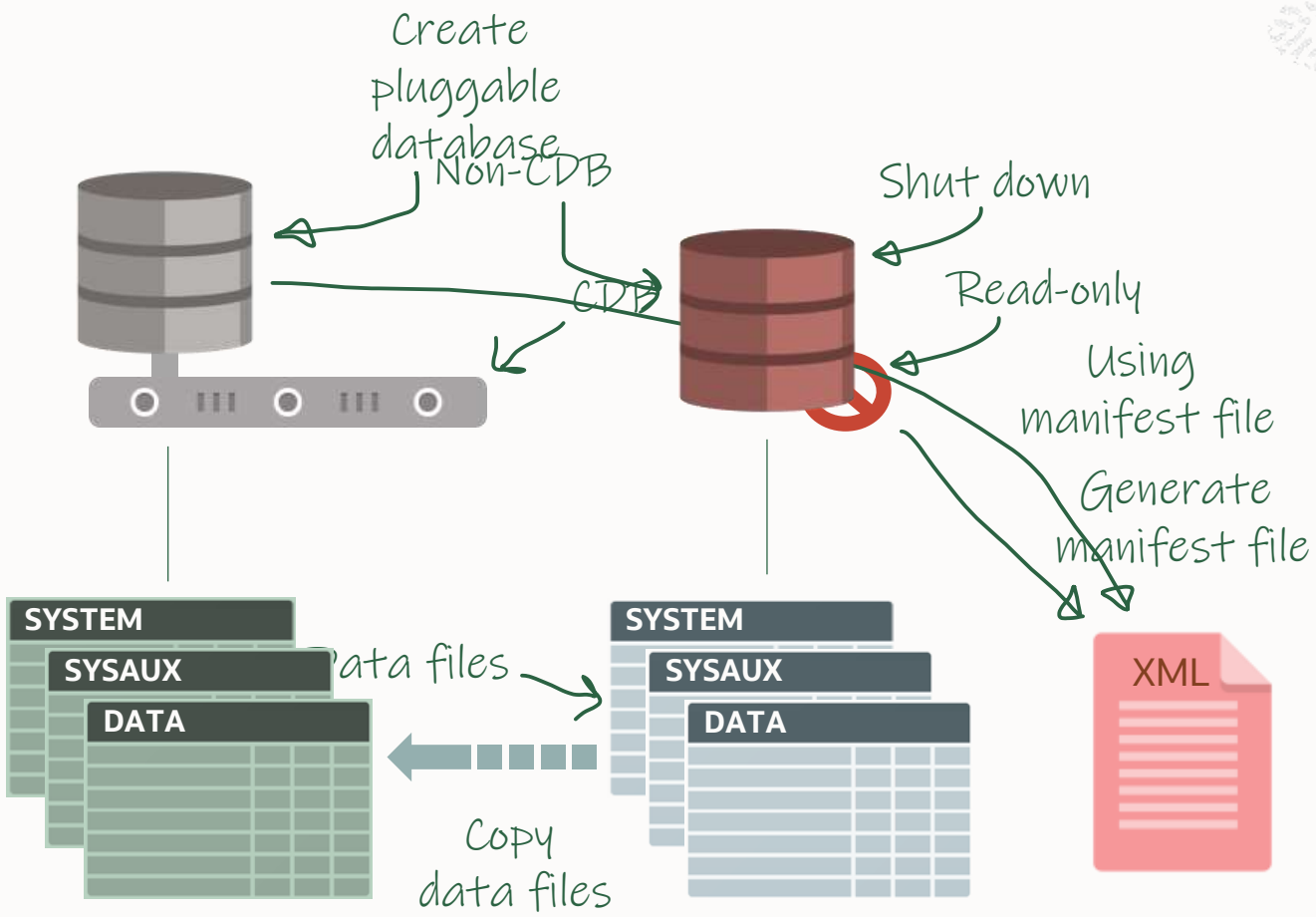
Transportable

Data Pump

Plug-in Copy

Plug-in NoCopy

Plug-in Copy | Concept



Plug-in Copy | Create

Copy data files

```
SQL> CREATE PLUGGABLE DATABASE DB19 ... COPY FILE_NAME_CONVERT= ... ;
```

Rename data files with FILE_NAME_CONVERT

- Regular search/replace
  `FILE_NAME_CONVERT=('DB19', 'SALES')`
- OMF
  `FILE_NAME_CONVERT=NONE`

Pro tip: Use the same `FILE_NAME_CONVERT` clause for plug-in with `MOVE` keyword

Plug-in Copy | Clone non-CDB

Plug in and copy data files over network link

```
SQL> CREATE DATABASE LINK CLONELNK ... ;
SQL> CREATE PLUGGABLE DATABASE DB19 FROM NON$CDB@CLONELNK ... ;
```

Prerequisites:

- Source must be 12.1.0.2 or newer
- Block size must match
- Blog post

Plug-in Copy | AutoUpgrade

Fully automated plug-in

```
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=DB19
upg1.target_cdb=CDB2
upg1.target_pdb_name=SALES
#Copy files and perform search/replace on file names
upg1.target_pdb_copy_option=file_name_convert=('DB19','SALES')
#Copy files and generate new OMF file names
#upg1.target_pdb_copy_option=file_name_convert=none
```

Command

```
java -jar autoupgrade.jar -config DB19.cfg -mode deploy
```


Plug-in Copy | Nice to know

Fallback option

- Original data files are preserved

Slow and requires additional disk space

Cross-platform

- Potentially roll off patches before unplug
- But can't go across Endian format

other

MIGRATION

options

GoldenGate

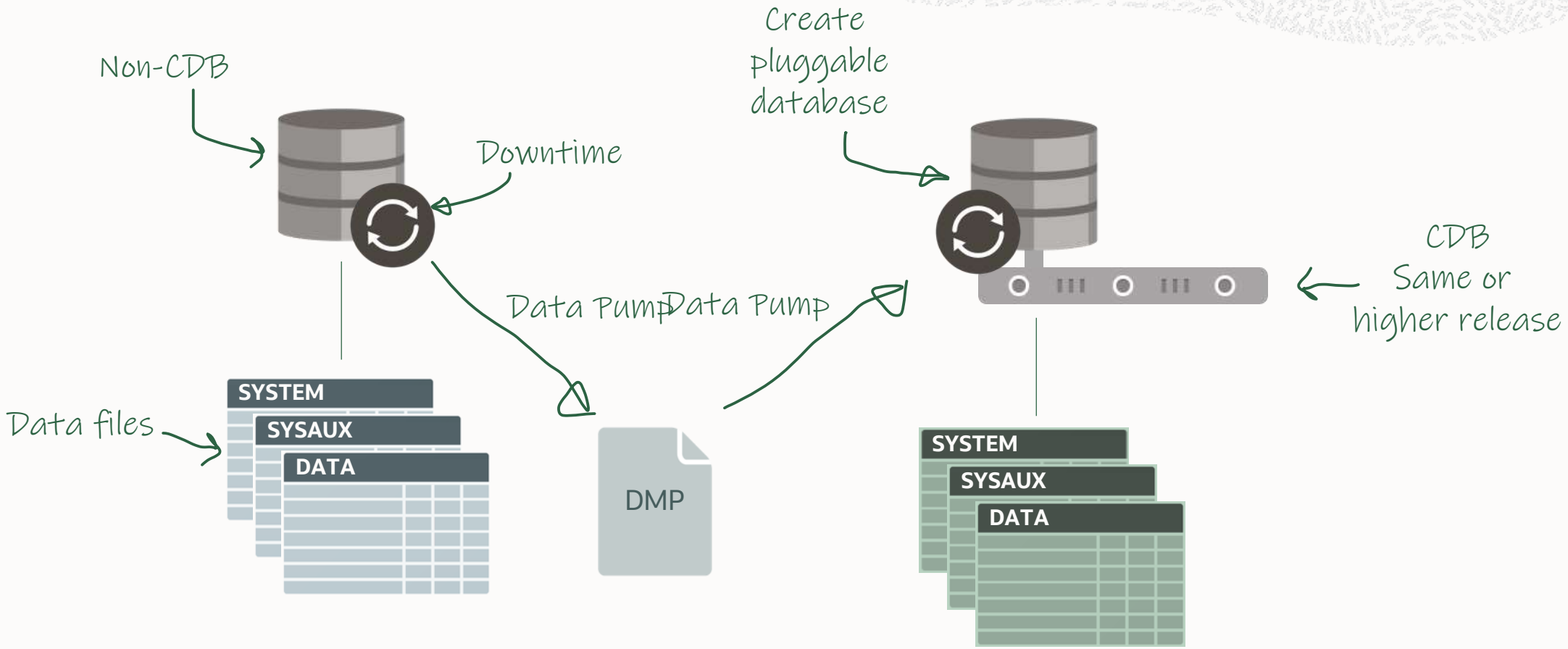
Transportable

Data Pump

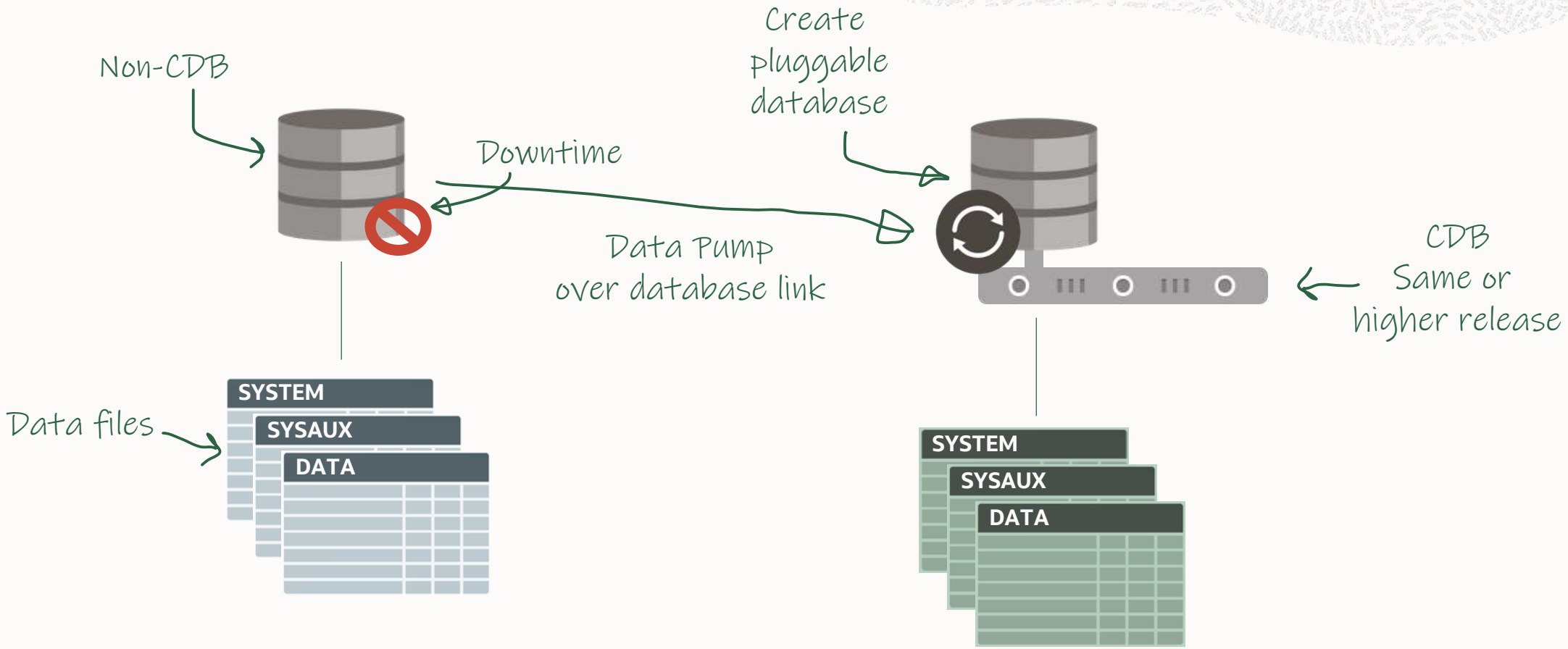
Plug-in Copy

Plug-in NoCopy

Data Pump | Concept Dump File Mode



Data Pump | Concept Network Mode



Data Pump | Concept

Well-known and proven method

Dump file or network mode

- Network mode has fewer parallel capabilities

The power and flexibility of Data Pump

- Change character set
- Implement partitioning
- Convert to SecureFile LOBs
- Everything or subset of data
- Filters and transform
- … and so much more

