ORACLE

Oracle 21c Cool Features

2021 APAC Groundbreakers Virtual Tour

Photo by Griffin Wooldridge on Unsplash

Mike Dietrich

Distinguished Product Manager
Database Upgrade and Migrations

https://MikeDietrichDE.com
MikeDietrich
@MikeDietrichDE

Daniel Overby Hansen

Senior Principal Product Manager
Database Cloud Migrations

https://dohdatabase.com
dohdatabase
@dohdatabase

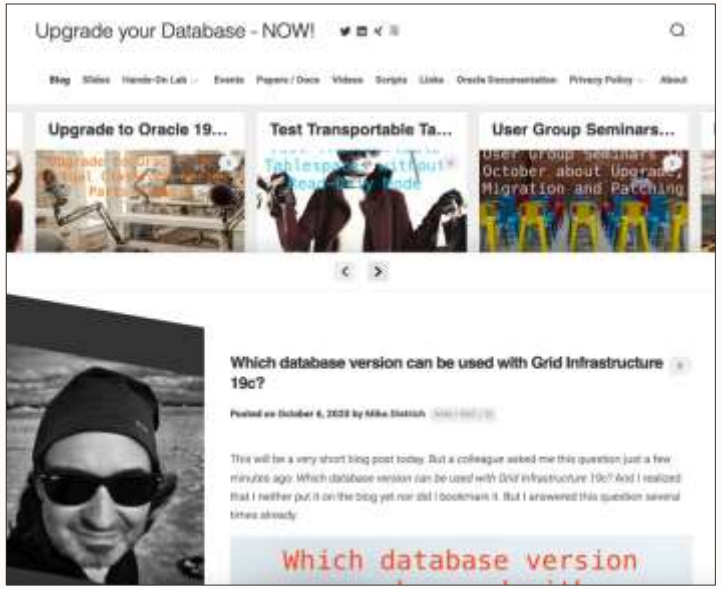
Get the slides

https://dohdatabase.com/slides

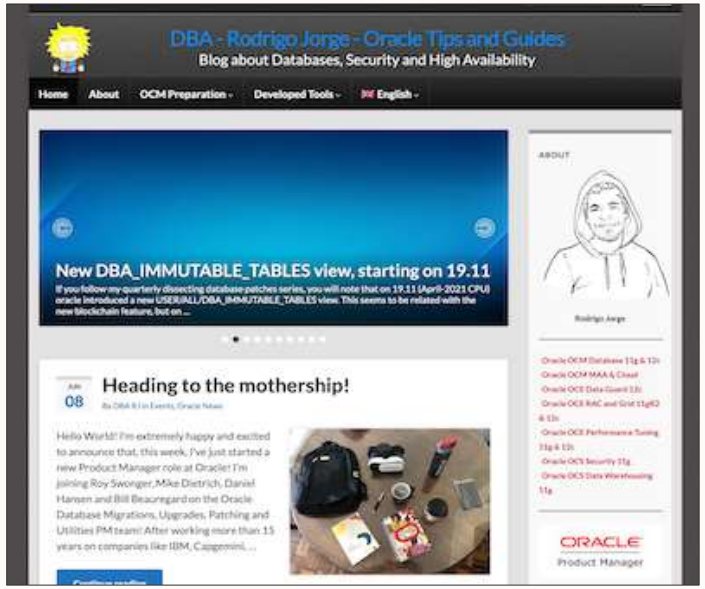
https://MikeDietrichDE.com/slides

Visit our Blogs

https://MikeDietrichDE.com



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



https://DOHdatabase.com

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

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

Restore

Gradual Password Rollover

Read Only Oracle Homes | Overview

Not a new feature, but as of Oracle Database 21 it is the default Oracle Home type

