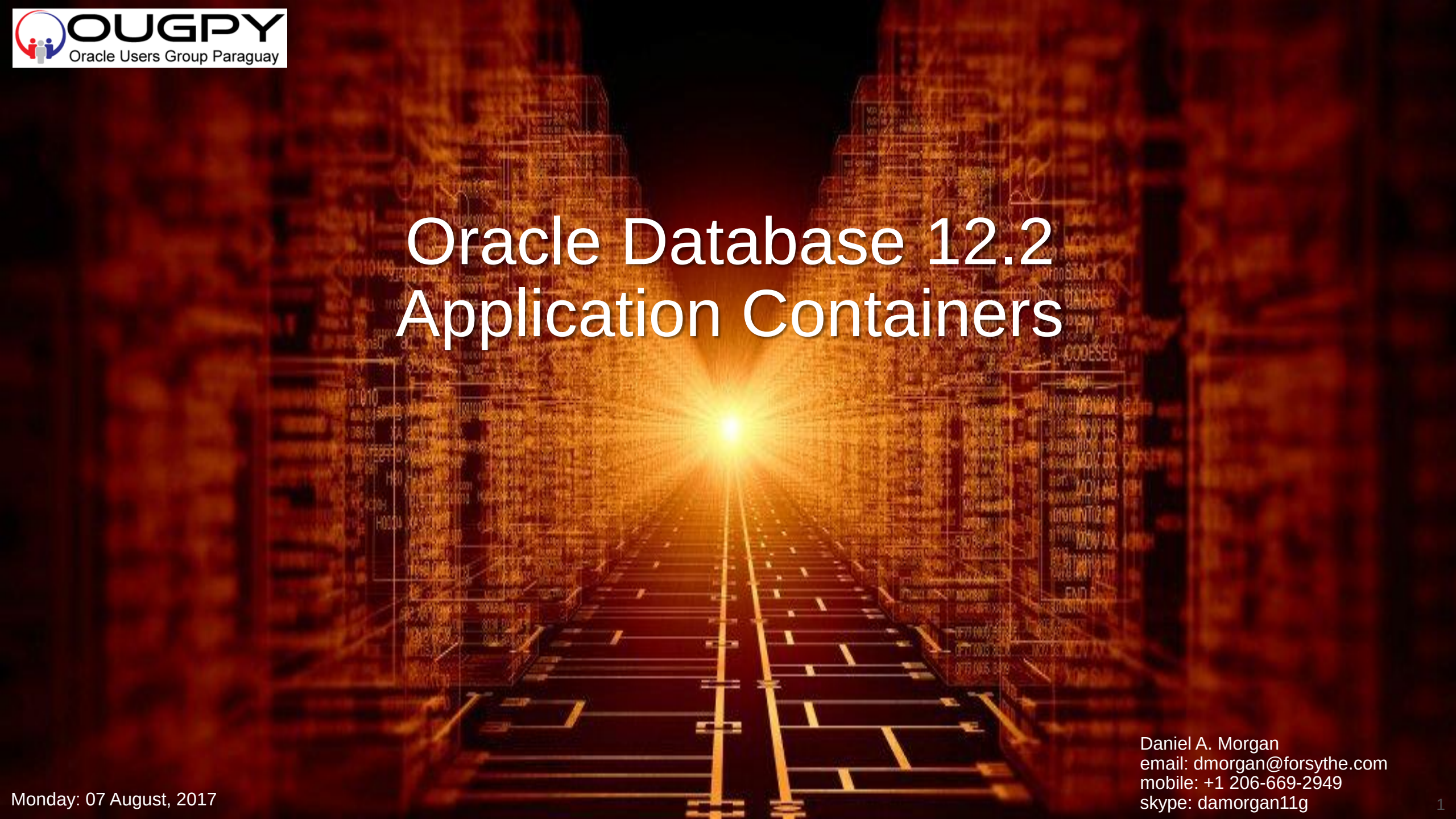


Oracle Database 12.2 Application Containers

Daniel A. Morgan
email: dmorgan@forsythe.com
mobile: +1 206-669-2949
skype: damorgan11g

Monday: 07 August, 2017

Unsafe Harbor

- This room is an unsafe harbor
- You can rely on the information in this presentation to help you protect your data, your databases, your organization, and your career
- No one from Oracle has previewed this presentation
- No one from Oracle knows what I'm going to say
- No one from Oracle has supplied any of my materials
- Everything I will present is existing, proven, functionality

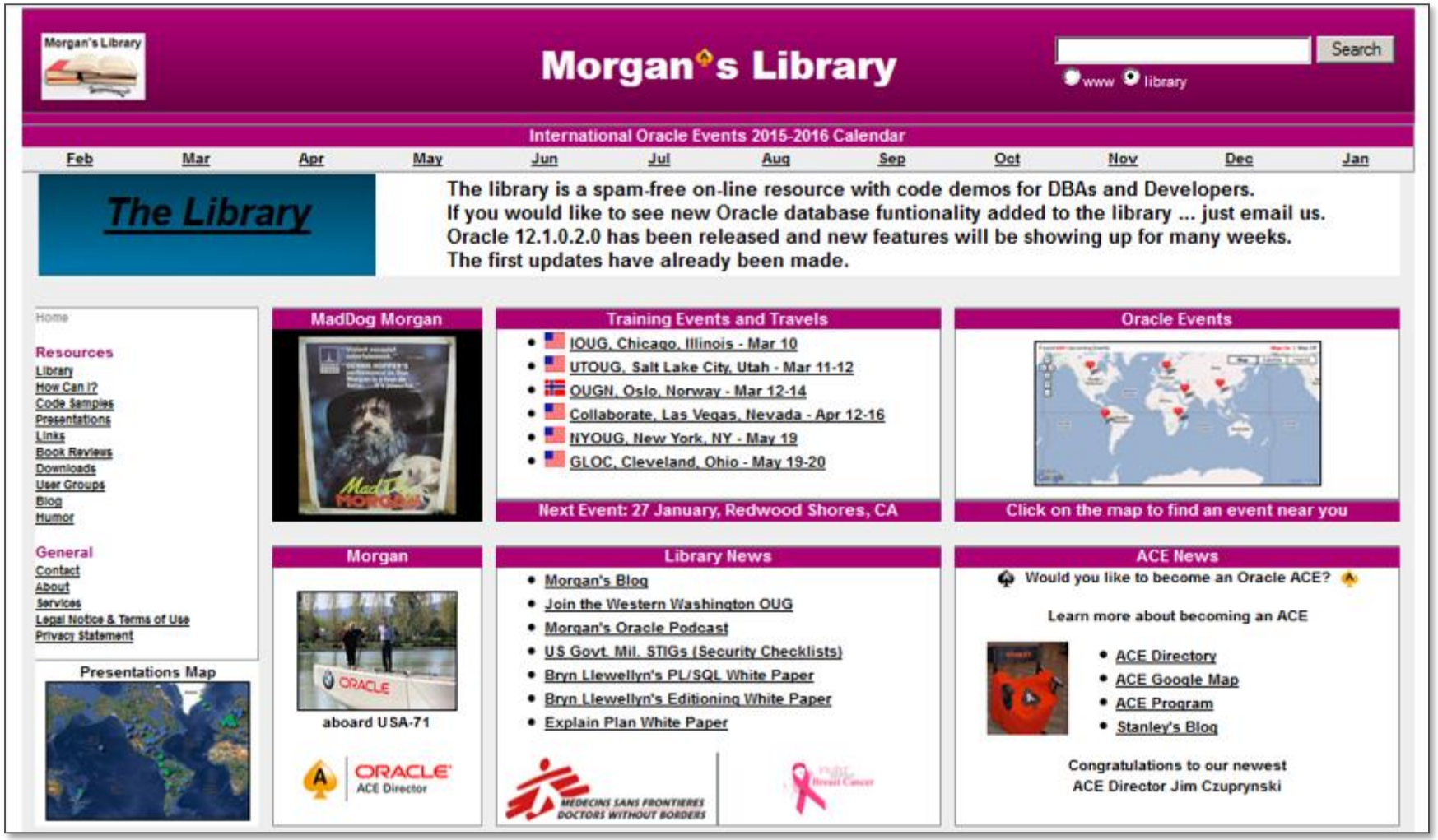
Introduction

Daniel Morgan

- Oracle ACE Director Alumni
- Oracle Educator
  - Curriculum author and primary program instructor at University of Washington
  - Consultant: Harvard University
  - University Guest Lecturers
    - APAC: University of Canterbury (NZ)
    - EMEA: University of Oslo (Norway)
    - Latin America: Universidad Cenfotec, Universidad Latina de Panama, Technologico de Costa Rica
- IT Professional
  - First computer: IBM 360/40 in 1969: Fortran IV
  - Oracle Database since 1988-9 and Oracle Beta tester
  - The Morgan behind www.morganslibrary.org
  - Member Oracle Data Integration Solutions Partner Advisory Council
  - Vice President Twin Cities Oracle Users Group (Minneapolis-St. Paul)
  - Co-Founder International GoldenGate Oracle Users Group
- Principal Adviser: Forsythe **Meta7**