Data Pump | Nice to know

Fallback option

- Original database is preserved

Slow - and requires disk space

Cross-platform and cross-endian format

Migrate from lower release without upgrading

other

MIGRATION

options

GoldenGate

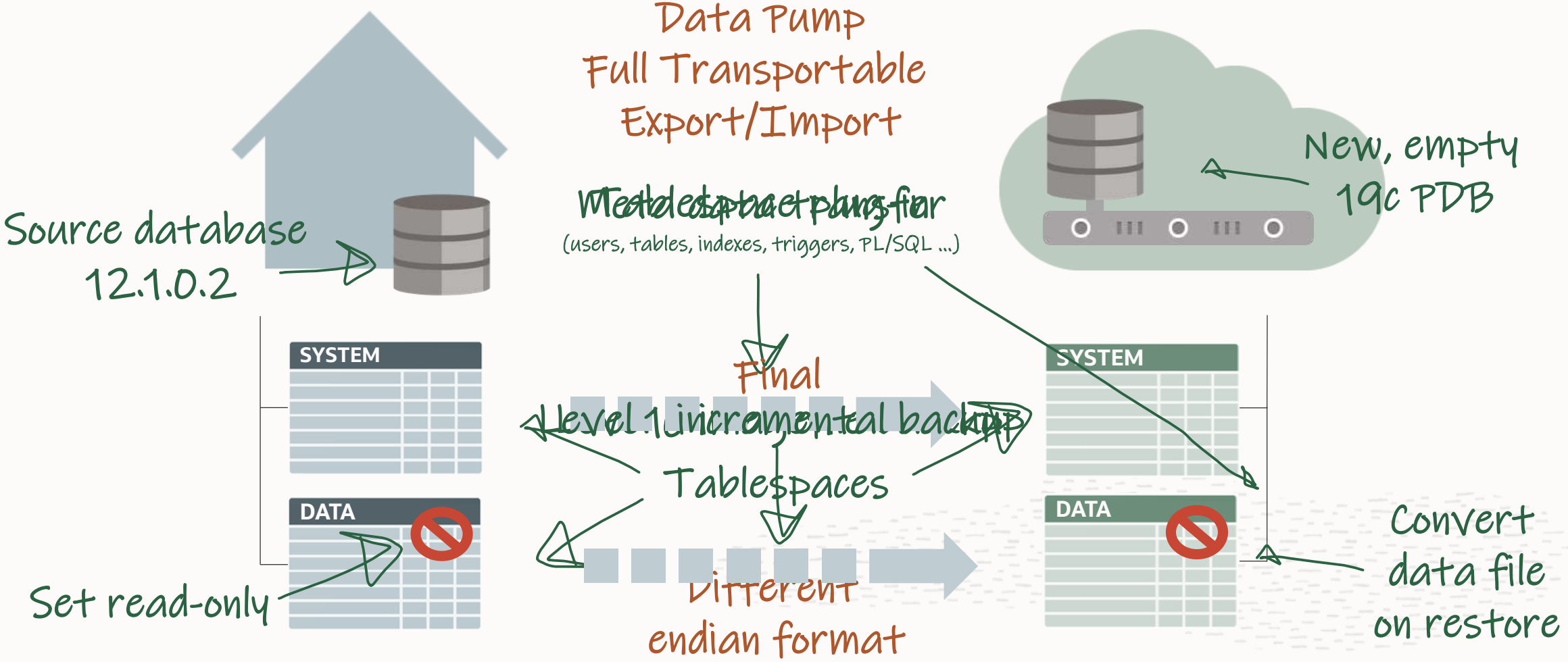
Transportable

Data Pump

Plug-in Copy

Plug-in NoCopy

Transportable | Concept



Transportable | Overview

For minimal downtime

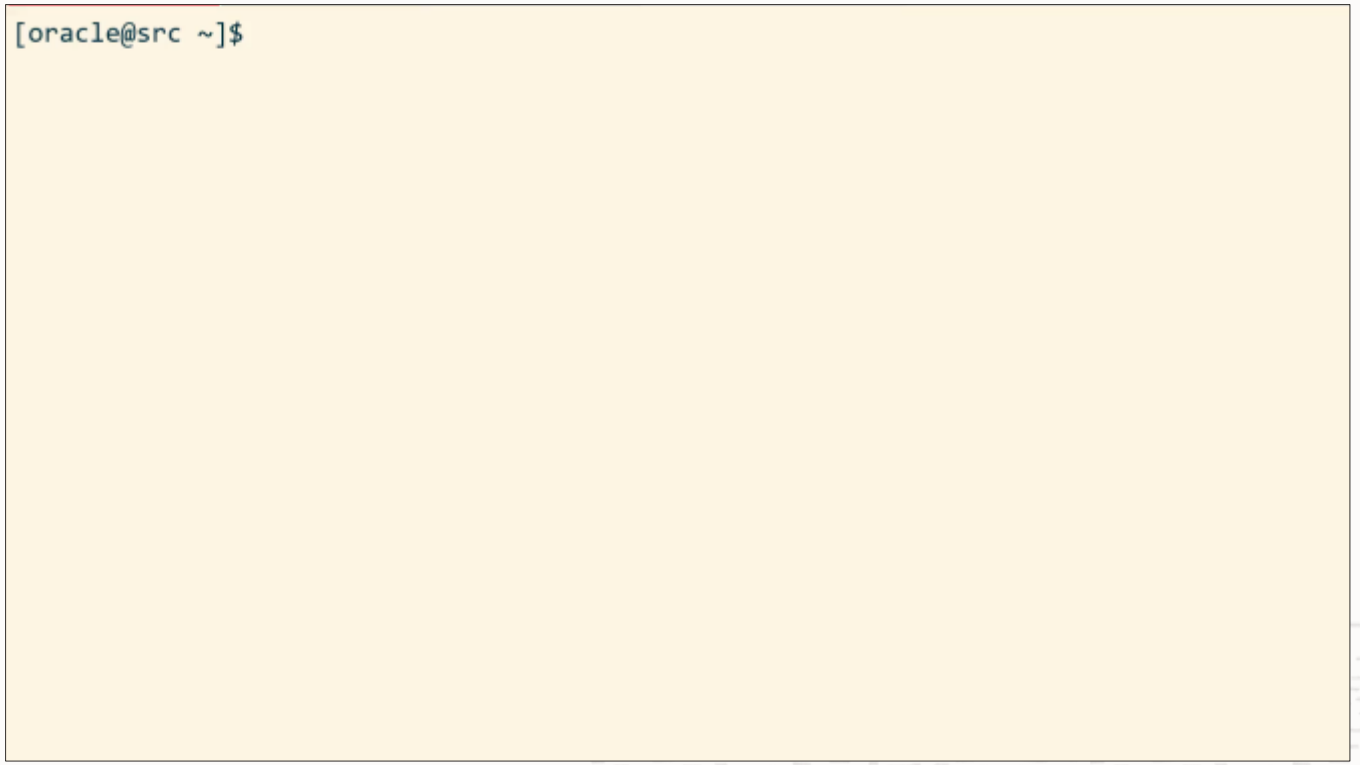
V4 PERL Scripts to reduce Transportable Tablespace Downtime using Cross Platform Incremental Backup (doc ID 2471245.1)

Blog posts:

- Overview
- Step-by-step

Other options with Transportable Tablespace exist

Transportable | Demo



Watch on YouTube

Transportable | Nice to know

Fallback option

- Original database is preserved

Fast - just final incremental and FTEX import

Cross-platform and cross-endian format

Migrate from lower release without upgrading

other MIGRATION options

GoldenGate

Transportable

Data Pump

Plug-in Copy

Plug-in NoCopy

Photo by Ludovic Migneault on Unsplash

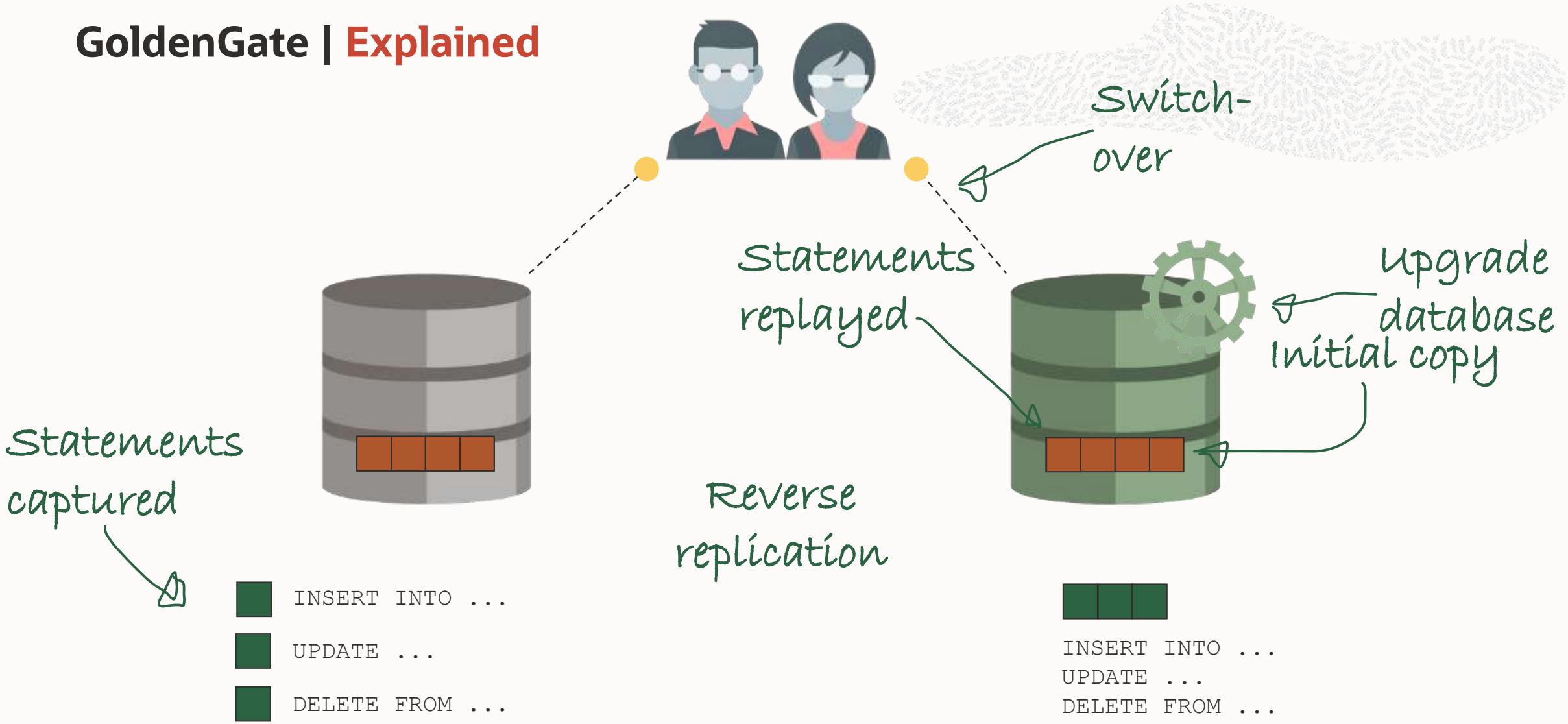
But what if you have no downtime allowed?

Oracle GoldenGate

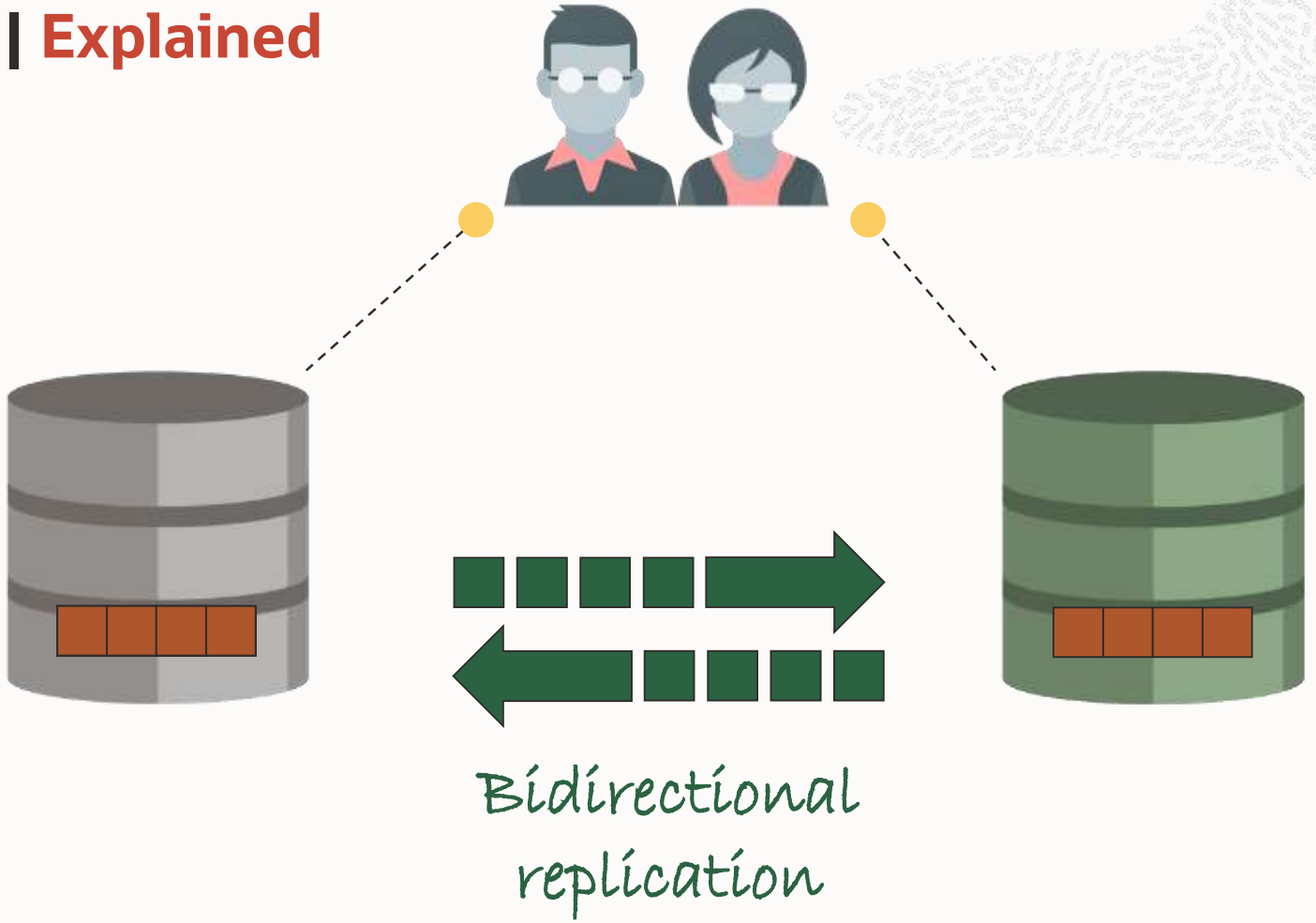
If you need upgrade with no downtime, there is only one option

Oracle GoldenGate

GoldenGate | Explained



GoldenGate | Explained



GoldenGate | Overview

True - and only - Zero Downtime option

Use on top of:

- Data Pump
- Transportable Tablespaces

Migration:

- From lower version
- Across endianness

GoldenGate | Nice to know

Fallback option

- Original database is preserved
- Optionally, reverse replication for fallback even after go-live

Zero downtime

Cross-platform and cross-endian format

"It is too expensive"

No, it is not!

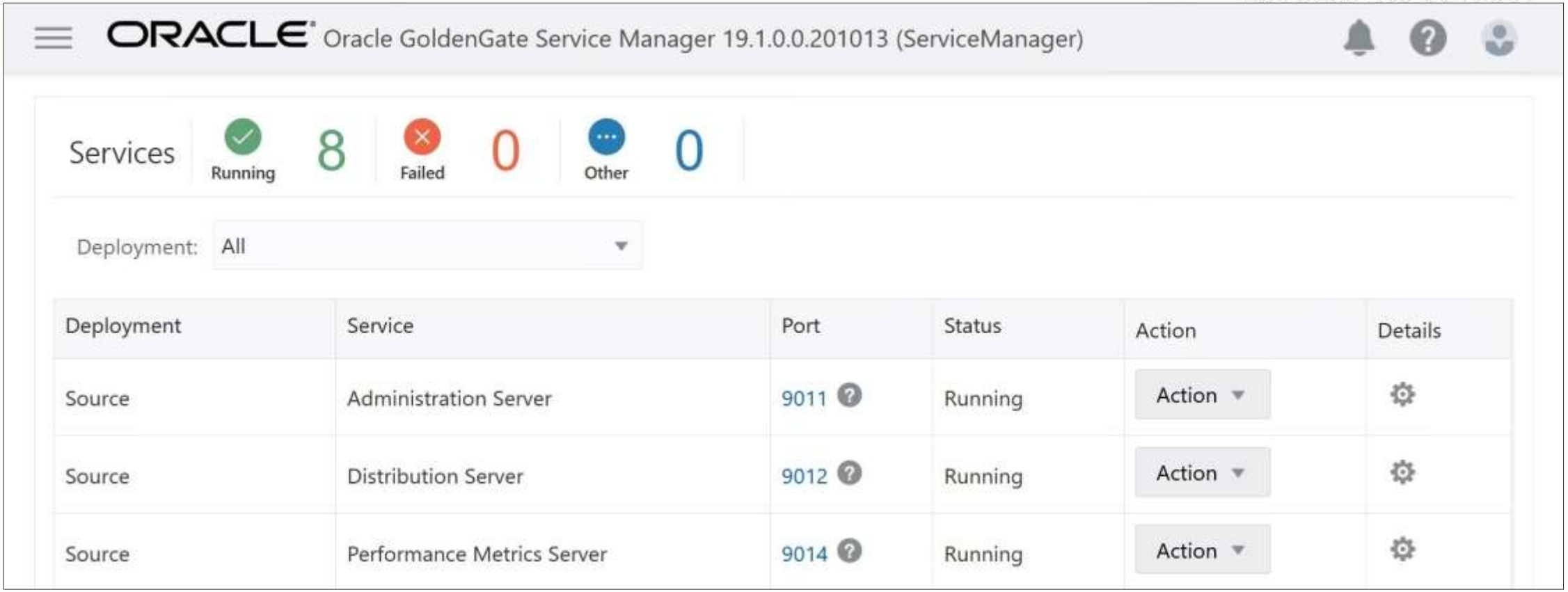
"It is too complex"

GoldenGate Microservices Architecture is a lot more intuitive and user friendly

GoldenGate | Before

```
[oracle@ggsource gghome]$ ./ggsci

Oracle GoldenGate Command Interpreter for Oracle
Version 12.2.0.1.1 OGGCORE_12.2.0.1.0_PLATFORMS_151211.1401_FBO
Linux, x64, 64bit (optimized), Oracle 12c on Dec 12 2015 02:56:48
Operating system character set identified as UTF-8.

Copyright (C) 1995, 2015, Oracle and/or its affiliates. All rights reserved.



GGSCI (ggsource.doyensys.com) 1> dblogin userid gguser,password gguser
Successfully logged into database.
```

GoldenGate | Now



Oracle GoldenGate

comparing MIGRATION options

	Plug-in NoCopy	Plug-in Copy	Data Pump	Transportable	GoldenGate
Downtime	Considerable	Less	Considerable	Minimal	None
Fallback	No	Yes	Yes	Yes	Yes
Cross-endian	No	No	Yes	Yes	Yes
Cross-version	No	No	Yes	Yes	Yes
Complexity	Low	Low	Medium	Medium	High