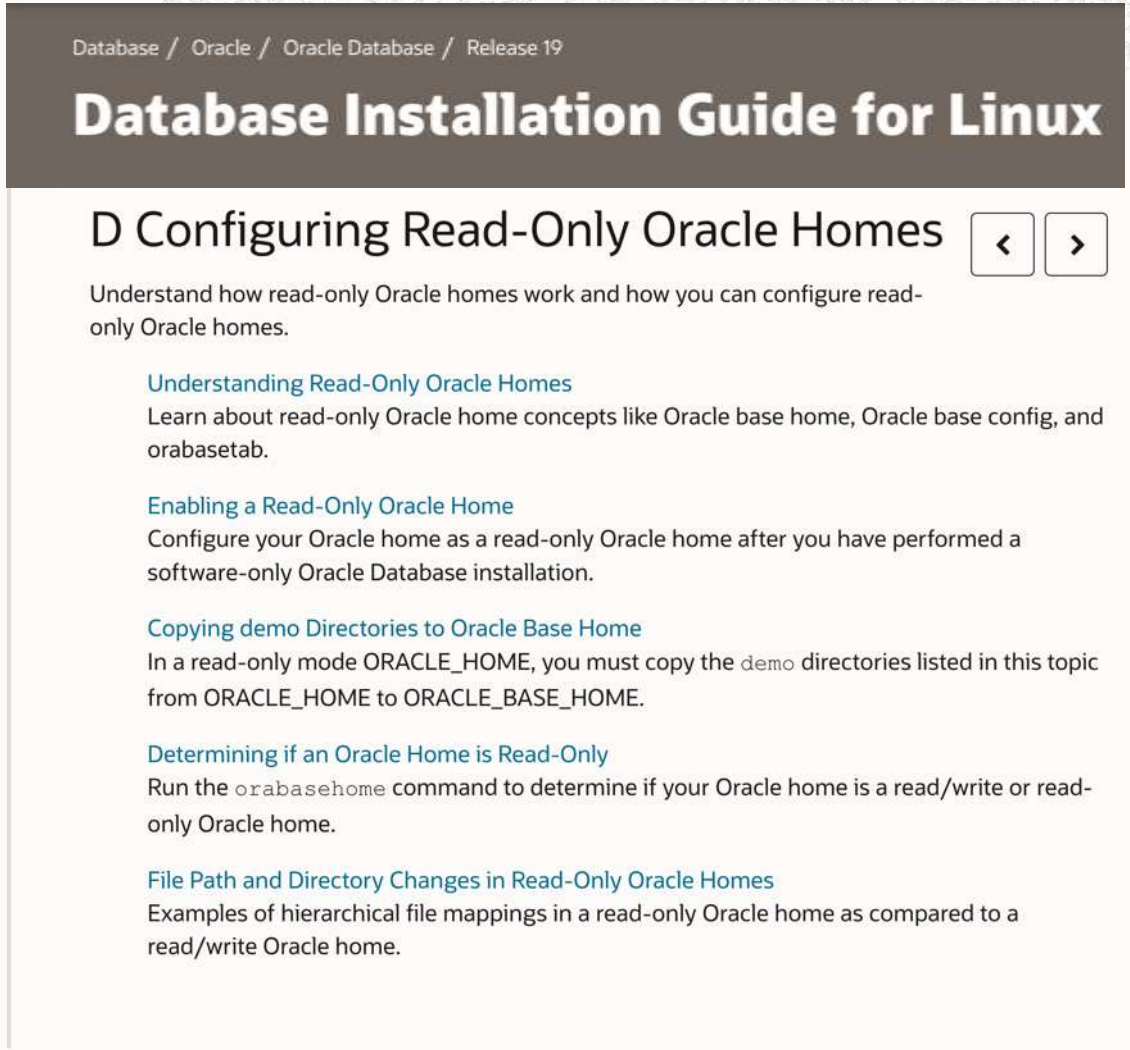
Read Only Oracle Homes | Overview

Simple and easy cloning and provisioning

Configuration and log files stay outside $OH

Documentation:

- https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/configuring-read-only-oracle-homes.html#GUID-906DA159-AC83-4ACC-A8A6-5B4A39EB72E1



