SOA Suite on Oracle Cloud Infrastructure Marketplace

Disaster Recovery

Production and Disaster Recovery in the Oracle Cloud Infrastructure (OCI)

July, 2022  |  Version 16
Copyright © 2022, Oracle and/or its affiliates
Confidential - Public
PURPOSE STATEMENT

This document provides a description, a summary of requirements, and the setup procedure to configure a Disaster Recovery solution for Oracle SOA Suite Cloud on Marketplace. This document is oriented to a technical audience having knowledge of Oracle Cloud, Oracle SOA Suite, Oracle WebLogic, Oracle Database, Data Guard and Oracle Database backup and recovery.

REVISION HISTORY

The following revisions have been made to this white paper:

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>June 2020</td>
<td>1</td>
<td>Initial publication</td>
</tr>
<tr>
<td>September 2020</td>
<td>2</td>
<td>Added Best Practices point</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added Appendix C for additional Lifecycle Operations (scale-out, open stby for validation)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added support for RAC for DRS tool and manual DG configuration for RAC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Appendix A, added note about restoration in automated Data Guard</td>
</tr>
<tr>
<td>December 2020</td>
<td>3</td>
<td>Updated table in “Provisioning Secondary SOA Suite on Marketplace”</td>
</tr>
<tr>
<td>January 2021</td>
<td>4</td>
<td>Added note to “Open secondary Site for validation”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Corrected typos and improved some wordings.</td>
</tr>
<tr>
<td>March 2021</td>
<td>5</td>
<td>Added “Recreating the standby DB System” in “Appendix C – Additional Lifecycle Operations”</td>
</tr>
<tr>
<td>April 2021</td>
<td>6</td>
<td>Enhancement to include an additional DR method for WLS domain configuration replication using OCI FSS with rsync. Several sections have been updated.</td>
</tr>
<tr>
<td>July 2021</td>
<td>7</td>
<td>Added support to the “MFT Cluster” service type too.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Enhancement to include an additional DR method for WLS domain configuration replication using Block Volume Cross-region Replication. Added the “Appendix D – Disaster Recovery Based on Block Volume Cross-Region Replication”.</td>
</tr>
<tr>
<td>August 2021</td>
<td>8</td>
<td>Added the DRS package version that can be run from an OEL8 box</td>
</tr>
<tr>
<td>September 2021</td>
<td>9</td>
<td>Added footnote in page 5.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added OCI DNS switchover example in switchover operations.</td>
</tr>
<tr>
<td>October 2021</td>
<td>10</td>
<td>Updated diagrams. Additional info in Assumptions &gt; Database</td>
</tr>
<tr>
<td>January 2022</td>
<td>11</td>
<td>Added point “RTO and RPO Overview”</td>
</tr>
<tr>
<td>February 2022</td>
<td>12</td>
<td>Updated Data Guard manual setup scripts. Added note for custom resource names.</td>
</tr>
<tr>
<td>April 2022</td>
<td>13</td>
<td>Added specific point “Considerations for EXACS”</td>
</tr>
<tr>
<td>June 2022</td>
<td>14</td>
<td>Added point “Patching” to “Appendix C – Additional Lifecycle Operations”</td>
</tr>
<tr>
<td>June 2022</td>
<td>15</td>
<td>Updated links Removed references to Oracle Site Guard due to Site Guard deprecation (see Doc ID 2875372.1)</td>
</tr>
<tr>
<td>July 2022</td>
<td>16</td>
<td>Added DNS views approach for the required host aliases</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated links</td>
</tr>
</tbody>
</table>
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purpose Statement</td>
<td>1</td>
</tr>
<tr>
<td>Revision History</td>
<td>1</td>
</tr>
<tr>
<td>Introduction</td>
<td>3</td>
</tr>
<tr>
<td>SOA Suite on Marketplace Disaster Recovery</td>
<td>5</td>
</tr>
<tr>
<td>Topology Description</td>
<td>5</td>
</tr>
<tr>
<td>Assumptions</td>
<td>9</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>9</td>
</tr>
<tr>
<td>Database</td>
<td>9</td>
</tr>
<tr>
<td>Requirements</td>
<td>10</td>
</tr>
<tr>
<td>Front-end address</td>
<td>10</td>
</tr>
<tr>
<td>Instance Name Prefix</td>
<td>10</td>
</tr>
<tr>
<td>Network communication between sites</td>
<td>10</td>
</tr>
<tr>
<td>Staging filesystems for the WebLogic domain config replication</td>
<td>11</td>
</tr>
<tr>
<td>Custom files</td>
<td>11</td>
</tr>
<tr>
<td>SLA requirements</td>
<td>11</td>
</tr>
<tr>
<td>SOA Suite on Marketplace Disaster Recovery Setup</td>
<td>13</td>
</tr>
<tr>
<td>1. Choose a virtual front-end name</td>
<td>14</td>
</tr>
<tr>
<td>2. Prepare primary mid-tier for the virtual front-end</td>
<td>14</td>
</tr>
<tr>
<td>3. Setup the Database in Secondary Site</td>
<td>16</td>
</tr>
<tr>
<td>5. Prepare Secondary mid-tier for the virtual front-end</td>
<td>23</td>
</tr>
<tr>
<td>6. Create the mid-tier hosts aliases</td>
<td>23</td>
</tr>
<tr>
<td>7. Configure the staging mounts for WebLogic domain config replication</td>
<td>24</td>
</tr>
<tr>
<td>8. Run the Disaster Recovery Setup utils (DRS)</td>
<td>27</td>
</tr>
<tr>
<td>SOA Suite on Marketplace Disaster Recovery Lifecycle Procedures</td>
<td>29</td>
</tr>
<tr>
<td>Configuration Replication</td>
<td>29</td>
</tr>
<tr>
<td>Switchover</td>
<td>36</td>
</tr>
<tr>
<td>Failover</td>
<td>38</td>
</tr>
<tr>
<td>RTO and RPO Overview</td>
<td>39</td>
</tr>
<tr>
<td>Expected RTO</td>
<td>39</td>
</tr>
<tr>
<td>Expected RPO</td>
<td>40</td>
</tr>
<tr>
<td>Best Practices</td>
<td>42</td>
</tr>
<tr>
<td>Conclusion</td>
<td>42</td>
</tr>
<tr>
<td>Appendix A – DB System Backups on manually configured Data Guard</td>
<td>43</td>
</tr>
<tr>
<td>Appendix B – Summary of networking requirements for DR Setup</td>
<td>44</td>
</tr>
<tr>
<td>Appendix C – Additional Lifecycle Operations</td>
<td>46</td>
</tr>
<tr>
<td>Appendix D – Disaster Recovery Based On Block Volume Cross-Region Replication</td>
<td>56</td>
</tr>
</tbody>
</table>
INTRODUCTION