Photo by Tim Rebkavets on Unsplash

Migration Options

Getting There Safely

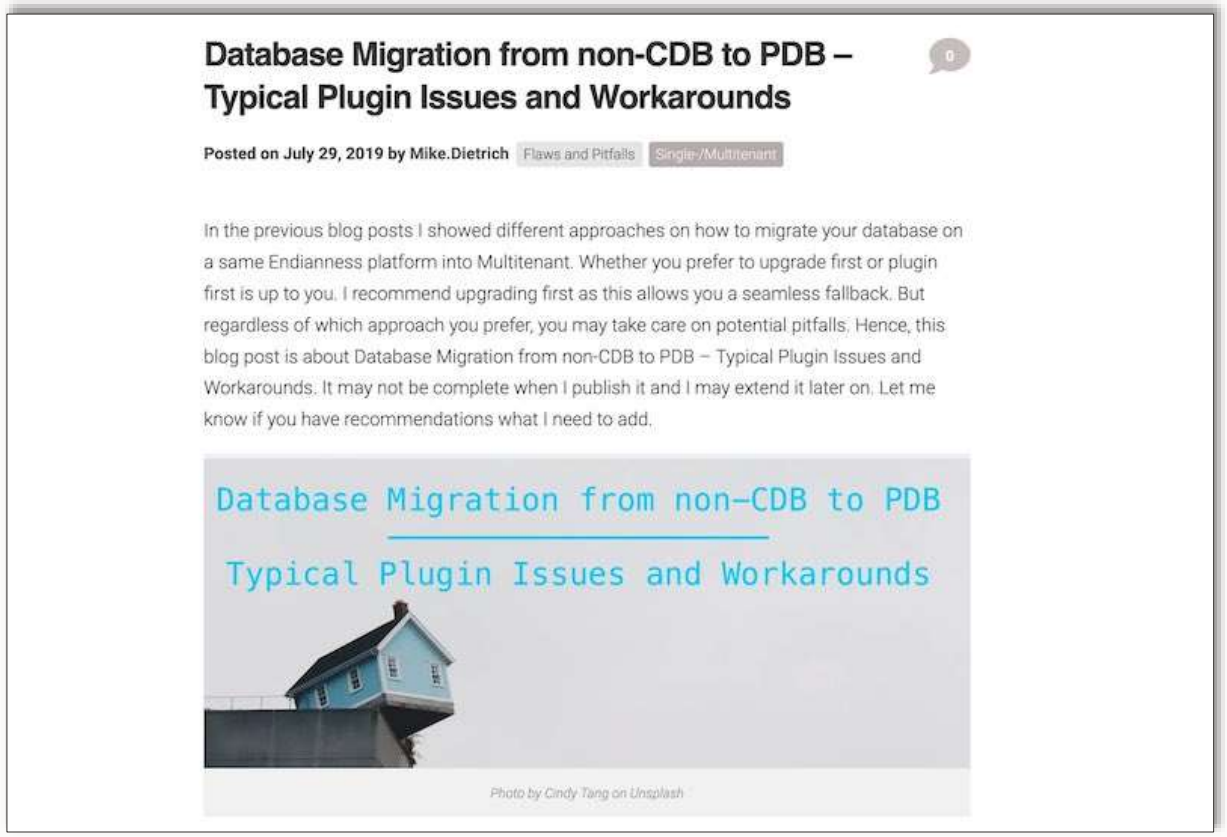
Migration | Avoid the Pitfalls

Complete overview

- non-CDB has more or different components than CDB
  - Remove components from non-CDB or install missing components into CDB$ROOT
- non-CDB has higher time zone value than CDB
  - You must install the higher time zone patch into the CDB's home
- non-CDB has a higher patch level than CDB
  - Install matching higher patch into CDB's home or rollback patch from non-CDB
- non-CDB has a different `DB_BLOCK_SIZE` than CDB
  - Use matching `DB_n_CACHE_SIZE` parameters in CDB's spfile

Photo by Casey Horner on Unsplash

Pitfalls | Overview

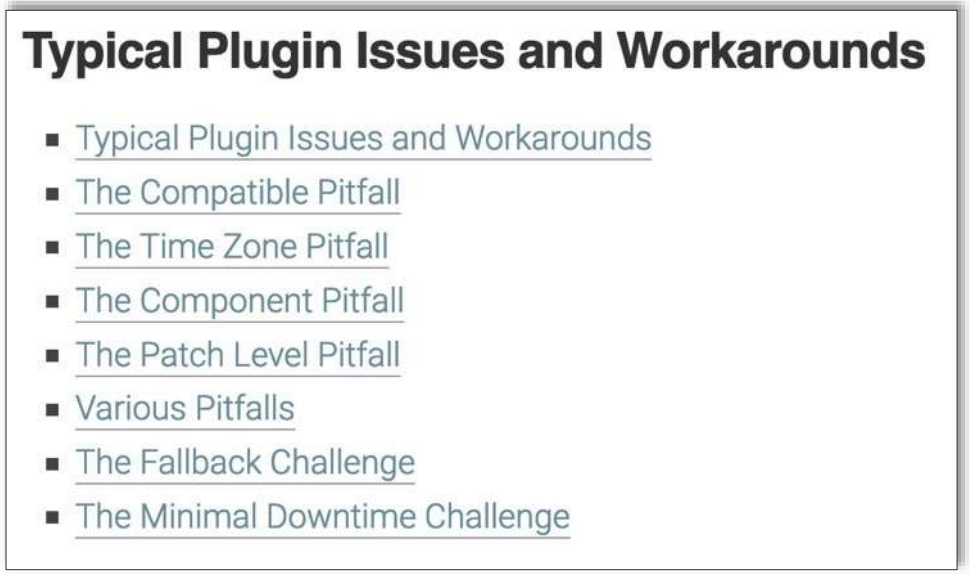


Database Migration from non-CDB to PDB – Typical Plugin Issues and Workarounds

Posted on July 29, 2019 by Mike.Dietrich Flaws and Pitfalls Single-/Multitenant

In the previous blog posts I showed different approaches on how to migrate your database on a same Endianness platform into Multitenant. Whether you prefer to upgrade first or plugin first is up to you. I recommend upgrading first as this allows you a seamless fallback. But regardless of which approach you prefer, you may take care on potential pitfalls. Hence, this blog post is about Database Migration from non-CDB to PDB – Typical Plugin Issues and Workarounds. It may not be complete when I publish it and I may extend it later on. Let me know if you have recommendations what I need to add.

Photo by Cindy Tang on Unsplash



Typical Plugin Issues and Workarounds

Blog Post: Typical Plugin Issues and Workarounds

Migration | Last Words

Every migration

- Is an architectural change
- Requires downtime
- Requires a fallback
- Ends with a backup

Photo by Oliver Roos on Unsplash

and finally

Multitenant Upgrades

when you adopted the CDB architecture

Everything at Once

Upgrade the entire CDB with all PDBs

Unplug / Plug / Upgrade

Upgrade one or multiple PDBs in a higher version, new CDB

More Options

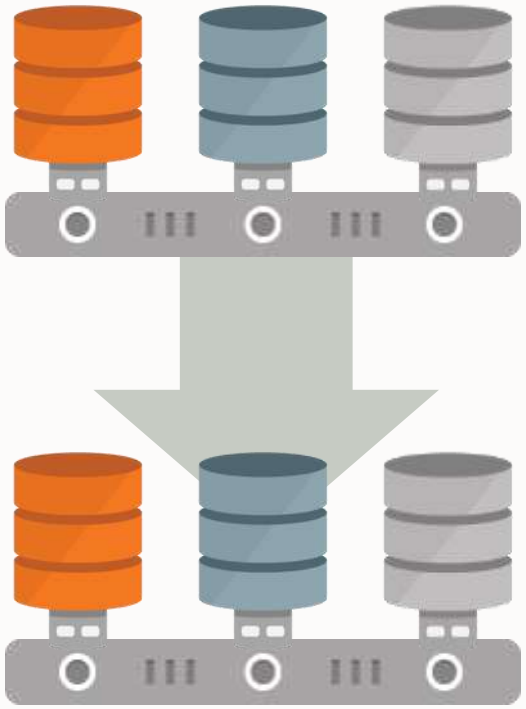
Refreshable Clones as an efficient way to test and perform upgrades

More Power

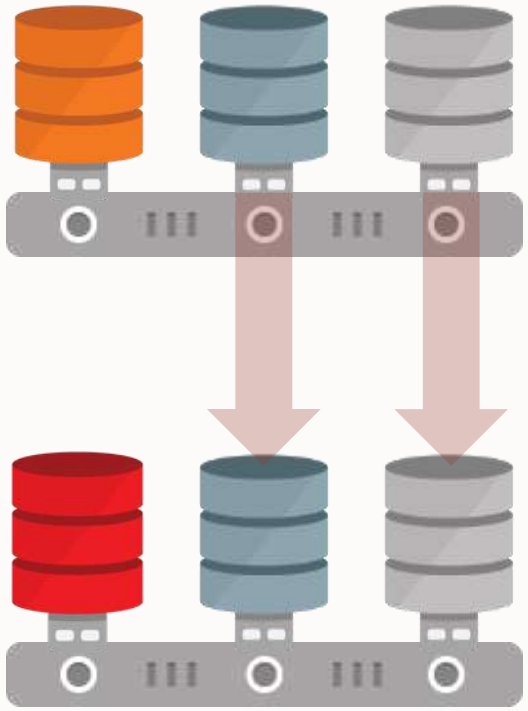
Speed Up "Everything at Once" CDB upgrades

