Photo by Joshua Kettle on Unsplash

How To Upgrade to Oracle AI Database 26ai

DOUG, March 2026

Daniel Overby Hansen

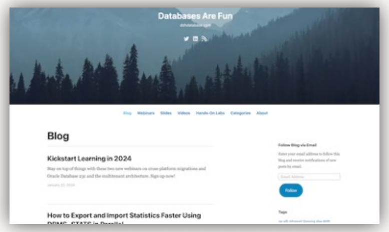
Distinguished Product Manager

in dohdatabase

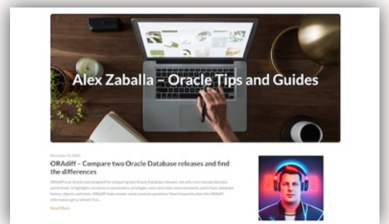
🦋 @dohdatabase.com

B https://dohdatabase.com

Find Slides and Much More on Our Blogs


Databases Are Fun
Blog
Kickstart Learning in 2024
How to Export and Import Statistics Faster Using

DBA - Rodrigo Jorge - Oracle Tips and Guides
What is new in Oracle Database 19.21 RU
How to check the code of the most important Oracle internal objects without having a database
ORACLE


Alex Zaballa – Oracle Tips and Guides
ORAdiff – Compare two Oracle Database releases and find the differences

MikeDietrichDE.com

Mike.Dietrich@oracle.com

dohdatabase.com

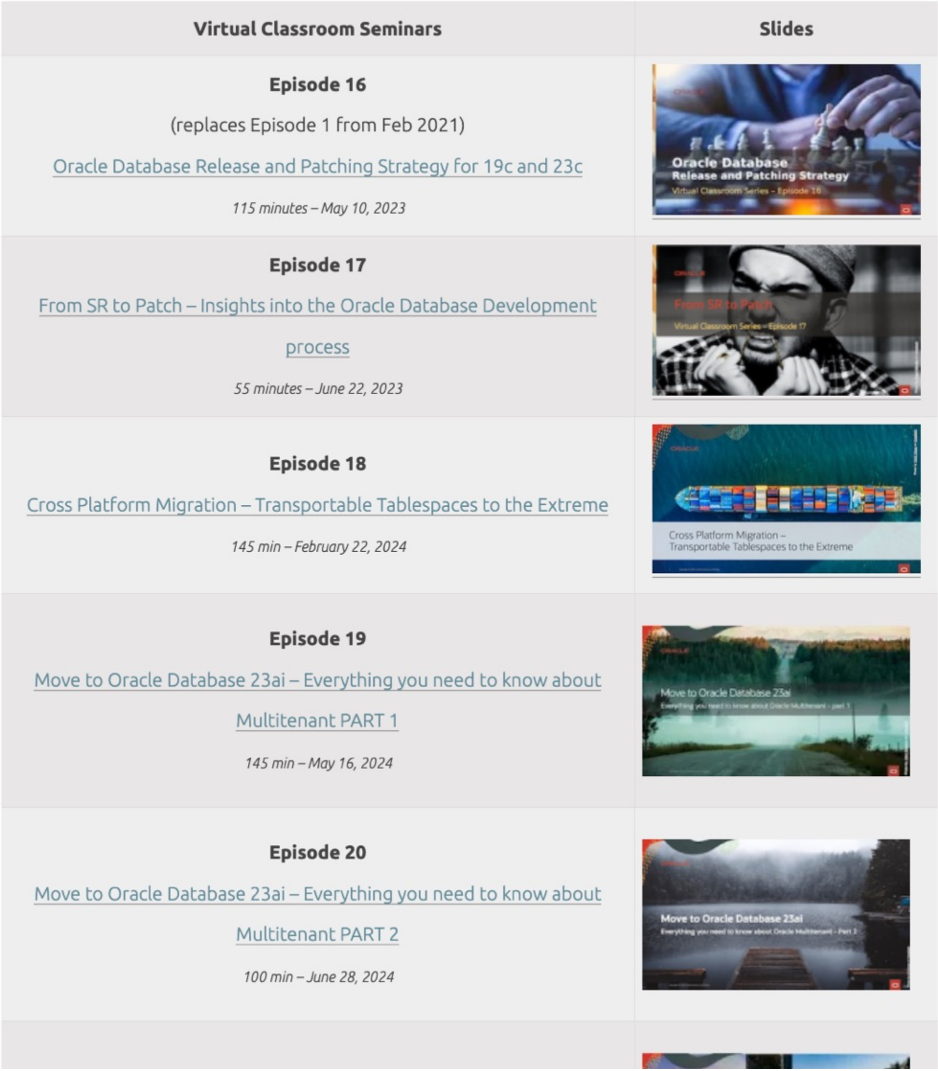
Daniel.Overby.Hansen@oracle.com

DBArj.com.br

Rodrigo.R.Jorge@oracle.com

AlexZaballa.com

Alex.Zaballa@oracle.com



Virtual Classroom Seminars	Slides
Episode 16 (replaces Episode 1 from Feb 2021) Oracle Database Release and Patching Strategy for 19c and 23c *115 minutes – May 10, 2023*	Oracle Database Release and Patching Strategy Virtual Classroom Series – Episode 16
Episode 17 From SR to Patch – Insights into the Oracle Database Development process *55 minutes – June 22, 2023*	From SR to Patch Virtual Classroom Series – Episode 17
Episode 18 Cross Platform Migration – Transportable Tablespaces to the Extreme *145 min – February 22, 2024*	Cross Platform Migration – Transportable Tablespaces to the Extreme
Episode 19 Move to Oracle Database 23ai – Everything you need to know about Multitenant PART 1 *145 min – May 16, 2024*	Move to Oracle Database 23ai Everything you need to know about Oracle Multitenant – part 1
Episode 20 Move to Oracle Database 23ai – Everything you need to know about Multitenant PART 2 *100 min – June 28, 2024*	Move to Oracle Database 23ai Everything you need to know about Oracle Multitenant – Part 2