Oracle’s Maximum Availability Architecture (Oracle MAA) is the best practices blueprint for data protection and availability of Oracle products (Database, Fusion Middleware, Applications) deployed on on-premises, private, public or hybrid clouds. Implementing Oracle Maximum Availability Architecture best practices is one of the key requirements for any Oracle deployment. Oracle Fusion Middleware and Oracle Databases include an extensive set of high availability features which can protect application deployments from unplanned downtime and minimize planned downtime. These features include: process death detection and restart, clustering, server migration, clusterware integration, GridLink datasources, load balancing, failover, backup and recovery, rolling upgrades, and rolling configuration changes.

Oracle SOA Suite on Marketplace (SOAMP) provides a Platform as a Service (PaaS) computing platform solution for running the SOA applications in the cloud (Oracle SOA Suite, Oracle Service Bus, Oracle B2B, Oracle Managed File Transfer, etc.). Oracle SOA Suite on Marketplace is a new PaaS solution that relies completely on Oracle Cloud Infrastructure, it is unique and different from Oracle SOA Cloud Service. It is provisioned using the OCI Console Marketplace and is fully integrated with other OCI components (like OCI Load Balancer) and OCI infrastructure life cycle procedures (like backup and recovery). It uses Oracle Compute Infrastructure, Oracle Cloud Infrastructure Database, and Oracle WebLogic as its basic infrastructure. SOAMP requires an Oracle Database to store Oracle Platform Security Services information, instance tracking, composite and document metadata and other Oracle FMW Infrastructure schemas. In a typical Oracle SOA deployment the application data (such as application-specific schemas, jms stores etc.) and the SOA-specific schemas are stored in the same database for transactional consistency and simplified administration reasons. In a SOA Suite on Marketplace instance an Oracle Cloud Infrastructure Database instance is used to store these schemas.