CDB Upgrades | Options

Everything at Once

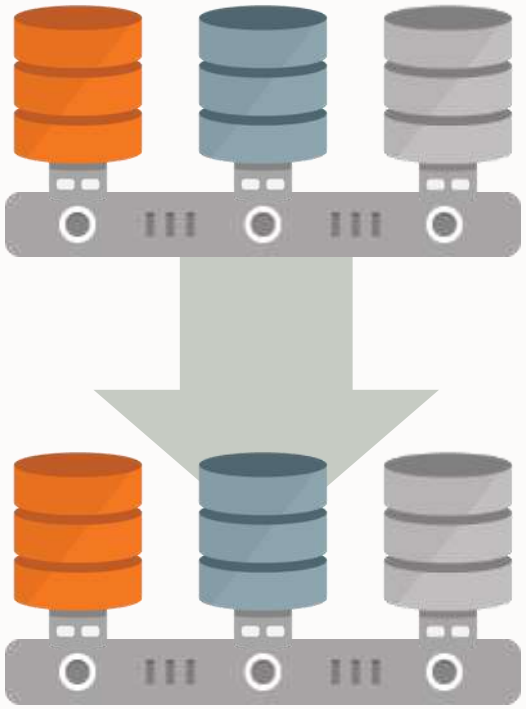


Unplug / Plug / Upgrade

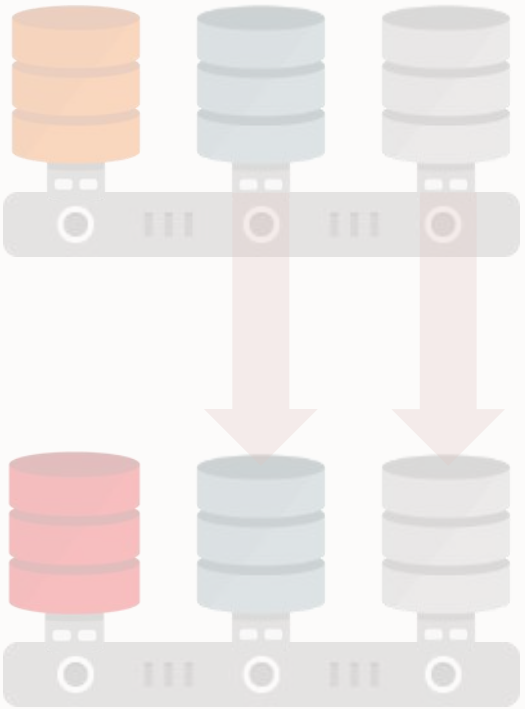


CDB Upgrades | Option 1

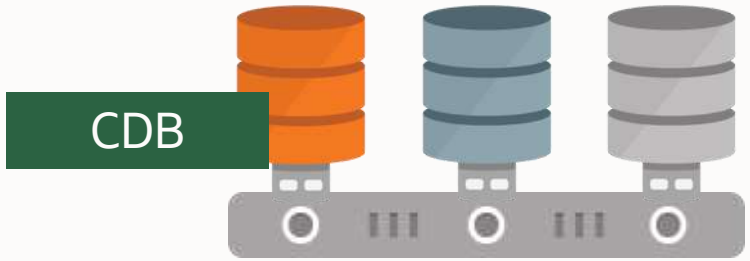
Everything at Once



Unplug / Plug / Upgrade



Parallel Upgrade | Container Database

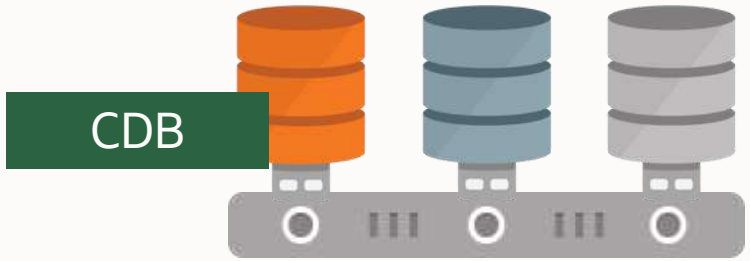


A number of processors are assigned

- Minimum 4
- Maximum unlimited
- Default CPU count

```
$ dbupgrade -n 4
```

Parallel Upgrade | Container Database

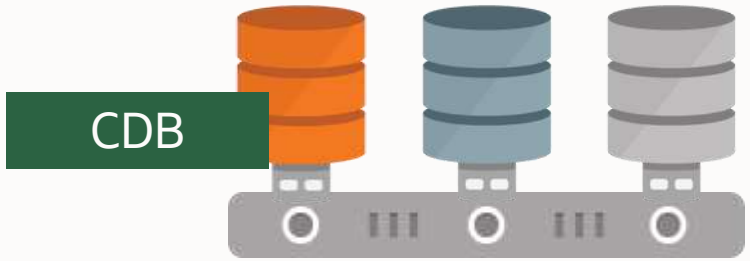


Each PDB gets a number of parallel processes

- Minimum 1
- Maximum 8
- Default 2

```
$ dbupgrade -N 2
```

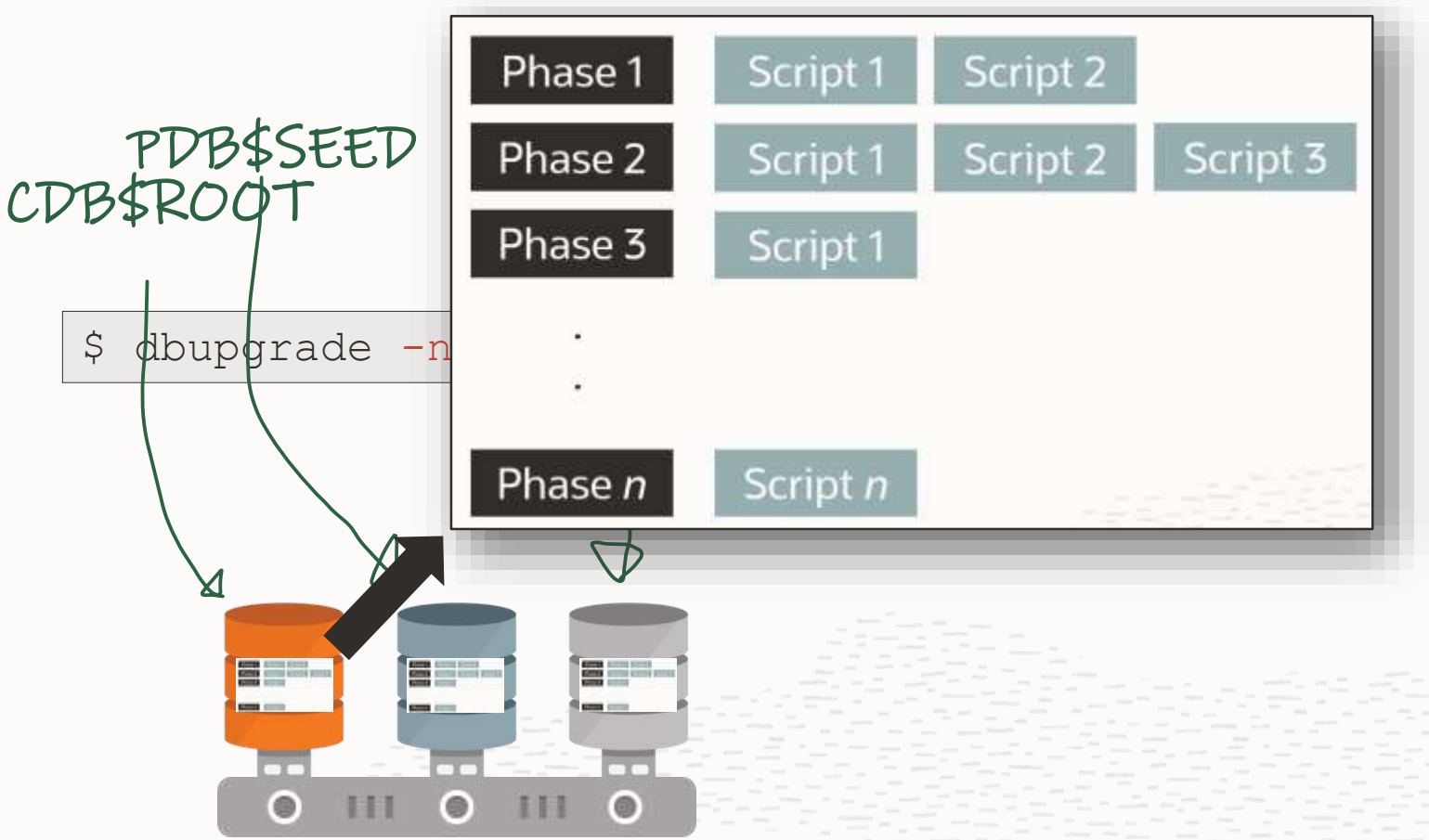
Parallel Upgrade | Container Database



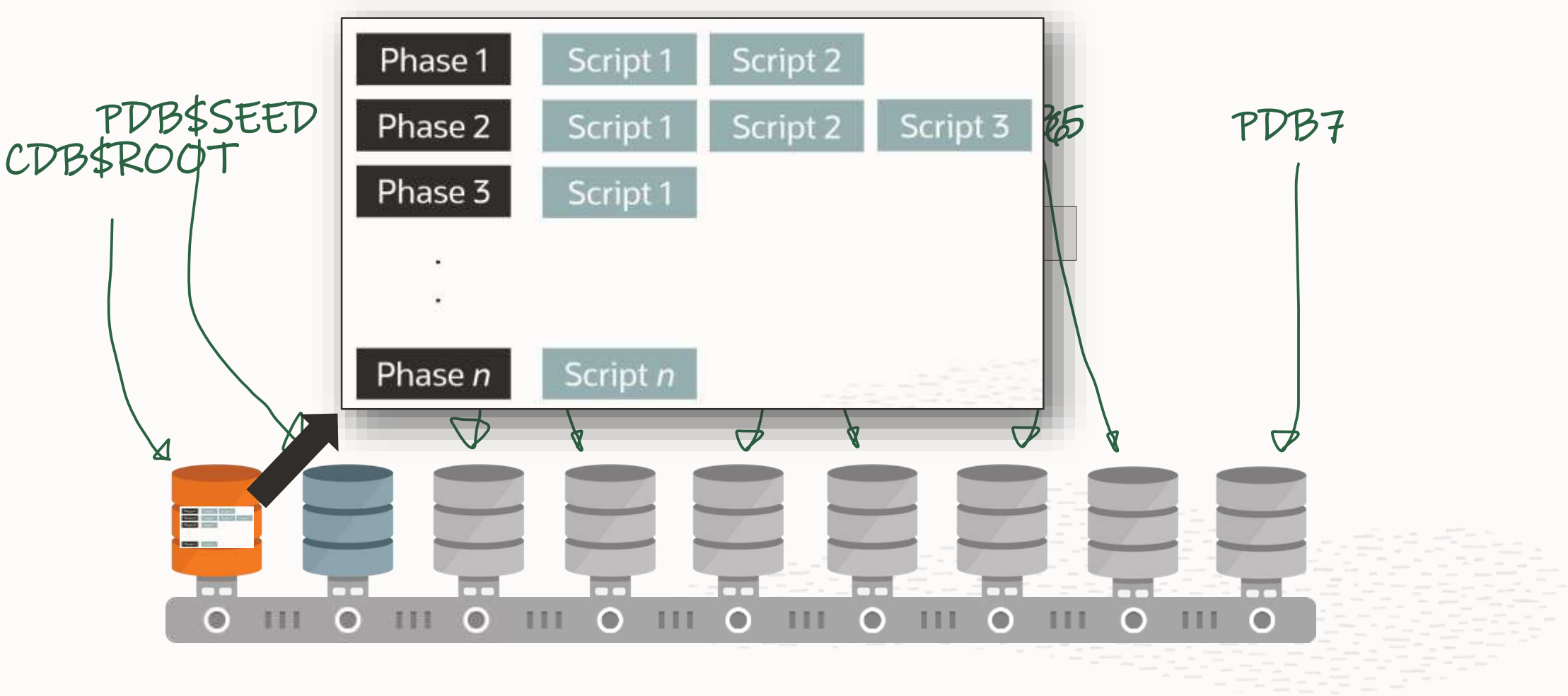
But - there is another limit

$$\frac{\text{Total number of processors (n)}}{\text{Processor per PDB (N)}} = \text{PDBs upgraded simultaneously}$$

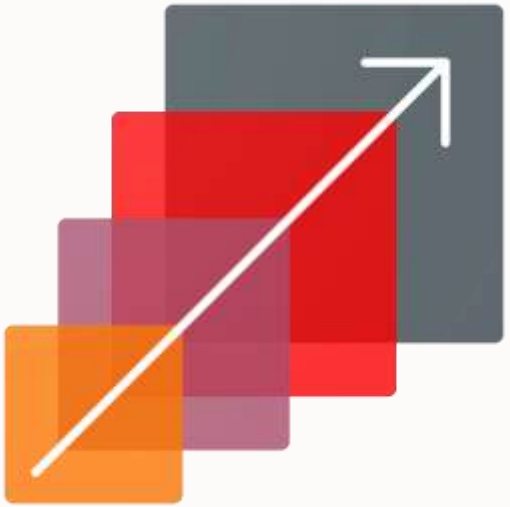
Parallel Upgrade | Single Tenant



Parallel Upgrade | Multitenant



Parallel Upgrade | Multitenant

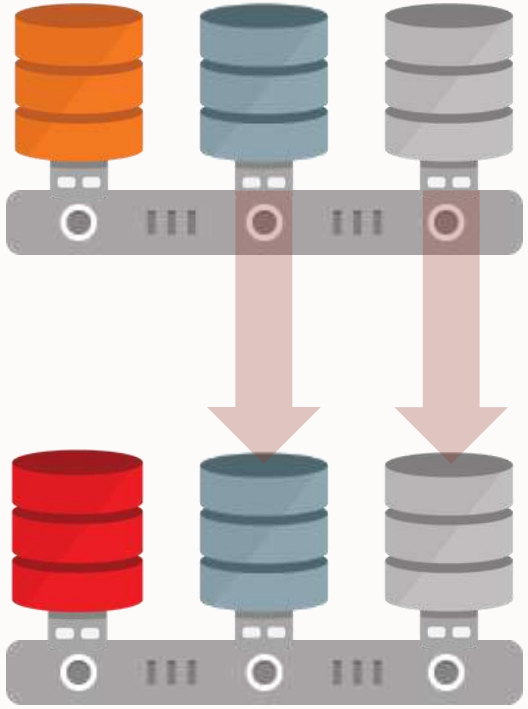


Scale by upgrading
more PDBs simultaneously

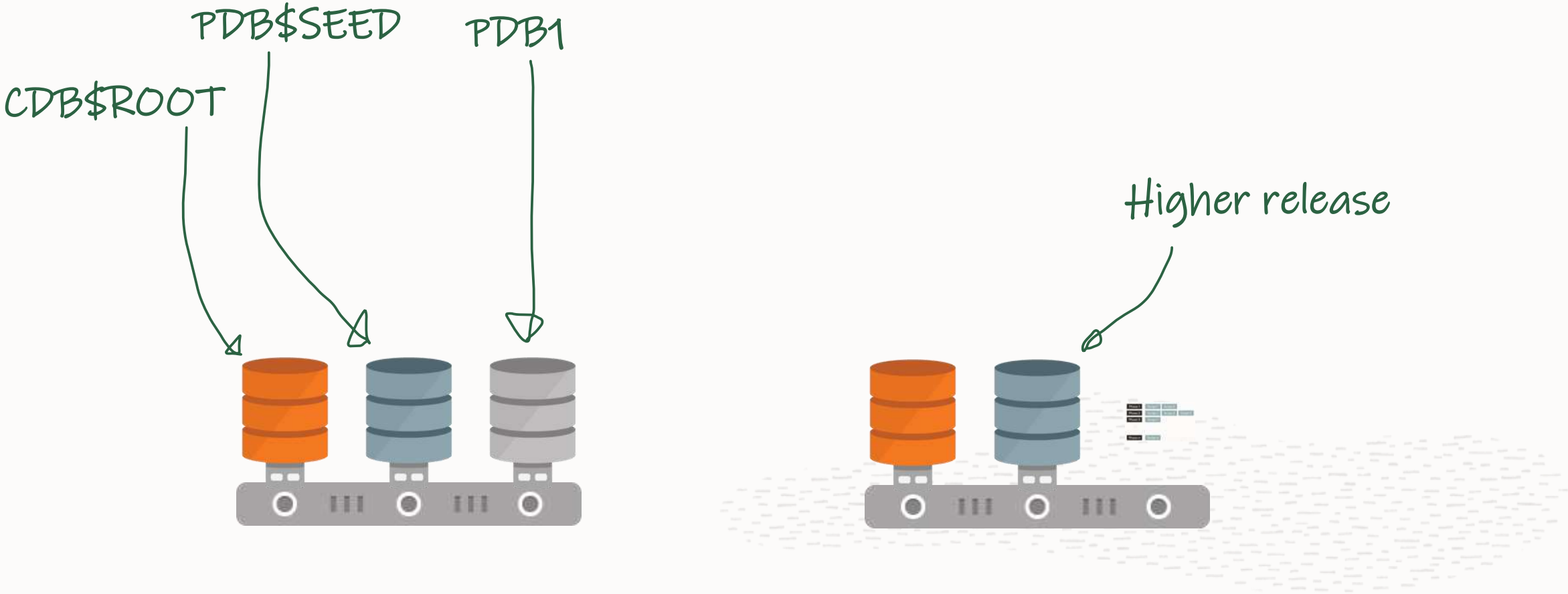
CDB Upgrades | Option 2

Everything at Once

Unplug / Plug / Upgrade



Parallel Upgrade | Unplug-plug Upgrade



Parallel Upgrade | Unplug-plug

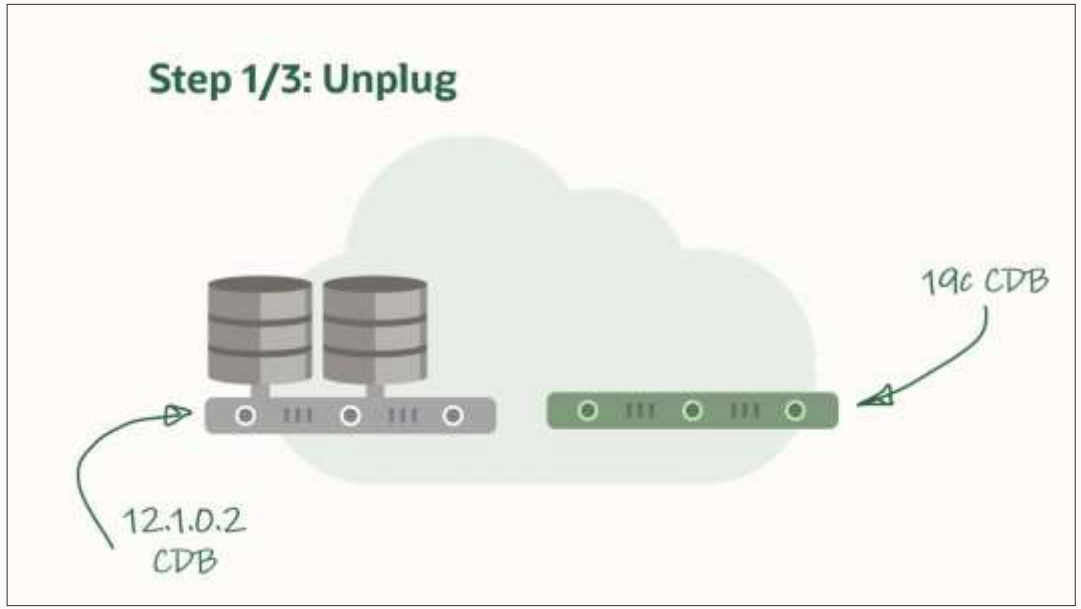
Unplug-plug always faster than

Non-CDB
Single Tenant
Multitenant

AutoUpgrade | Unplug-plug Upgrade

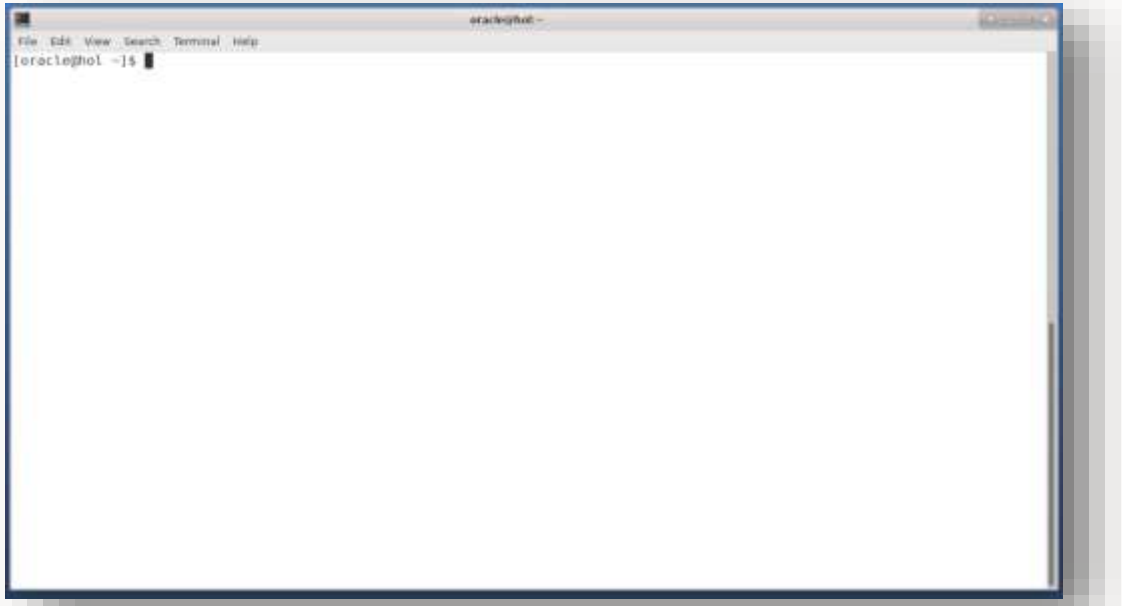
Upgrade a single PDB

- Faster
- More flexible
- Requires compatible target CDB
- Not compatible with Flashback Database
  - Consider using Refreshable PDBs
  - Copy data files (`target_pdb_copy_option`)



AutoUpgrade | Unplug-plug Upgrade

```
upg1.sid=CDB12102
upg1.target_cdb=CDB19
upg1.pdbs=pdb1
upg1.source_home=/u01/app/oracle/product/12102
upg1.target_home=/u01/app/oracle/product/19
```