Recorded Web Seminars

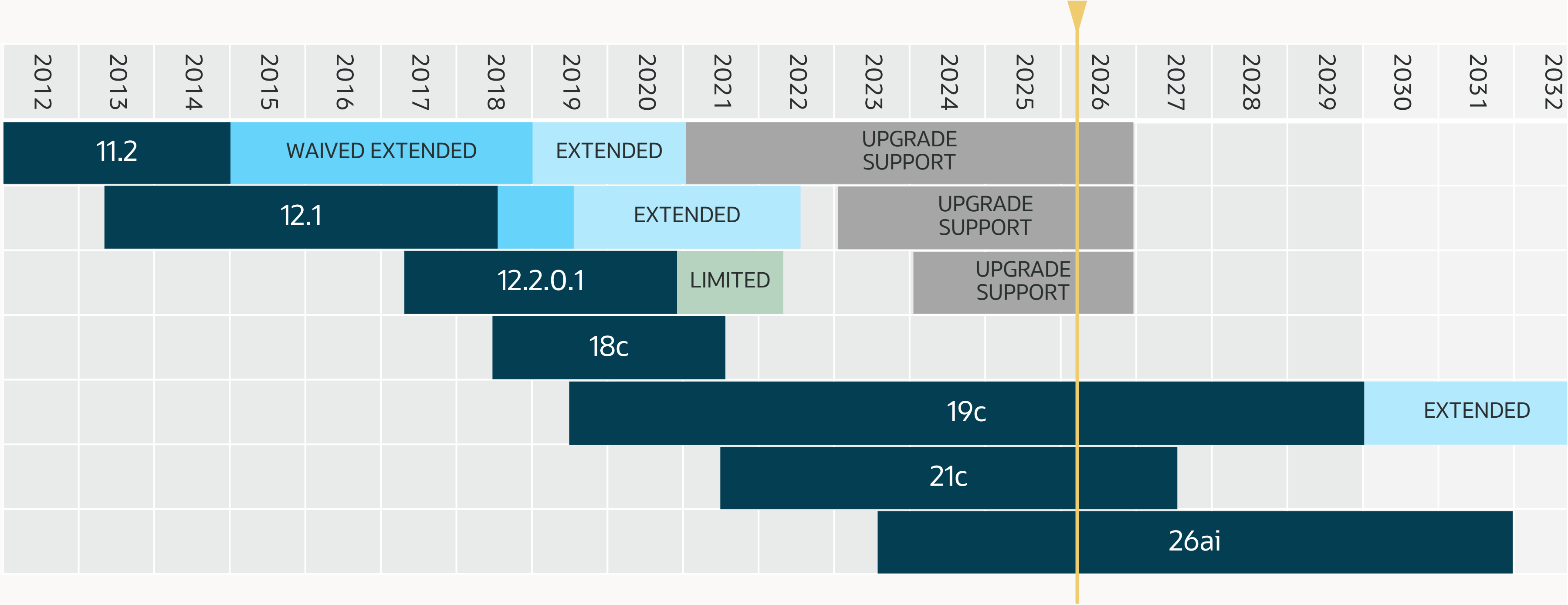
https://MikeDietrichDE.com/videos

More than 40 hours of technical content, on-demand, anytime, anywhere

Release Strategy

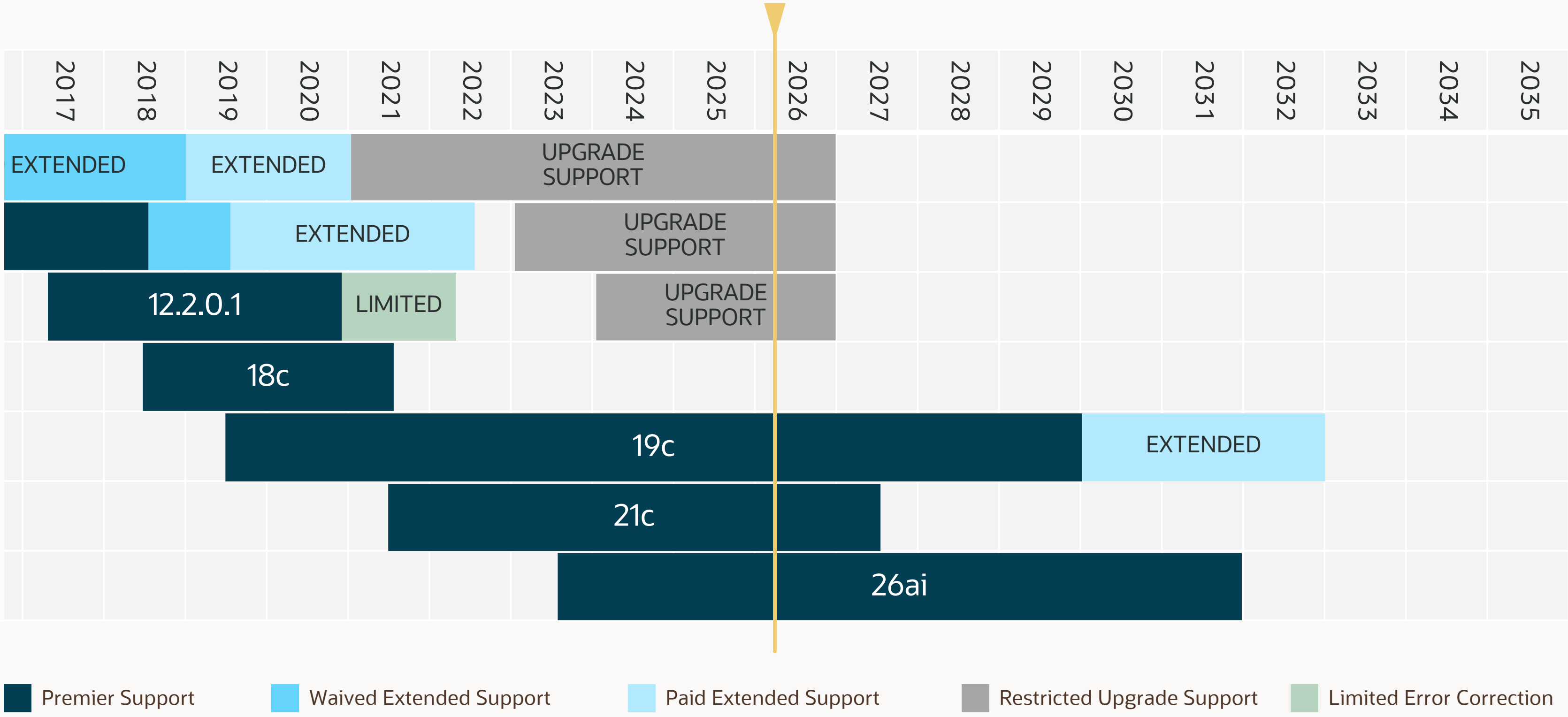
Lifetime Support Policy

Lifetime Support Policy



🟦 Premier Support 🟦 Waived Extended Support 🟦 Paid Extended Support ⬜ Restricted Upgrade Support 🟩 Limited Error Correction

Lifetime Support Policy



26ai

What about the other platforms?