All Oracle SOA deployments need protection from unforeseen disasters and natural calamities, including within Oracle Cloud Infrastructure. This disaster recovery protection needs to address the middle tier (Oracle SOA Suite on Marketplace), the data tier (Oracle Cloud Infrastructure Database) and LBR tier (OCI LBR or 3rd-party). The solution involves setting up a standby system at a geographically different Oracle cloud data center than the primary production site. Although the standby system may have equal or fewer services and resources compared to the production site, Oracle recommends to run a mirror configuration with the same capacity. The standby system is normally in a passive mode and is activated when the primary site becomes unavailable. This deployment model is sometimes referred to as an active-passive model.

This whitepaper has been particularly created to address Disaster Recovery (DR) for Oracle SOA Suite on Marketplace. The overall topology and setup procedure is very similar to the SOA Cloud Service DR1, but the steps and scripts have been updated to address SOA Suite on Marketplace specifics.

---

1 SOA Cloud Service Disaster Recovery on OCI - Production and DR in the Cloud
Oracle SOA Suite on Marketplace (SOA MP) can satisfy the most demanding **Recovery Time Objective (RTO)** and **Recovery Point Objective (RPO)** by utilizing high availability and disaster protection capabilities provided by Oracle Fusion Middleware and Oracle Database. 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. This Oracle MAA blueprint details Oracle MAA Best Practices and provides a procedural overview for deploying DR for SOA Suite on Marketplace. Oracle SOA on Marketplace Service Disaster Recovery solution is achieved by replicating a limited set of configuration files that are required to bootstrap SOA components. The application may require additional configuration files to be replicated. Options are provided in this paper to suit different application paradigms. Disaster protection for Oracle Cloud Infrastructure Database used by Oracle SOA is provided through Oracle Data Guard.

This document applies to **“SOA with SB & B2B Cluster”** and **“MFT Cluster”** service types of the Oracle SOA Suite on Marketplace.

This document is intended for a technical audience that has knowledge of **Oracle Weblogic Server, Oracle FMW SOA, Oracle Database, Data Guard, Oracle Database backup and recovery**, and a basic understanding of services offered on the **Oracle Cloud**[^1].

[^1]: [https://cloud.oracle.com/home](https://cloud.oracle.com/home)
SOA SUITE ON MARKETPLACE DISASTER RECOVERY

Topology Description

The Disaster Recovery solution for Oracle SOA Suite on Oracle Cloud Marketplace is an **active-passive model**. There is a **primary system** that consists on a SOA Suite on Marketplace deployment, load balancer, and Oracle Cloud Infrastructure DB system in one region, and a **standby system**, that consist in a SOA Suite WLS domain, load balancer, and Oracle Cloud Infrastructure DB system in a different region. This same topology may be implemented in the scope of a single region with multiple availability domains, although Oracle recommends the use of different regions for maximum Disaster recovery protection.

The terms “region”, “data center” or “site” are used in this document indistinctly to refer to an Oracle OCI region. “Region”, “data center” or “sites” are physical location entities that are (far enough) geographically separated not be affected by the same disaster event. For example: Ashburn and Phoenix are two differnet data centers, sites or regions in context of this paper.

The primary and standby Oracle Cloud Infrastructure DB Systems are configured with Data Guard. By relying on Data Guard features, all the changes applied to primary database are replicated to secondary database (which acts as the “standby” database).

The standby SOA Suite on Marketplace domain is a replica of the primary domain, that uses the same name, schemas, passwords, etc. but points to the secondary database. The listener addresses of the WebLogic Servers are configured with the primary midtier host names, so in secondary midtier hosts the pertinent aliases are created in the hosts file to resolve them with the secondary IPs. This document provides the steps to create and configure this standby system.

