ORACLE

WebLogic for OCI Disaster Recovery

Overview

MAA PaaS team

WebLogic for OCI Disaster Recovery

WebLogic for OCI Disaster Recovery

WebLogic for OCI Disaster Recovery

Introduction

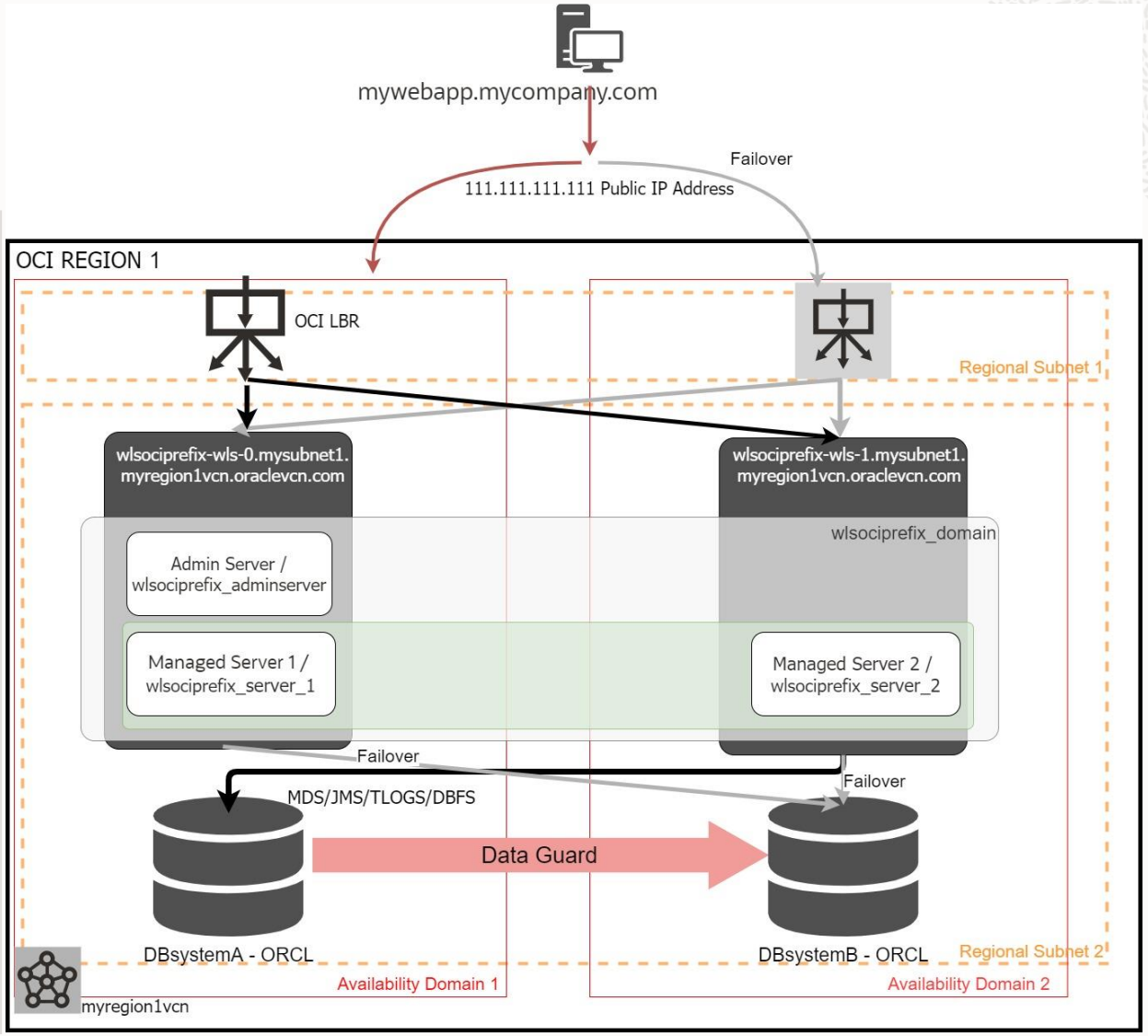
- Oracle Maximum Availability Architectures (MAA)
  - Oracle's **best practices** to provide optimal **high availability**, **data protection** and **disaster recovery** for Oracle customers at the **lowest cost and complexity,** by minimizing the **RTO** and **RPO** of the system.
  - Consist of reference architectures, configuration, and HA Life Cycle operational best practices, for all the Oracle Stack (cloud, non-cloud, engineered systems, etc.).

- Disaster Recovery (DR)
  - DR are MAA architectures intended to protect critical mission systems by **providing a secondary system** in another **geographically-separated** area.
  - DR is **additional protection** to High Availability. WLS for OCI provides High Availability out-of-the-box (inside the region).

WebLogic for OCI Disaster Recovery

Introduction – High Availability in scope of a region

- WLs for OCI uses the Active **high availability** (HA) policy **for compute** when it provisions instance compute nodes: virtual machines (VM) fail over automatically to another physical compute node in the same compute zone in case the primary compute node fails.
- A **different Fault Domain** is used by default for each compute instance used by the WLS cluster
- When using **regional subnets**, the provisioning process places **each compute** instance used by the WLS cluster in a **different Availability Domain**
- Additionally, the **front-end LBR** used by WLS for OCI **is regional and failover** across ADs provided **OOTB** for regions with more than one AD
- The **Database** can also be protected **against AD failures** by using **Oracle Data Guard** and placing the standby in a different ADs (see on-prem MDC AA for Datasource configuration)
- This configuration, however, **does not provide protection against** disasters that affect an **entire region**



WebLogic for OCI Disaster Recovery

Introduction

- The DR solution for WLS for OCI involves setting up an standby system at a **geographically-separated** Oracle Cloud Data Center, in a **active-passive** model.

Based on solid and proven DR technologies

- While there are some unique considerations to a cloud disaster recovery configuration, it follows the same Oracle MAA best practices as any Oracle Fusion Middleware (FMW) and Oracle Database deployment
- Based on Data Guard (more than 20 years providing DR)

Cross-region

- The DR solution for WLS for OCI involves setting up an standby system at a **geographically** different Oracle Cloud Data Center, in a **active-passive** model.
- Cross-region DR is a real protection for any unforeseen (natural or man-made) event that can put your organization at risk

Provides the best RTO and RPO

- By utilizing high availability and disaster protection capabilities provided by Oracle Fusion Middleware and Oracle Database. RTO for a typical switchover: 15-30 minutes

WebLogic for OCI Disaster Recovery
Introduction

- WLS for OCI is a **customer managed** service:
  - the Disaster Recovery initial configuration and lifecycle ops will be performed by the customer.
- Oracle provides:

A **framework** with:

- Automation scripts to configure the mid-tiers in an active/passive topology with a standby ready to take over.
- An automated procedure for replicating WLS configuration changes to standby during the lifecycle .
- Scripts to setup Data Guard in the DB layer across regions (NOW this can be performed with the OCI Console)

A description of how to operate on the system (**lifecycle**):

- Switchover & failover
- Open secondary for validation
- Syncing WLS configuration
- Scale-in/out etc.