Platforms

Microsoft Windows x64 (64-bit)

IBM AIX on POWER Systems

- CY2026

Oracle Solaris SPARC (64-bit)

IBM Linux on System z

- CY2027

Linux on Arm

- TBA

Release Schedule of Current Database Releases (PNEWS1360)

26^{ai}

There is no need to re-certify your app

- Certification for Oracle Database 23ai applies to Oracle AI Database 26ai

Types of Upgrades

Types of Upgrades

CDB
upgrade

PDB
unplug-plug

Non-CDB
upgrade and convert

CDB Upgrade

- Upgrades entire CDB including all PDBs
- *Many-as-one* principle
- Less work, more automation
- In-place upgrade, no extra resources needed
- Supports Flashback Database
- Seamless Data Guard + RMAN integration
- At least 30-45 minutes, possibly hours

CDB Upgrade

- Less control
- Common SLAs needed

PDB Upgrade

- Upgrades single PDB
- More flexibility
- More control
- Typically 10-30 minutes

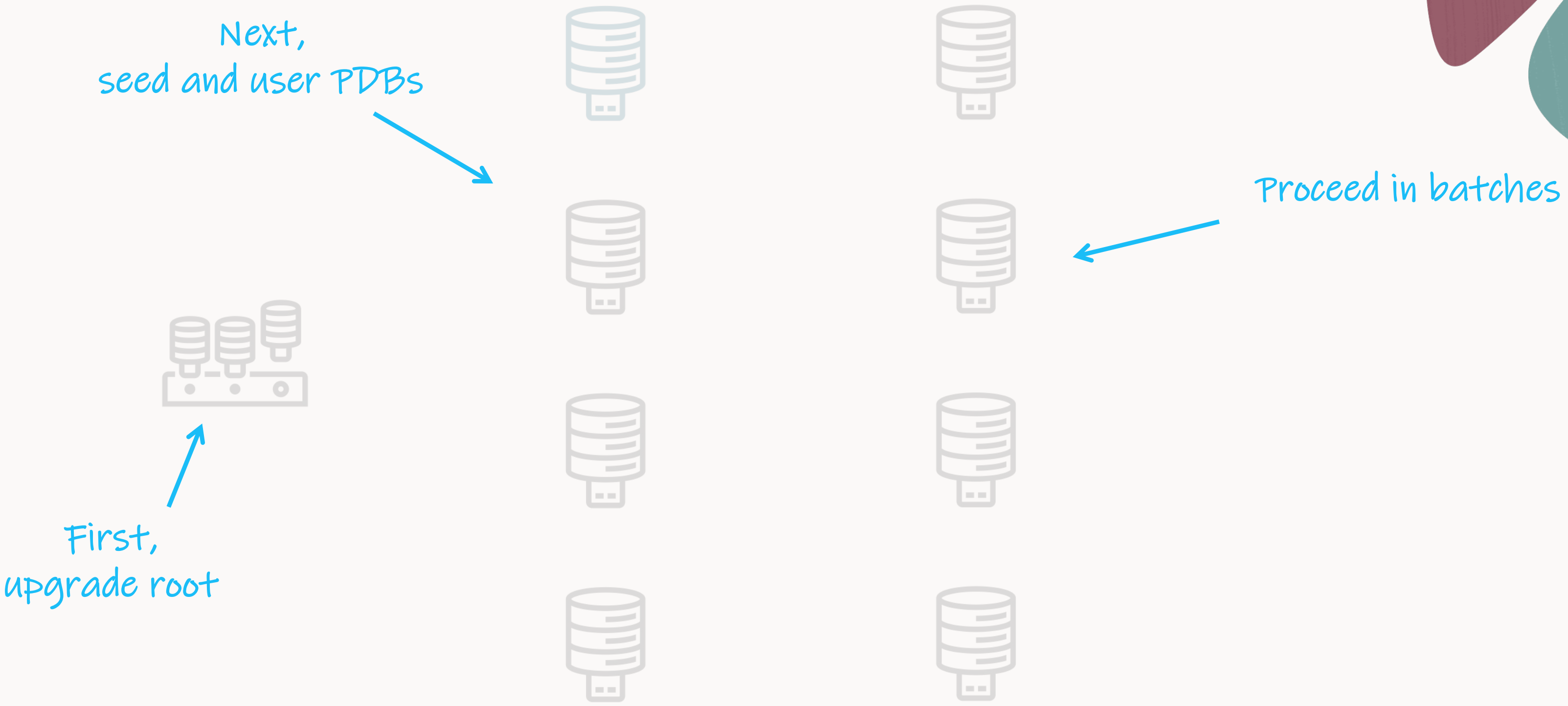
PDB Upgrade

- Additional CDB needed
- Out-of-place upgrade, extra resources needed
- PDB is moved or cloned
- No support for Flashback Database
- Extra work for Data Guard
- Restore between containers require pre-plugin backups

Non-CDB to PDB

- Similar to unplug-plug upgrade
- One-time conversion
- Irreversible
- No support for Flashback Database
- Extra work for Data Guard
- Typically 5-10 minutes
- Restore require pre-plugin backups