Read Only Oracle Homes | Configuration

Setup

1. Install as usual

2. `$ORACLE_HOME/bin/roohctl -enable`

Documentation:

- https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/configuring-read-only-oracle-homes.html#GUID-906DA159-AC83-4ACC-A8A6-5B4A39EB72E1

```
[oracle@hol ~]$ cd /u01/app/oracle/product/ROOH19/
[oracle@hol ROOH19]$ cd bin

[oracle@hol bin]$ ./roohctl -enable

Enabling Read-Only Oracle home.
Update orabasetab file to enable Read-Only Oracle home.
Orabasetab file has been updated successfully.
Create bootstrap directories for Read-Only Oracle home.
Bootstrap directories have been created successfully.
Bootstrap files have been processed successfully.
Read-Only Oracle home has been enabled successfully.

Check the log file
/u01/app/oracle/cfgtoollogs/roohctl/roohctl-
201124PM045139.log for more details.
```

Read Only Oracle Homes | Demo

Watch on YouTube

Read Only Oracle Homes | Directories

Important directories

```
cd $(orabaseconfig)
/u01/app/oracle

cd $(orabasehome)
/u01/app/oracle/homes/OraDB19Home2
```

Read Only Oracle Homes | Directory Structure

```
$ tree -a $(orabaseconfig)/dbs

/u01/app/oracle/dbs
├── hc_ROOH19.dat
├── initROOH19.ora
├── lkROOH19
├── orapwROOH19
└── spfileROOH19.ora
```

```
$ tree -a -d $(orabasehome)

/u01/app/oracle/homes/OraDB19Home2
├── assistants
│   └── dbca
│       └── templates
├── dbs
├── install
├── network
│   ├── admin
│   ├── log
│   └── trace
└── rdbms
    ├── audit
    └── log
        └── opatch
            └── lsinv
```

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

Restore

Gradual Password Rollover

Expression Based Parameters | Overview

Numeric operation:

```
SQL> alter system set cpu_count='8/2' scope=both;
```

Other parameters:

```
SQL> alter system set sga_target=sga_max_size scope=both;
```

Combination:

```
SQL> alter system set shared_pool_size='sga_target*0.2' scope=both;
```

Expression Based Parameters | Overview

Min/max - and override operator precedence:

```
SQL> alter system set shared_pool_size='max(8000M, (sga_target-5000M)*0.2)';
```

Environment variable:

```
SQL> alter system set cpu_count='$NUMBER_OF_PROCESSORS/2';
```

Pro tip: Applies to `ALTER SESSION` commands as well

Expression Based Parameters | Overview

PFile:

```
*.cpu_count=(${NUMBER_OF_PROCESSORS} / 2)
*.aq_tm_processes=MIN(40, PROCESSES*0.1)
*.job_queue_processes=processes
```

Documentation: Syntax

Expression Based Parameters | Demo

```
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
```