System/370-145 system console

My Websites: Morgan's Library



Stability: IT Fire Fighting

Oracle Stack Security

Scalability: VLDBs and Partitioning

Database Performance


Zero Downtime Migration

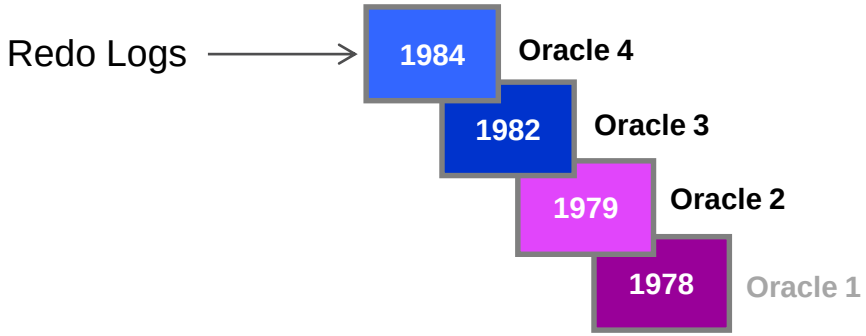
Just In Time IT Procurement

with the Oracle Bare Metal Cloud and Infrastructure as Code

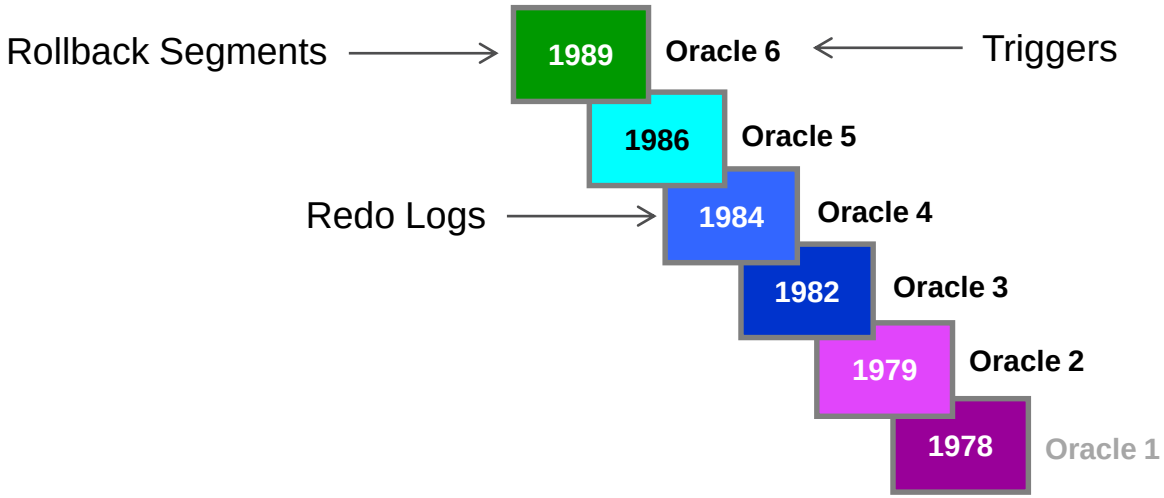
Learning Experience Alert

Introduction to Database 12c

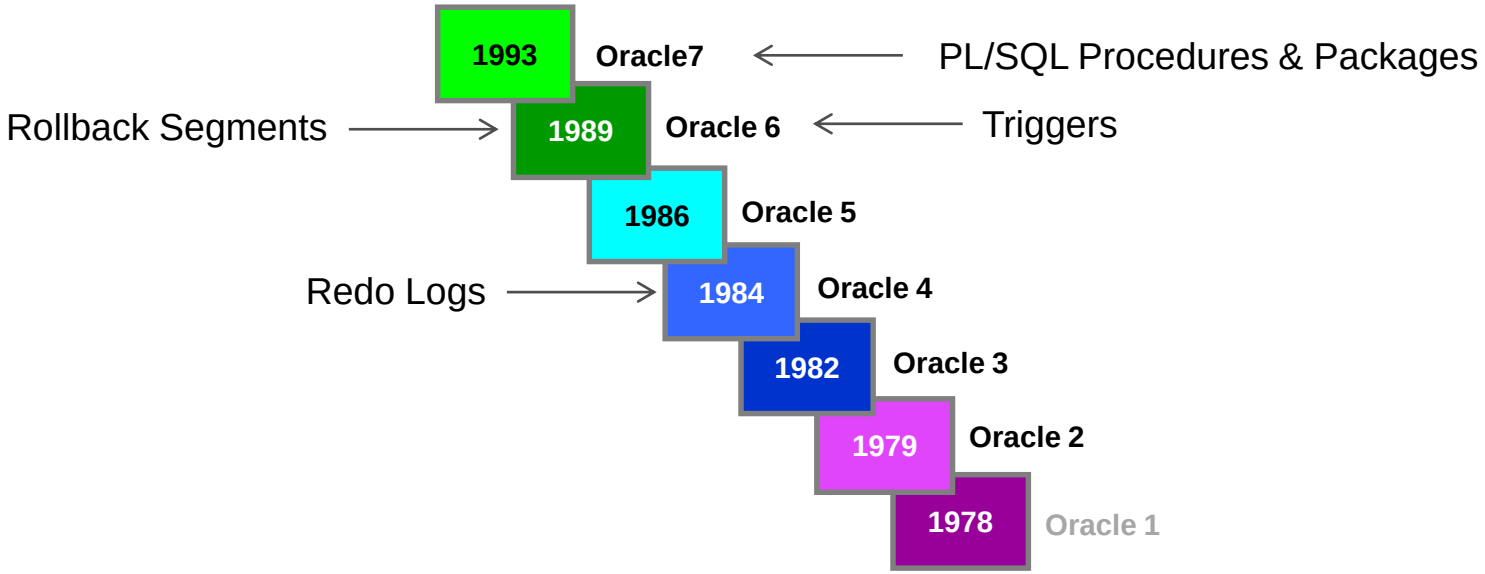
A Brief History of the Oracle Database



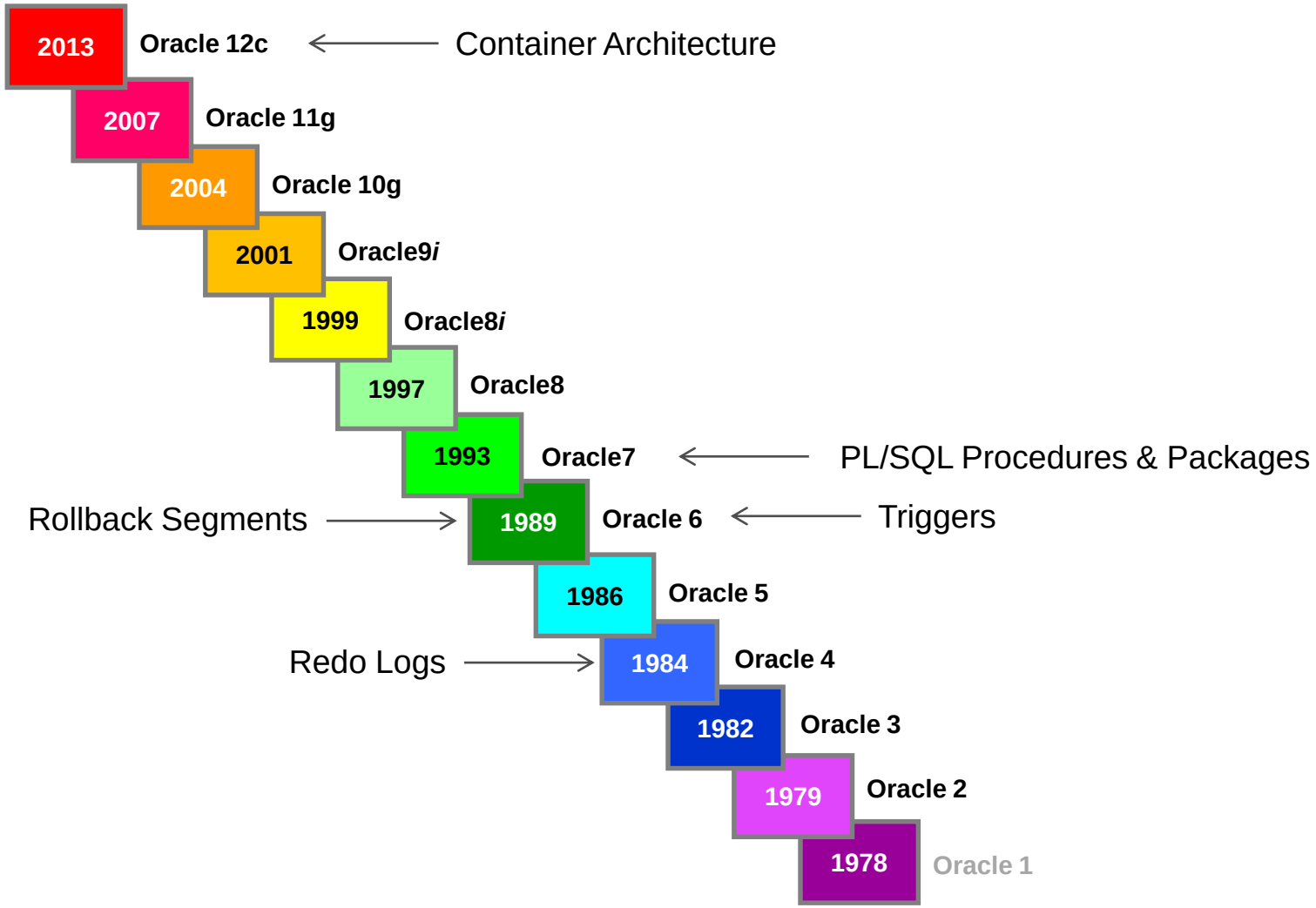