All described in the public WLS for OCI DR **whitepaper**:

- https://www.oracle.com/a/otn/docs/middleware/maa-wls-mp-dr.pdf

WebLogic for OCI Disaster Recovery

Introduction

Assumptions

- WebLogic Editions
  - Oracle WebLogic Suite Edition and WebLogic Enterprise Edition
  - When using RAC: only to Suite Edition, because it uses GridLink Datasources.
- Authentication
  - Both default and IDCS authentication are supported.
  - IDCS's own DR is out-of-scope of this paper.
- Load Balancer
  - Assumed an OCI LBR is used in each WLS for OCI instance.
- Database
  - Assumed Database is used.
  - RAC is supported.
  - Autonomous database is supported (but documented in a separate playbook).

Requirements

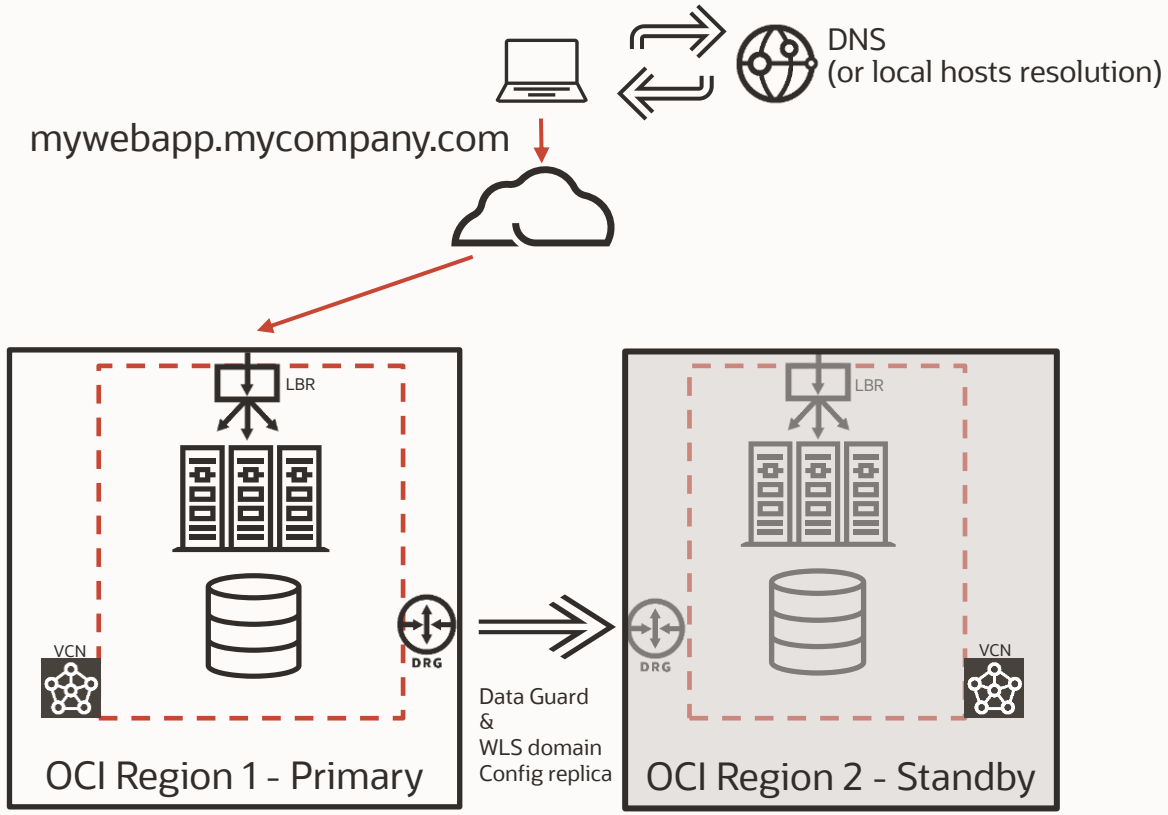
- Unique Frontend Address
  - The frontend address name used to access the system must be unique. Usually referred to as "virtual frontend" or "vanity url".
- Same WebLogic Resource Name Prefix
  - It must be the same value in the primary and secondary WLS for OCI systems.
- Network Communication between sites
  - Required. Dynamic Routing Gateway and remote peering recommended.
- Staging File System
  - Staging file systems are required for WLS Config replication. They can be DBFS or FSS (more details later).

WebLogic for OCI Disaster Recovery

WebLogic for OCI Disaster Recovery

DR Topology - Overview

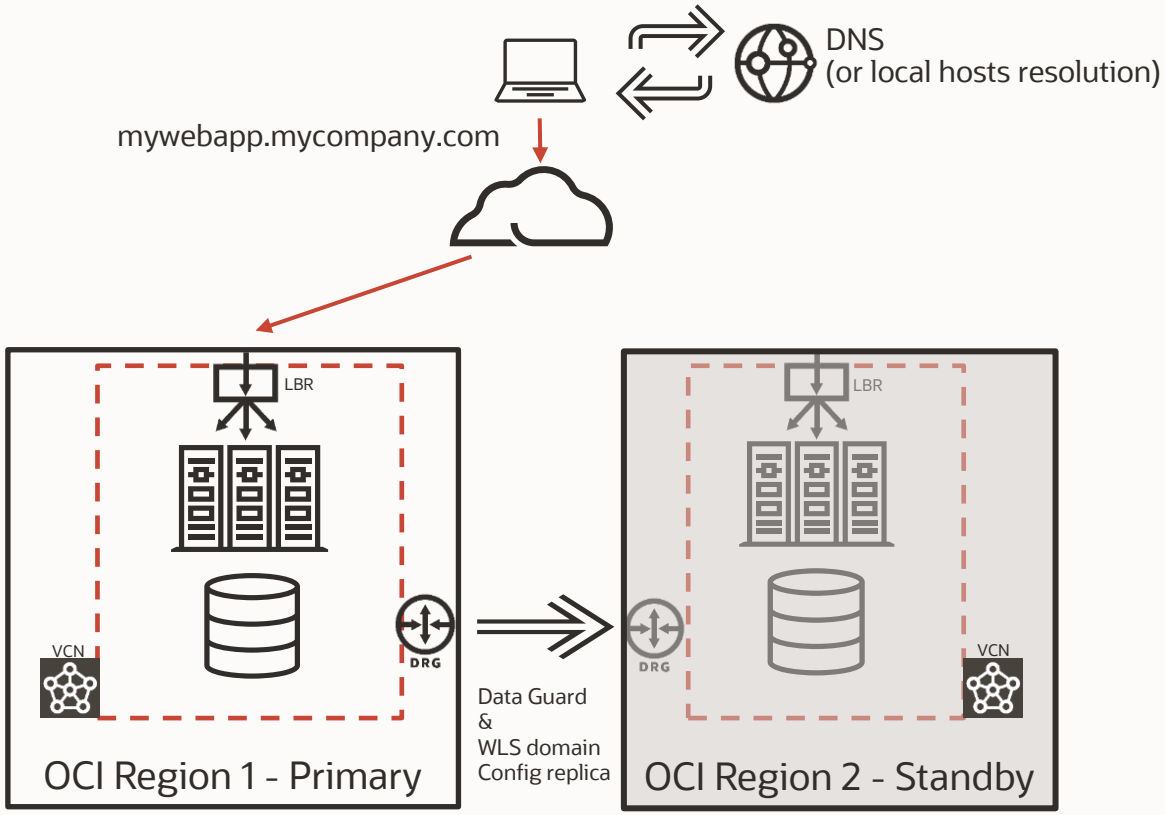
- **Active-Passive** model
  - Primary WebLogic for OCI & DB system in one region
  - Standby WebLogic for OCI & DB system in a different region (cross-AD deployment is NOT considered DR protection)
- DB systems configured with **Data Guard**
- **Standby WLS domain is a replica** of the primary domain (same name, schemas, passwords, etc.). Two options for the WLS config replica:
  - DBFS based method
  - FSS with RSYNC method
- **Unique frontend hostname** to access to the system. Is a “virtual name” that points to the IP of the LBR of the site with primary role
- Network communication between primary and secondary networks via **Dynamic Routing Gateway** (recommended)