Watch on YouTube

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

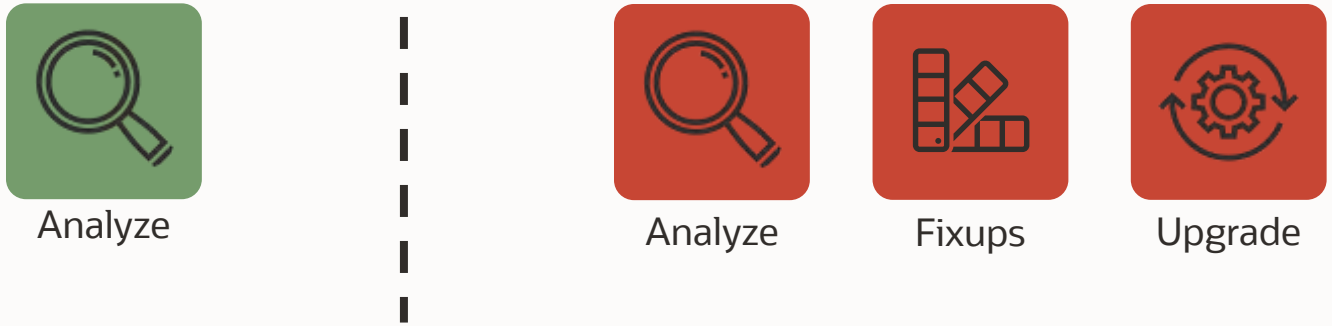
Restore

Gradual Password Rollover

Fast Deploy | Overview

Upgrade with less downtime by running preupgrade fixups in advance

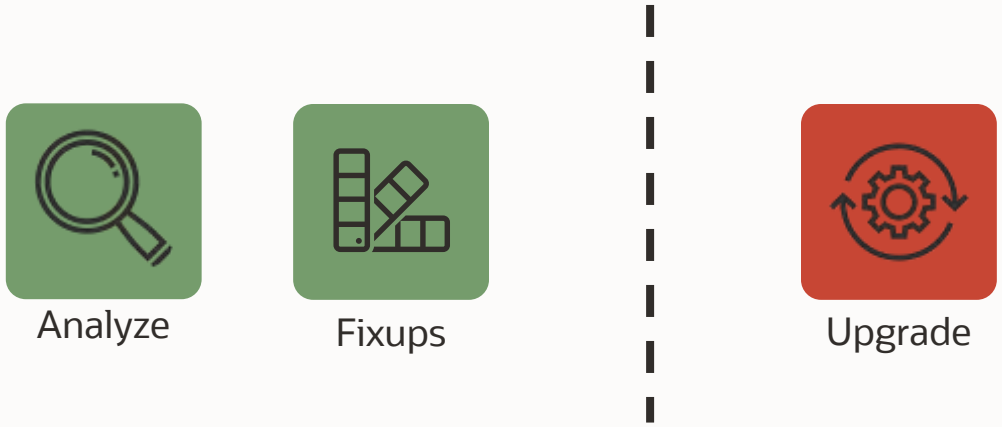
Fast Deploy | Traditional




```
$ java -jar autoupgrade.jar -mode analyze

$ java -jar autoupgrade.jar -mode deploy
```

Fast Deploy | Faster




```
$ java -jar autoupgrade.jar -mode analyze
$ java -jar autoupgrade.jar -mode fixups

$ java -jar autoupgrade.jar -mode upgrade
```

Fast Deploy | Caution

Between fixups and downtime there is a risk that new, undetected issues are introduced

Pro tip: Blog post with more details

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

Restore

Gradual Password Rollover

Data Pump | Checksum

Calculate the checksum of dump files on export, and verify the integrity of dump files on import

Data Pump | Checksum

What can happen to a dump file when it is transferred?

- Tampering
- Corruption

Pro tip: Corruptions usually manifests as `ORA-31693`, `ORA-29913` or `ORA-29104`

Data Pump | Checksum

How to detect corruption or alteration?

```
[oracle@hol]$ md5sum metal*.dmp



5edf66ed92086b4f69580fc27b75f662  metal_01.dmp
59eb25ff2a0f648c051a9212e0861979  metal_02.dmp
29951a56abe074d9151c27728d88e9eb  metal_03.dmp
c8860e7a71e74f8013068240b598c116  metal_04.dmp
0d05d258e4b501c657cd9490b7e48715  metal_05.dmp
1e367394a31e2ce45d2aeb6a3d4f9507  metal_06.dmp
9c276aa580c0e57c0829f274d04d15de  metal_07.dmp
0d560d0ce57c47425424e17604d8ec49  metal_08.dmp
```

```
SQL> select object_name, checksum
     from dbms_cloud.list_objects(
          '<credential_name>',
          '<location_uri>');

metal_01.dmp 5edf66ed92086b4f69580fc27b75f662
metal_02.dmp 59eb25ff2a0f648c051a9212e0861979
metal_03.dmp 29951a56abe074d9151c27728d88e9eb
metal_04.dmp c8860e7a71e74f8013068240b598c116
metal_05.dmp 0d05d258e4b501c657cd9490b7e48715
metal_06.dmp 1e367394a31e2ce45d2aeb6a3d4f9507
metal_07.dmp 9c276aa580c0e57c0829f274d04d15de
metal_08.dmp 0d560d0ce57c47425424e17604d8ec49
```