On the front-end, there is a unique name configured to access the applications that run in the system. This “virtual” front-end name will point to the IP of the OCI Load Balancer of the primary site. In case of a switchover, this front-end name is updated to point to the IP of the OCI Load Balancer of the secondary site. It always must point to the LBR IP of the site that has the primary role in each time.

In normal business operation, the standby database is a physical standby. It is either in the mount state, or opened in read-only mode when Active Data Guard is used. The standby database receives and applies redo from primary, but cannot be

---

3 SOA Marketplace provides out-of-the-box protection for the middle tier against Availability Domain’s failures in regions with more than one Availability Domain. This protection requires that SOAMP be deployed on a regional subnet. When deploying in this type of network, the nodes in an Oracle SOA Suite on Marketplace cluster are distributed evenly across all available availability domains. Notice, however, that Oracle’s Database does not support placing instances of the same RAC cluster in different availability domains, so when cross-region DR protection is not used, it is still necessary to use Oracle Data Guard to protect the Database tier against failures at the Availability Domain level.
opened in read-write mode. For some actions, in the DR setup and lifecycle steps described in this document, the standby database will be converted from a physical standby to a snapshot standby. A database in **snapshot standby** mode is fully updateable database. A snapshot standby database receives and archives, but does not apply, the redo data from a primary database. All the changes performed to a snapshot standby are discarded when it is converted again into a physical standby.

All the information that resides in the database is automatically replicated to the secondary site by Data Guard. This includes: SOA schemas, OPSS information, custom schemas, TLOGs, JDBC persistent stores, etc. The WebLogic Domain configuration, located on the local filesystem in each site, needs to be replicated from the primary site to the secondary as well. This replication is required and performed during the initial DR setup, and also necessary during the system’s operations lifecycle, whenever configuration change maintenance is performed in the primary domain. Oracle supports three different methods to perform this WebLogic domain configuration replication: **DBFS (Oracle Database File System)**, **OCI File Storage Service (FSS) with rsync**, and **Block Volume Cross-Region Replication**. All the methods use the same topology and similar mechanics. The difference is how the information is transferred from the primary site to the standby.

- In the **DBFS-based method**, a copy of the domain configuration is staged to a DBFS filesystem mount point and replicated to the secondary site via Data Guard. A DBFS mount is a filesystem that resides in the database and that can be mounted like an NFS volume. The primary domain configuration is copied to that DBFS mount, and then, it is automatically replicated to the standby via the underlying Data Guard functionality. In the secondary site, the midtier hosts can mount the same DBFS mount point from the standby database. The replicated domain configuration data is now available and copied from the DBFS mount to the secondary domain. This paper provides a script that automatizes this process in primary and in standby.

This method takes advantage of the **robustness of the Data Guard replica**. It can be used in any scenario, and it is strongly recommended for DR scenarios that have medium or high latencies between the regions.

![Figure 2 SOA in Marketplace DR topology, that uses DBFS method for WLS Domain config replication](image)
In the **OCI File Storage Service (FSS) with rsync method**, the domain configuration is transferred to the secondary site by using direct rsync between two file systems, one in primary and another in secondary. This approach, like the DBFS method, uses a shared filesystem as an intermediate "staging" point. For this, an FSS mount is used in each region. To replicate the primary domain config, the WLS domain is copied first to the local staging FSS mount, and then, via rsync, to the remote FSS mount. Then, in secondary, the domain configuration is copied from the FSS in the secondary environment's data center to the secondary domain directory. This paper provides a script that automatizes this process in primary and in standby. This method is recommended only when the latency between the data centers is low and the connection is reliable, as it is the case between Oracle OCI regions that communicate with Dynamic Routing Gateways.

Figure 4 SOA Suite in Marketplace DR topology, that uses OCI FSS with rsync method for WLS Domain config replication. The blue arrows just represent the logical flow of the configuration copy. The rsync commands run either in primary or standby site’s WebLogic Administration hosts. I.e., for the remote copy, primary site’s WebLogic Administration host connects to standby WebLogic Administration host with rsync.
The **DBFS method** delivers better availability through Oracle Driver’s retry logic and provides a more resilient behavior than FSS with rsync. It can be used in any scenario, and it is strongly recommended for DR scenarios that have medium or high latencies between the data centers. However, the use of DBFS for configuration replication has also additional implications from the setup, database storage and lifecycle perspectives.

