

Cross-Site Load Balancing: Data agility with Oracle Multitenant

A Proof of Concept

ORACLE WHITE PAPER | JANUARY 2015





Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

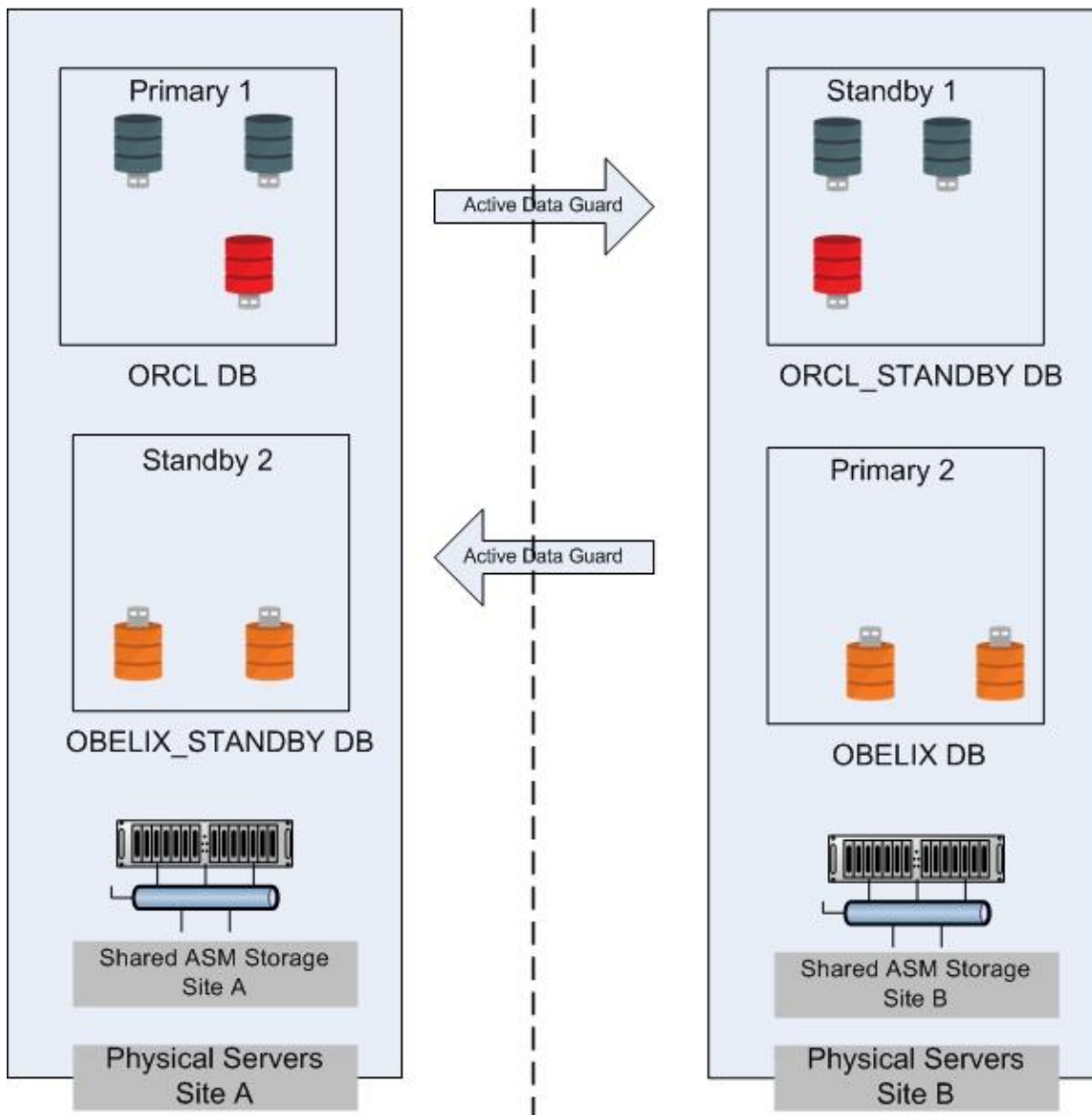


Table of Contents

Disclaimer	1
Description	2
Pre-requisites	4
Create a new PDB	4
Cross-Site load balancing procedure	6
Ensure the PDB is closed on both primary and standby	6
Unplug the PDB from Primary1	6
Copy the XML Manifest to Site2	6
Create Aliases on OBELIX and OBELIX_STANDBY	6
Ensure Media Recovery is active on OBELIX_STANDBY	8
Plug PDB_XSTE in to OBELIX	9
Confirm PDB_XSTE was created on OBELIX_STANDBY	10
Open PDB_XSTE on OBELIX	11
Optionally delete PDB from original primary, ORCL	11
Conclusion	13

Description

A single standby database architecture has the primary database in Site A and the standby database in Site B. Active Data Guard allows the standby database to be opened read-only and used for reporting, ad-hoc queries and the like. There is no restriction as to where the primary and standby can be placed, and many customers will place the standby database on the same physical servers as the production (primary) instances of another database. The diagram below describes such a configuration, and is used as the basis for this proof of concept.