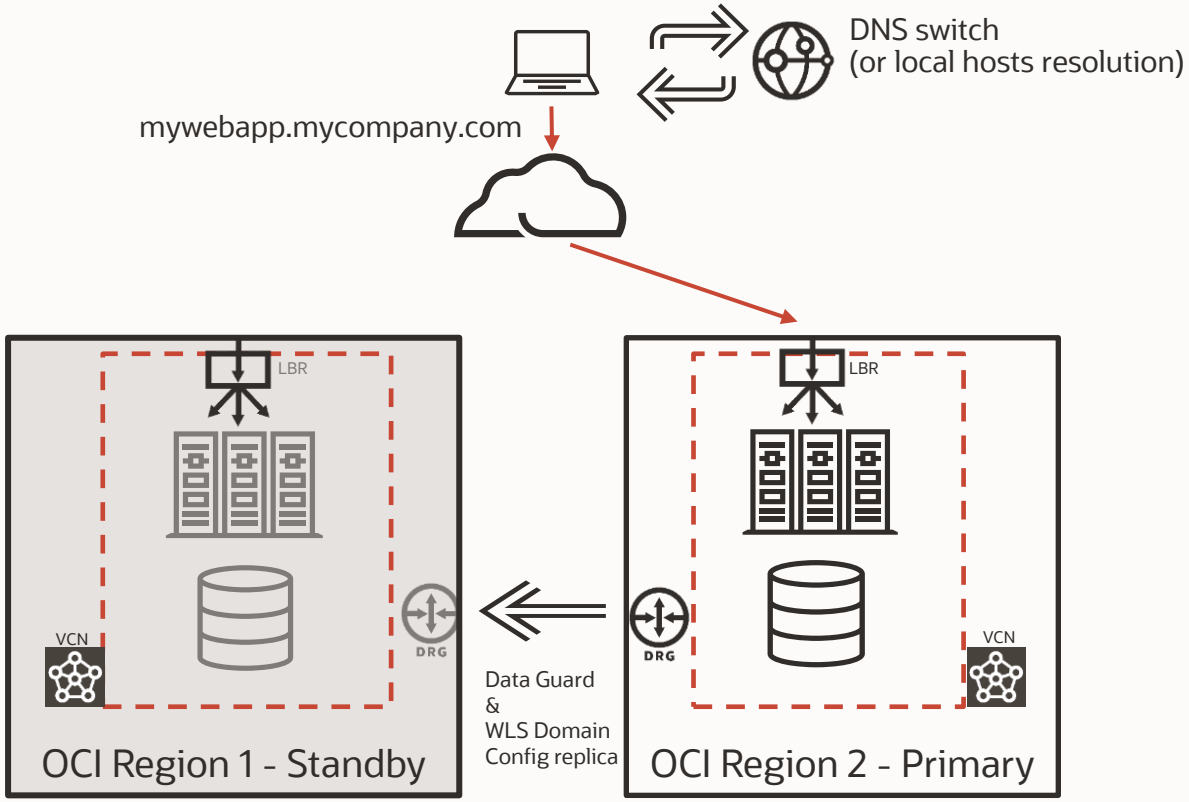
WebLogic for OCI Disaster Recovery

DR Topology - Overview

Normal Operation



After a Switchover



WebLogic for OCI Disaster Recovery

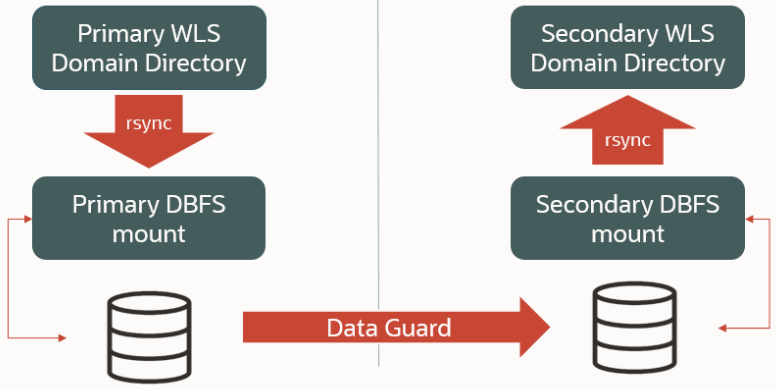
DR Topology – DBFS method vs FSS with RSYNC method to replicate WLS domain config

DBFS based method

- Database File System (DBFS) mount as staging file system for **a copy of the WLS domain**.
- Uses underlying Data Guard replica to copy the domain to standby region.
- Recommended for any latency (high or low)

✓ Takes advantage of the robustness of the DG replica
More resilient behavior through Oracle Driver's retry logic

✗ More complex to configure (db client required) and maintain

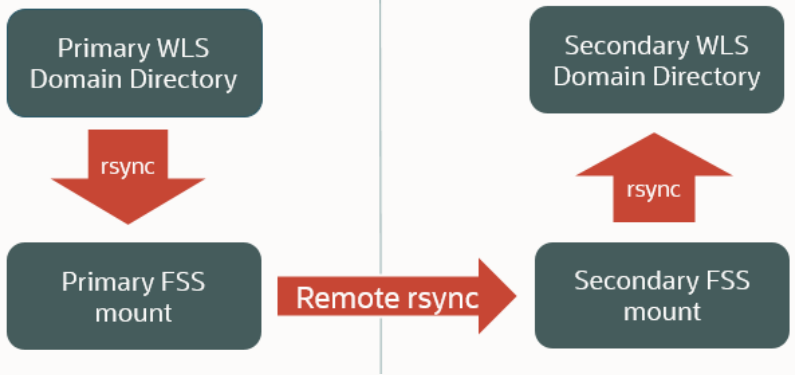


FSS with RSYNC

- File Storage Service (FSS) as staging file system **for a copy of the WLS domain**.
- Uses rsync to copy the domain to standby region.
- Recommended when latency is low