The **FSS with rsync method** is easier to maintain and configure. However, it is recommended only in DR scenarios between Oracle OCI data centers using Dynamic Routing Gateways. Data centers that communicate over the public internet, for example, may not have sufficiently low latency for a reliable behavior of FSS with rsync. Notice also that the FSS with rsync method can incur in additional cost due to the FSS usage and to the connectivity requirements between primary and standby middle tiers (customer billing conditions are out-of-scope of this document, contact your Oracle license team in order to get details on this).

Regardless which method is used to replicate the domain configuration, note that if the standby database is in shutdown status during normal business operation, it will not receive updates from primary and it will become out-of-sync. This can result in a data loss in case a switchover needs to be performed, thus it is not recommended to have the standby database stopped during normal business operation. The standby midtier hosts could be stopped, however, the configuration changes that are replicated from the primary site will not be pushed to the secondary domain configuration if the standby site’s admin server host is stopped. In case of a switchover event, the RTO is increased if the standby midtier hosts need to be started and the domain synchronized with primary changes.

- **There is a third DR model based in Block Volume Cross-region replication.** In this approach, the entire Block Volume of the mid-tier hosts that contains the WebLogic Domain configuration is replicated to the secondary site using the OCI Cross-Region Volume Replication feature. This capability allows you to perform ongoing automatic asynchronous replication of block volumes and boot volumes to other regions. No stage location is used for configuration replication, hence the set up and ongoing replication process differs significantly from the dbfs and fss approaches. This model provides worse RTO, and the switchover operations are more complex than in DBFS and FSS with rsync methods. However, it provides a general-purpose solution applicable not only to middleware-based PaaS services but also to all the data that may reside in block volumes attached to a compute instance. It also provides a continued and automated replica. See the [Appendix D – Disaster Recovery Based On Block Volume Cross-Region Replication](#) for specific details on advantages and disadvantages, topology and setup of this model.

---

*In SOA Marketplace, the billing of the stopped compute instances follow the OCI compute model and depend on the compute shape. See [https://docs.cloud.oracle.com/en-us/iaas/Content/Compute/Tasks/restartinginstance.htm#resource-billing](https://docs.cloud.oracle.com/en-us/iaas/Content/Compute/Tasks/restartinginstance.htm#resource-billing)*
Assumptions

Load Balancer
The Disaster Recovery solution assumes that the SOA Suite in Oracle Cloud Marketplace is configured with an OCI Load Balancer. A load balancer is mandatory when the cluster has more than one server, so the incoming requests can be balanced between them.

The default OCI Load Balancer that is created during the SOA Suite Marketplace provisioning is regional in scope. If your region includes multiple availability domains, it creates a primary load balancer and a standby load balancer, both in the same region but each one in a different availability domain. If the primary load balancer fails, the public IP address switches to the standby load balancer that is in the same region. The service treats the two load balancers as equivalent and you cannot specify which one is “primary”. This way, the load balancer provides local (inside a region) high availability for the load balancer layer.

The same topology will exist in the secondary region: the OCI LBR for the standby site’s domain will have one primary load balancer in one of the availability domains of the standby site’s region and another load balancer in a second availability domain of the same region.

This configuration is sufficient for the disaster recovery configuration. No configuration replication is required between primary and standby site’s Load Balancers, as each needs to route only to its local WebLogic cluster. Only in the case when the default load balancer configuration is modified manually in the primary site should the same configuration modifications be made manually to the secondary site’s load balancer.

See documentation for OCI Load Balancing for additional details.

Database
Oracle SOA Suite on Marketplace requires a database to store Oracle Platform Security Services information, SOA instance tracking, composite and document metadata, and other Oracle FMW Infrastructure schemas. It is also an MAA best practice to use a database for any persistent information stored by the WebLogic Server domain, including JMS persistent stores and JTA logs. This best practice is included in the default out-of-the-box configuration for SOA Marketplace. This is especially valuable and critical in Disaster Recovery topologies, where this information becomes automatically available in the standby site in fail-over and switch-over cases thanks to the Data Guard replication.