- Windows: `Get-FileHash *.dmp -Algorithm MD5`

Data Pump | Checksum

Data Pump can calculate checksum on export

```
$ expdp system directory=dmpdir ... checksum_algorithm=sha384
```

Verify dump file integrity on import

```
$ impdp system directory=dmpdir ... verify_only=yes


$ impdp system directory=dmpdir ... verify_checksum=yes
```

Pro tip: Multiple checksum algorithms are available

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

Restore

Gradual Password Rollover

Restore | Overview

AutoUpgrade can now revert a database upgrade

Restore | AutoUpgrade

Use AutoUpgrade to:

- Flashback the database
- Revert a plug-in operation (only when data files are copied)
- Revert a non-CDB to PDB conversion (only when data files are copied)

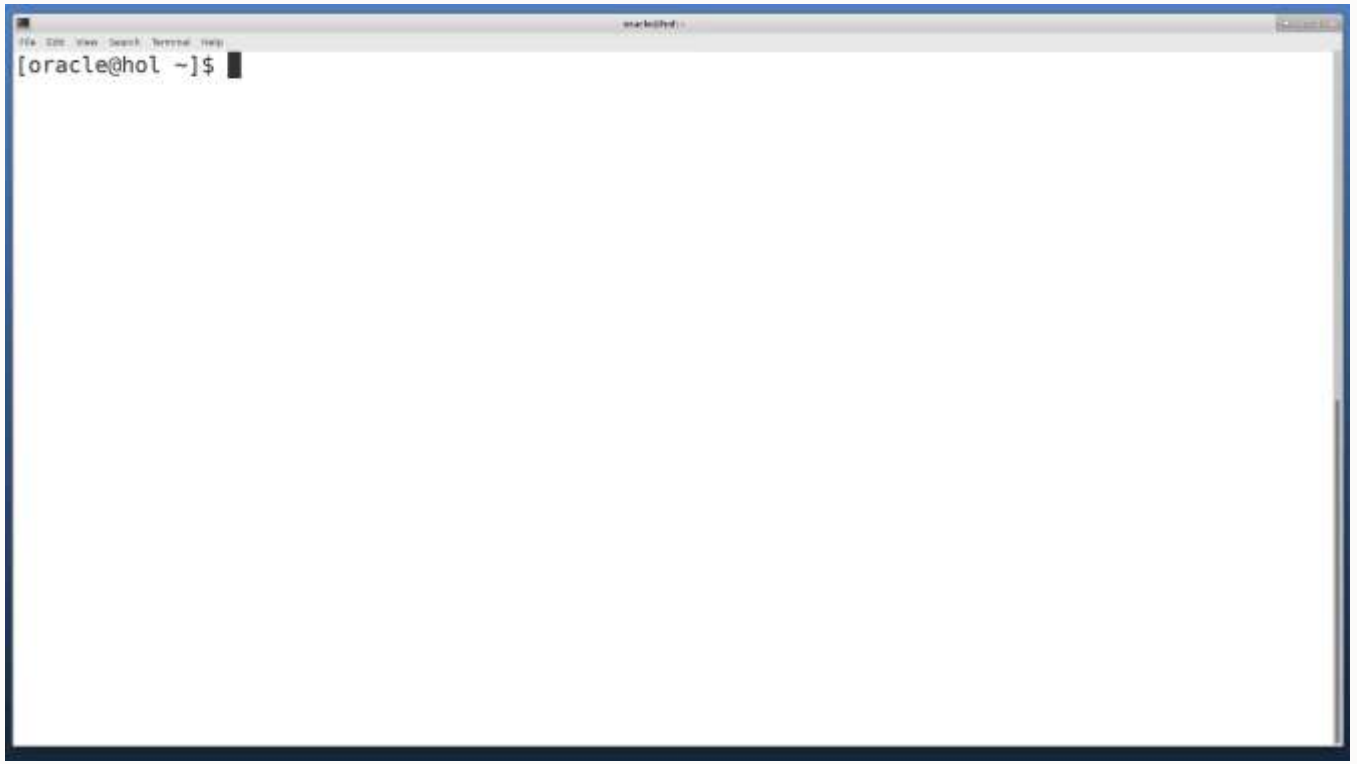
Restore | Command

```
$ java -jar autoupgrade.jar -restore -jobs n
```