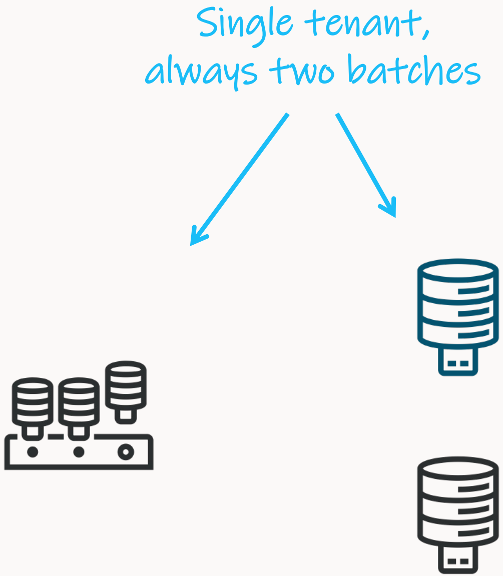
Comparing CDB and PDB upgrades



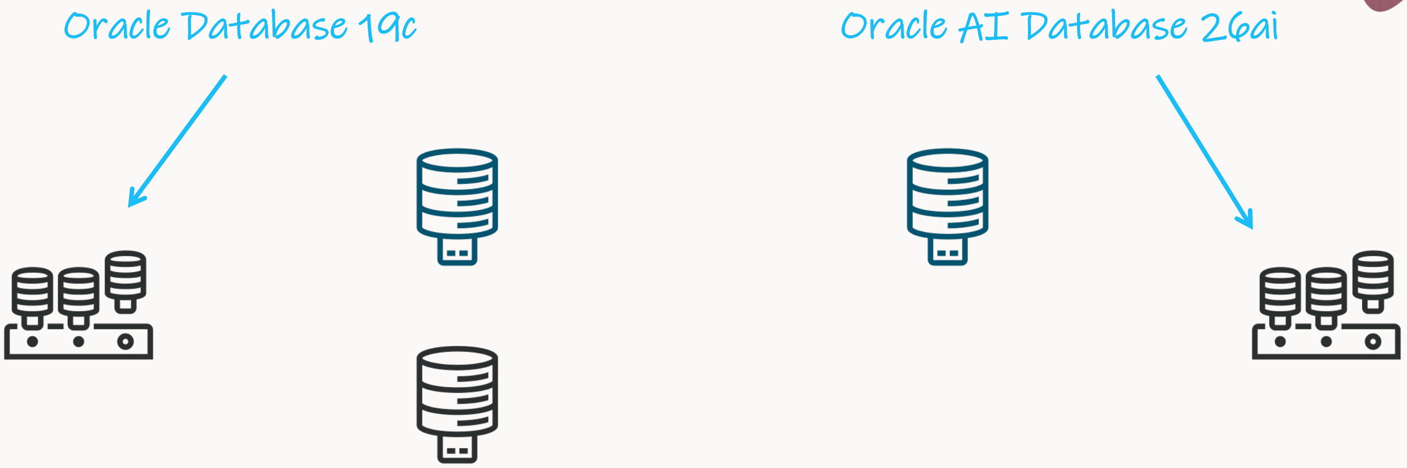
Next,
seed and user PDBs

Proceed in batches

First,
upgrade root



Single tenant,
always two batches



Oracle Database 19c

Oracle AI Database 26ai

👍

Unplug-plug upgrades are faster than upgrading an entire CDB

How long time does it take?

Things that

Matter a lot

- Number of components
- Dictionary size
- Dictionary complexity
- Some feature/version combinations

Things that

Matter a lot

- Number of components
- Dictionary size
- Dictionary complexity
- Some feature/version combinations

Matter a little

- CPU speed
- I/O capabilities
- Memory
- SGA/PGA

Things that

Matter a lot

- Number of components
- Dictionary size
- Dictionary complexity
- Some feature/version combinations

Matter a little

- CPU speed
- I/O capabilities
- Memory
- SGA/PGA

Don't matter

(usually)

- Physical size
- Amount of user data

Comparing UPGRADE types

	CDB upgrade	PDB upgrade	PDB conversion
Time	Longest	Shorter	Very short
Complexity	Easy	Moderate	Moderate
Outage	One big outage	Many smaller outages	Many smaller outages
Rollback	Flashback Database	Other rollback options	Other rollback options
Data Guard + RMAN	Seamless integration	Extra work	Extra work

How To Upgrade

your key to

Successful Database Upgrades

AutoUpgrade

Only supported method for upgrading to Oracle AI Database 26ai

One single tool for everything
- on all platforms

```
-- Always use the latest version of AutoUpgrade
-- Available as a direct download from oracle.com or MOS (KB123450)

wget https://download.oracle.com/otn-pub/otn_software/autoupgrade.jar
```

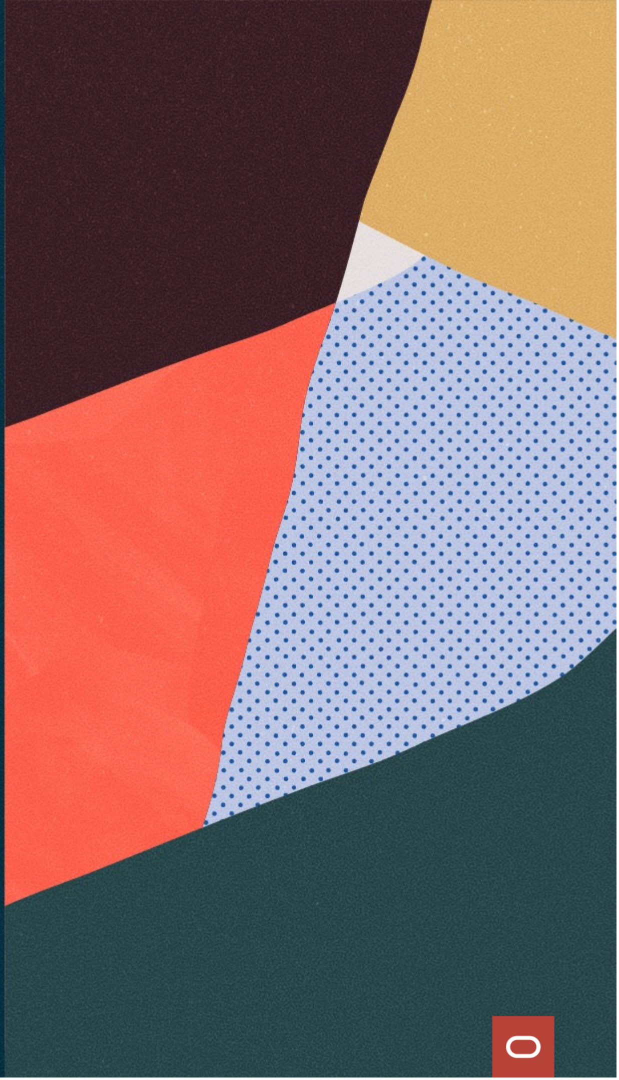
Simplify downloading and installing Oracle software and patches