Watch on YouTube

AutoUpgrade | Unplug-plug Upgrade

Upgrade several PDBs

```
upg1.pdbs=pdb1,pdb2,pdb3
```

Rename a PDB

```
upg1.pdbs=pdb1
upg1.target_pdb_name.pdb1=sales
```

Copy data files on plug-in

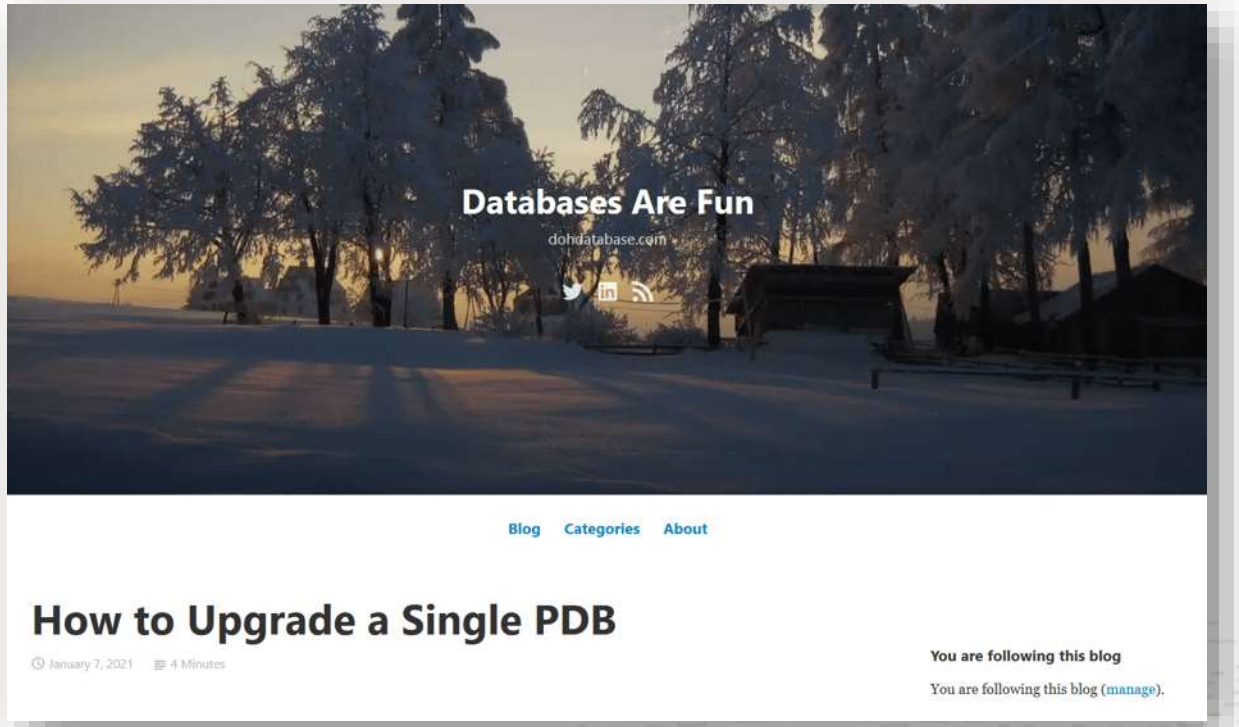
```
upg1.pdbs=pdb1
upg1.target_pdb_copy_option.pdb1=file_name_convert=('pdb1','sales')
```


AutoUpgrade | Unplug-plug Upgrade

Current limitations:

- Does not support Data Guard
- Does not support TDE Tablespace Encryption

AutoUpgrade | Unplug-plug Upgrade



https://dohdatabase.com/how-to-upgrade-a-single-pdb

Upgrade – Not only for tests

Refreshable Clone PDB

COMING SOON!

Upgrade via refreshable clone PDB

Unplug-Plug | Refreshable Clone PDB

Clone User

```
CREATE USER c##borg
IDENTIFIED BY oracle
DEFAULT TABLESPACE users
TEMPORARY TABLESPACE temp
CONTAINER=ALL;
```

```
GRANT
CREATE SESSION,
CREATE PLUGGABLE DATABASE,
SELECT_ANY_CATALOG
TO c##borg
CONTAINER = ALL;
```

Unplug-Plug | Refreshable Clone PDB

Database link into source PDB

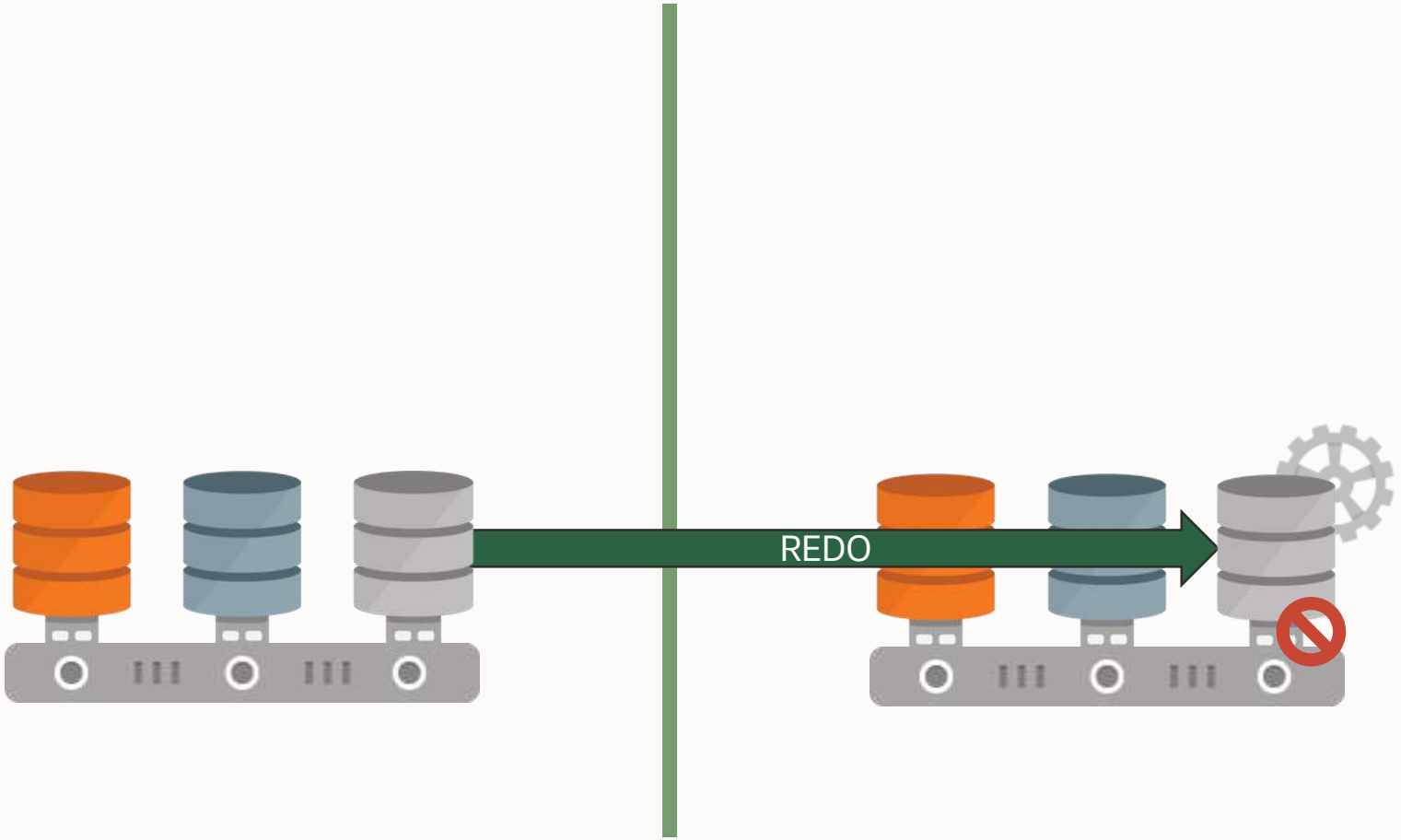
```
CREATE DATABASE LINK clonePDB1
CONNECT TO c##borg
IDENTIFIED BY oracle
USING 'tns-or-ezconnect';
```

Unplug-Plug | Refreshable Clone PDB

Fully automated relocation with upgrade

```
upg1.source_home=/u01/app/oracle/product/12.2
upg1.target_home=/u01/app/oracle/product/19
upg1.sid=CDB1
upg1.pdbs=PDB1
upg1.target_cdb=CDB2
upg1.source_dblink.PDB1=clonePDB1 600
```

Unplug-Plug | Refreshable Clone PDB



AutoUpgrade uses
`CREATE PLUGGABLE DATABASE` statement
which automatically adjusts parallel degree

Photo by Nathan Dumlao on Unsplash

I Need More Power

I can't hold her together, Captain!

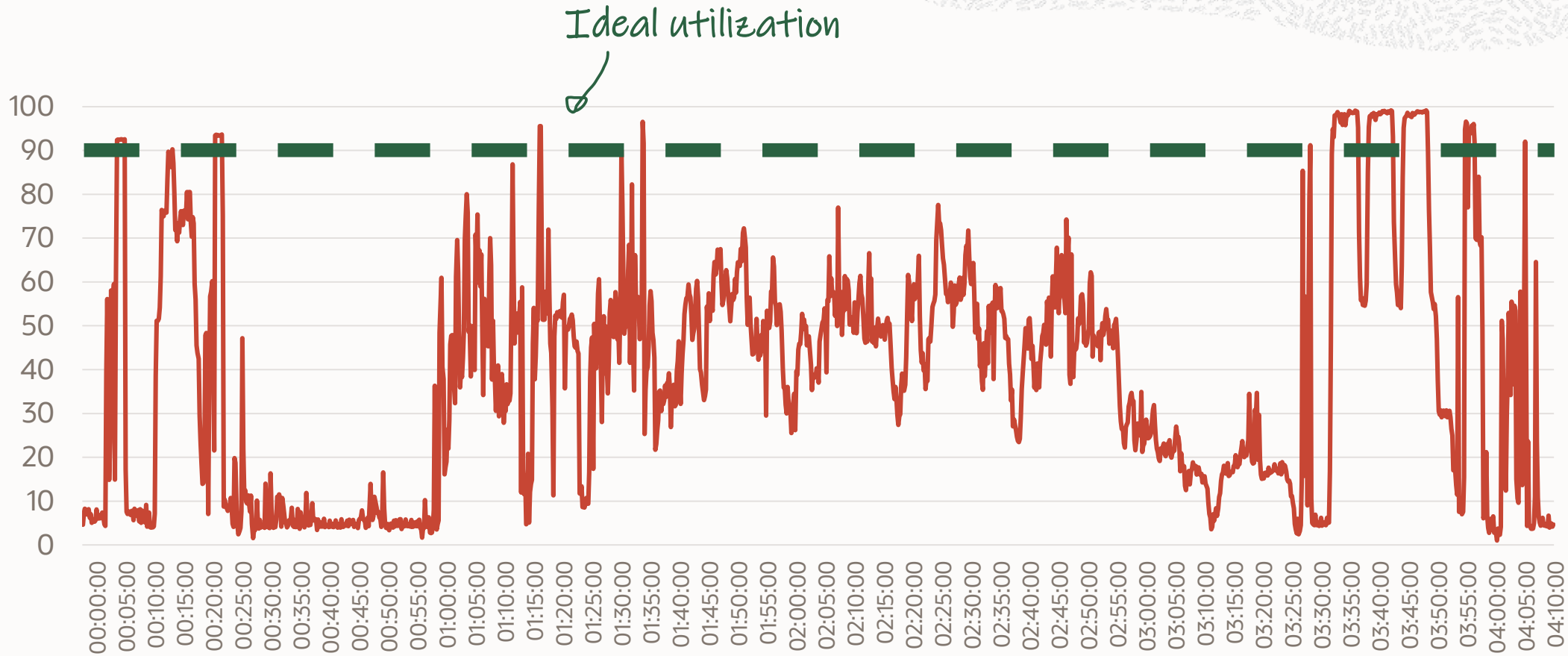
During upgrade CPU is a vital resource

Faster Upgrades | Overview

- OCI Bare Metal host
  - 16 OPCUs
  - 768 GB memory
  - NVMe disks
- CDB with 52 PDBs
  - `CPU_COUNT` = 32
  - `SGA_TARGET` = 80G
  - `PGA_AGGREGATE_TARGET` = 20G
- Many database components (17 in total)