A Brief History of the Oracle Database



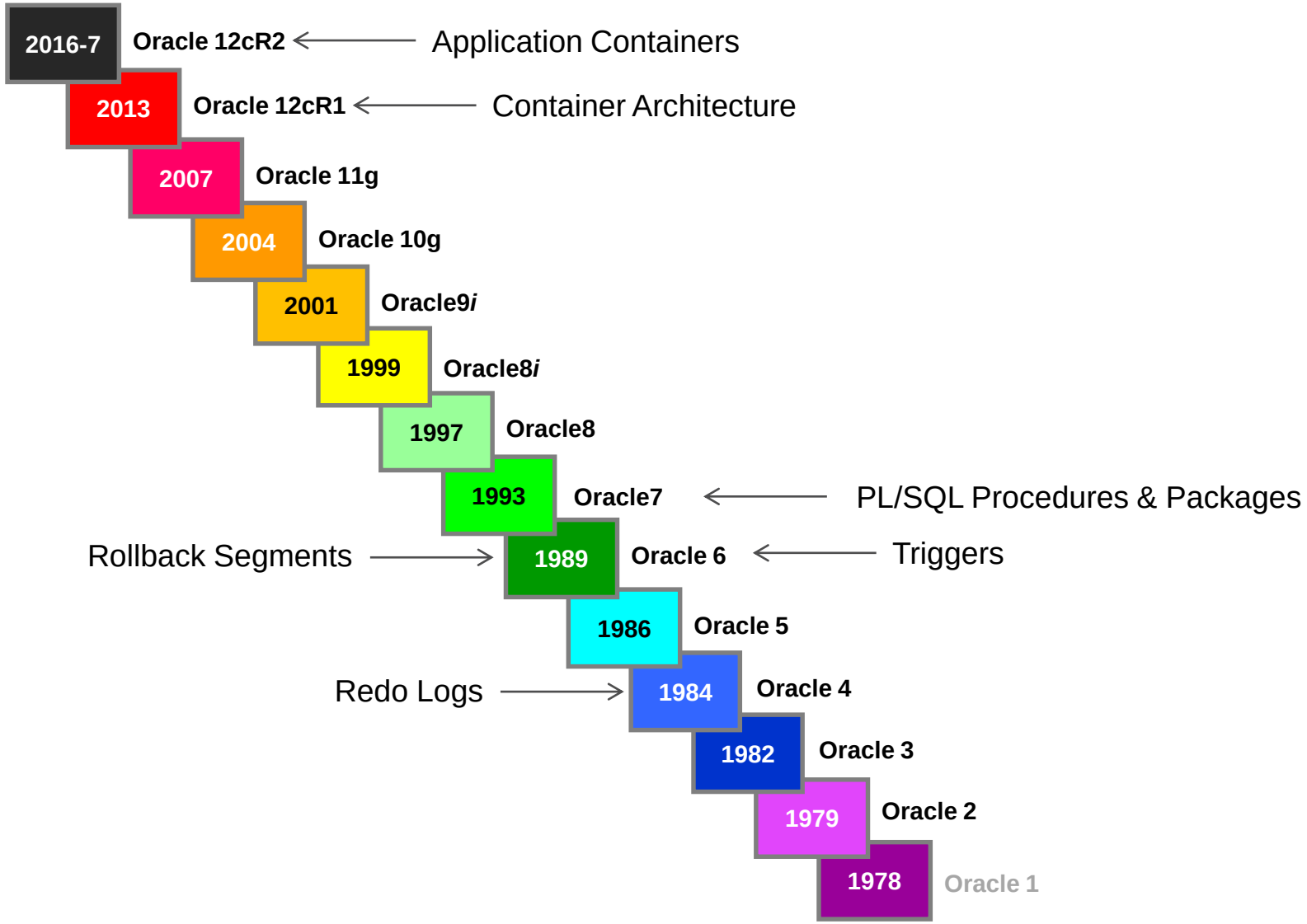
A Brief History of the Oracle Database



A Brief History of the Oracle Database

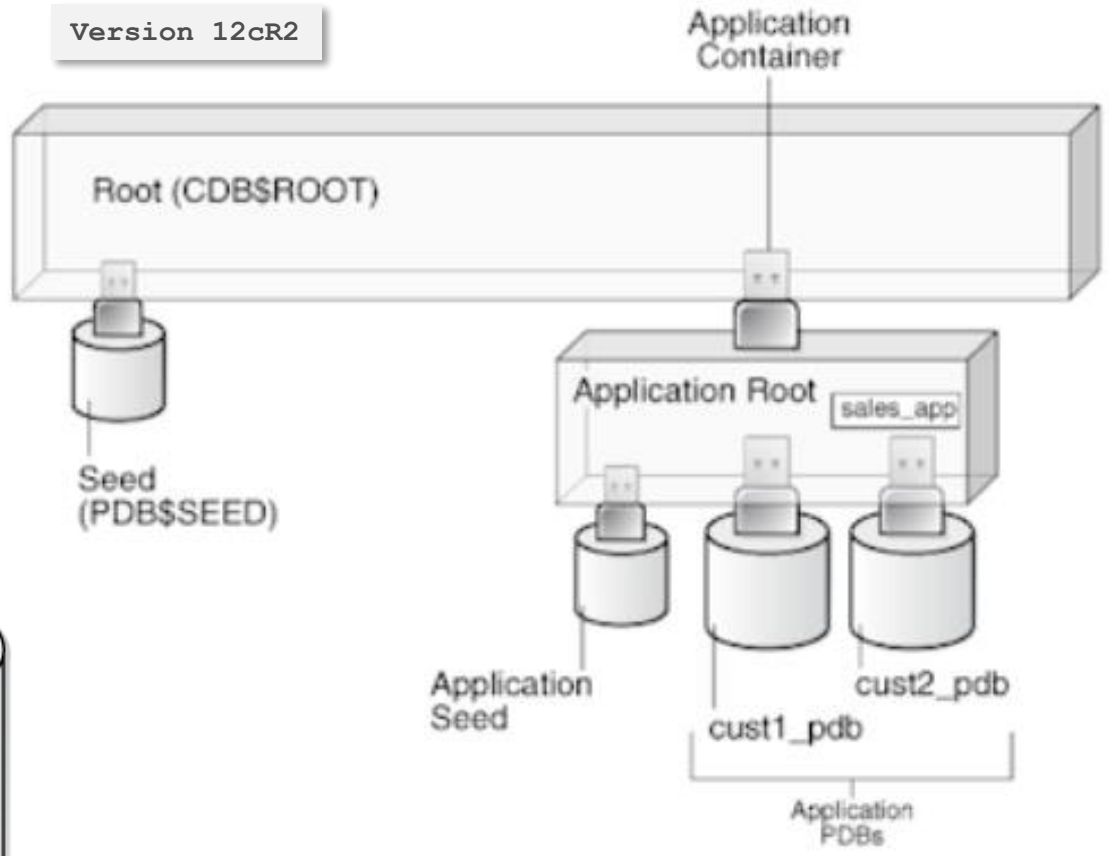


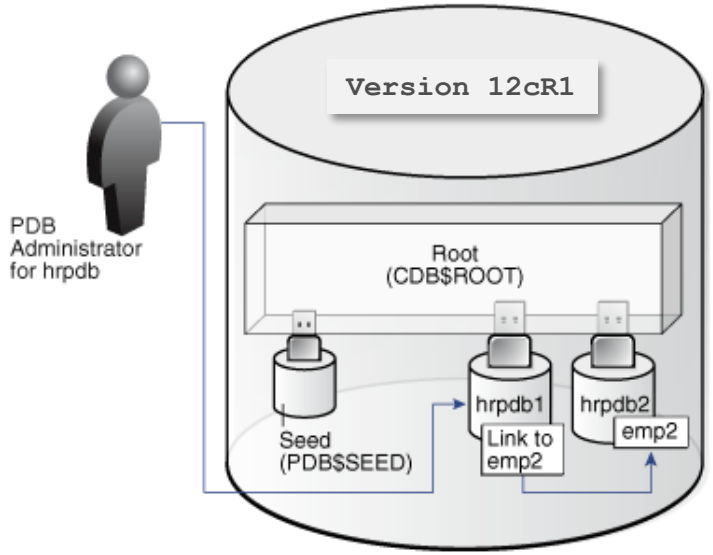
A Brief History of the Oracle Database



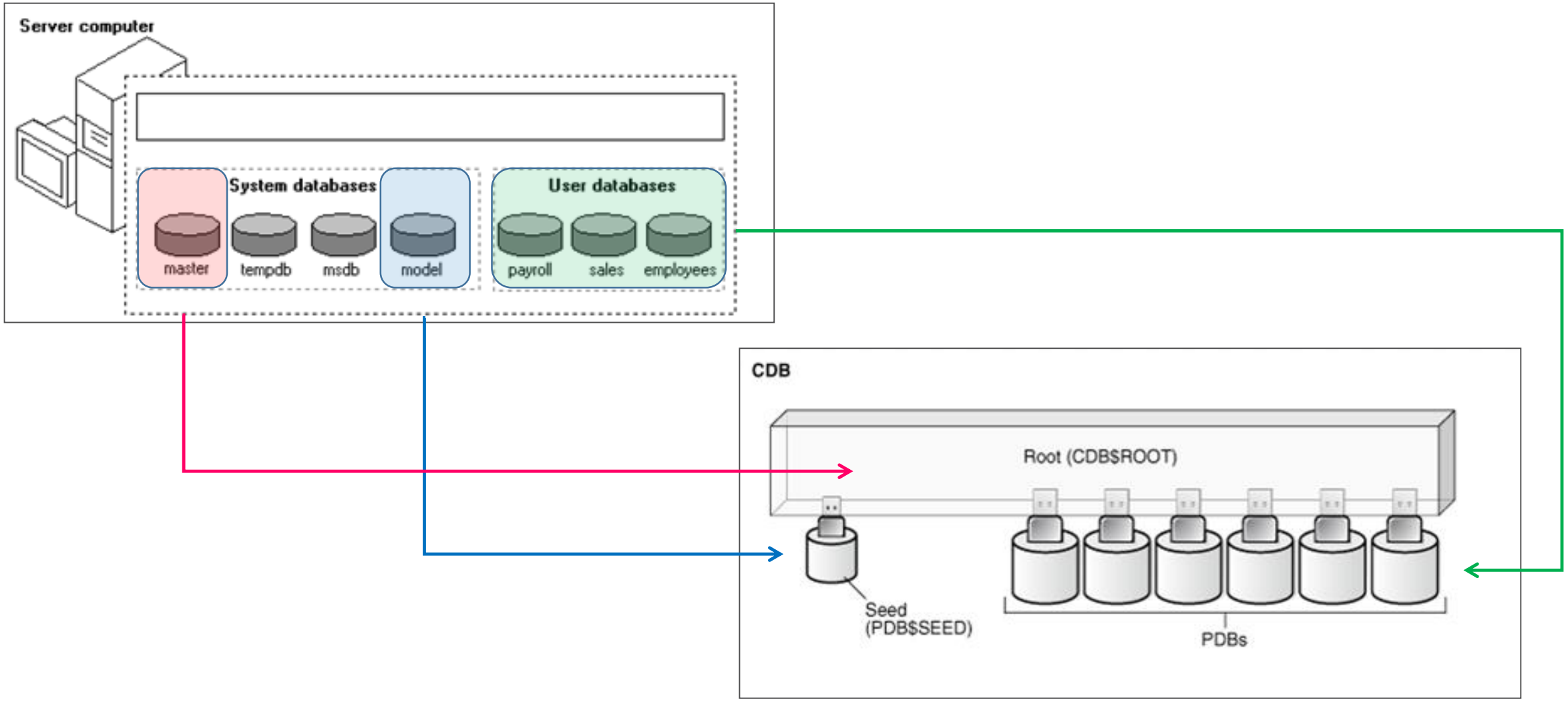
New Physical and Logical Architecture

New 12cR2 Container Database Architecture

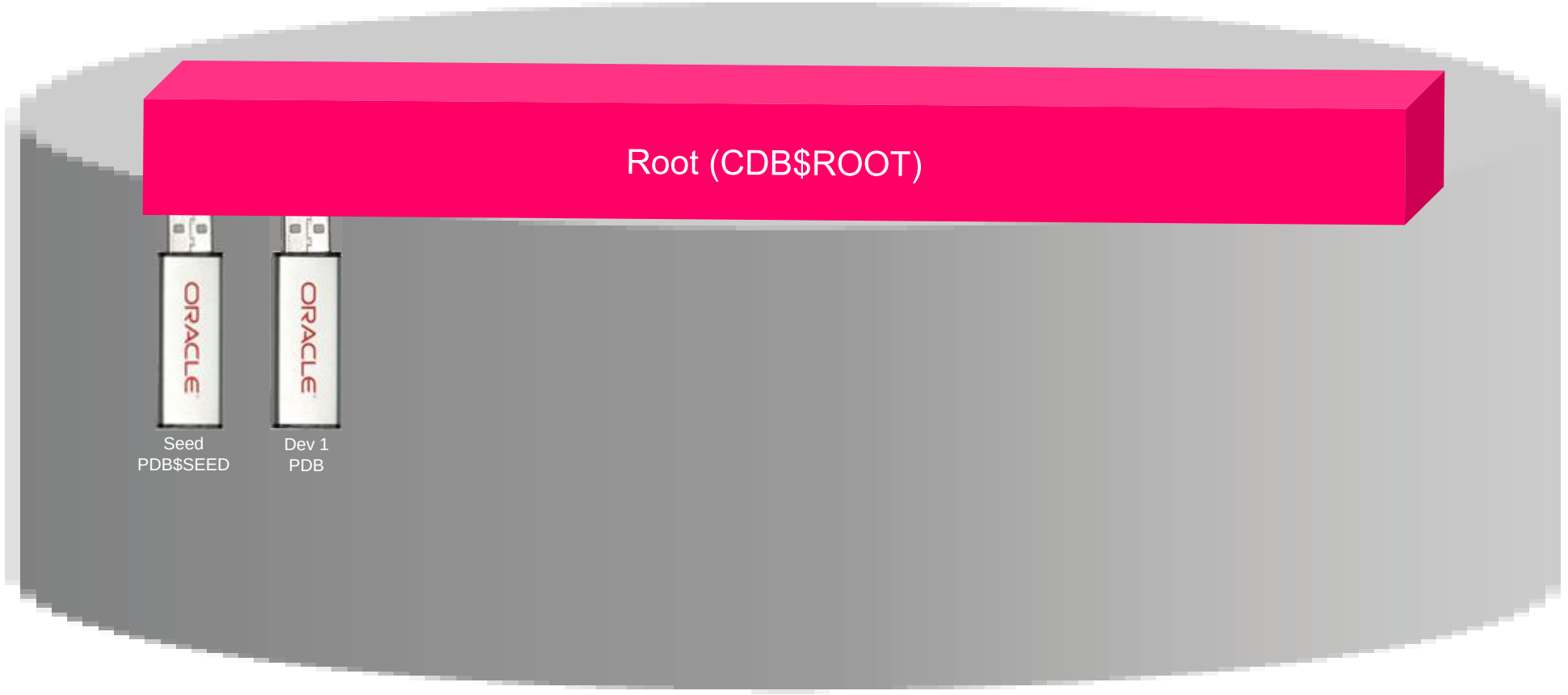




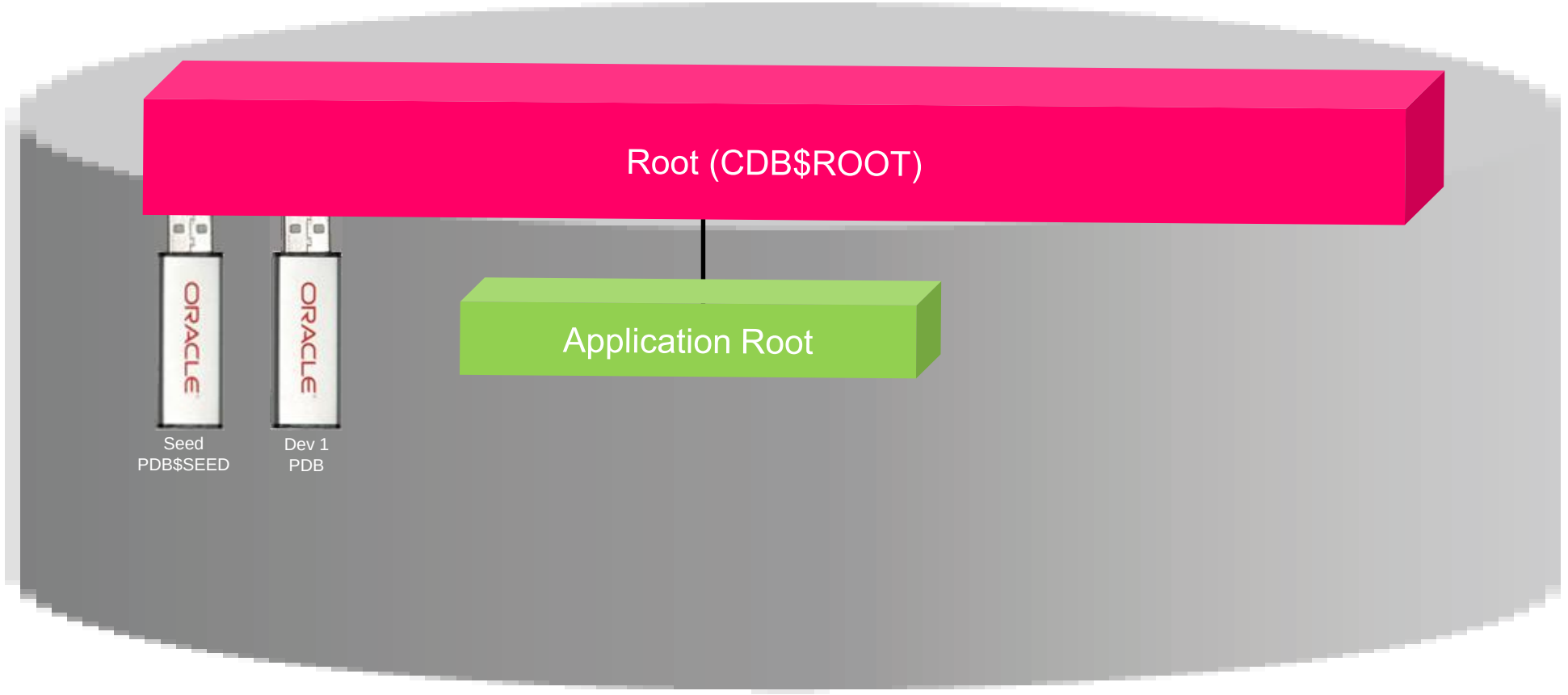
New 12cR1 Container Database Architecture