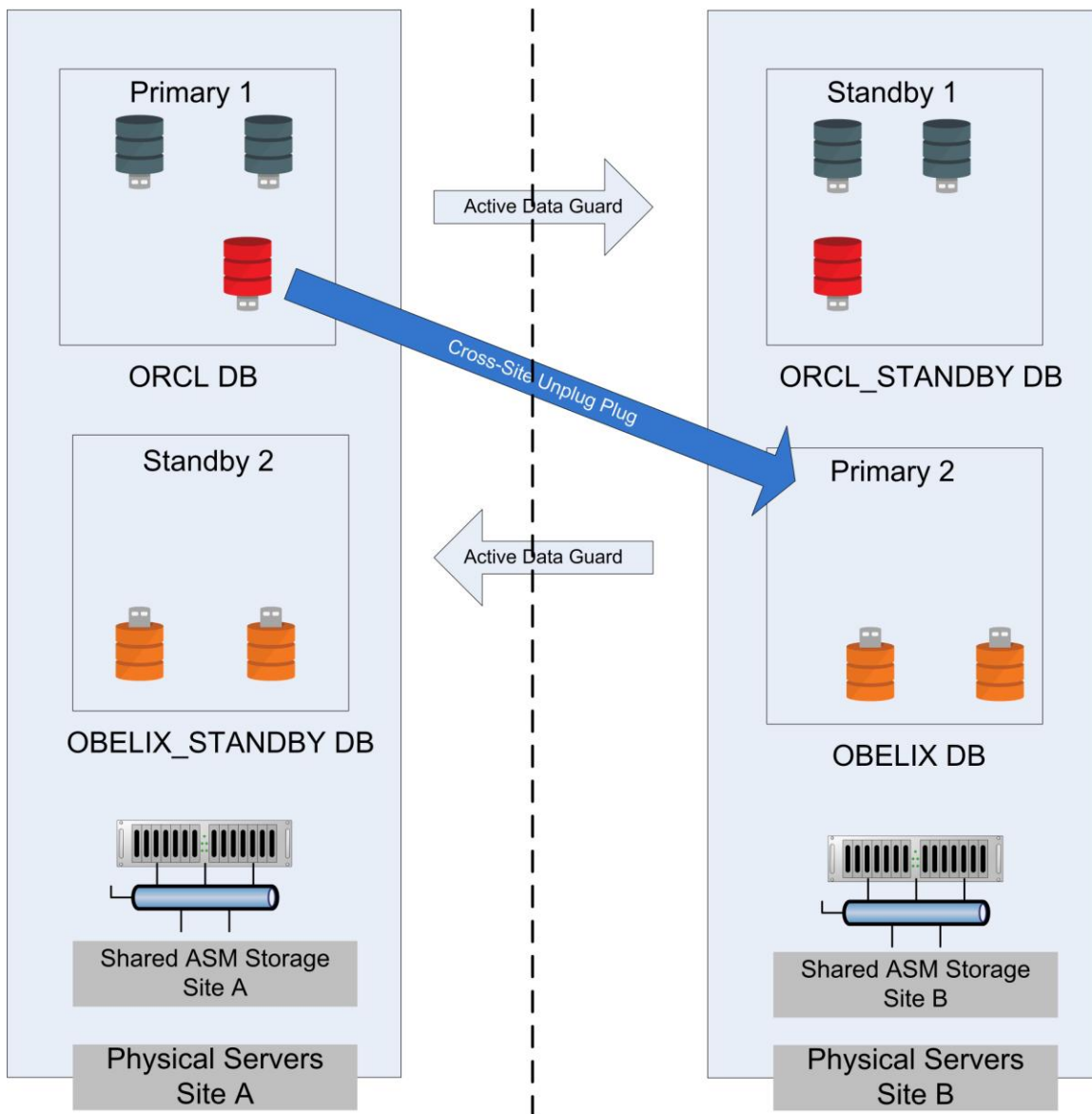
Site A contains the production instance `ORCL` whose standby instance `ORCL_STANDBY` is on Site B. Site B contains the production instance `OBELIX` whose standby instance `OBELIX_STANDBY` is on Site A.