Restore | Flashback Database

Pre Upgrade Environment	Post Upgrade Environment
`CREATE RESTORE POINT grpt GUARANTEE FLASHBACK DATABASE;`	
	
	`SHUTDOWN IMMEDIATE`
	`STARTUP MOUNT;`
	`FLASHBACK DATABASE TO RESTORE POINT grpt;`
	`SHUTDOWN IMMEDIATE`
`STARTUP MOUNT;`	
`ALTER DATABASE OPEN RESETLOGS;`	
`DROP RESTORE POINT grpt;`	

Restore | Demo



Watch on YouTube

AutoUpgrade | Restore

AutoUpgrades handles everything, including

- `/etc/oratab`
- Clusterware registration
- Moving files
  - PFile
  - SPFile
  - Password file
  - Etc.

Pro tip: If restoring a primary database, you must manually handle the standby database

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

Restore

Gradual Password Rollover

Gradual Password Rollover | Overview

- Allow a user to have two passwords for a limited amount of time

```
SQL> CREATE PROFILE app_profile LIMIT
       PASSWORD_ROLLOVER_TIME 1;

SQL> CREATE USER app_user
       ...
       PROFILE app_profile;
```

- Minimum one hour (1/24), maximum 60 days

Originally a 21c feature, but backported with 19.12.0

Documentation

Gradual Password Rollover | Overview

- Which users are using the old password?

```
SQL> select authentication_type
     from   unified_audit_trail
     where  action_name='LOGON' and dbusername='APP_USER';
```

- The `authentication_type` tells which password is used:

```
(TYPE=(DATABASE));(CLIENT ADDRESS=((PROTOCOL=tcp)(HOST=10.0.1.225)(PORT=24974)));
 (LOGON_INFO=((VERIFIER=12C-NEW)(CLIENT_CAPABILITIES=O5L_NP,O7L_MR,O8L_LI)));

(TYPE=(DATABASE));(CLIENT ADDRESS=((PROTOCOL=tcp)(HOST=10.0.1.225)(PORT=24983)));
 (LOGON_INFO=((VERIFIER=12C-OLD)(CLIENT_CAPABILITIES=O5L_NP,O7L_MR,O8L_LI)));
```