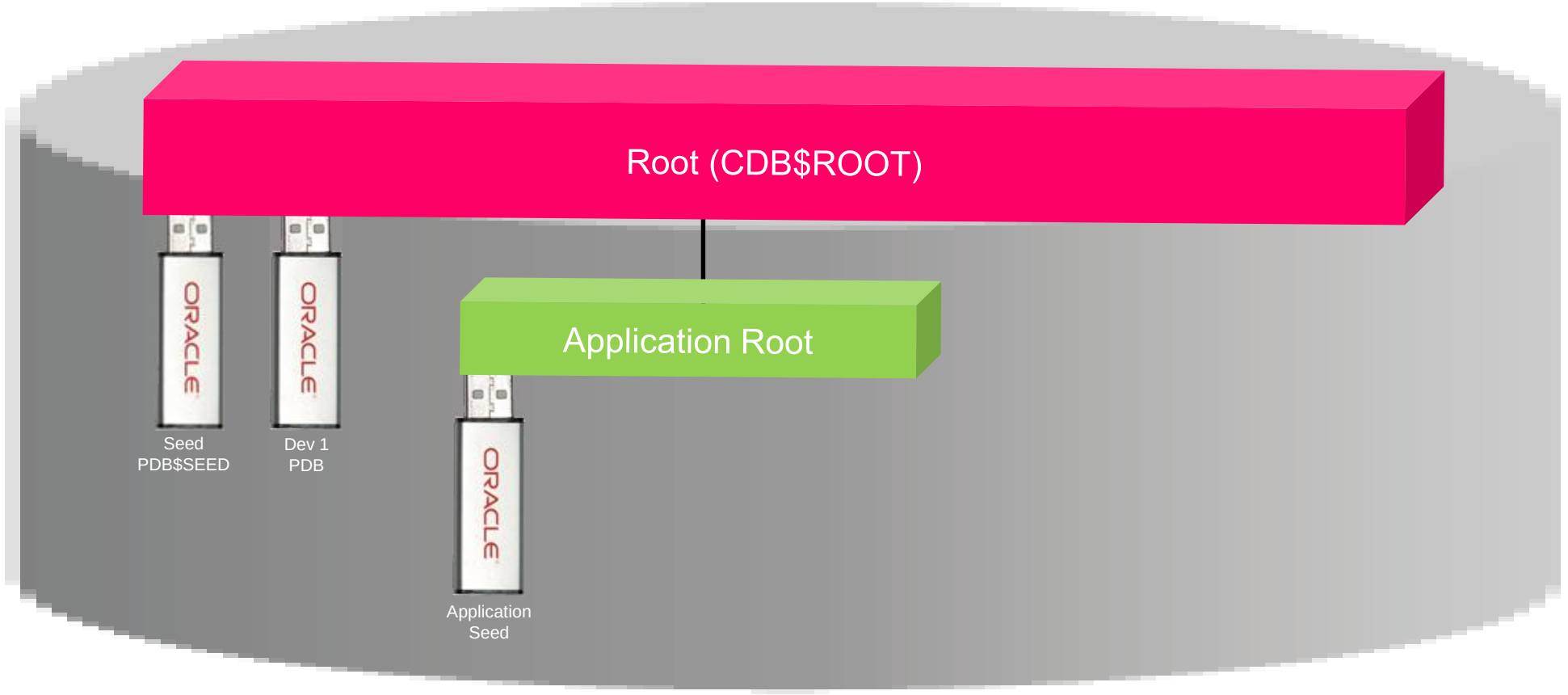
New 12cR1 Container Database Architecture



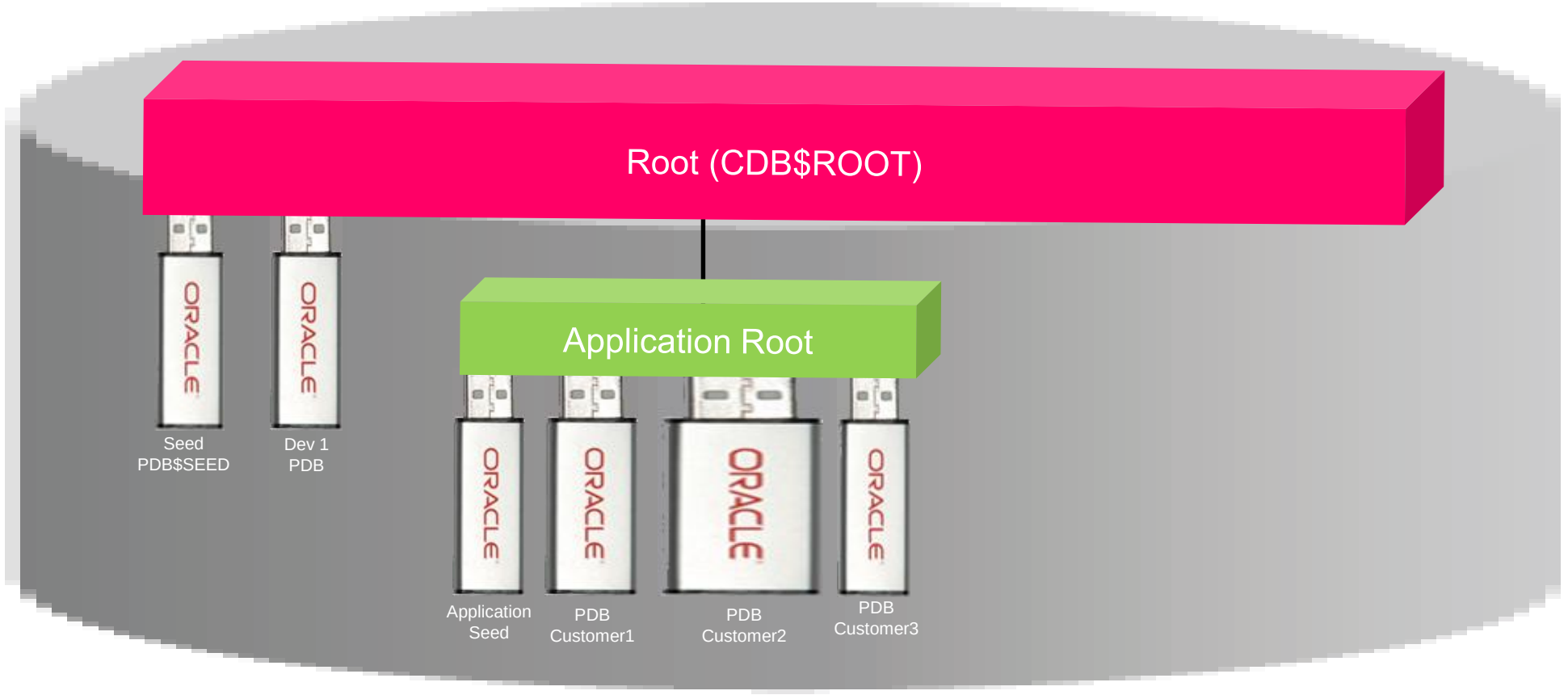
New 12cR2 Container Database Architecture



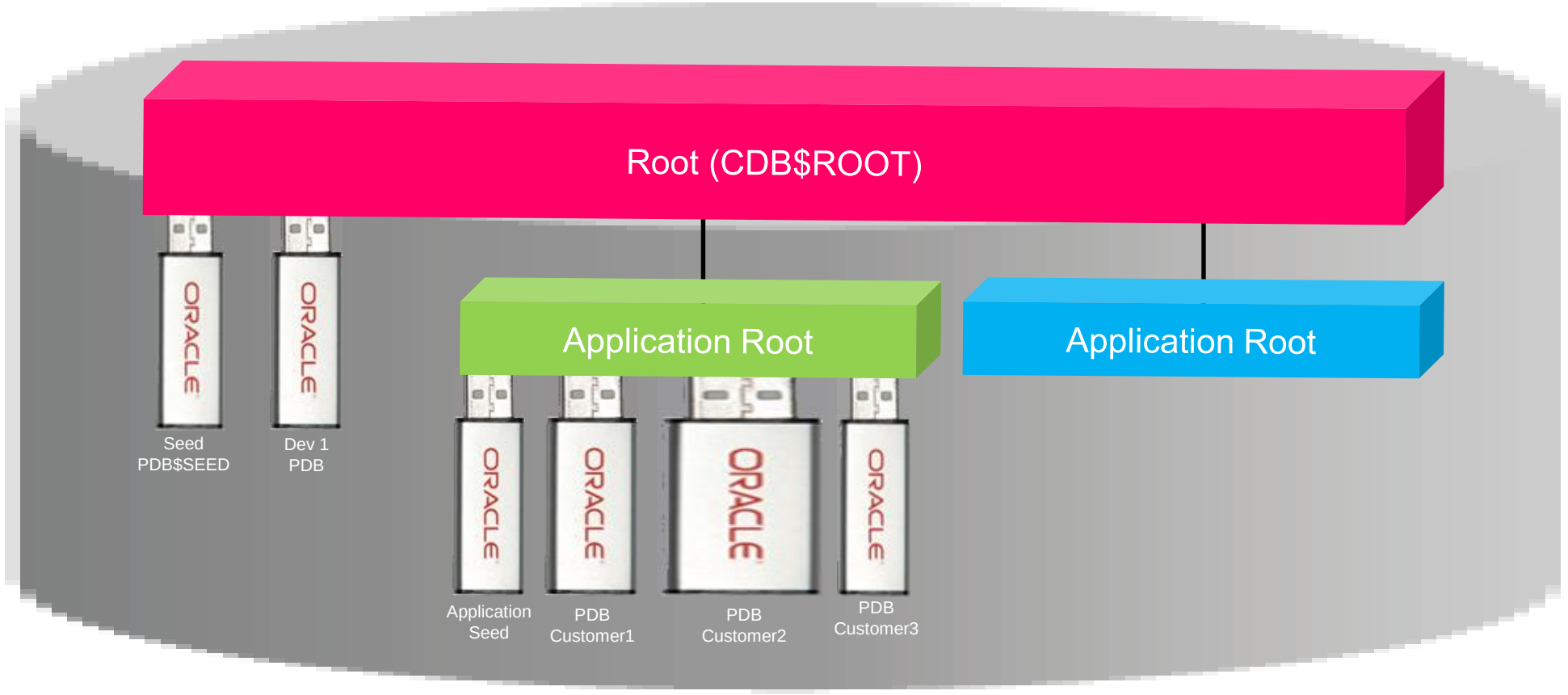
New 12cR2 Container Database Architecture