The Disaster Recovery solution assumes that the Oracle SOA Suite on OCI Marketplace is configured with an Oracle Cloud Infrastructure DB System. This document precisely focuses and uses Database Systems VM on OCI for the examples and configuration provided. Note that only one standby database per primary database is supported in the DR topology provided by this document. This is consistent with the OCI Console, which limits the DG configuration to only one standby database for each primary database.5

The DR Setup procedure described in this paper is certified with single instance and Oracle RAC database DB Systems, and with Exadata Cloud Service (EXACS) database.

Oracle Autonomous Processing (ATP) is out of the scope of this document. Although ATP now supports cross-region Data Guard, it does not provide a number of features required for PaaS DR, (like snapshot standby conversion, dgbroker access and others) Oracle ATP cannot be used in SOA Marketplace disaster recovery topologies.

---

Requirements

Front-end address
The access from clients to the SOAMP system must be agnostic to the site that is being used as primary. To accomplish this, the front-end address name used to access the system must be unique and point always to the system that is the primary at the moment. This name is usually referred to as “virtual front-end” or “vanity url”.

It is possible to reuse the existing system’s front-end host name address (if such exists already) as the virtual front-end for disaster protection. For example, if the original system was using “soampdrs.mycompany.com” as the vanity url for primary, this same virtual hostname can be re-mapped to the second site’s load balancer IP after a switchover or failover.

Appropriate DNS services (Oracle Cloud DNS, other commercial DNS, local DNS, or local hosts resolution) need to be used for the front-end name to be mapped to either site. Later in this document it is explained how to configure the SOA WebLogic domain to use the virtual front-end name.

Instance Name Prefix
When you provision a SOA Suite on Marketplace, an “Instance Name Prefix” needs to be provided. This property is used to construct the names of all the resources, including: the WebLogic Server domain name, the cluster name, the Weblogic server names, the VM’s hostnames, etc.

This property must be set to the same value in the primary and secondary SOA systems, so that both systems have the same name for the WebLogic resources. Using the same name guarantees consistency and is required for the recovery of JMS messages and TLogs. It also simplifies customizations and operations in both sites.

There is no problem to use the same “Instance Name Prefix” in multiple instances in the same Cloud tenancy, as long as they are created in different regions and/or compartment. Each instance is shown only in its specific region and compartment.

The SOAMP provisioning process provides an optional feature that allows to configure custom names for the domain, the cluster, the admin server, the managed server’s prefix, etc. In that case, the names are not derived from the “Instance Name Prefix”. They take the values provided instead. It is possible to use this feature in the Disaster Recovery topology described in this whitepaper, as long as the custom names provided are the same in primary and standby.

Network communication between sites
The primary and standby databases need to communicate with each other over their listener port for redo transport. Secondary middle tier hosts need to communicate with the primary database for the initial setup also.

When you use the FSS with rsync method approach, WebLogic Administration host at each site needs to communicate via ssh (TCP/22) with the remote WebLogic Administration host for the rsync copy.

See the Appendix B – Summary of networking requirements for DR Setup in this document for more details on specific networking requirements.

The communication between primary and secondary sites can be through Oracle internal networks, by using Dynamic Routing Gateway, which is the recommended approach (refer to the Dynamic Routing Gateway documentation for additional details on the network configuration). The communication between sites can also happen over an Internet Gateway (Oracle Net’s traffic is encrypted), but this is not the recommended approach. Depending on which method is used, the appropriate ingress rules need to be enabled. Security rules are configured in the Security Lists for each Virtual Cloud Network (in the OCI console, Core Infrastructure > Networking Section > Virtual Cloud Network). More information about this is available in Security Rules section on the OCI documentation.