- Oracle AI Database 26ai comes as fully updated gold images

DEMO

Download patches
Install Oracle home

▶ Watch on YouTube



```
$ cat get-software.cfg

global.global_log_dir=/home/oracle/autoupgrade/logs
global.keystore=/home/oracle/autoupgrade/keystore
install1.target_version=26
install1.patch=RECOMMENDED,37393792
install1.folder=/u01/app/oracle/software
```

```
$ cat get-software.cfg

global.global_log_dir=/home/oracle/autoupgrade/logs
global.keystore=/home/oracle/autoupgrade/keystore
install1.target_version=26
install1.patch=RECOMMENDED,37393792,JDK
install1.folder=/u01/app/oracle/software
```

21c

OJVM is embedded in Release Updates

- No separate download
- Complete RAC Rolling patching support

```
$ cat get-software.cfg

global.global_log_dir=/home/oracle/autoupgrade/logs
global.keystore=/home/oracle/autoupgrade/keystore
install1.target_version=26
install1.patch=RECOMMENDED,37393792,JDK
install1.folder=/u01/app/oracle/software
```

```
java -jar autoupgrade.jar -config get-software.cfg -mode download
```

```
$ cat install-OH.cfg

global.global_log_dir=/home/oracle/autoupgrade/logs
install1.target_version=26
install1.patch=RECOMMENDED,37393792,JDK
install1.folder=/u01/app/oracle/software
install1.download=no
install1.target_home=/u01/app/oracle/product/dbhome_26
install1.home_settings.edition=EE
install1.home_settings.oracle_base=/u01/app/oracle
```

```
$ cat install-OH.cfg

global.global_log_dir=/home/oracle/autoupgrade/logs
install1.target_version=26
install1.patch=RECOMMENDED,37393792,JDK
install1.folder=/u01/app/oracle/software
install1.download=no
install1.target_home=/u01/app/oracle/product/dbhome_26
install1.home_settings.edition=EE
install1.home_settings.oracle_base=/u01/app/oracle
```

```
java -jar autoupgrade.jar -config install-OH.cfg -mode create_home
```

Downloading patches and installing Oracle home

- Blog post: The Easiest Way to Download 19.27 Release Update
- Blog post: Install Oracle Home on Brand-New, Empty Server
- Webinar: Database Patching for DBAs – Patch smarter, not harder

Upgrading an entire CDB

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB19
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
upg1.run_dictionary_health=full
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
upg1.run_dictionary_health=full
upg1.raise_compatible=yes
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
upg1.run_dictionary_health=full
upg1.raise_compatible=yes
upg1.emcli_path=/u01/app/oracle/oem
upg1.em_target_name=CDB1_myhost.domain.int
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
upg1.run_dictionary_health=full
upg1.raise_compatible=yes
upg1.emcli_path=/u01/app/oracle/oem
upg1.em_target_name=CDB1_myhost.domain.int
upg1.rman_catalog_connect_string=catalogdb
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
upg1.run_dictionary_health=full
upg1.raise_compatible=yes
upg1.emcli_path=/u01/app/oracle/oem
upg1.em_target_name=CDB1_myhost.domain.int
upg1.rman_catalog_connect_string=catalogdb
upg1.add_after_upgrade_pfile=/home/oracle/scripts/CDB1_add.ora
upg1.delete_before_upgrade_pfile=/home/oracle/scripts/CDB1_remove.ora
```

Do you remember to gather
fixed objects statistics after upgrade?

```
--Gather fixed objects statistics after the upgrade
--The database must be warmed up and subjected to a representative workload

exec dbms_stats.gather_fixed_objects_stats;
```

```
cat sched_gfos.sql
cat sched_gfos.sql

BEGIN
   DBMS_SCHEDULER.CREATE_JOB (
      job_name => '"SYS"."GATHER_FIXED_OBJECTS_STATS_ONE_TIME"',
        job_type => 'PLSQL_BLOCK',
        job_action => 'BEGIN DBMS_STATS.GATHER_FIXED_OBJECTS_STATS; END;',
        start_date => SYSDATE+7,
        auto_drop => TRUE,
        comments => 'Gather fixed objects stats after upgrade - one time',
         enabled => TRUE
   );
END;
/
```

```
cat sched_gfos.sql

BEGIN
   DBMS_SCHEDULER.CREATE_JOB (
      job_name => '"SYS"."GATHER_FIXED_OBJECTS_STATS_ONE_TIME"',
         job_type => 'PLSQL_BLOCK',
         job_action => 'BEGIN DBMS_STATS.GATHER_FIXED_OBJECTS_STATS; END;',
         start_date => SYSDATE+7,
         auto_drop => TRUE,
         comments => 'Gather fixed objects stats after upgrade - one time',
         enabled => TRUE
   );
END;
/
```

```
cat sched_gfos.sql

BEGIN
   DBMS_SCHEDULER.CREATE_JOB (
      job_name => '"SYS"."GATHER_FIXED_OBJECTS_STATS_ONE_TIME"',
        job_type => 'PLSQL_BLOCK',
        job_action => 'BEGIN DBMS_STATS.GATHER_FIXED_OBJECTS_STATS; END;',
        start_date => SYSDATE+7,
        auto_drop => TRUE,
        comments => 'Gather fixed objects stats after upgrade - one time',
         enabled => TRUE
   );
END;
/
```

```
cat sched_gfos.sql

BEGIN
   DBMS_SCHEDULER.CREATE_JOB (
      job_name => '"SYS"."GATHER_FIXED_OBJECTS_STATS_ONE_TIME"',
         job_type => 'PLSQL_BLOCK',
         job_action => 'BEGIN DBMS_STATS.GATHER_FIXED_OBJECTS_STATS; END;',
         start_date => SYSDATE+7,
         auto_drop => TRUE,
         comments => 'Gather fixed objects stats after upgrade - one time',
         enabled => TRUE
   );
END;
/
```