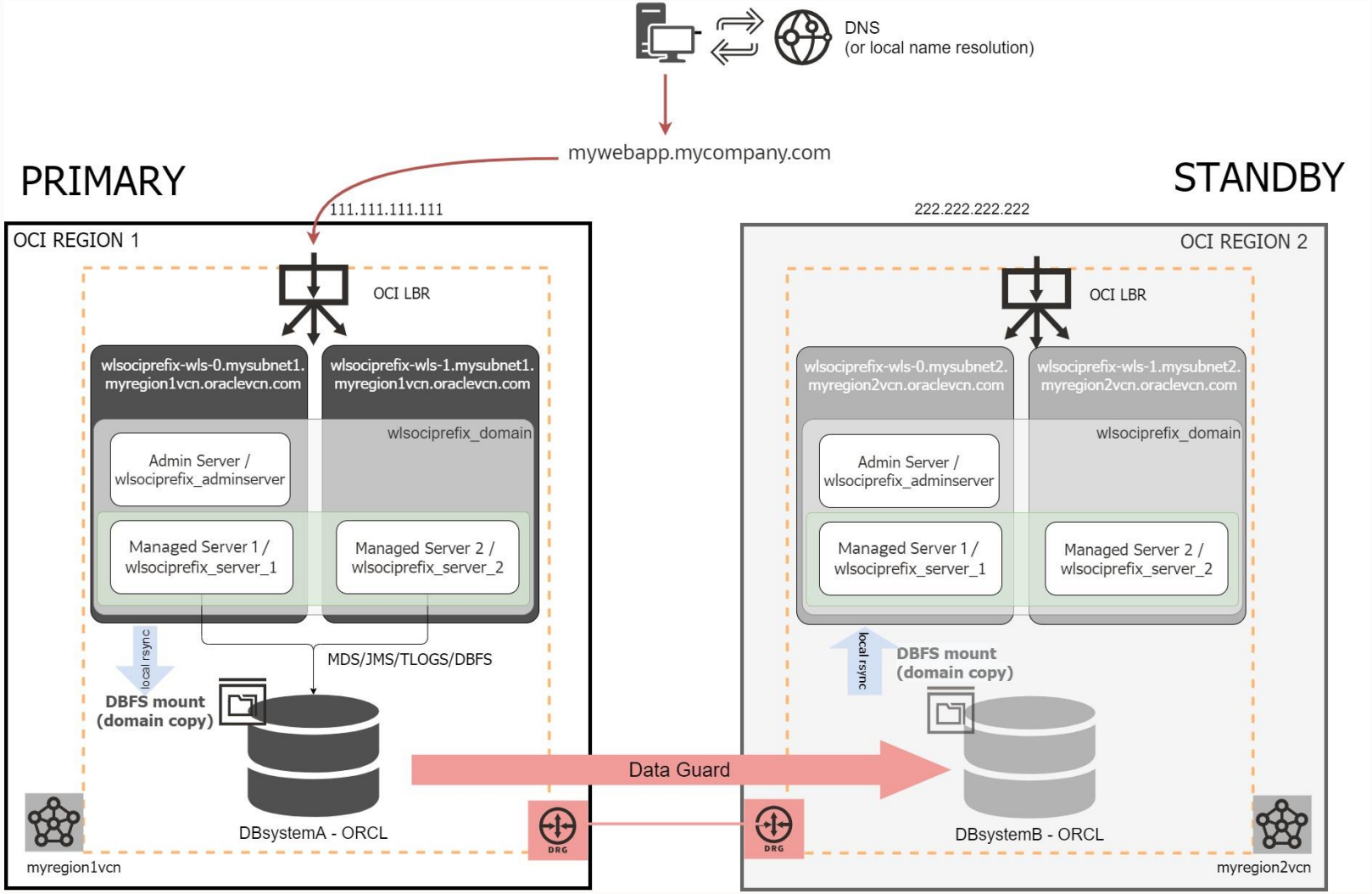
✓ Easy to configure and maintain

✗ More sensible to latency and jitter



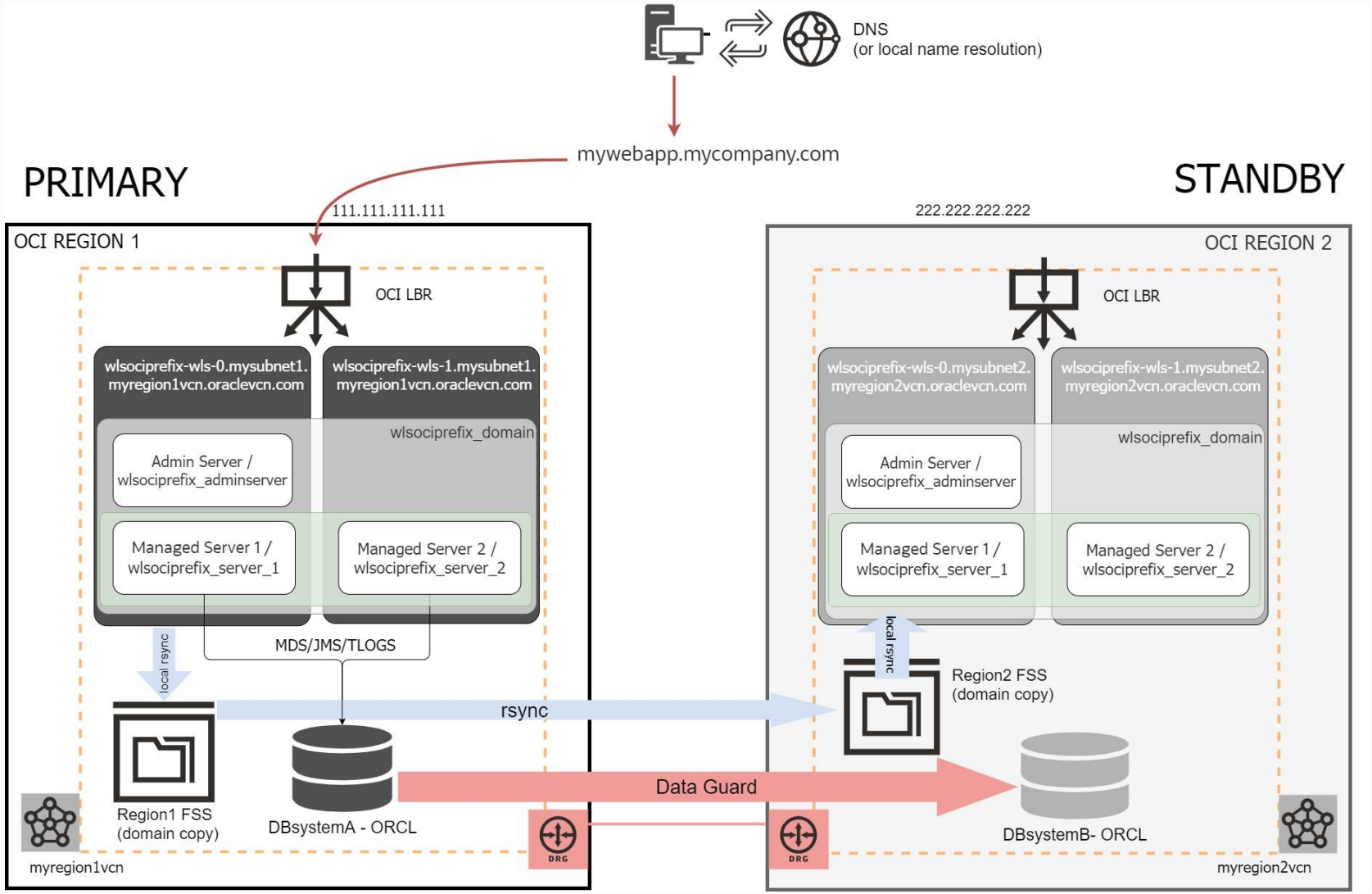
WebLogic for OCI Disaster Recovery

DR Topology – Detailed (DBFS based method)