`ORCL` and `OBELIX` are container databases containing several pluggable databases (PDBs).

Site A and Site B are both RAC clusters, though this is not strictly required for this proof of concept. Storage management at both sites is provided by ASM. The use of ASM allows for several optimizations in the cross-site operations, but as for RAC, ASM is not strictly required.

It is conceivable to imagine that in a dynamic environment, where new PDBs are created, or workload requirements change, that load on a given site will increase to the extent that resources become scarce and the production database is impacted.

The procedure outlined in this proof of concept shows how load on a site may be reduced by moving a PDB from Site A to Site B. This process could also apply to moving a given PDB such that more resources are available to it – if, for example, it was not meeting its service level agreement (SLA), or possibly during maintenance scenarios.



Pre-requisites

In order to complete the procedure outlined in this document you will need to be able to connect to all of the database instances (both primaries and both standbys) with Oracle SQL*Plus, and if you will implement the optimization using ASM aliases, `asmcmd`.

Requirements for transfer of PDBs (unplug and plug) between different CDBs must also be met.

Create a new PDB

Create a new PDB in the container `ORCL`. In this example we name the PDB: `PDB_XSTE`.

```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl as sysdba

SQL> create pluggable database PDB_XSTE admin user pdbxste identified by pdbxste
roles=(connect, resource) tempfile reuse;

Pluggable database created.

SQL> show pdbs

  CON_ID  CON_NAME          OPEN MODE  RESTRICTED
-----  -
         2  PDB$SEED          READ ONLY  NO
         3  PDB1              READ WRITE NO
...
        23  PDB_XSTE          MOUNTED    NO

SQL> alter pluggable database PDB_XSTE open
Pluggable database altered

SQL> show pdbs

  CON_ID  CON_NAME          OPEN MODE  RESTRICTED
-----  -
         2  PDB$SEED          READ ONLY  NO
         3  PDB1              READ WRITE NO
...
        23  PDB_XSTE          READ WRITE NO
```

The standby site, `SITE B`, will show corresponding information

```
[racuser@SITEB] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl_standby as
sysdba

SQL> show pdbs

  CON_ID  CON_NAME          OPEN MODE  RESTRICTED
-----  -
         2  PDB$SEED          READ ONLY  NO
         3  PDB1              READ WRITE NO
...
        23  PDB_XSTE          MOUNTED    NO
```

Identify the datafiles associated with the PDB `PDB_XSTE` on both the primary and the standby:



```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl as sysdba
SQL> select name from v$datafile where con_id = 23;
NAME
-----
+RWSBIJ1314_DATA_DG/ORCL/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system.581.86
0972039

+RWSBIJ1314_DATA_DG/ORCL/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux.758.86
0972041
```

```
[racuser@SITEB] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl_standby as
sysdba
SQL> select name from v$datafile where con_id = 23;
NAME
-----
+RWSBIJ1314_DATA_DG/ORCL_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/syste
m.755.860972045

+RWSBIJ1314_DATA_DG/ORCL_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysau
x.588.860972047
```

Cross-Site load balancing procedure

Ensure the PDB is closed on both primary and standby

```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl as sysdba
SQL> alter pluggable database PDB_XSTE close;
Pluggable database altered.
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PDB1	READ WRITE	NO
...			
23	PDB_XSTE	MOUNTED	NO

Unplug the PDB from Primary1

```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl as sysdba
SQL> alter pluggable database pdb_xste unplug into
  2  '/scratch/oracle/oracle_base/dbhome/dbs/pdb_xste.xml';
Pluggable database altered.
```

Copy the XML Manifest to SiteB

```
[racuser@SITEA] scp /scratch/oracle/oracle_base/dbhome/dbs/pdb_xste.xml racusr@SITEB:
/scratch/oracle/oracle_base/dbhome/dbs/pdb_xste.xml
[racusr@SITEA]
```

Create Aliases on OBELIX and OBELIX_STANDBY

We will be moving the PDB to a different primary database container. In our configuration the original standby and the other primary are on the same server, and use the same ASM diskgroups. We can eliminate the need to transfer files (to the new primary site) as they are already present, albeit as part of the old standby.

We are using Oracle-Managed Files with this configuration, which means the full path names to the database files contain the database name. ASM identifies the files associated with a database through the disk and file header information. The pathname is a logical construct, and does not clearly identify which database a given datafile belongs to. This can be confusing to DBAs and other humans. Therefore when we create the aliases in this section we will also change the path to better represent the database these files are now part of.

In order for the redo apply service to continue to operate an alias has to be created. On the new primary instance, OBELIX, create an alias that points to files on ORCL_STANDBY:

First, add path entries with `asmcmd` for the OBELIX database on Site B, and then connect to the ASM instance to add the alias to the diskgroup:

```
[racuser@SITEB] export ORACLE_SID=+ASM2

[racusr@SITEB] export ORACLE_HOME=/scratch/oracle/12.1.0/grid

[racusr@SITEB]$ $ORACLE_HOME/bin/asmcmd
ASMCMD> cd +RWSBIJ1314_DATA_DG
ASMCMD> cd OBELIX
ASMCMD> mkdir 0570BC129DBE51BBE053F501D10A9471
ASMCMD> cd 0570BC129DBE51BBE053F501D10A9471
ASMCMD> mkdir DATAFILE
ASMCMD> cd DATAFILE
ASMCMD> ls
ASMCMD> quit
[racusr@SITEB]$ $ORACLE_HOME/bin/sqlplus sys/oracle as sysasm

SQL> alter diskgroup RWSBIJ1314_DATA_DG add alias
'+RWSBIJ1314_DATA_DG/OBELIX/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system_755_8609
72045' FOR
'+RWSBIJ1314_DATA_DG/ORCL_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system.75
5.860972045';

Diskgroup altered.

SQL> alter diskgroup RWSBIJ1314_DATA_DG add alias
'+RWSBIJ1314_DATA_DG/OBELIX/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux_588_8609
72047' FOR
'+RWSBIJ1314_DATA_DG/ORCL_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux.58
8.860972047';

Diskgroup altered.

SQL> quit
```

The datafile path corresponds to that which we read from the standby site earlier. The alias name has the same path, except that underscores (“_”) have replaced periods (“.”) in the file names, and we replaced the standby database name (ORCL_STANDBY) with the new primary name (OBELIX).

Now we create aliases for the new standby instance (OBELIX_STANDBY) on Site A.

```

[racuser@SITEA] export ORACLE_SID=+ASM1

[racusr@SITEA] export ORACLE_HOME=/scratch/oracle/12.1.0/grid

[racusr@SITEB]$ $ORACLE_HOME/bin/asmcmd
ASMCMD> cd +RWSBIJ1314_DATA_DG
ASMCMD> cd OBELIX_STANDBY
ASMCMD> mkdir 0570BC129DBE51BBE053F501D10A9471
ASMCMD> cd 0570BC129DBE51BBE053F501D10A9471
ASMCMD> mkdir DATAFILE
ASMCMD> cd DATAFILE
ASMCMD> ls
ASMCMD> quit
[racusr@SITEB]$ $ORACLE_HOME/bin/sqlplus sys/oracle as sysasm

SQL> alter diskgroup RWSBIJ1314_DATA_DG add alias
'+RWSBIJ1314_DATA_DG/OBELIX_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system_
755_860972045' FOR
'+RWSBIJ1314_DATA_DG/ORCL/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system.581.860972
039';

Diskgroup altered.

SQL> alter diskgroup RWSBIJ1314_DATA_DG add alias
'+RWSBIJ1314_DATA_DG/OBELIX_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux_
588_860972047' FOR
'+RWSBIJ1314_DATA_DG/ORCL/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux.758.860972
041';

Diskgroup altered.

SQL> quit

```

Note that the alias names in the standby use the filenames recently created on the new primary. In this example this corresponds to the names containing the underscores, that is `system_755_860972045`. The aliases created for these files point to files that belong to the original primary, `ORCL`.

Ensure Media Recovery is active on OBELIX_STANDBY

Before plugging the PDB in to the new primary, OBELIX, ensure that the Media Recovery Process is active on the OBELIX_STANDBY instance:

```

[racuser@SITEA] export ORACLE_SID=obelix_standby1

[racusr@SITEA] export ORACLE_HOME=/scratch/oracle/oracle_base/dbhome

[racusr@SITEA]$ $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/obelix_standby
as sysdba