```
cat sched_gfos.sh
cat sched_gfos.sh

export ORACLE_SID=CDB1
export ORACLE_HOME=/u01/app/oracle/product/26
export PATH=$ORACLE_HOME/bin:$PATH
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl \
   -n 4 -C 'PDB$SEED' -e \
   -b sched_gfos -d /home/oracle/sched_gfos/ sched_gfos.sql
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB1
upg1.timezone_upg=no
upg1.run_dictionary_health=full
upg1.raise_compatible=yes
upg1.emcli_path=/u01/app/oracle/oem
upg1.em_target_name=CDB1_myhost.domain.int
upg1.rman_catalog_connect_string=catalogdb
upg1.add_after_upgrade_pfile=/home/oracle/scripts/CDB1_add.ora
upg1.delete_before_upgrade_pfile=/home/oracle/scripts/CDB1_remove.ora
upg1.after_action=/home/oracle/scripts/sched_gfos.sh N
```

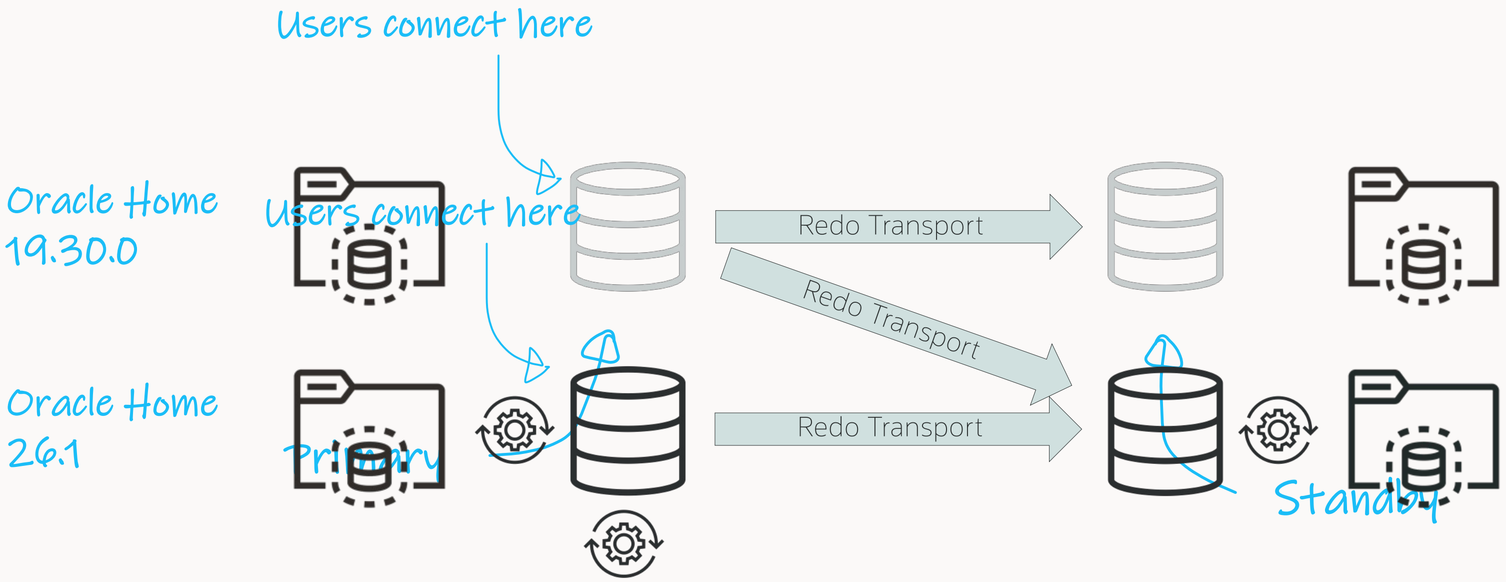
```
java -jar autoupgrade.jar -config CDB1.cfg -mode analyze
java -jar autoupgrade.jar -config CDB1.cfg -mode analyze
```

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

We have an Oracle RAC Database

We have Oracle Data Guard

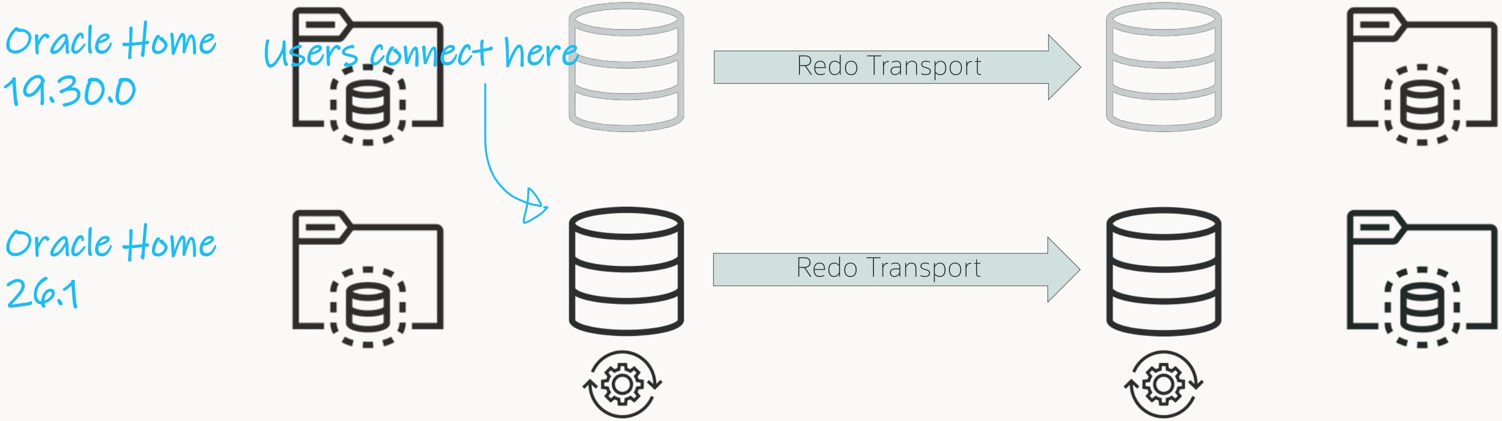
Upgrade Data Guard - MAA



```
java -jar autoupgrade.jar \
   -config ... \
   -mode deploy
```

```
java -jar autoupgrade.jar \
   -config ... \
   -mode deploy
```

Upgrade Data Guard - Standby Offline



```
java -jar autoupgrade.jar \
   -config ... \
   -mode deploy
```

```
java -jar autoupgrade.jar \
   -config ... \
   -mode deploy
```

Upgrading a single PDB

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES,HR,REPORTING
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
upg1.target_pdb_name.SALES=SALES26
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
upg1.target_pdb_name.SALES=SALES26
upg1.raise_compatible=yes
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
upg1.target_pdb_name.SALES=SALES26
~~upg1.raise_compatible=yes~~
```

Think about your rollback plans

- Unplug-plug upgrades doesn't support Flashback Database

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
upg1.target_pdb_copy_option.SALES=FILE_NAME_CONVERT=NONE
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
upg1.target_pdb_copy_option.SALES=FILE_NAME_CONVERT=('/CDB19','/CDB26')
```