Gradual Password Rollover | Demo

Watch on YouTube

Read-only Oracle Home

Expression Based Parameters

Fast Deploy

Checksum

Unplug Plug

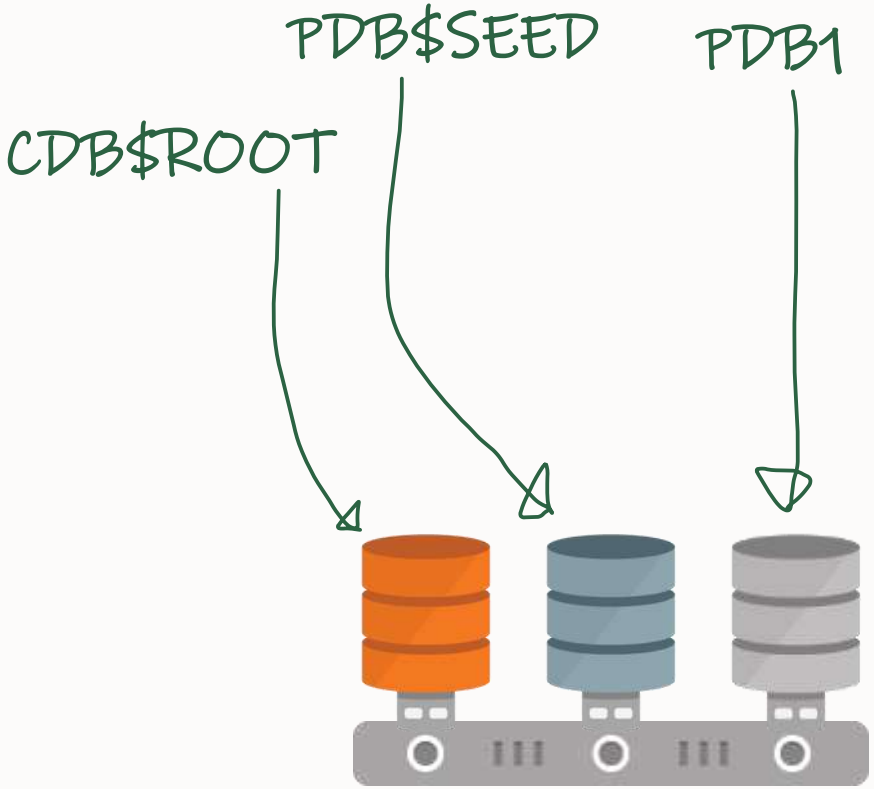
Restore

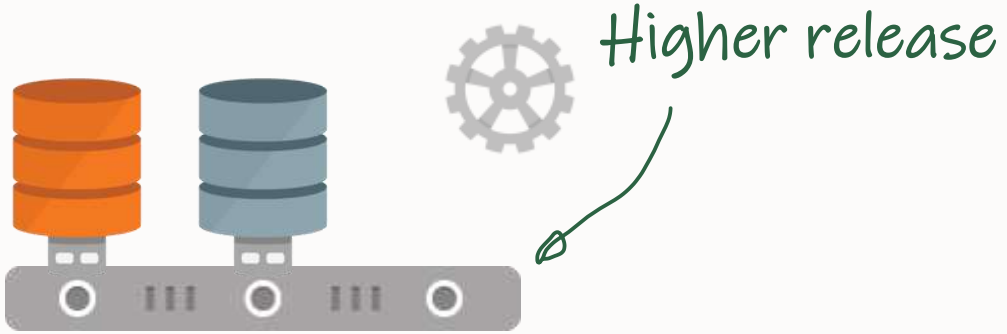
Gradual Password Rollover

Unplug-Plug | Overview

AutoUpgrade can now perform unplug-plug upgrades

Unplug-Plug | Concept





Unplug-Plug | AutoUpgrade

Fully automated unplug-plug 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
```

Command

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

Pro tip: The CDB must be created in advance

Unplug-Plug | AutoUpgrade

Flashback Database doesn't work for unplug-plug upgrade, and the existing data files are re-used

Unplug-Plug | AutoUpgrade

Fully automated unplug-plug upgrade - data files are copied

```
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.target_pdb_copy_option.PDB1=file_name_convert=NONE
```

Command

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


Unplug-Plug | AutoUpgrade

Fully automated unplug-plug upgrade - data files are copied