SQL> set lines 120
SQL> set pages 9999
SQL> select * from v$managed_standby where process='MRP0';

```

PROCESS	PID		STATUS	CLIENT_P	CLIENT_PID

CLIENT_DBID			GROUP#		
RESETLOG_ID	THREAD#	SEQUENCE#			

BLOCK#	BLOCKS	DELAY_MINS	KNOWN_AGENTS	ACTIVE_AGENTS	CON_ID

MRP0	13359		APPLYING_LOG	N/A	N/A
N/A			N/A		
854748558	1	836			
43146	204800	0	25	25	0

If the MRP0 process is not running, examine the alert log and/or trace files as to why it has stopped.

Plug PDB_XSTE in to OBELIX

Plugin the PDB_XSTE on OBELIX primary using the directive `source_file_directory=<Full directory of OBELIX primary datafile location>`

The PLUG operation will use the location specified in the path `source_file_directory` to search for the datafiles with which to create the pluggable database. The file names will not exist, but the header information stored in the XML manifest can be used to match the actual files.

On the standby site the MRP process will look for files in the default location (`<diskgroup_name/db_unique_name/guid/DATAFILE>`) and again try to match files based on header information passed in the redo stream.

```
[racuser@SITEB] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/obelix as
sysdba

SQL> create pluggable database PDB_XSTE using
  2  '/scratch/oracle/oracle_base/dbhome/dbs/pdb_xste.xml'
  3
source_file_directory='+RWSBIJ1314_DATA_DG/OBELIX/0570BC129DBE51BBE053F501D10A9471/DA
TAFILE' tempfile reuse
  4  nocopy;

Pluggable database created.

SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
  2 PDB$SEED                                     READ ONLY  NO
  3 PDB1                                         READ WRITE NO
...
  18 PDB_XSTE                                   MOUNTED

SQL> select name from v$datafile where con_id=18;

NAME
-----
+RWSBIJ1314_DATA_DG/OBELIX/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system_755_
860972045

+RWSBIJ1314_DATA_DG/OBELIX/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux_588_
860972047
```

Confirm PDB_XSTE was created on OBELIX_STANDBY

```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/obelix_standby
as sysdba

SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
  2 PDB$SEED                                     READ ONLY  NO
  3 PDB1                                         READ WRITE NO
...
  18 PDB_XSTE                                   MOUNTED

SQL> select name from v$datafile where con_id=18;

NAME
-----
+RWSBIJ1314_DATA_DG/OBELIX_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/system_7
55_860972045

+RWSBIJ1314_DATA_DG/OBELIX_STANDBY/0570BC129DBE51BBE053F501D10A9471/DATAFILE/sysaux_5
88_860972047
```

Open PDB_XSTE on OBELIX

The PDB can now be opened **READ WRITE** on the new primary, OBELIX, and **READ ONLY** on OBELIX_STANDBY.

On SITE2

```
[racuser@SITEB] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/obelix as sysdba

SQL> alter session set container=pdb_xste;

Session altered.

SQL> startup restrict
Pluggable Database opened.
SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      18 PDB_XSTE                                READ WRITE YES
SQL> exec dbms_pdb.sync_pdb();

PL/SQL procedure successfully completed.

SQL> alter system disable restricted session;

System altered.
SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      18 PDB_XSTE                                READ WRITE NO
```

On SITE1

```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/obelix_standby as sysdba

SQL> alter session set container=pdb_xste;
Session altered.

SQL> alter pluggable database pdb_xste open read only;
Pluggable database altered.

SQL> alter tablespace temp add tempfile;

Tablespace altered.

SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      18 PDB_XSTE                                READ ONLY  NO
```

Optionally delete PDB from original primary, ORCL

The PDB can now be dropped from the ORCL database, remembering to KEEP the datafiles, as they are now associated with the database OBELIX_STANDBY.

```
[racuser@SITEA] $ORACLE_HOME/bin/sqlplus sys/oracle@myscan.oracle.com/orcl as sysdba

SQL> drop pluggable database pdb_xste keep datafile;
Pluggable database dropped.

SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PDB1	READ WRITE	NO
...			
22	PDB20	MOUNTED	



Conclusion

This proof-of-concept shows that load balancing across sites is possible using a combination of Oracle Multitenant and Oracle Data Guard.

When coupled with the flexibility offered by Oracle Real Application Clusters for local site management it is a complete demonstration of the agility and flexibility required by those customers offering Database as a Service.







Oracle Corporation, World Headquarters

500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries

Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Hardware and Software, Engineered to Work Together

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0115

White Paper Title: Cross Site Load Balancing: Data agility with Oracle Multitenant, A proof of concept
January 2015
Author: Silviu Teodoru
Contributing Authors: Troy Anthony, Frank Kobylanski, Ciprian Pustianu



Oracle is committed to developing practices and products that help protect the environment