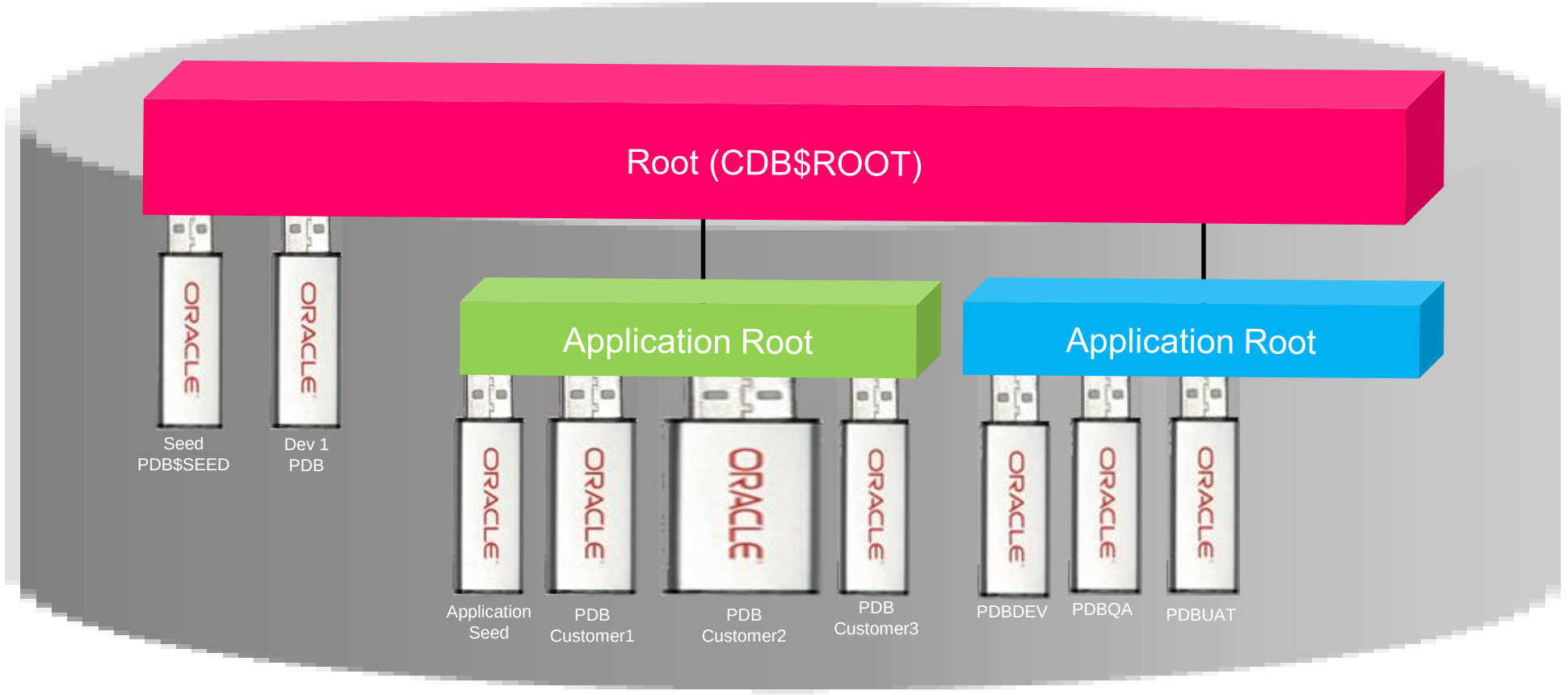
New 12cR2 Container Database Architecture



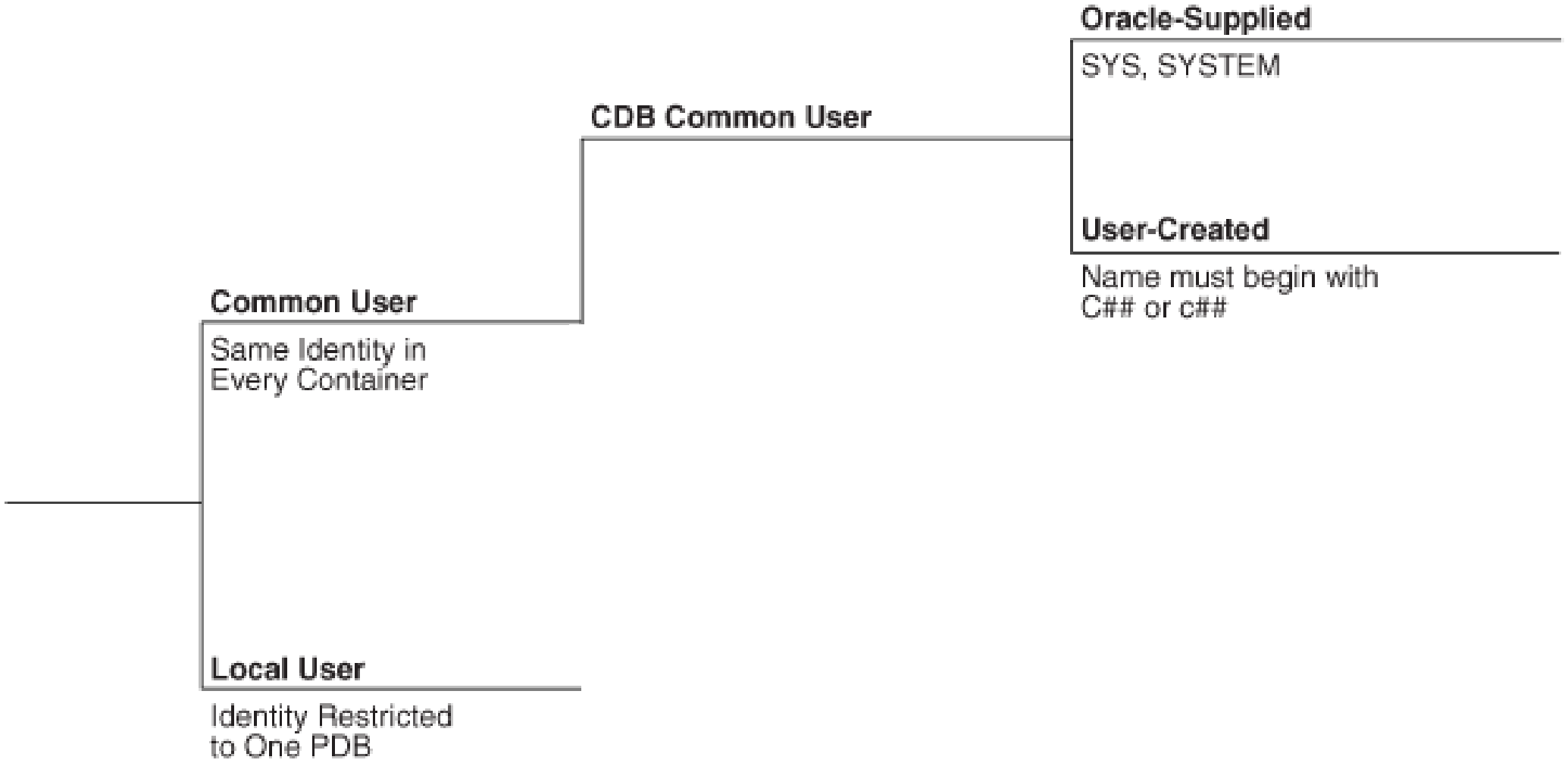
New 12cR2 Container Database Architecture



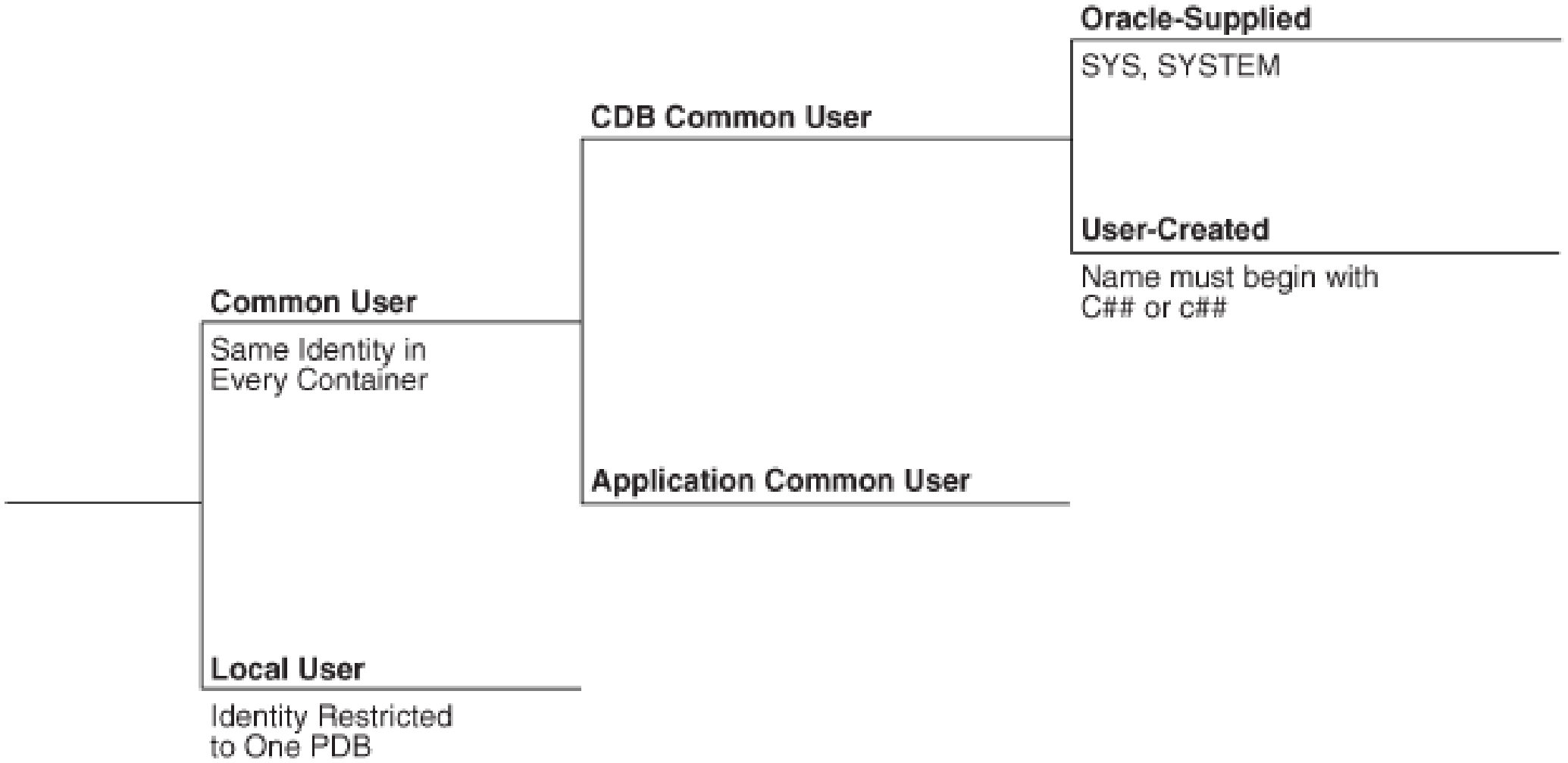
New 12cR2 Container Database Architecture



New 12cR1 Container Database User Architecture



New 12cR2 Container Database User Architecture



Users

New: 12cR1

AUDSYS
GSMADMIN_INTERNAL
GSMCATUSER
GSMUSER
PDBADMIN
SYSBACKUP
SYSDG
SYSKM

New: 12cR2

APEX_050100
APEX_INSTANCE_ADMIN_USER
APEX_LISTENER
APEX_REST_PUBLIC_USER
DBJSON
DBSFWUSER
GGSYS
HRREST
OBE
ORDS_METADATA
ORDS_PUBLIC_USER
PDBADMIN
REMOTE_SCHEDULER_AGENT
RESTFUL
SYS$UMF
SYSRAC
XDBEXT
XDBPM
XFILES

New Users With Escalated Privs

USERNAME	Usage
GGSYS	The internal account used by Oracle GoldenGate. It should not be unlocked or used for a database login.
SYSBACKUP	This privilege allows a user to perform backup and recovery operations either from Oracle Recovery Manager (RMAN) or SQL*Plus.
SYSDG	This privilege allows a user to perform Data Guard operations can use this privilege with either Data Guard Broker or the DGMGRL command-line interface.
SYSKM	This privilege allows a user to perform Transparent Data Encryption keystore operations.
SYSRAC	This privilege allows the Oracle agent of Oracle Clusterware to perform Oracle Real Application Clusters (Oracle RAC) operations. SYSRAC facilitates Oracle Real Application Clusters (Oracle RAC) operations by connecting to the database by the Clusterware agent on behalf of Oracle RAC utilities such as SRVCTL.