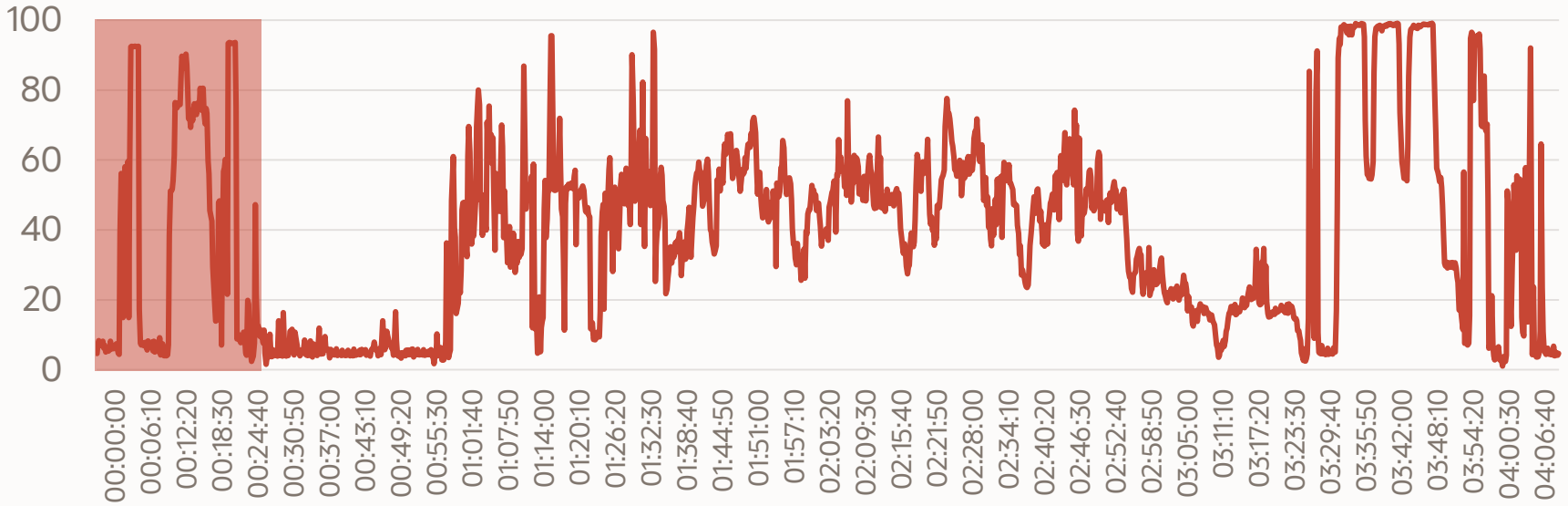
- **Upgrade from 12.1.0.2 to 19**

Faster Upgrades | CPU Utilization



Total upgrade time: 4 hours 8 minutes

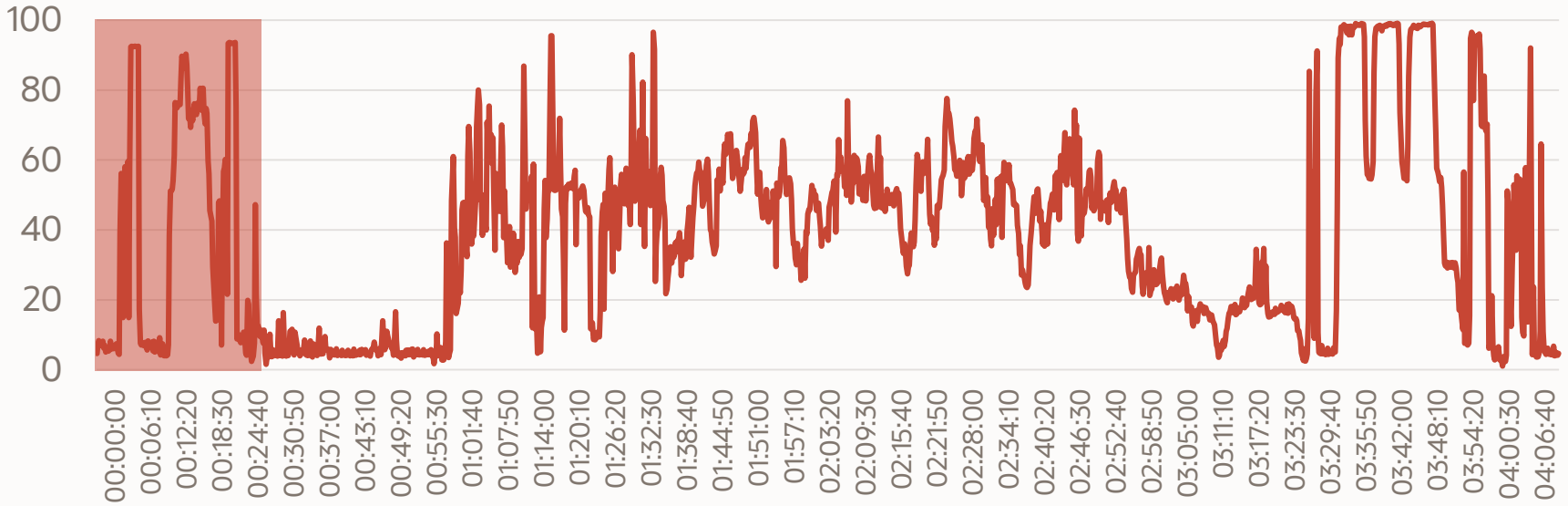
Faster Upgrades | CPU Utilization

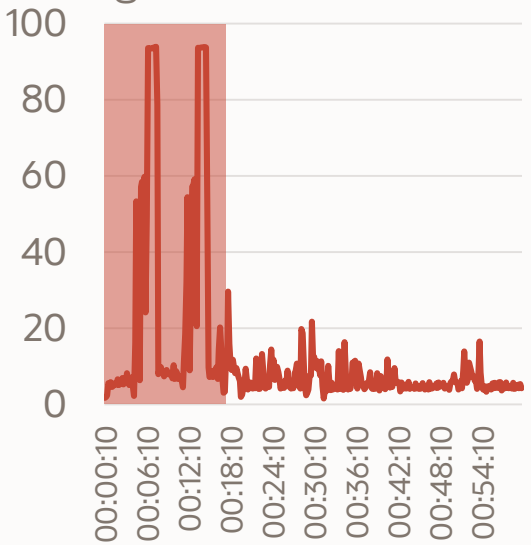


Preupgrade check and fixups

- Gather dictionary and fixed objects stats in advance (7 days)

Faster Upgrades | CPU Utilization

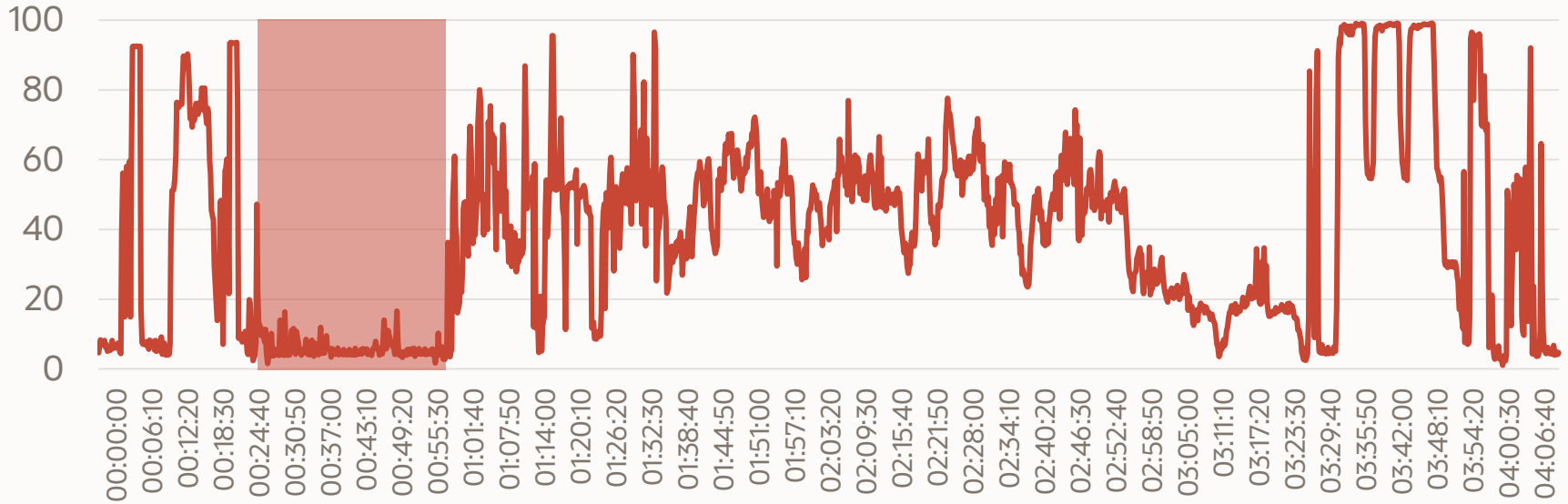




Gathering stats in advance saves 12 minutes

Dictionary and fixed objects

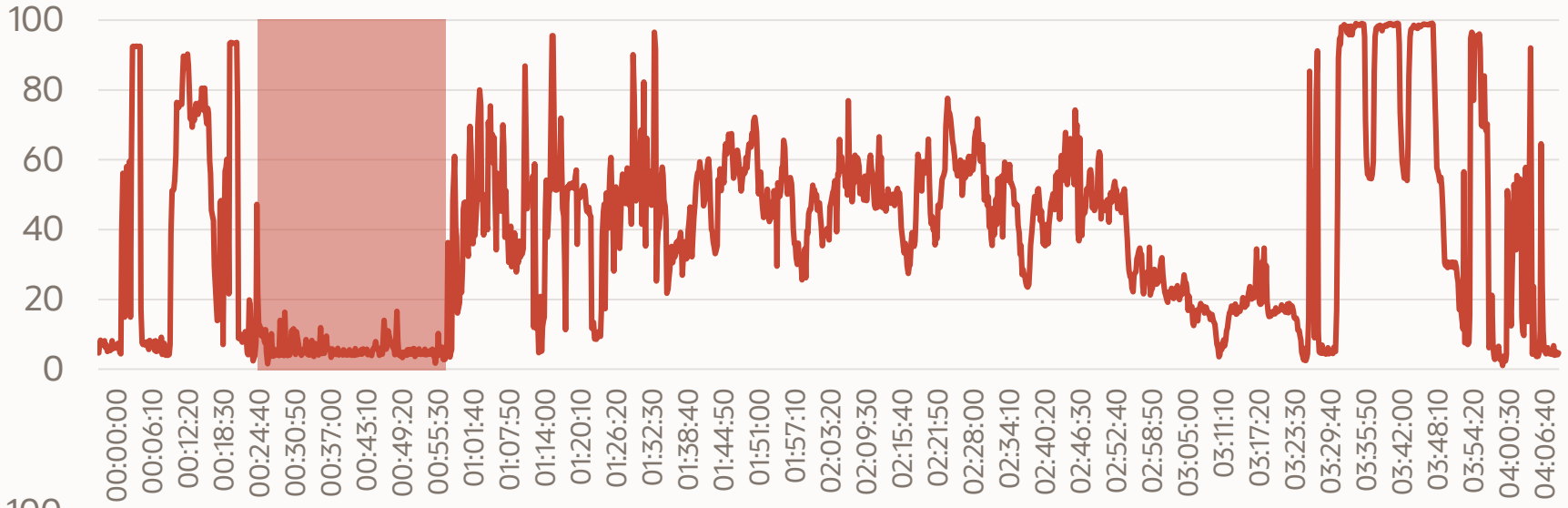
Faster Upgrades | CPU Utilization



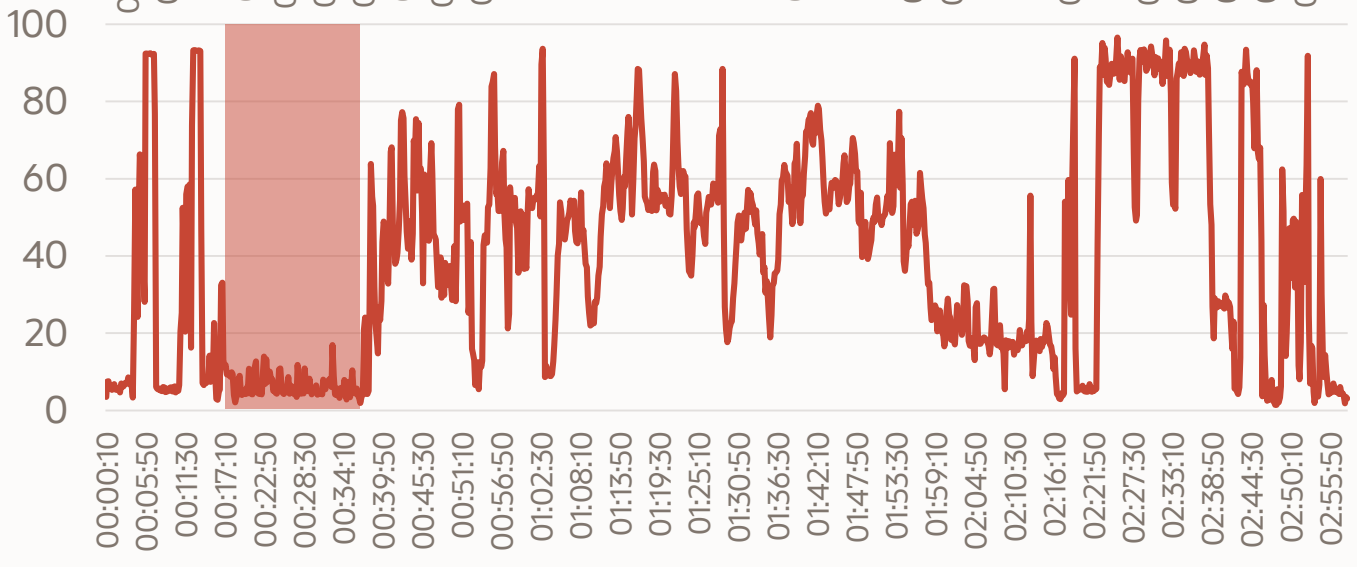
Upgrade CDB$ROOT

- Remove components
- AutoUpgrade automatically assigns 8 parallel processes to CDB$ROOT upgrade