WebLogic for OCI Disaster Recovery

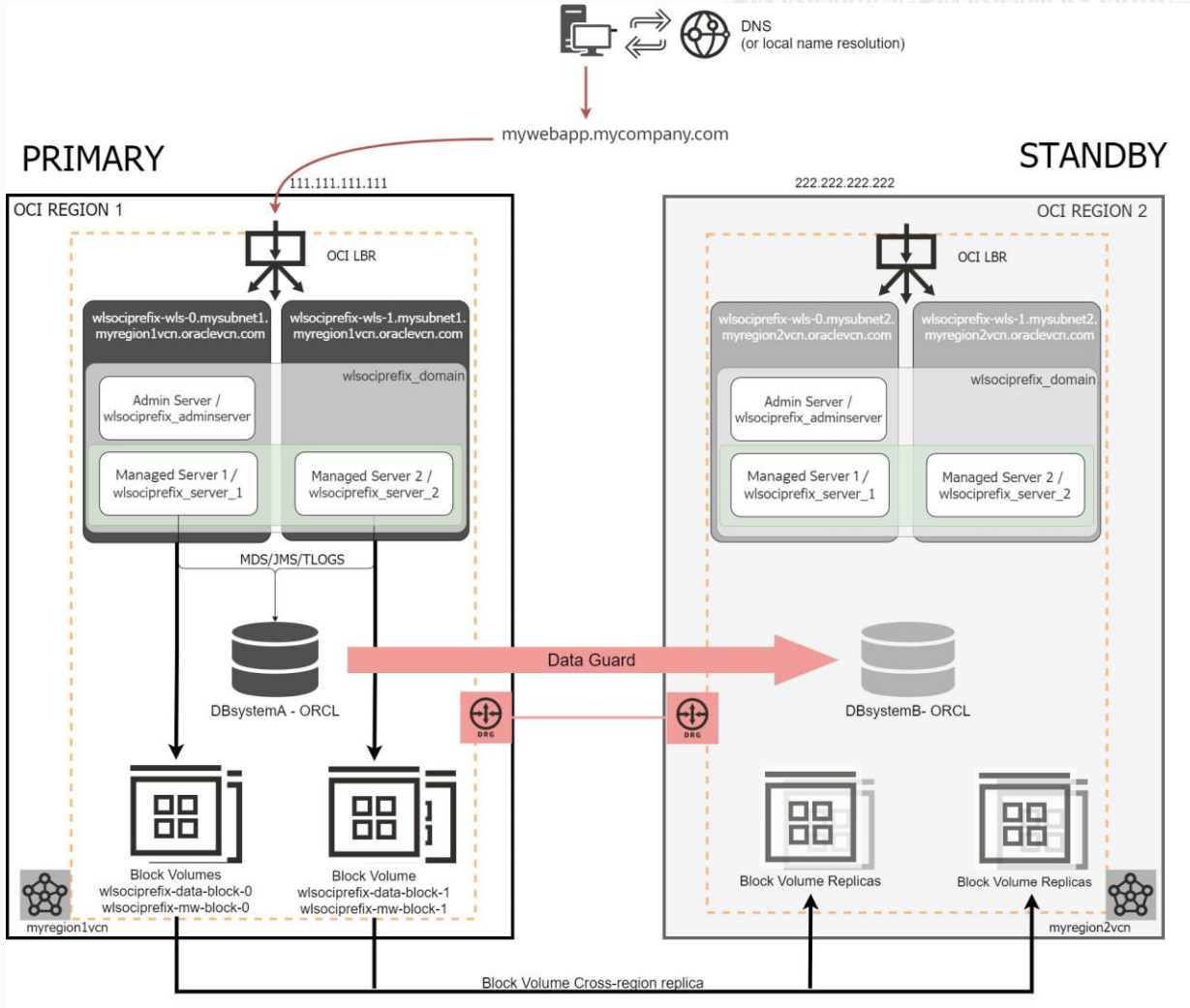
DR Topology – Detailed (FSS with rsync method)



WebLogic for OCI Disaster Recovery

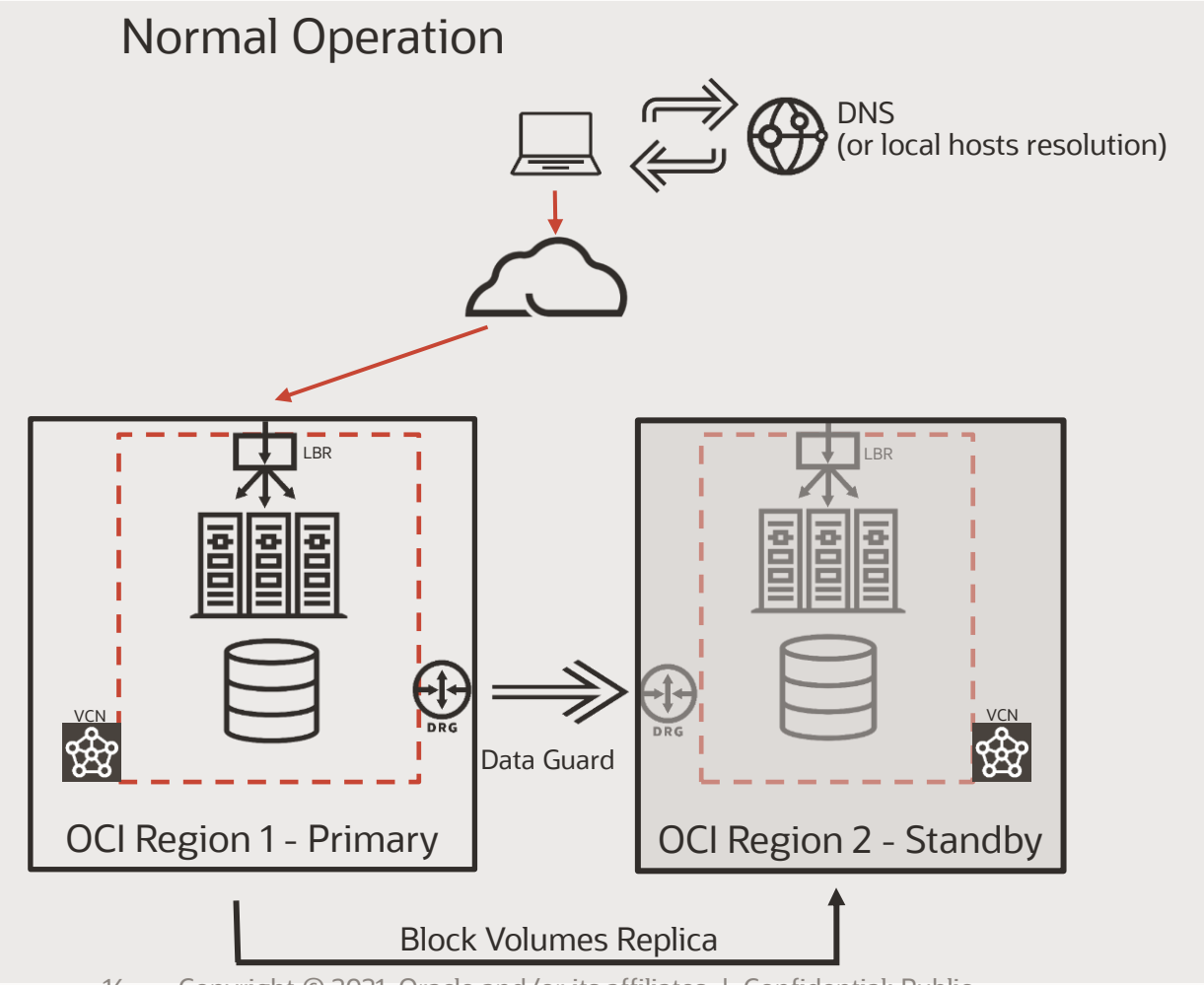
DR Topology – Block Volume cross-region replica (NEW since July 2021!)