If you don't think this will affect you consider the following

- Oracle 18.0 will be released this Fall with more than 180 new features
  - Oracle has changed `./runInstaller` and OUI to make them more Cloud compatible
- Oracle 18.1 will be released in January
- Oracle 18.2 will be released in April
- Oracle is moving to eliminate SYSDBA
  - We now have users named GGSYS, SYSBACKUP, SYSDG, SYSKM, SYSRAC?
- The Oracle Database on Linux may move to rpms with installation and patching fully automated from a YUM server
- The database's kernel is being rewritten so that almost all patches will not require an outage
- Expect OEM and other tools to manage the database through the REST API
- Do not be surprised if SQL*Plus and Server Control are merged into a single tool and DBAs will be strongly discouraged for using it for everything except troubleshooting and root cause analysis

Oracle Multitenant

Multitenant Business Case

- Database 12c was architected specifically to meet business needs related to the move away from n-tier on-premise deployment to Software Defined Everything
- An architecture able to support multitenant pluggable containers is one that provides
  - Rapid provisioning
  - Application portability
  - Physical and logical customer isolation
  - Granular control
  - Improved scaling and resource utilization
  - Improved security

Create Application Containers

Application Containers Demo 1: Create Application Root (1:12)

```
conn sys@orabase as sysdba
Enter password:
Connected.

sho con_id

CON_ID
-------
     1

show con_name

CON_NAME
------------------------------
CDB$ROOT

SELECT name, open_mode, application_root,
application_pdb, application_seed, pdb_count
FROM v$containers
ORDER BY con_id;

NAME       OPEN_MODE  APP APP APP PDB_COUNT
---------- ---------- --- --- --- ----------
CDB$ROOT   READ WRITE NO  NO  NO           2
PDB$SEED   READ ONLY  NO  NO  NO           0
PDBDEV     READ WRITE NO  NO  NO           0
```

Application Containers Demo 1: Create Application Root (2:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (3:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (4:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (5:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
ROLES = (CDB_DBA)
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (6:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
ROLES = (CDB_DBA)

DEFAULT TABLESPACE uwapp_tbs
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (7:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
ROLES = (CDB_DBA)

DEFAULT TABLESPACE uwapp_tbs

FILE_NAME_CONVERT = ('/pdbseed/', '/uwapp/')
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (8:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
ROLES = (CDB_DBA)

DEFAULT TABLESPACE uwapp_tbs

FILE_NAME_CONVERT = ('/pdbseed/', '/uwapp/')



USER_TABLESPACES = NONE
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (9:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[PARALLEL <integer>]
[DEFAULT TABLESPACE <tablespace_name>]
[<pdb_storage_clause>]
[<file_name_convert_clause>]
[<service_name_convert_clause>]
[<path_prefix_clause>]
[TEMPFILE REUSE]
[<user_tablespace_clause>]
[<standby_database_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>]
[<create_file_dest_clause>]
[HOST = '<host_name>']
[PORT = <port_number>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
ROLES = (CDB_DBA)

DEFAULT TABLESPACE uwapp_tbs

FILE_NAME_CONVERT = ('/pdbseed/', '/uwapp/')



USER_TABLESPACES = NONE

LOGGING;
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (10:12)

```
CREATE PLUGGABLE DATABASE <pdb_name>
AS <APPLICATION CONTAINER | SEED>
ADMIN USER <admin_user_name> IDENTIFIED BY <password>
[ROLES = (<comma_delimited_list_of_roles>)]
[DEFAULT TABLESPACE <tablespace_name>]
[<file_name_convert_clause>]
[<user_tablespace_clause>]
[<LOGGING | NOLOGGING | FILESYSTEM_LIKE_LOGGING>];
```

```
CREATE PLUGGABLE DATABASE uwapp_root
AS APPLICATION CONTAINER
ADMIN USER uwAdmin IDENTIFIED BY uwAdmin
ROLES = (CDB_DBA)
DEFAULT TABLESPACE uwapp_tbs
FILE_NAME_CONVERT = ('/pdbseed/', '/uwapp/')
USER_TABLESPACES = NONE
LOGGING;

Pluggable database created.
```

- As desirable as it would be to do so ... you cannot use a special character when naming a container

Application Containers Demo 1: Create Application Root (11:12)

```
SELECT name, open_mode, application_root,
application_pdb, application_seed, pdb_count
FROM v$containers
ORDER BY con_id;

NAME       OPEN_MODE  APP APP APP PDB_COUNT
---------- ---------- --- --- --- ----------
CDB$ROOT   READ WRITE NO  NO  NO          3
PDB$SEED   READ ONLY  NO  NO  NO          0
PDBDEV     READ WRITE NO  NO  NO          0
UWAPP_ROOT MOUNTED    YES NO  NO          0

ALTER PLUGGABLE DATABASE uwapp_root OPEN;

Pluggable database altered.

SELECT name, creation_date, clb_goal, pdb
FROM v$services
ORDER BY 1;

NAME             CREATION_DATE        CLB_G PDB
---------------- -------------------- ----- ------------
SYS$BACKGROUND   26-JAN-2017 13:54:44 SHORT CDB$ROOT
SYS$USERS        26-JAN-2017 13:54:44 SHORT CDB$ROOT
uwapp_root       26-MAR-2017 17:09:28 LONG  UWAPP$ROOT
pdbdev           02-MAR-2017 07:57:37 LONG  PDBDEV
orabase          02-MAR-2017 07:52:46 LONG  CDB$ROOT
orabaseXDB       02-MAR-2017 07:52:46 LONG  CDB$ROOT
```

Application Containers Demo 1: Create Application Root (12:12)

```
ALTER SESSION SET CONTAINER=uwapp_root;

Session altered.

sho con_id

CON_ID
-------
      4
show con_name

CON_NAME
------------------------------
UWAPP$ROOT

SELECT tablespace_name TBS_NAME, file_name
FROM dba_data_files
UNION
SELECT tablespace_name, file_name
FROM dba_temp_files
ORDER BY 1;

TBS_NAME FILE_NAME
-------- ---------------------- ----------------------------------------------------
SYSAUX    /u01/app/oracle/oradata/orabase/uwapp/sysaux01.dbf
SYSTEM    /u01/app/oracle/oradata/orabase/uwapp/system01.dbf
TEMP      /u01/app/oracle/oradata/orabase/uwapp/temp012017-03-02_07-53-20-031-AM.dbf
UNDOTBS1  /u01/app/oracle/oradata/orabase/uwapp/undotbs01.dbf
```

Application Installation

Application Containers Demo 2: Application Installation (1:5)

```
ALTER PLUGGABLE DATABASE APPLICATION
{ { app_name
{ BEGIN INSTALL 'app_version' [ COMMENT 'comment' ]
| END INSTALL [ 'app_version' ]
| BEGIN PATCH number [ MINIMUM VERSION 'app_version' ] [ COMMENT 'comment' ]
| END PATCH [ number ]
| BEGIN UPGRADE 'start_app_version' TO 'end_app_version' [ COMMENT 'comment' ]
| END UPGRADE [ TO 'end_app_version' ]
| BEGIN UNINSTALL
| END UNINSTALL
| SET PATCH number
| SET VERSION 'app_version'
| SET COMPATIBILITY VERSION { 'app_version' | CURRENT }
| SYNC }
|
{ ALL SYNC }
}
```

Application Containers Demo 2: Application Installation (2:5)

```
ALTER PLUGGABLE DATABASE APPLICATION uw_app BEGIN INSTALL '1.0';

Pluggable database altered.

-- create application tablespace
CREATE TABLESPACE uwapp_tbs
DATAFILE '/u01/app/oracle/oradata/orcl12c/uwapp/uwapp_tbs'
SIZE 25M AUTOEXTEND ON NEXT 25M;

Tablespace created.

-- create user
CREATE USER uwapp_user IDENTIFIED BY uwapp_user
DEFAULT TABLESPACE uwapp_tbs
TEMPORARY TABLESPACE temp
QUOTA UNLIMITED ON uwapp_tbs;

User created.

-- grant system privileges
GRANT create session TO uwapp_owner;
GRANT create procedure TO uwapp_owner
GRANT create table TO uwapp_owner;
GRANT create view TO uwapp_owner;

Grant succeeded.
```

Application Containers Demo 2: Application Installation (3:5)

```
CREATE OR REPLACE PROCEDURE uwapp_owner.who_am_i AUTHID DEFINER IS
BEGIN
  dbms_output.put_line('I do not know');
END who_am_i;
/

CREATE TABLE uwapp_owner.t1 (
tid       NUMBER(10),
last_name VARCHAR2(20));

ALTER TABLE uwapp_owner.t1
ADD PRIMARY KEY (tid);

CREATE TABLE uwapp_owner.t2(
tid NUMBER(10),
last_name VARCHAR2(20));

ALTER TABLE uwapp_owner.t2
ADD PRIMARY KEY (tid);

CREATE VIEW uwapp_owner.t1t2_view AS
SELECT t1.tid, t2.last_name
FROM uwapp_user.t1, uwapp_user.t2
WHERE t1.tid = t2.tid;
```

Application Containers Demo 2: Application Installation (4:5)

```
INSERT INTO uwapp_owner.t1 VALUES (1, 'MORGAN');
INSERT INTO uwapp_owner.t1 VALUES (2, 'KYTE');
INSERT INTO uwapp_owner.t1 VALUES (3, 'LEWIS');
INSERT INTO uwapp_owner.t2 VALUES (1, 'TOWNSEND');
INSERT INTO uwapp_owner.t2 VALUES (2, 'KURIAN');
COMMIT;

SELECT * FROM uwapp_user.t1t2_view;

TID        LAST_NAME
---------- --------------------
         1 TOWNSEND
         2 KURIAN

ALTER PLUGGABLE DATABASE APPLICATION uw_app END INSTALL;

Pluggable database altered.
```

Application Containers Demo 2: Application Installation (5:5)

```
col app_name format a10
col app_version format a12

SELECT app_name, app_version, app_status, app_implicit
FROM dba_applications
WHERE app_name = 'UW_APP';

APP_NAME   APP_VERSION  APP_STATUS   A
---------- ------------ ------------ -
UW_APP     1.0          NORMAL       N

-- if there are application PDBs, under the application root
-- container, exit and synchronize them with their root with
-- ALTER APPLICATION PDB
```

Application Upgrade

Application Containers Demo 3: Sharable Objects (1:3)

```
SQL> CREATE TABLE servers (
  2  srvr_id NUMBER(10),
  3  network_id NUMBER(10),
  4  status VARCHAR2(1),
  5  latitude FLOAT(20),
  6  longitude FLOAT(20),
  7  netaddress VARCHAR2(15));

Table created.

SQL> CREATE TABLE serv_inst
  2  SHARING=METADATA (
  3  siid NUMBER(10),
  4  si_status VARCHAR2(15),
  5  type VARCHAR2(5),
  6  installstatus VARCHAR2(1),
  7  location_code NUMBER(10),
  8  custacct_id VARCHAR2(10),
  9  srvr_id NUMBER(10),
 10* ws_id NUMBER(10));
SHARING=METADATA
*
ERROR at line 9:
ORA-00922: missing or invalid option

SQL> show parameter default_sharing

NAME                 TYPE        VALUE
-------------------- ----------- ---------
default_sharing      string      METADATA
```

Application Containers Demo 3: Sharable Objects (2:3)

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION uw_app BEGIN INSTALL '1.0';
ALTER PLUGGABLE DATABASE APPLICATION uw_app BEGIN INSTALL '1.0'
*
ERROR at line 1:
ORA-65221: application UW_APP exists already

SQL> ALTER PLUGGABLE DATABASE APPLICATION uw_app
  2  BEGIN UPGRADE '1.0' TO '2.0'
  3  COMMENT 'Adding New Table With Sharing';

SQL> CREATE TABLE serv_inst
  2  SHARING=METADATA (
  3  siid NUMBER(10),
  4  si_status VARCHAR2(15),
  5  type VARCHAR2(5),
  6  installstatus VARCHAR2(1),
  7  location_code NUMBER(10),
  8  custacct_id VARCHAR2(10),
  9  srvr_id NUMBER(10),
 10* ws_id NUMBER(10));