```
global.global_log_dir=/home/oracle/autoupgrade/logs/CDB1
upg1.source_home=/u01/app/oracle/product/19
upg1.target_home=/u01/app/oracle/product/26
upg1.sid=CDB19
upg1.target_cdb=CDB26
upg1.pdbs=SALES
~~upg1.target_pdb_copy_option.SALES=FILE_NAME_CONVERT=('/SALES','/SALES26')~~
```

We have an Oracle RAC Database

We have Oracle Data Guard

Copy Data Files

- If you want to copy the data files, you must use *deferred recovery*

```
upg1.target_pdb_copy_option.SALES=FILE_NAME_CONVERT=NONE
upg1.manage_standbys_clause=standbys=none
```

- Restore PDB on standby after upgrade

```
RMAN> restore pluggable database SALES from service ... ;

SQL> alter pluggable database enable recovery;
SQL> alter database datafile ... online;
```

Reuse Data Files

- If you want to reuse the data files, you must use *enabled recovery*

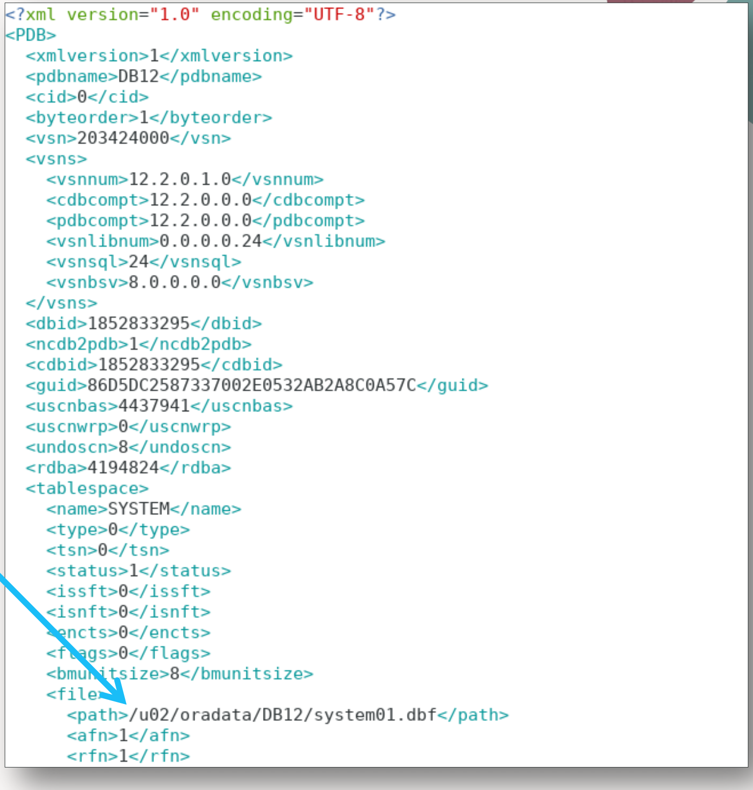
~~`upg1.target_pdb_copy_option.SALES=FILE_NAME_CONVERT=NONE`~~

`upg1.manage_standbys_clause=standbys=all`

- But how does the standby know where the data files are?

Reuse Data Files

- The plug-in happens on the primary database
- The plug-in uses the *manifest file*
- The manifest file contains information on data files from the primary database only



```
<?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>
```

Enabled Recovery

How does the standby database know which files to plug in?

1 Regular files

2 OMF in regular file system

3 ASM

AutoUpgrade

- Documentation: Upgrade Parameters for the AutoUpgrade Configuration File
- Blog post: How to Detect and Repair Corruption in Your Oracle Database
- Blog post: Changing COMPATIBLE Parameter and Data Guard
- Blog post: AutoUpgrade New Features: Update Enterprise Manager Configuration
- Blog post: AutoUpgrade New Features: Upgrade RMAN Catalog Schema
- Blog post: Upgrade to Oracle AI Database 26ai

How to generate the config file

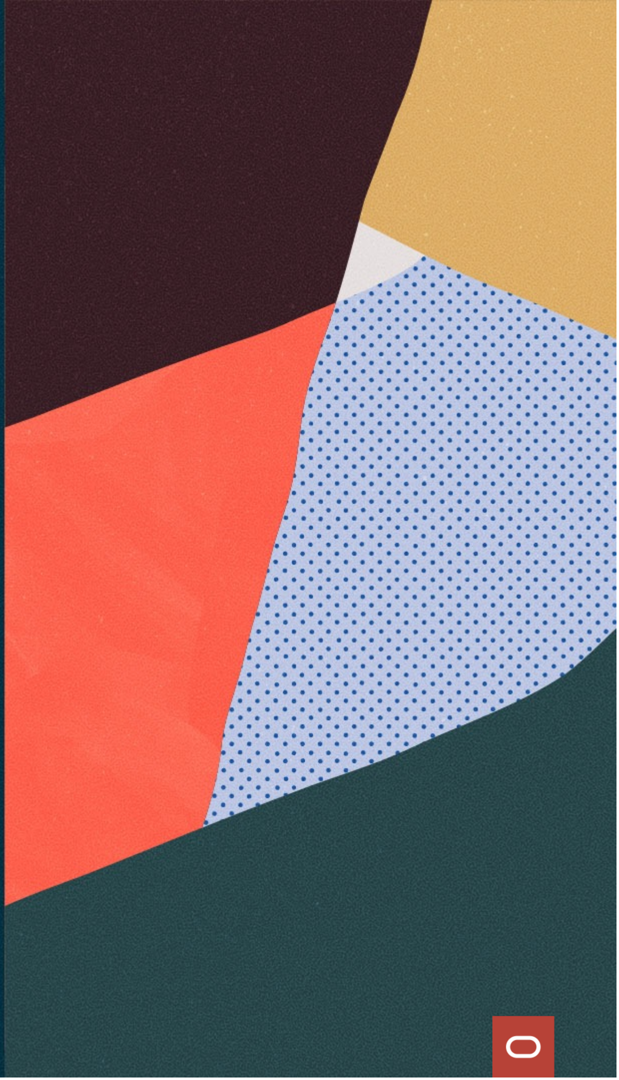
```
$ env | grep ORA

   ORACLE_SID=CDB19
   ORACLE_BASE=/u01/app/oracle
   ORACLE_HOME=/u01/app/oracle/product/19
```

```
$ java -jar autoupgrade.jar -auto_config
```

DEMO

Generate config file

Watch on YouTube



AutoUpgrade Composer

Marcus Vinicius Miguel Pedro

Practice Director LATAM at Accenture

Oracle ACE Director

Other Upgrade Methods

Other Upgrade Methods

- Refreshable clone PDB

- Data Pump
- Transportable Tablespaces
- Classic export/import

- Rolling upgrades
- Oracle GoldenGate

Other Upgrade Methods

- Webinar: Virtual Classroom #10: Zero Downtime Operations of Oracle Database

Replay Upgrade

```
--The database automatically starts an upgrade
--when you plug in a lower-release PDB

SQL> alter pluggable database pdb1 open;

Pluggable database altered.