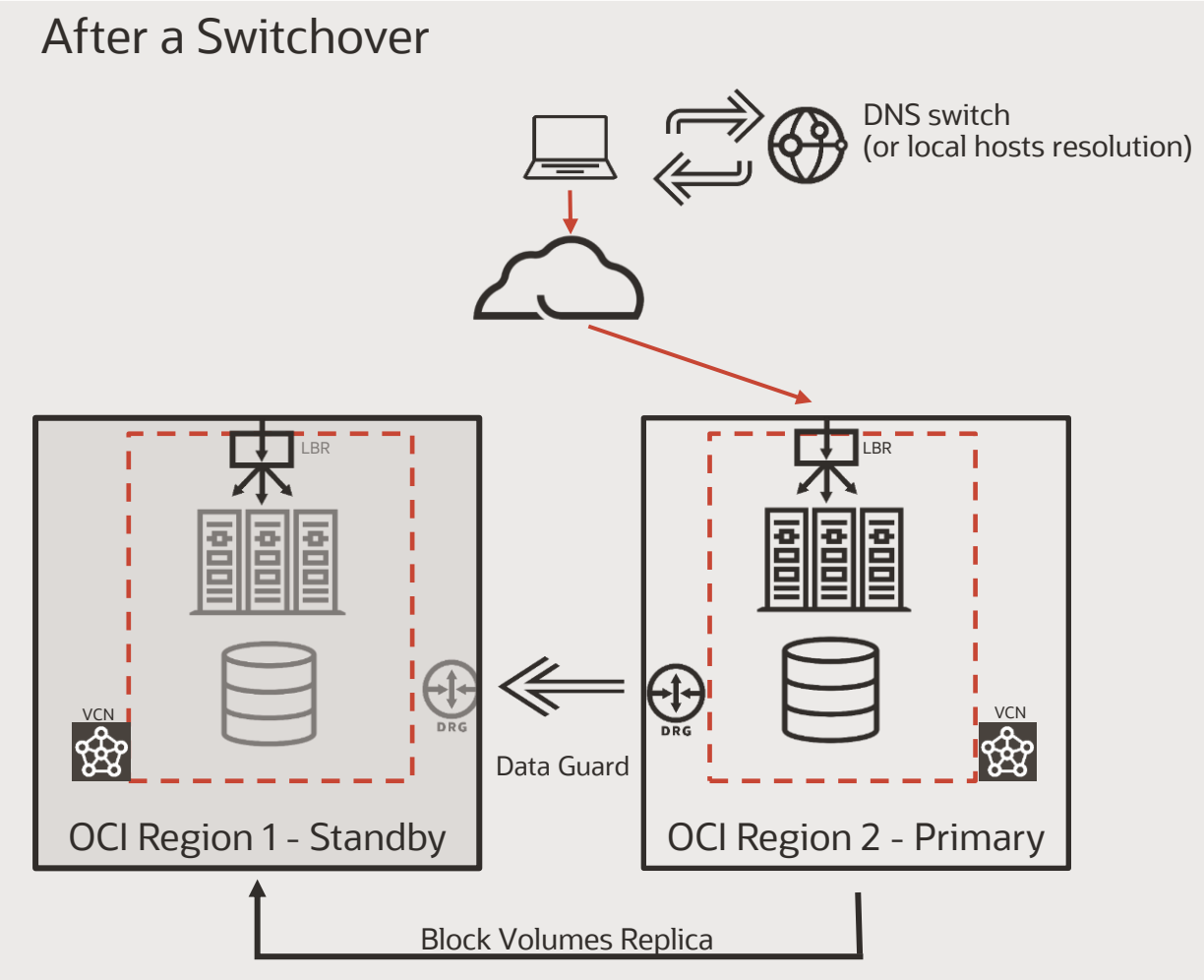
- The **Block Volume** containing the WLS Domain **is replicated** using Cross-Region Block Volume Replication feature (automatic asynchronous replication to other region)
- No stage location is used, hence, the **setup** and ongoing **replication differs significantly** from the DBFS and FSS-rsync approaches.
- Considerations of this model:
  - General-purpose solution applicable to other systems
  - Uses a continuous and unattended replica process
  - Replication is not limited to the domain configuration
  - Management complexity, more complicated as the number of block volumes replicated increases.
  - Switchover RTO is same as in other approaches.
  - For Failover RTO, additional steps required increment the downtime.
- More details in the Appendix E of the whitepaper



WebLogic for OCI Disaster Recovery

DR Topology – Block Volume cross-region replica (NEW since July 2021!)



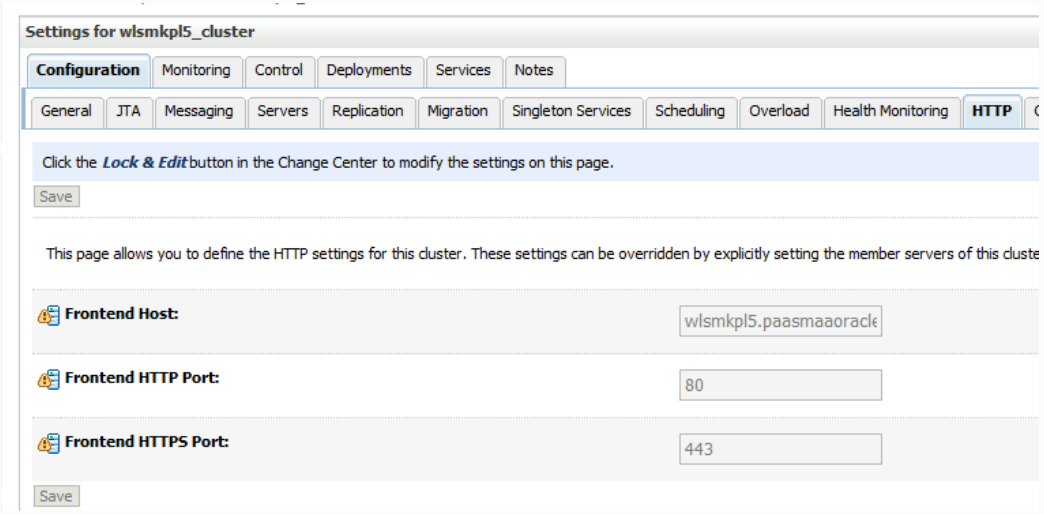


WebLogic for OCI Disaster Recovery

WebLogic for OCI Disaster Recovery

DR Setup

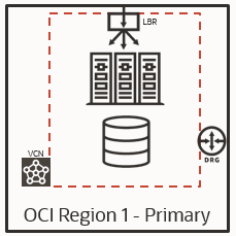
- Starting point is **the primary** WLS for OCI **system already exists** (along with its LBR and DB system)
- The impact of the DR setup on the existing system **minimal**:
  - Down time needed (a restart of the managed servers) only in case the frontend name was not already configured/used frontend is not going to be re-used for DR

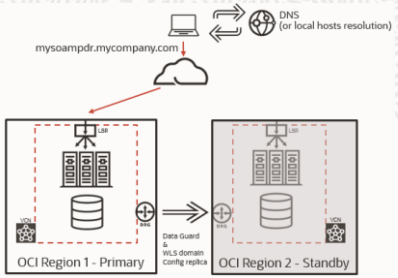


- DR setup process has been designed to be **idempotent**: each step can be retried.