Table created.
```

Application Containers Demo 3: Sharable Objects (3:3)

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION uw_app END UPGRADE;

SQL> desc dba_applications
Name                      Null?    Type
------------------------- -------- --------------
APP_NAME                           VARCHAR2(128)
APP_ID                             NUMBER
APP_VERSION                        VARCHAR2(30)
APP_STATUS                         VARCHAR2(12)
APP_IMPLICIT                       VARCHAR2(1)
APP_CAPTURE_SERVICE                VARCHAR2(64)
APP_CAPTURE_MODULE                 VARCHAR2(64)

col app_name format a37

SQL> SELECT app_name, app_id, app_version, app_status, app_implicit
  2  FROM dba_applications;

APP_NAME                                 APP_ID APP_VERSION  APP_STATUS   A
------------------------------------ ---------- ------------ ------------ -
APP$4BAF1A01C5964D55E0530100007F821B          2 1.0          NORMAL       Y
UW_APP                                       21 2.0          NORMAL       N
```

Application Seed Creation

Application Containers Demo 4: Create Seed (1:4)

```
CREATE PLUGGABLE DATABASE AS SEED FROM uwapp_root
ADMIN USER uwappAdmin IDENTIFIED BY uwappAdmin
FILE_NAME_CONVERT = ('/uwapp/', '/uwappseed/')
LOGGING;

SQL> SELECT con_id, name, open_mode, application_root, application_pdb,
application_seed
  2  FROM v$containers
  3  ORDER BY con_id;

CON_ID  NAME             OPEN_MODE   APP APP APP
------- ---------------- ----------- --- --- ---
     4  UWAPP_ROOT       READ WRITE  YES NO  NO
     6  UWAPP_ROOT$SEED  MOUNTED     NO  YES YES

SQL> ALTER PLUGGABLE DATABASE uwapp_root$seed OPEN;

Warning: PDB altered with errors.

SQL> ALTER PLUGGABLE DATABASE uwapp_root$seed OPEN READ ONLY;

Warning: PDB altered with errors.
```

Application Containers Demo 4: Create Seed (2:4)

```
SQL> host
[oracle@vbgeneric ~]$ cd $ORACLE_BASE/diag/orabase/orabase/trace
[oracle@vbgeneric trace]$ tail alert_orabase.log
UWAPP_ROOT$SEED(6):Opatch validation is skipped for PDB UWAPP_ROOT$SEED (con_id=0)
UWAPP_ROOT$SEED(6):************************************************************
UWAPP_ROOT$SEED(6):WARNING: Pluggable Database UWAPP_ROOT$SEED with pdb id - 6 is
UWAPP_ROOT$SEED(6): altered with errors or warnings. Please look into
UWAPP_ROOT$SEED(6): PDB_PLUG_IN_VIOLATIONS view for more details.
UWAPP_ROOT$SEED(6):************************************************************
2017-04-19T18:45:58.662039-04:00
UWAPP_ROOT$SEED(6):Opening pdb with no Resource Manager plan active
Pluggable database UWAPP_ROOT$SEED opened read only
UWAPP_ROOT(4):Completed: ALTER PLUGGABLE DATABASE uwapp_root$seed OPEN READ ONLY
```

```
[oracle@vbgeneric ~]$exit
SQL> col time format a29
SQL> col name format a16
SQL> col type format a6
SQL> col cause format a30
SQL> col action format a22

SQL> SELECT time, name, cause, status, action
  2* FROM pdb_plug_in_violations;

TIME                          NAME             CAUSE                                    STATUS   ACTION
----------------------------- ---------------- ---------------------------------------- -------- ----------------------
19-APR-17 06.45.57.958082 PM  UWAPP_ROOT$SEED  Non-Application PDB to Application PDB  PENDING  Run pdb_to_apppdb.sql.
```

Application Containers Demo 4: Create Seed (3:4)

```
SQL> @?/rdbms/admin/pdb_to_apppdb.sql
SQL> SET FEEDBACK 1
SQL> SET NUMWIDTH 10
SQL> SET LINESIZE 80
SQL> SET TRIMSPOOL ON
SQL> SET TAB OFF
SQL> SET PAGESIZE 100
SQL>
SQL> WHENEVER SQLERROR EXIT;
SQL>
SQL> VARIABLE cdbname VARCHAR2(128)
SQL> VARIABLE pdbname VARCHAR2(128)
SQL> VARIABLE appname VARCHAR2(128)
SQL> BEGIN
2 -- Disallow script in non-CDB
3 SELECT sys_context('USERENV', 'CDB_NAME')
4 INTO :cdbname
5 FROM dual
6 WHERE sys_context('USERENV', 'CDB_NAME') is not null;
7 -- Disallow script in CDB Root
8 -- Disallow script in PDB$SEED (Bug 22550952)
9 SELECT sys_context('USERENV', 'CON_NAME')
10 INTO :pdbname
11 FROM dual
12 WHERE sys_context('USERENV', 'CON_NAME') <> 'CDB$ROOT'
13 AND sys_context('USERENV', 'CON_NAME') <> 'PDB$SEED';
14 -- Disallow script outside of Application Container
15 SELECT sys_context('USERENV', 'APPLICATION_NAME')
16 INTO :appname
17 FROM dual
18 WHERE sys_context('USERENV', 'APPLICATION_NAME') is not null;
19 -- Disallow script in Proxy PDB (Bug 22550952). This query works
20 -- because remote mapping in Proxy PDB has been disabled using
21 -- the underscore parameter.
22 SELECT /*+ OPT_PARAM('_ENABLE_VIEW_PDB', 'FALSE') */ name
23 INTO :pdbname
24 FROM v$pdbs
25 WHERE proxy_pdb='NO';
26 END;
27 /
BEGIN
*
ERROR at line 1:
ORA-01422: exact fetch returns more than requested number of rows
ORA-06512: at line 22

Disconnected from Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production
[oracle@vbgeneric ~]$
```

Application Containers Demo 4: Create Seed (4:4)

- On dissecting the pdb_to_apppdb.sql script the following was identified as the root cause of the error

```
SQL> SELECT /*+ OPT_PARAM('_ENABLE_VIEW_PDB', 'FALSE') */ name
  2  FROM v$pdbs
  3  WHERE proxy_pdb='NO';

NAME
------------------------------
UWAPP_ROOT
UWAPP_ROOT$SEED
```

- Further examination of the script(s) found numerous examples of this

```
select TO_NUMBER('NOT_IN_APPLICATION_PDB') from v$pdbs
where con_id=sys_context('USERENV', 'CON_ID') and application_pdb<>'YES';
```

- Clearly the script has no choice but to fail

- More news on this in our Slack group as it becomes available

SQL*Net

TNSNAMES Configuration

- Every time you add a new PDB ... you must also make a manual entry to TNSNAMES.ORA

```
# tnsnames.ora Network Configuration File:
C:\app\oracle\product\12.1.0\dbhome_1\network\admin\tnsnames.ora
# Generated by Oracle configuration tools.

PDBDEV =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = 127.0.0.1)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = pdbdev)
    )
  )

PDBTEST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 127.0.0.1)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = pdbtest)
    )
  )

ORACLR_CONNECTION_DATA =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
    (CONNECT_DATA =
      (SID = CLRExtProc)
      (PRESENTATION = RO)
    )
  )
```

```
PDBPROD =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 127.0.0.1)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = pdbprod)
    )
  )

ORABASE =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 127.0.0.1)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orabase)
    )
  )
```

Lockdown Profiles

Lockdown Profile Enhanced Security

- Lockdown profiles compliment to GRANTs
- GRANTs, alone, are all or nothing
- A Lockdown Profile makes a system permission grant more granular

```
SQL> CREATE LOCKDOWN PROFILE devel_pdbs;

Lockdown Profile created.

SQL> ALTER LOCKDOWN PROFILE devel_pdbs
  2  DISABLE STATEMENT=('ALTER SYSTEM')
  3  CLAUSE=('SET')
  4* OPTION ALL EXCEPT = ('plsql_code_type, plsql_debug, plsql_warning');

SQL> ALTER SYSTEM SET pdb_lockdown = 'DEVEL_PDBS' SID='*' scope=BOTH;

System altered.

SQL> show parameter lockdown

NAME          TYPE    VALUE
------------- ------- -----------
pdb_lockdown  string  DEVEL_PDBS

SQL> ALTER SESSION SET CONTAINER=PDBDEV;

Session altered.

SQL> ALTER SYSTEM SET pdb_lockdown = 'DEVEL_PDBS' SID='*' scope=BOTH;

System altered.

SQL> GRANT alter system TO pdb_user;
```

Built-in PL/SQL Objects

12.1 DBMS_PDB (1:2)

- Provides an interface to examine and manipulate data about pluggable databases
- Subprograms
  - CHECK_PLUG_COMPATIBILITY
  - DESCRIBE
  - RECOVER
  - REMOVE_LINK
  - SET_DATA_LINKED
  - SET_EXT_DATA_LINKED
  - SET_METADATA_LINKED
  - SET_PROFILE_EXPLICIT
  - SET_ROLE_EXPLICIT
  - SET_USER_EXPLICIT

12.1 DBMS_PDB (2:2)

```
conn / as sysdba

ALTER PLUGGABLE DATABASE pdbdev CLOSE;

ALTER PLUGGABLE DATABASE pdbdev OPEN READ ONLY;

exec dbms_pdb.describe('/home/oracle/pdbdev.xml', 'PDBDEV');

BEGIN
  IF dbms_pdb.check_plug_compatibility('/home/oracle/pdbdev.xml', 'PDBDEV') THEN
    dbms_output.put_line('TRUE');
  ELSE
    dbms_output.put_line('FALSE');
  END IF;
END;
/

SELECT *
FROM pdb_plug_in_violations;
```

12.1 DBMS_PDB_EXEC_SQL

- Stand-alone stored procedure
- Calls DBMS_PDB.EXEC_AS_ORACLE_SCRIPT
- Executes a statement as an Oracle Script

```
CREATE OR REPLACE NONEDITIONABLE PROCEDURE sys.dbms_pdb_exec_sql(sql_stmt IN VARCHAR2) AS
BEGIN
  dbms_pdb.exec_as_oracle_script(sql_stmt);
END;
/
```

```
-- $ORACLE_HOME/rdbms/admin/catadmprvs.sql

@@?/rdbms/admin/sqlsessstart.sql

alter session set "_ORACLE_SCRIPT"=true;

-----------------------------------------------------
-- Grant required privileges to administrative users
-----------------------------------------------------
-------------
-- SYSBACKUP
-------------
-- To perfrom backup and recovery tasks.
GRANT alter database                          TO sysbackup;
GRANT alter session                           TO sysbackup;
...
@?/rdbms/admin/sqlsessend.sql
```

12.2 DBMS_PDB_ALTER_SHARING

- This package can set a database object to one of the following types of common objects in a PDB: data-linked object, extended data-linked object, or metadata-linked object
- An application can be migrated to CDB$ROOT or to an application PDB. For example, an application can be migrated from an application installed in a PDB plugged into a 12.1 CDB to a PDB in a 12.2 CDB
- Subprograms
  - REMOVE_LINK
  - SET_DATA_LINKED
  - SET_EXT_DATA_LINKED
  - SET_METADATA_LINKED
  - SET_PROFILE_EXPLICIT
  - SET_ROLE_EXPLICIT
  - SET_USER_EXPLICIT

```
exec dbms_pdb_alter_sharing.set_metadata_linked('C##UWCLASS','ACCOUNTS', 1);
```

12.2 DBMS_SYNC_REFRESH (1:2)

- Synchronous refresh is a refresh method introduced in Oracle Database Release 12c to enable synchronizing a set of tables and dependent materialized views
- Subprograms
  - ABORT_REFRESH
  - ALTER_REFRESH_STATS_RETENTION
  - CAN_RESYNC_TABLE
  - EXECUTE_REFRESH
  - GET_ALL_GROUP_IDS, GET_GROUP_ID, GET_GROUP_ID_LIST
  - PREPARE_REFRESH
  - PREPARE_STAGING_LOG
  - PURGE_REFRESH_STATS
  - REGISTER_MVIEWS
  - REGISTER_PARTITION_OPERATION
  - UNREGISTER_MVIEWS
  - UNREGISTER_PARTITION_OPERATION