Elapsed: 00:06:01.95
```

```
SQL> select property_name, property_value
     from   database_properties
     where  property_name like '%OPEN%';

PROPERTY_NAME          PROPERTY_VALUE
---------------------- --------------------
CONVERT_NONCDB_ON_OPEN true
UPGRADE_PDB_ON_OPEN    true
```

```
SQL> alter pluggable database pdb1 open;
alter pluggable database pdb1 open
*
ERROR at line 1:
ORA-60510: encountered an error during Replay Upgrade
```

If Replay Upgrade fails

- Check for errors:
  - `SELECT * FROM dba_replay_upgrade_errors`
  - `SELECT * FROM dba_app_errors`
  - `SELECT * FROM dba_applications WHERE app_name='APP$CDB$CATALOG';`
  - Check alert log
  - Trace files

- Revert to classic upgrade
  - Use AutoUpgrade (`upg1.replay=no`)

```
--To disable replay upgrade
ALTER DATABASE UPGRADE SYNC OFF;

--Or
ALTER DATABASE PROPERTY SET UPGRADE_PDB_ON_OPEN='false';

--To disable convert on open
ALTER DATABASE PROPERTY SET CONVERT_NONCDB_ON_OPEN='false';
```

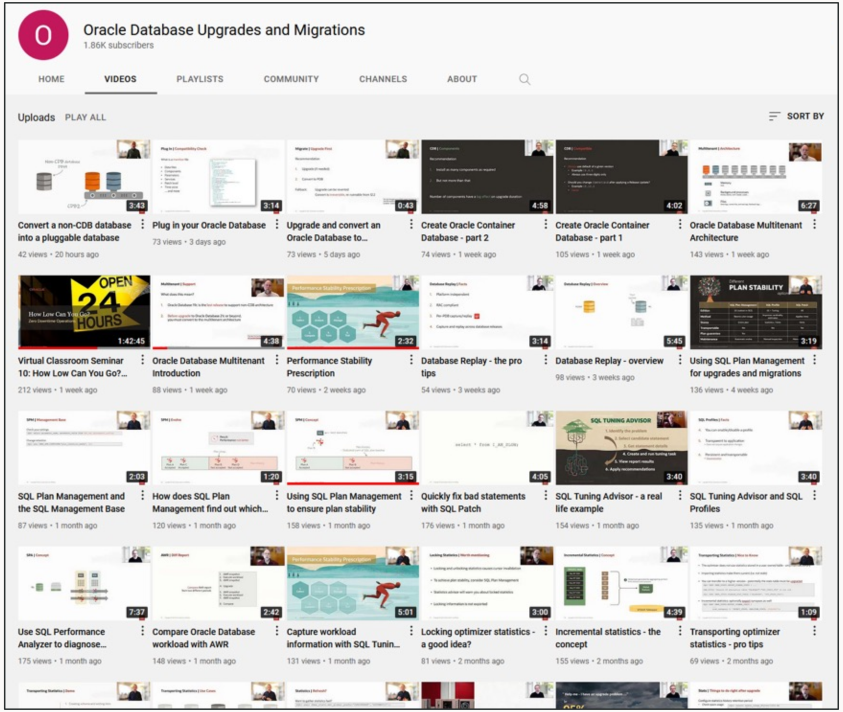
It's better to fail in our lab, than in production

Oracle LiveLabs:
Hitchhiker's Guide for Upgrading to Oracle AI Database

Key Learnings

1. Use AutoUpgrade
2. Easy or fast
3. Use AutoUpgrade Composer

YouTube | Oracle Database Upgrades and Migrations



Oracle Database Upgrades and Migrations
1.86K subscribers

- 3:43 Convert a non-CDB database into a pluggable database — 42 views • 20 hours ago
- 3:14 Plug in your Oracle Database — 73 views • 3 days ago
- 0:43 Upgrade and convert an Oracle Database to... — 73 views • 5 days ago
- 4:58 Create Oracle Container Database - part 2 — 74 views • 1 week ago
- 4:02 Create Oracle Container Database - part 1 — 105 views • 1 week ago
- 6:27 Oracle Database Multitenant Architecture — 143 views • 1 week ago
- 1:42:45 Virtual Classroom Seminar 10: How Low Can You Go?... — 212 views • 1 week ago
- 4:38 Oracle Database Multitenant Introduction — 88 views • 1 week ago
- 2:32 Performance Stability Prescription — 70 views • 2 weeks ago
- 3:14 Database Replay - the pro tips — 54 views • 3 weeks ago
- 5:45 Database Replay - overview — 98 views • 3 weeks ago
- 3:19 Using SQL Plan Management for upgrades and migrations — 136 views • 4 weeks ago
- 2:03 SQL Plan Management and the SQL Management Base — 87 views • 1 month ago
- 1:20 How does SQL Plan Management find out which... — 120 views • 1 month ago
- 3:15 Using SQL Plan Management to ensure plan stability — 158 views • 1 month ago
- 4:05 Quickly fix bad statements with SQL Patch — 176 views • 1 month ago
- 3:40 SQL Tuning Advisor - a real life example — 154 views • 1 month ago
- 3:40 SQL Tuning Advisor and SQL Profiles — 135 views • 1 month ago
- 7:37 Use SQL Performance Analyzer to diagnose... — 175 views • 1 month ago
- 2:42 Compare Oracle Database workload with AWR — 148 views • 1 month ago
- 5:01 Capture workload information with SQL Tunin... — 131 views • 1 month ago
- 3:00 Locking optimizer statistics - a good idea? — 81 views • 2 months ago
- 4:39 Incremental statistics - the concept — 155 views • 2 months ago
- 1:09 Transporting optimizer statistics - pro tips — 69 views • 2 months ago

https://www.youtube.com/@upgradenow

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

SCAN ME

ORACLE