WebLogic for OCI Disaster Recovery

DR Setup





Primary WLS for OCI exists

DR setup complete

1. Prepare primary mid-tier (frontend and TNS alias)

2. Setup secondary database

3. Provision secondary WLS for OCI

4. Prepare secondary mid-tier (frontend, TNS, and aliases

5. Configure the staging mounts for WLS config replication (DBFS or FSS)

6. Run DR setup scripts

WebLogic for OCI Disaster Recovery

DR Setup - Details on the step 2

1. Prepare virtual frontend name in DNS and in primary mid-tier
2. Setup secondary database
3. Provision secondary WLS for OCI
4. Prepare secondary mid-tier virtual frontend and aliases
5. Configure the staging mounts for WLS config replication (DBFS or FSS)
6. Run DR setup scripts

- Since March 2020, **OCI console** allows **to configure Data Guard cross-region** (before, only cross-ad was supported)
- Some requirements: same tenancy, same compartment, communication between Dynamic Routing Gateway

RECOMENDED

Option 1) Configuring using OCI Console (“auto DG”)

- For scenarios where *Option 1)* does not apply, it can be done manually.
- First, provision standby database as a regular DB System (same version, shape, password, etc. than primary)
- Second, use scripts provided in the whitepaper to configure it as standby (rman duplicate, dgmgrl commands, etc.)
  - *dataguardit_primary*.sh and *dataguardit_standby_root*.sh

Option 2) Configuring data guard manually (“manual DG”)

The secondary database is created as a Data Guard physical standby of the primary database. Two ways to do this.

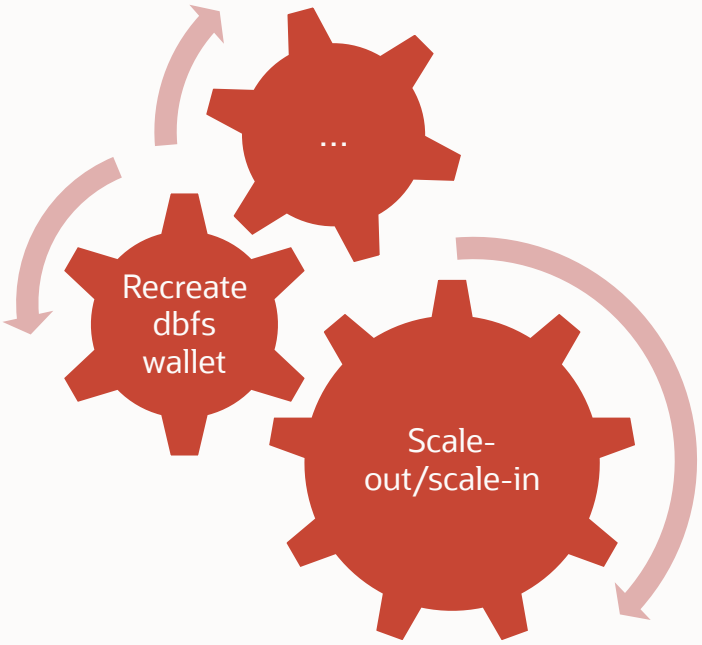
WebLogic for OCI Disaster Recovery

WebLogic for OCI Disaster Recovery

Main DR lifecycle operations

- WLS Domain config replication
- Switchover & Failover
- Open secondary site for validation

Other lifecycle operations



WebLogic for OCI Disaster Recovery

Main DR Lifecycle operations - WLS Domain config Replication

OPTION 1)

WHEN DOMAIN CHANGES ARE **INFREQUENT**

- Apply the configuration **manually twice**

	STEP
1	Apply the configuration change normally in the primary site
2	Convert the standby database to a snapshot standby
3	Start (if it wasn't started) the WebLogic Administration Server on the secondary site
4	Repeat the configuration change in the secondary site
5	Revert the database to physical standby

OPTION 2)

WHEN DOMAIN CHANGES ARE **FREQUENT**

- WLS Config can be automatically replicated from primary to standby,
  - using the DBFS approach
  - or the FSS with Rsync approach.
- Script **config_replica.sh** provided to automate this replication.
  - Run the script in **primary WLS Administration host**
  - Then run script in **secondary WLS Administration host**

WebLogic for OCI Disaster Recovery

Main DR Lifecycle operations - Switchover procedure

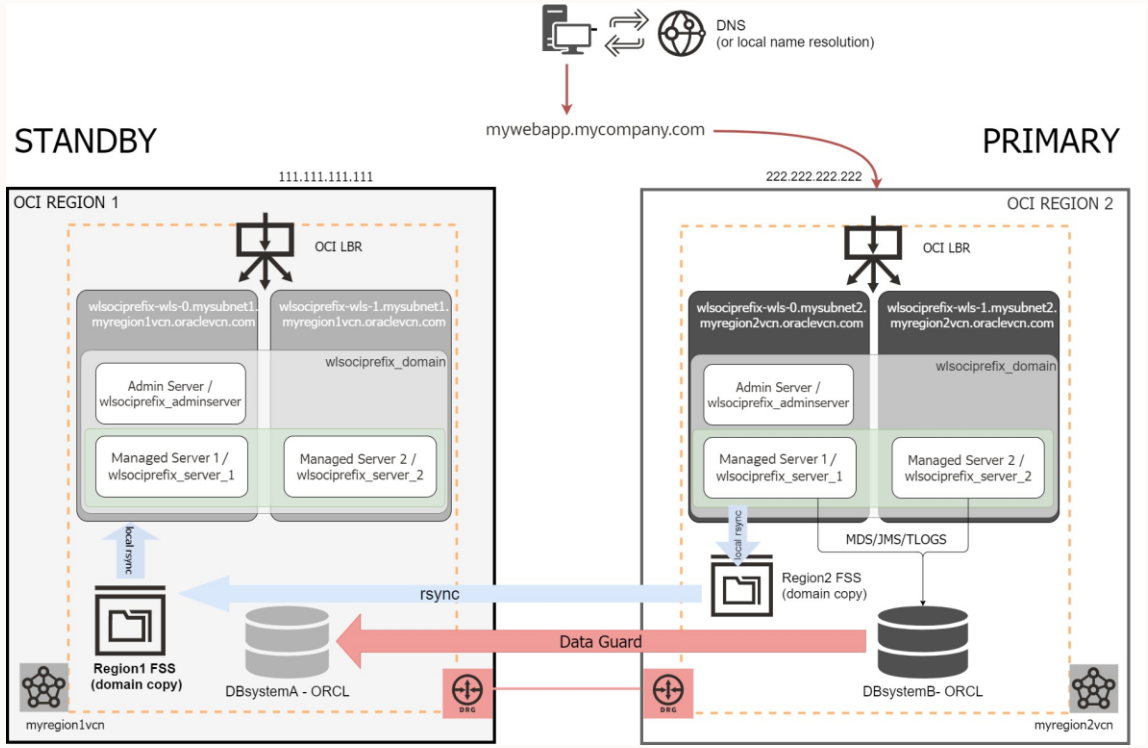
A switchover is planned operation where an administrator reverts the roles of the two sites.