12.2 DBMS_SYNC_REFRESH (2:2)

Value	Description
1 to 365,000	Valid range in days
0	Refresh history not saved by PREPARE_REFRESH
-1	Refresh history not purged by PREPARE_REFRESH
NULL	Return to default value (31)

```
col parameter_name format a35
col str_value format a20

SELECT *
FROM syncref$_parameters;

exec dbms_sync_refresh.alter_refresh_stats_retention(90);

SELECT *
FROM syncref$_parameters;
```

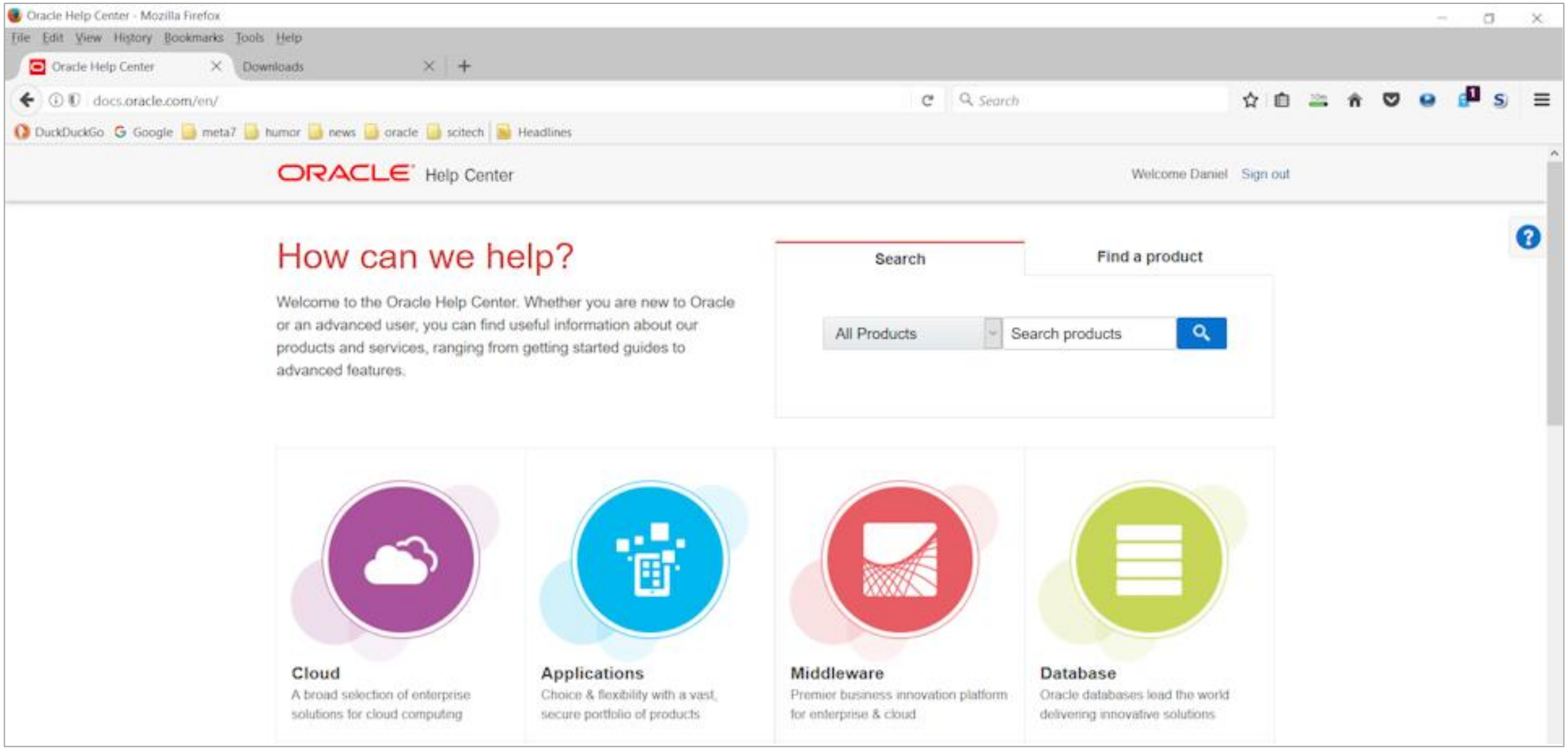
Conclusion

Conclusion

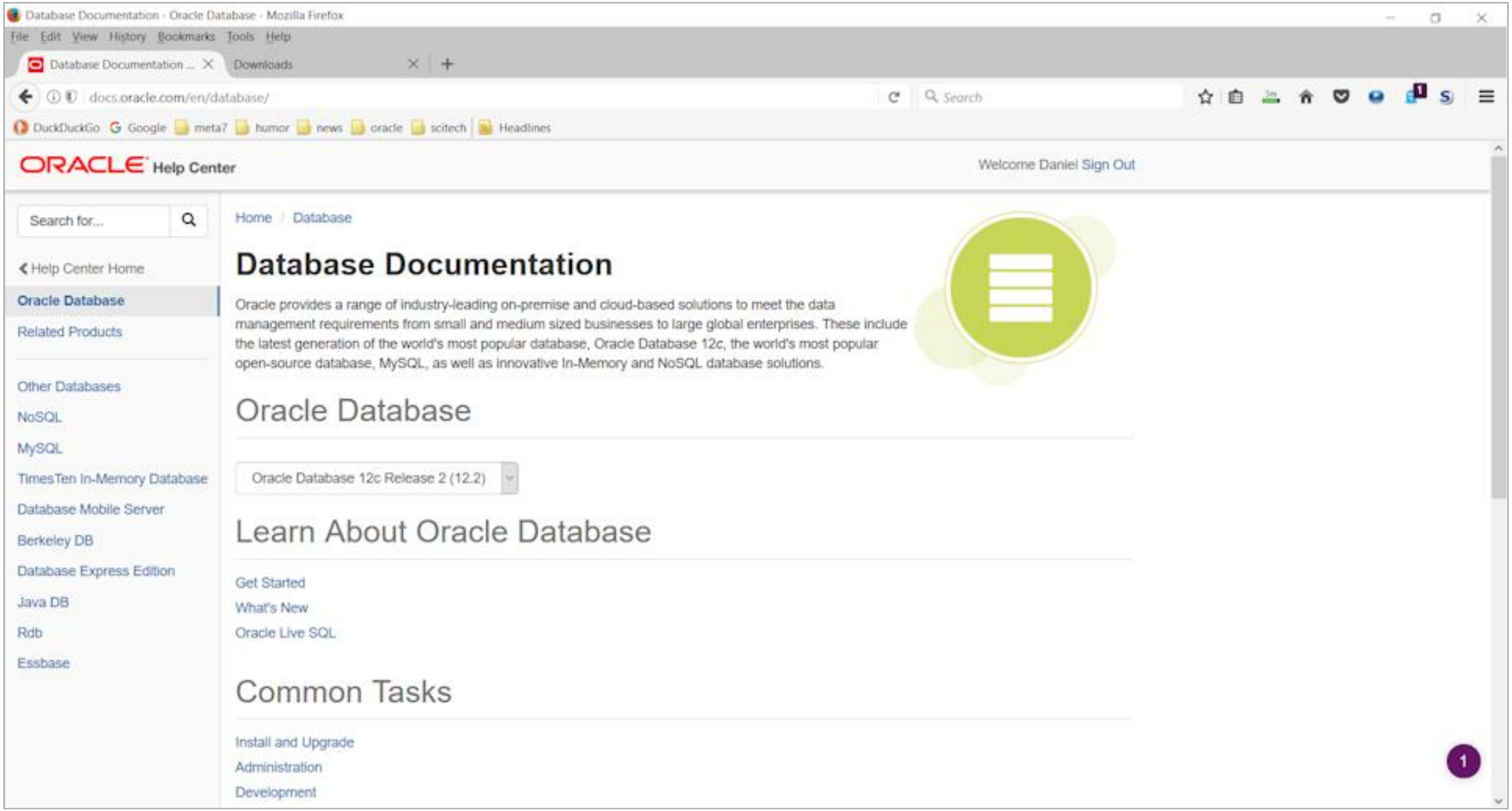
- Read the 12.2 docs
- Download the installation files
- Install it
- Learn it

- For demos that work in SQL*Plus
  www.morganslibrary.org/library.html

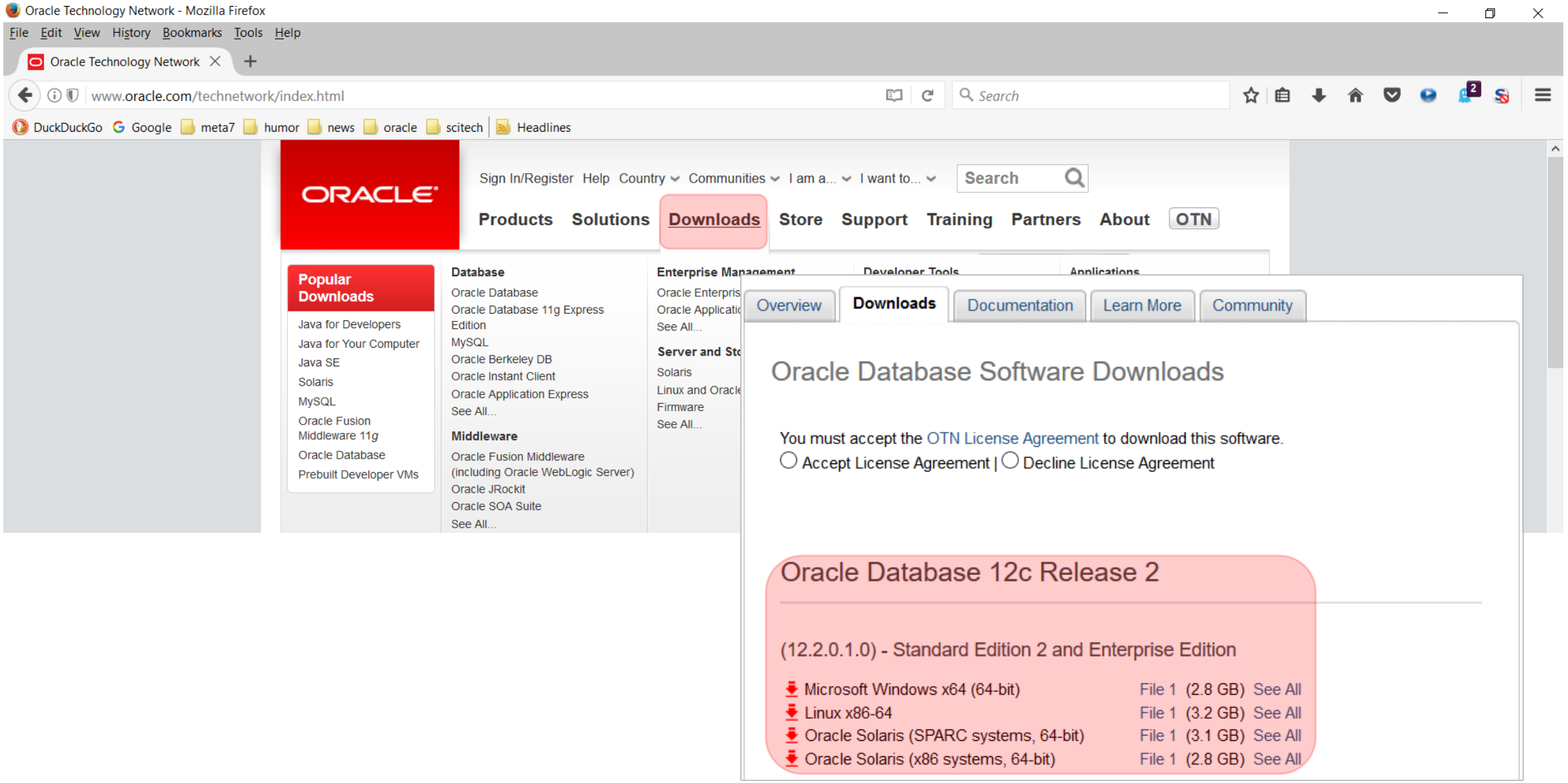
Docs: https://docs.oracle.com (1:2)



Docs: https://docs.oracle.com (2:2)



Downloads: https://otn.oracle.com



```
*
ERROR at line 1:
ORA-00028: your session has been killed
```

Thank You

Daniel A. Morgan
email: dmorgan@forsythe.com
mobile: +1 206-669-2949