Faster Upgrades | CPU Utilization



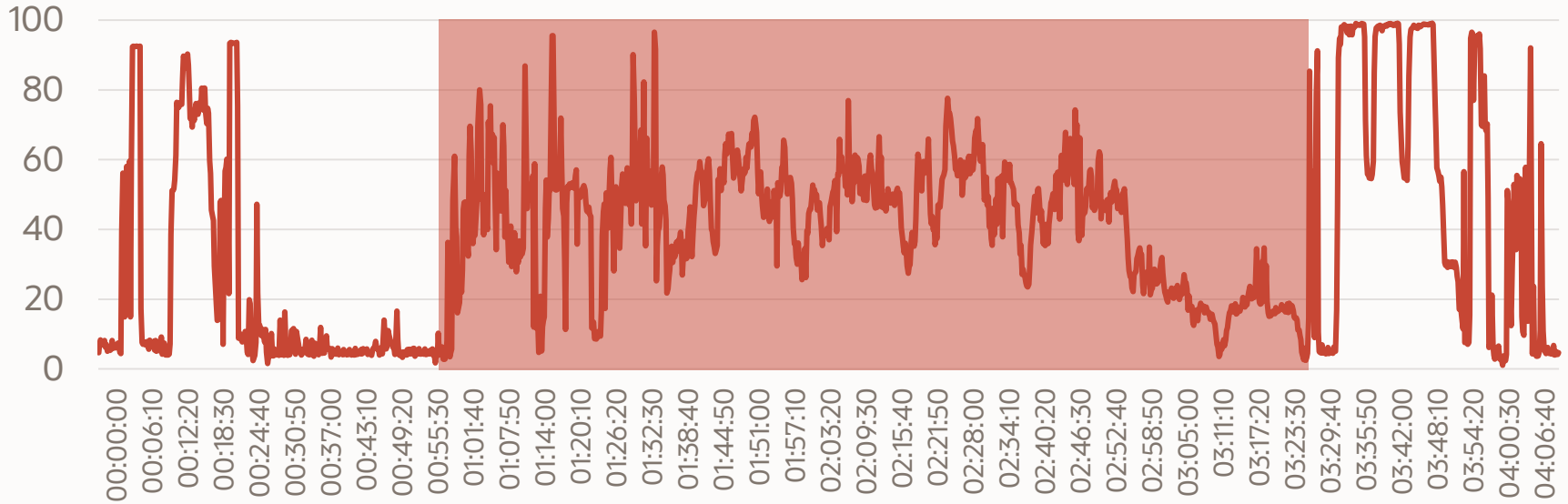
All components installed



Removing all components

13 minutes faster

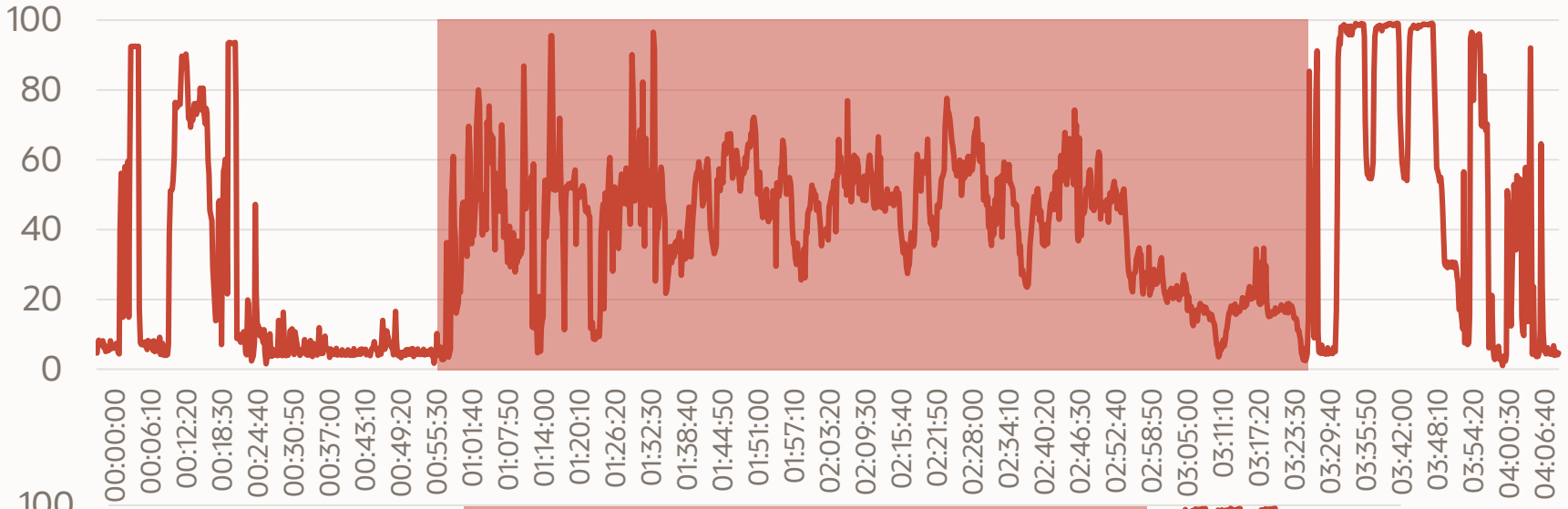
Faster Upgrades | CPU Utilization

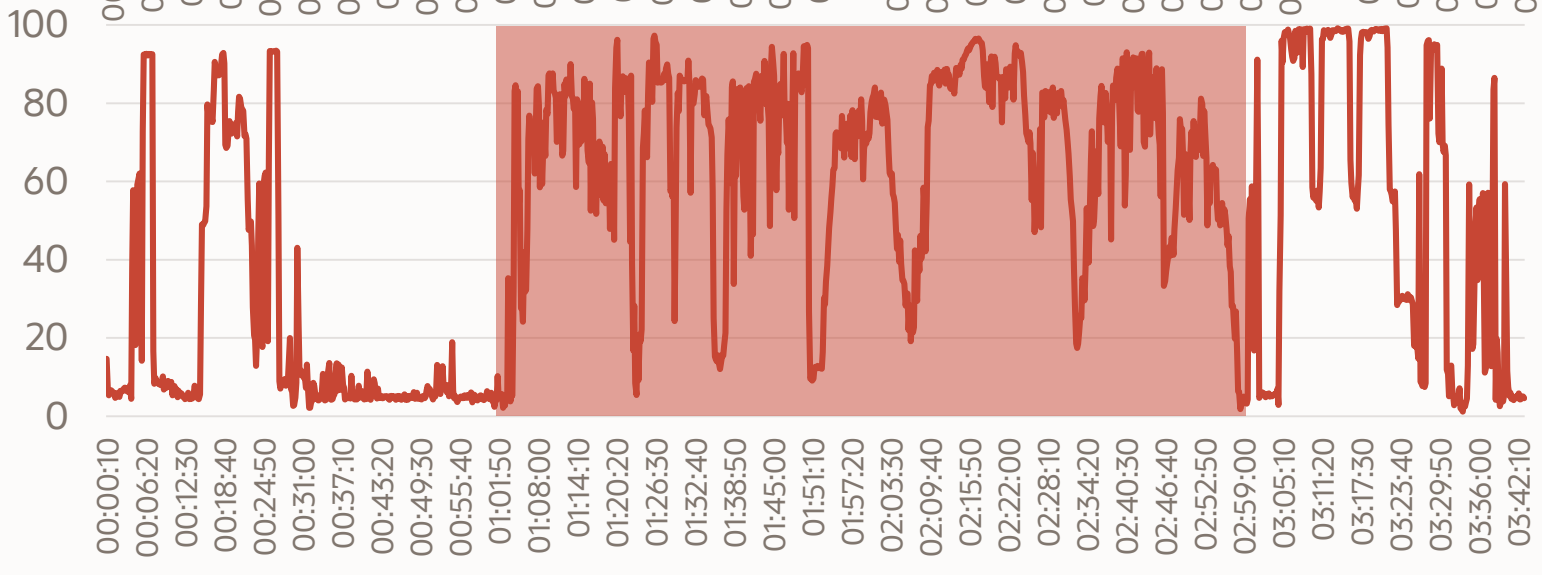


Upgrade PDB$SEED and user PDBs

- Add more PDBs (`catctl -n`)
- Keep parallel processes per PDB at default (2)
- Remove components from PDBs

Faster Upgrades | CPU Utilization





32 parallel processes

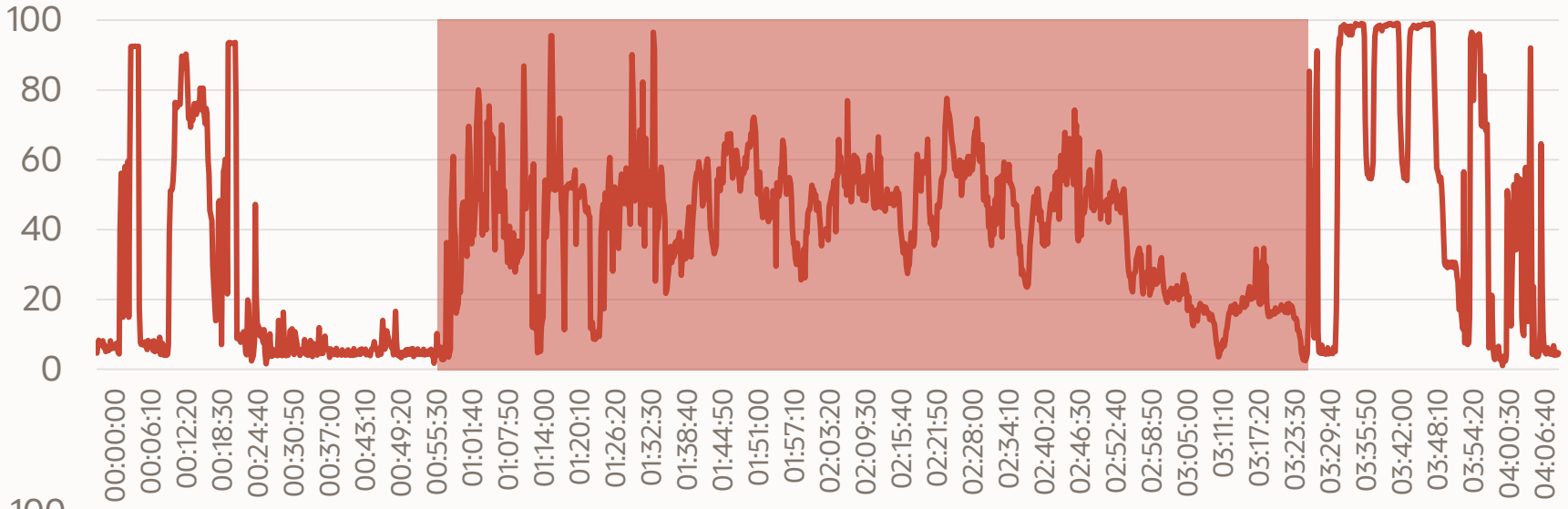
54 parallel processes

`upg1.catctl_options=-n 54`

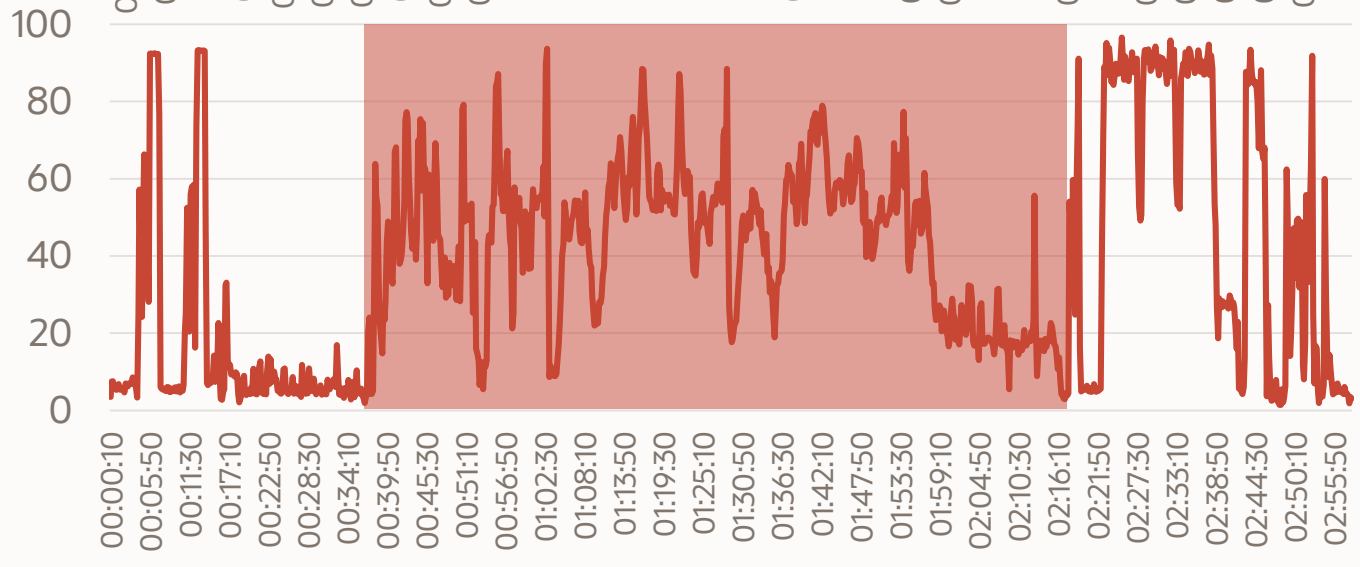
26 minutes faster

Pro tip: Remember to increase `PROCESSES` dramatically

Faster Upgrades | CPU Utilization



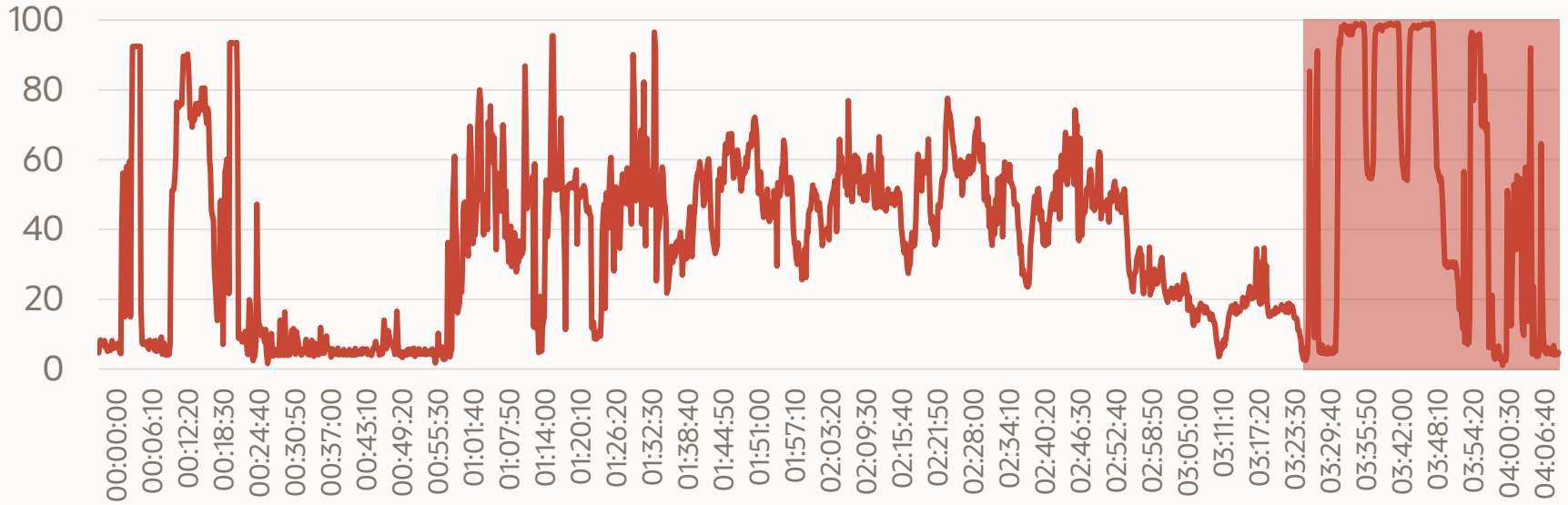
All components installed



Removing all components

48 minutes faster

Faster Upgrades | CPU Utilization

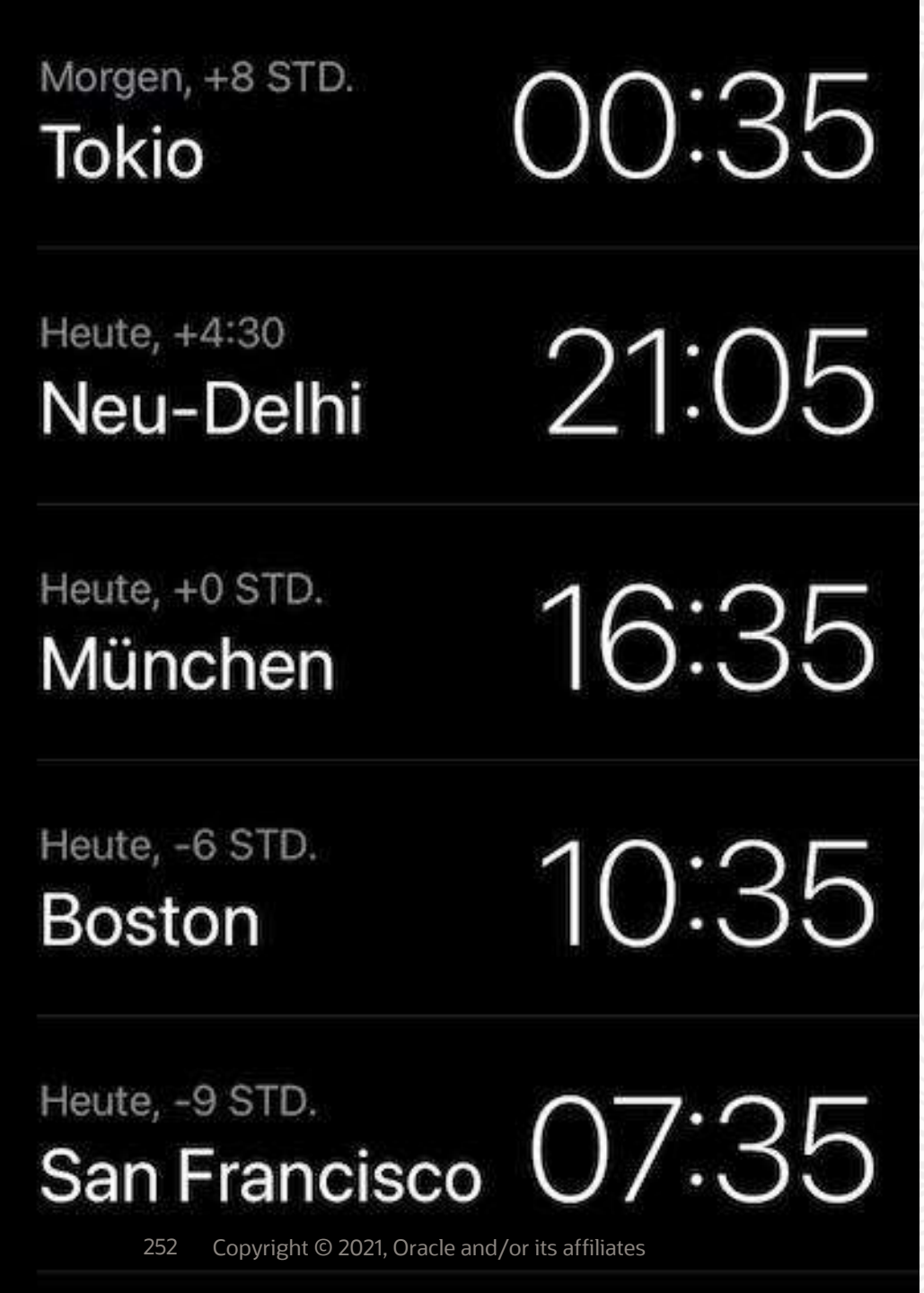


Post upgrade checks and fixups

- Recompilation (`utlrp`) already highly parallelized
- Postpone timezone file upgrade