SWITCHOVER Steps:

1) Propagate any pending WLS config changes
2) Stop WLS servers in primary Site
3) Switchover frontend name in DNS
4) Switchover Database
5) Start WLS servers in secondary Site

15-30 min*

AFTER switchover:

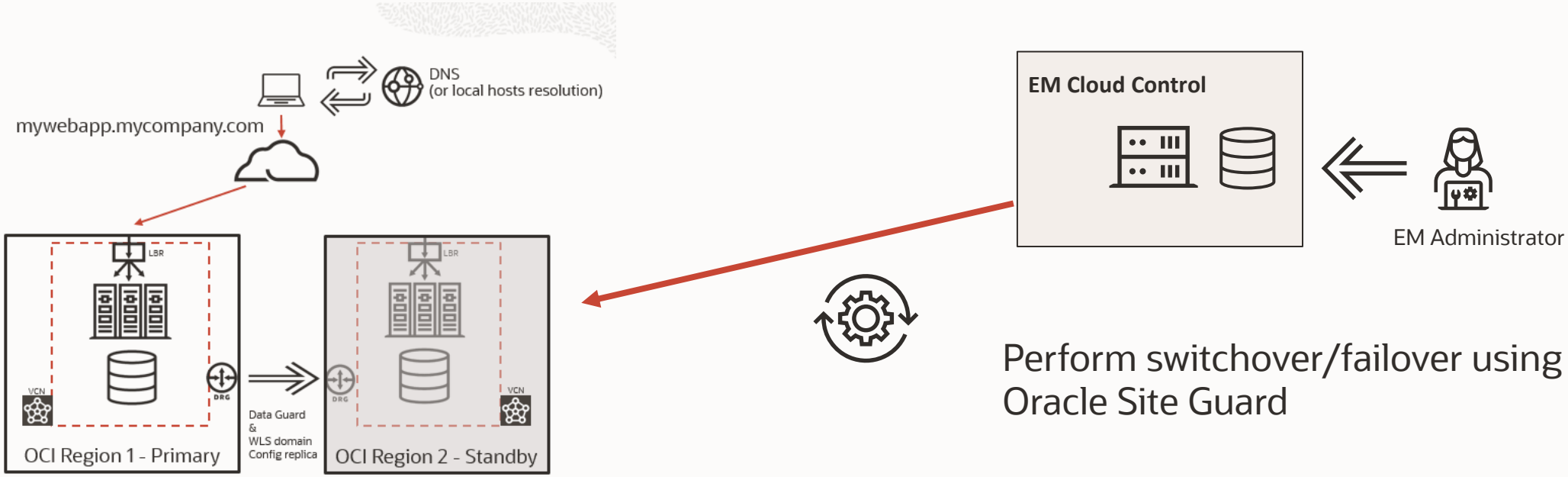


WebLogic for OCI Disaster Recovery

Main DR Lifecycle operations - Using Oracle Site Guard

Full stack switchover and failover can be orchestrated by **Oracle Site Guard.**

Required setup documented in separated [whitepaper](#) (common for WLS, SOAMP and SOACS DR)



WebLogic for OCI Disaster Recovery

References

Public Documents

- Oracle WebLogic Server for Oracle Cloud Infrastructure Disaster Recovery
https://www.oracle.com/a/otn/docs/middleware/maa-wls-mp-dr.pdf

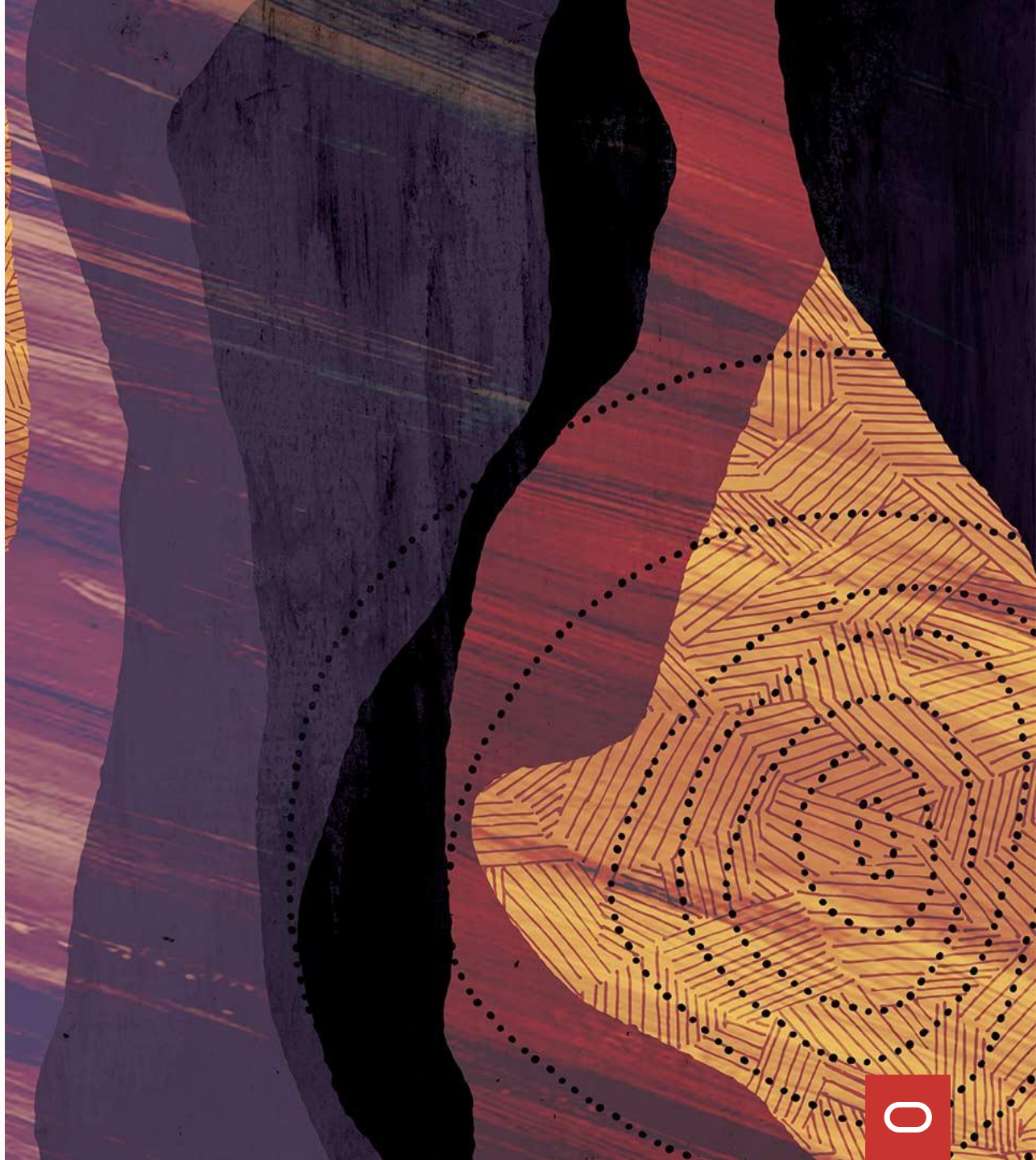
- Configure Oracle Fusion Middleware DR on Oracle Cloud with an autonomous database
https://docs.oracle.com/en/solutions/adb-refreshable-clones-dr

- MAA Best Practices for the Oracle Cloud - OTN page
https://www.oracle.com/database/technologies/high-availability/oracle-cloud-maa.html

- MAA Best Practices for Oracle Fusion Middleware - OTN page
https://www.oracle.com/database/technologies/high-availability/fusion-middleware-maa.html

- The WebLogic Server Blog
https://blogs.oracle.com/weblogicserver/disaster-recovery-in-oracle-weblogic-server-for-oracle-cloud-infrastructure

Thank you



ORACLE