```
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.target_pdb_copy_option.PDB1=file_name_convert=NONE
upg1.target_pdb_copy_option.PDB1=file_name_convert=('CDB1','CDB2')
```

Command

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


Unplug-Plug | AutoUpgrade

But it takes time to copy the data files

Unplug-Plug | AutoUpgrade

And the target CDB is located on a different host

Unplug-Plug | 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

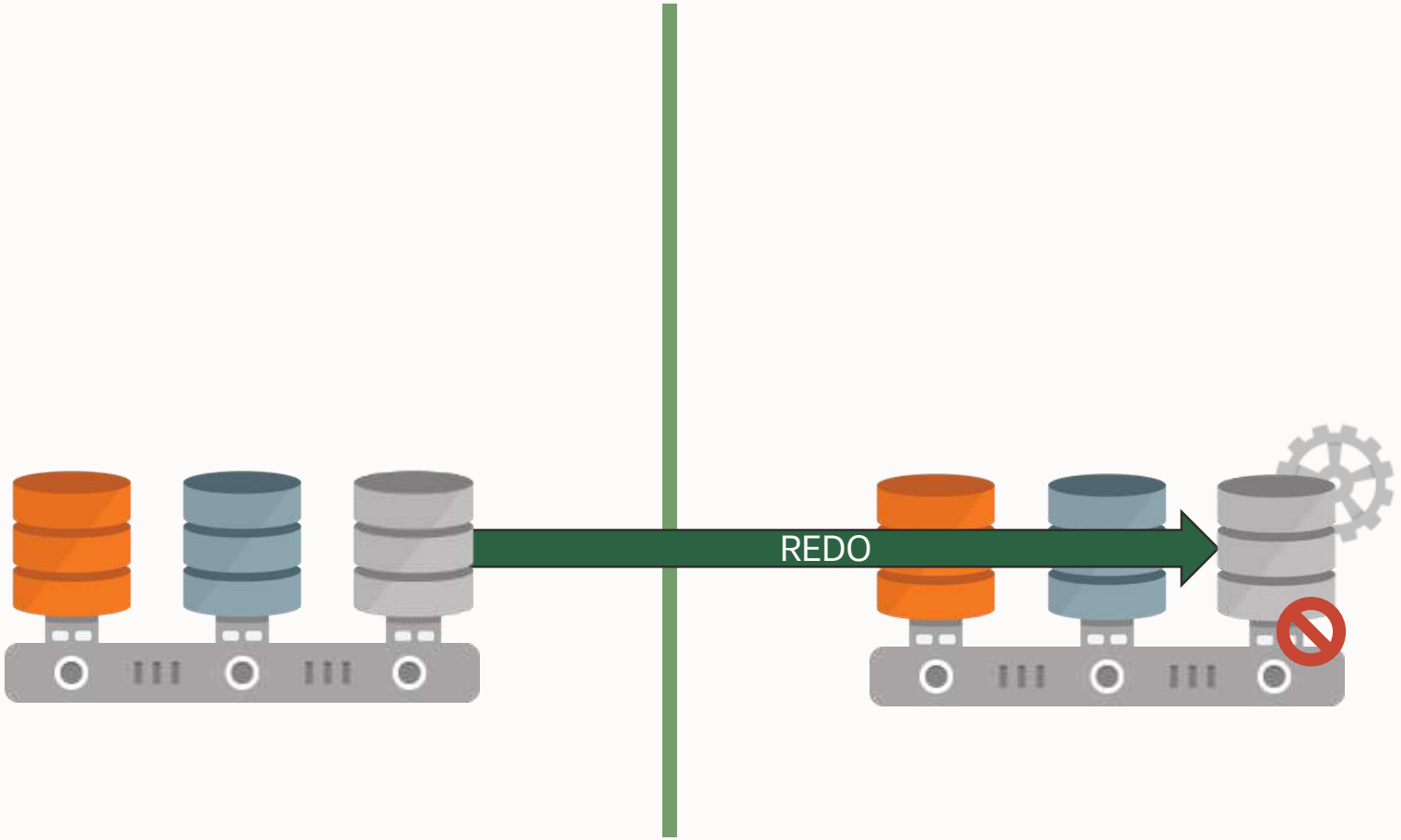
```
CREATE DATABASE LINK clonePDB1
CONNECT TO c##borg
IDENTIFIED BY oracle
USING 'tns-or-ezconnect';
```

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



Unplug-Plug | Refreshable Clone PDB

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

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

UPGRADE TO 19C FROM ZERO TO HERO
Monday 29 November

HUH? IS IT FIXED IN 23C?
Saturday 11 December

THANK YOU