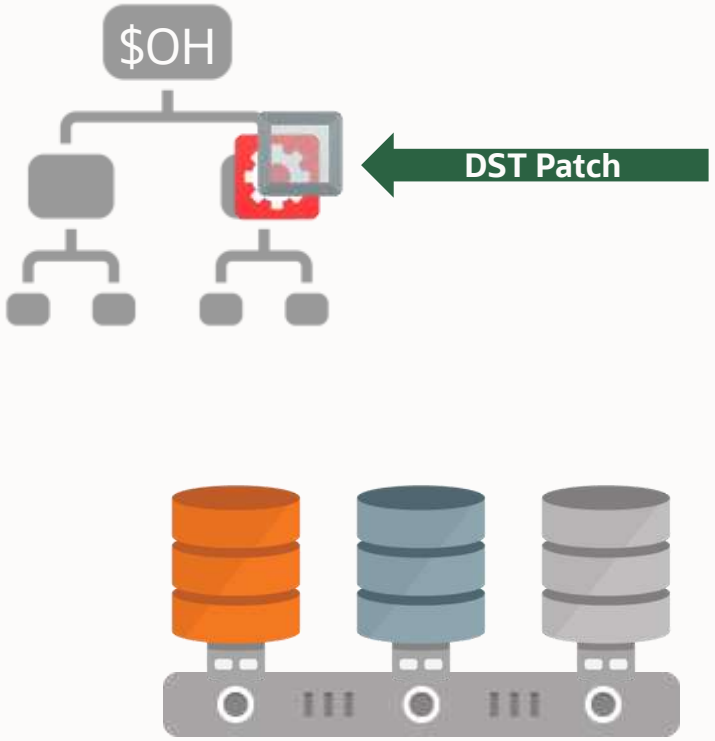
Faster Upgrades | Conclusion

- Gather stats in advance
- Allow more PDBs per cycle
- Remove components



Multitenant and Time Zone Patching

Time Zone | Multitenant DST Version and Patching

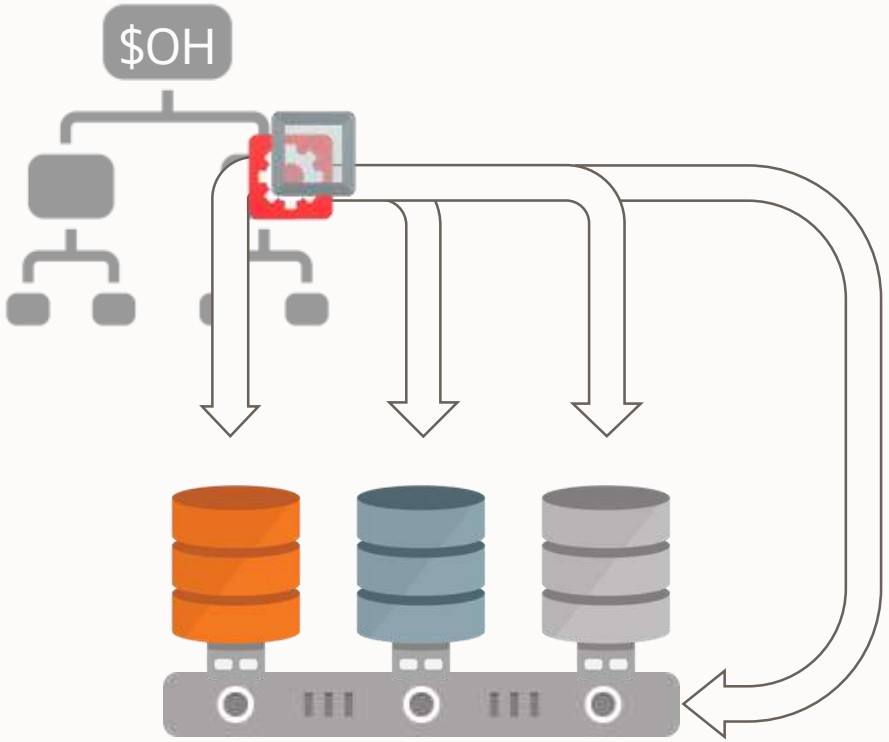


- New 19c CDB gets created with DST V32

Database Release	Default TZ Version
12.1.0.1, 12.1.0.2	DST V18
12.2.0.1	DST V26
18c	DST V31
19c	DST V32
21c	DST V35

- Patching $ORACLE_HOME
  - Containers need to be "TZ upgraded"
  - PDBs and CDB$ROOT can stay on different TZ values

Time Zone | Multitenant Time Zone Upgrade



- Upgrading by default will adjust time zone
  - AutoUpgrade default: `upg1.timezone_upg=yes`
  - 2 restarts will happen
  - Time zone upgrade happens for all containers

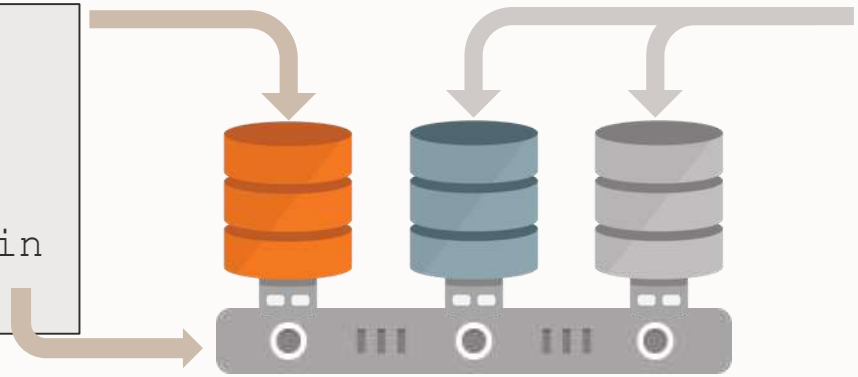
- Manual time zone upgrade is still possible
  - `?/rdbms/admin/utltz_countstar.sql`
  - `?/rdbms/admin/utltz_upg_check.sql`
  - `?/rdbms/admin/utltz_upg_apply.sql`

Time Zone | Updating Time Zone - Check

Download DST patch with: MOS Note:412160.1

Check script:

```
perl catcon.pl -n 1
-c 'CDB$ROOT PDB$SEED'
-l /home/oracle
-b tz_check_ROOT_SEED
-d $ORACLE_HOME/rdbms/admin
utltz_upg_check.sql
```



```
perl catcon.pl -n 1 -S
-l /home/oracle
-b tz_check_PDBs
-d $ORACLE_HOME/rdbms/admin
utltz_upg_check.sql
```

Time Zone | Updating Time Zone - Apply

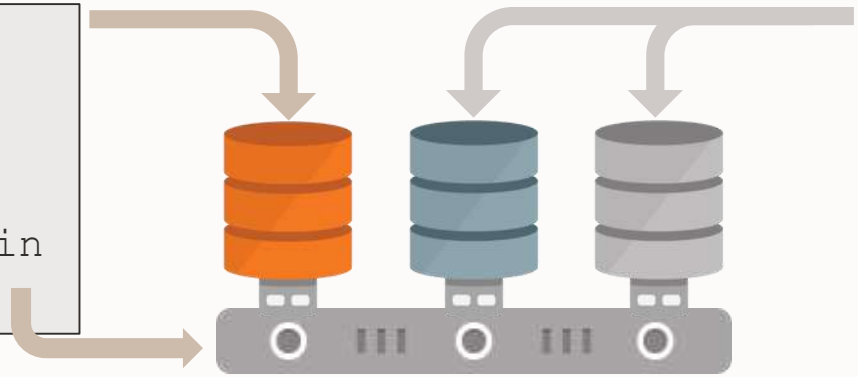
This will restart the database twice, first in UPGRADE mode, then in normal mode

- Exclusive locks may happen

Apply script:

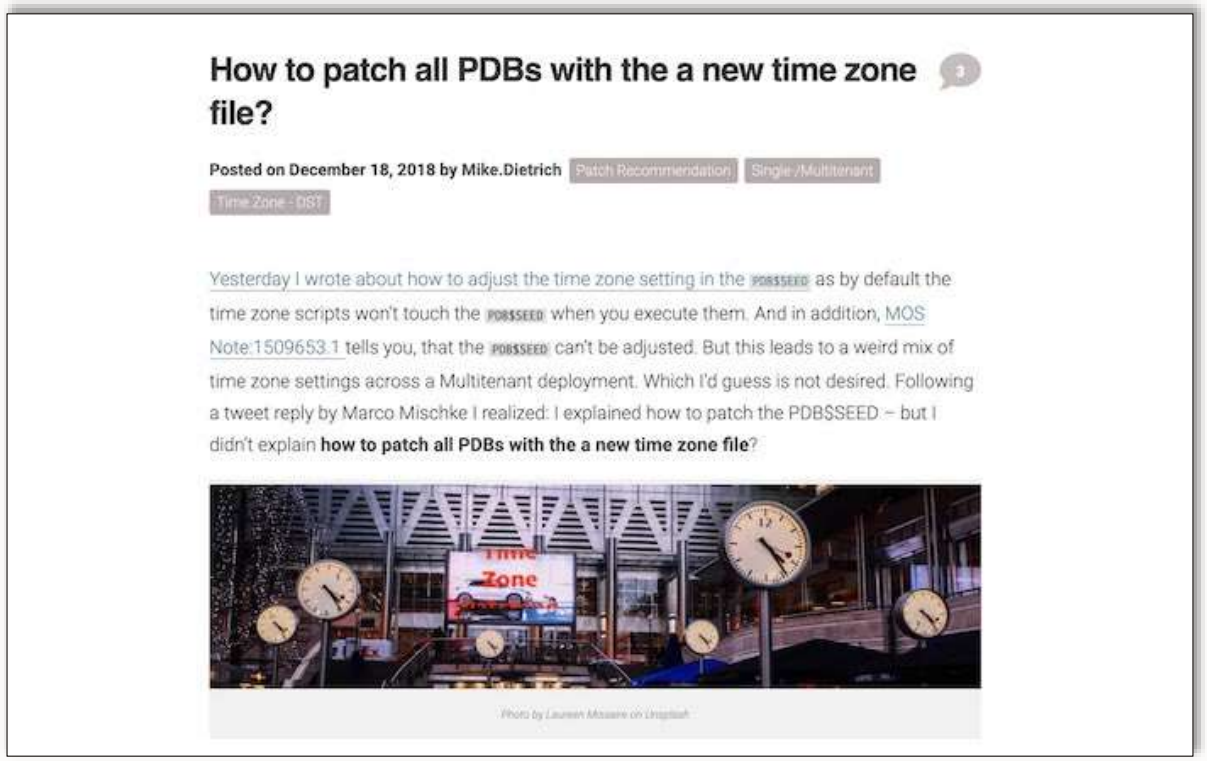
```
SQL> alter pluggable database all open;
```

```
perl catcon.pl -n 1
-c 'CDB$ROOT PDB$SEED'
-l /home/oracle
-b tz_apply_ROOT_SEED
-d $ORACLE_HOME/rdbms/admin
utltz_upg_apply.sql
```



```
perl catcon.pl -n 1 -S
-l /home/oracle
-b tz_apply_PDBs
-d $ORACLE_HOME/rdbms/admin
utltz_upg_apply.sql
```

Time Zone | Summary



How to patch all PDBs with the a new time zone file?

Posted on December 18, 2018 by Mike.Dietrich Patch Recommendation Single-/Multitenant Time Zone - DST

Yesterday I wrote about how to adjust the time zone setting in the PDB$SEED as by default the time zone scripts won't touch the PDB$SEED when you execute them. And in addition, MOS Note:1509653.1 tells you, that the PDB$SEED can't be adjusted. But this leads to a weird mix of time zone settings across a Multitenant deployment. Which I'd guess is not desired. Following a tweet reply by Marco Mischke I realized: I explained how to patch the PDBSSEED – but I didn't explain **how to patch all PDBs with the a new time zone file**?

Photo by Lauren Mitoire on Unsplash

How to patch all your PDBs with a new time zone patch?

Getting ready for the future

Upgrade / Migrate and Consolidate to Multitenant at La Mobilière, Switzerland

"We upgraded 735 databases to 19c, and the task was mostly relatively relaxed.
Start the AutoUpgrade tool and monitor the progress from time to time.

Sitting in front of the screen the whole time is not necessary."

Alain Fuhrer
Head IT Database Services
La Mobilière
Bern, Switzerland

Agenda Day 2

- 13:00-13:15h Review of Day 1 and Results
- 13:15-14:45h Many ways to plugin to Multitenant

- 14:45-15:00h Break

- 15:15-16:45h LAB DAY2:
  Plugin. Unplug/Plug/Upgrade. Migration

- 16:45-17:00h Wrap Up, Review and Goodbye

Agenda Day 2

- 13:00-13:30h Review of Day 1 and Results
- 13:30-14:45h Many ways to plugin to Multitenant

- 14:45-15:00h Break

- 15:15-16:45h LAB DAY2:
  Plugin. Unplug/Plug/Upgrade. Migration

- 16:45-17:00h Wrap Up, Review and Goodbye

Agenda Day 2

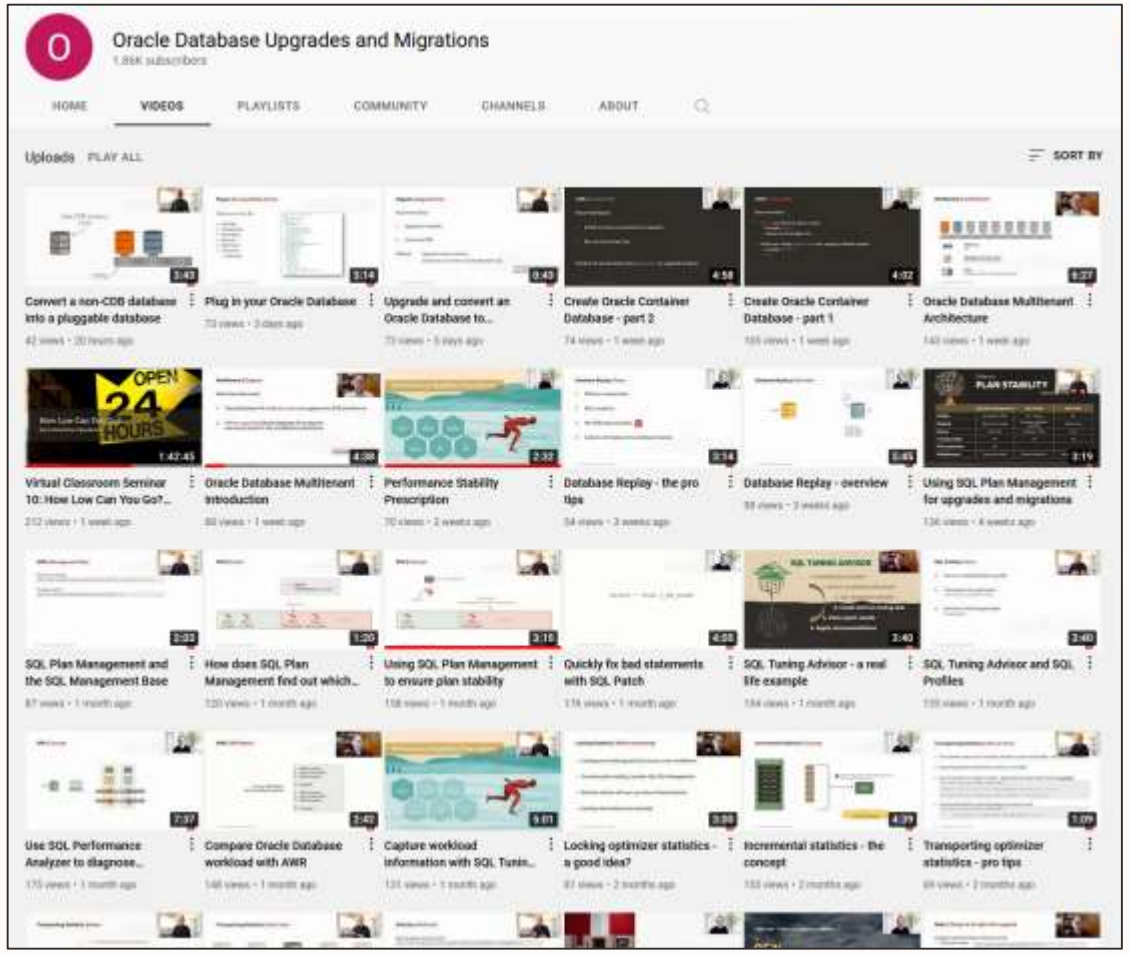
- 13:00-13:15h Review of Day 1 and Results
- 13:15-14:45h Many ways to plugin to Multitenant

- 14:45-15:00h Break

- 15:15-16:45h LAB DAY2:
  Plugin. Unplug/Plug/Upgrade. Migration

- 16:45-17:00h Wrap Up, Review and Goodbye

YouTube | Oracle Database Upgrades and Migrations



Link

- 100+ videos
- New videos every week
- No marketing
- No buzzword
- All tech

SCAN ME

THANK YOU

Visit our blogs:

https://MikeDietrichDE.com

https://DOHdatabase.com

https://www.dbarj.com.br/en

THANK YOU

Webinars:

https://MikeDietrichDE.com/videos

YouTube channel:

OracleDatabaseUpgradesandMigrations

THANK YOU

MIGRATING VERY LARGE DATABASES

Dec 9, 2021 – 10:00h CET

THANK YOU