The amount of data replicated depends on the redo generated by the primary database, and this is directly related with application load, its transactionality, concurrency, etc. The overhead of the DBFS to replicate configuration is typically irrelevant compared to the runtime data that Data Guard synchronizes. To ensure a timely delivery of the redo log files to the standby database, a suitable network connection between the primary site and the secondary site must be provided. Oracle Cloud Infrastructure regions are interconnected with high-bandwidth, fault-tolerant networks achieving ≥ 99.95 percent reliability (≤5 packets lost in 10,000), which provides also a consistent latency. See Oracle Cloud Infrastructure Data Center Regions for more details.
Staging filesystems for the WebLogic domain config replication

Two different methods are included in this document to replicate the SOA WLS domain configuration between sites (DBFS, and FSS with rsync). Both methods require an assistance filesystem, DBFS or FSS respectively, to be mounted in the WebLogic hosts. In the next sections of this document, more details and specific instructions are provided to configure the staging filesystem in each case.

Custom files

MDS, SOA Composite deployments and policies are automatically synchronized across sites by Data Guard since they are stored in the database. Most of the WebLogic Server domain configuration that Oracle SOA Suite on Marketplace uses is synced initially across sites with the following considerations:

- Each SOA system will maintain the original JDBC URLs used to connect to their local DB even after the DR set up has completed. Only the schema prefix will be altered so that both locations point to the same schemas.
- All the configuration under weblogic_domain_name/config is automatically distributed, by the Weblogic Infrastructure, to the other nodes in the same site by the WebLogic cluster features.
- Custom application deployments (workflow task ears, custom ear/war files, deployment plans, JMS resources, etc.) and everything that resides under the Administration Server WLS domain directory (except temp data) is synced initially across sites with the procedures described in this paper. In the next sections of this document more details are provided about how the configuration replica is performed.

In case that customer has any other data that resides in other node’s or outside the domain directory of the Weblogic Administration Server, it will have to be manually copied to the secondary location.

SLA requirements

Oracle SOA Suite on Marketplace is a user-managed environment. The user must determine service level expectations for availability, data protection, and performance that are practical for a given configuration and application. Service Levels must be established for each of three dimensions relevant to disaster recovery that are applicable to any Data Guard configuration:

- **Availability**: Recovery Time Objective (RTO) describes the maximum acceptable downtime should an outage occur. This includes time required to detect the outage and to failover the database, the Web tier and SOA servers so that service is resumed. More details about this in the section RTO and RPO Overview of this document.

- **Data Protection**: Recovery Point Objective (RPO) describes the maximum amount of data loss that can be tolerated. In SOA’s case this is especially related to transaction logs, JMS messages and SOA instance information which all resides in the same database. The actual achievable RPO depends upon:
  - Available network bandwidth.
  - Network reliability.
  - Data Guard transport method used: either asynchronous for near-zero data loss protection, or synchronous for zero data loss protection.

More details about this in the section RTO and RPO Overview of this document.

- **Performance**: Database and Middle Tier response time may be different after failover if less capacity – compute, memory, I/O, etc., are provisioned at the standby system than in the primary system. This occurs when users purposefully under-configure standby resources to reduce cost (accepting reduced service level while in DR model). MAA best practices recommend to configure symmetrical capacity at both primary and standby in the web, application and database tiers so there is no change in response time after failover. Rapid provisioning available with the cloud can enable a middle ground where less capacity is initially deployed, but where the new primary is rapidly scaled-up should a failover be required.

In addition, the status of the hosts and services in the secondary site can impact on the RTO and RPO. As mentioned in the introduction, if the standby database is in shutdown status during normal business operation, it will not receive updates from primary and it will become out-of-sync. This can result in a data loss (impact on RPO) in case a switchover needs to be performed, thus it is not recommended to have the standby database stopped during normal business operation. The standby midtier hosts could be stopped, however, the configuration changes that are replicated from the primary site will not be pushed to the secondary domain configuration while they are stopped. In case of a switchover event, the RTO is increased if the midtier hosts need to be started and synchronized with primary. Thus, it is recommended to have the
secondary midtier hosts up (with WebLogic managed servers stopped). See About having compute instances stopped in standby site in Appendix C for more details.

Note: Independent of the service levels related to DR, all database instances created in the Oracle cloud conform to the service descriptions defined by the applicable Database Cloud Service\(^6\).
SOA SUITE ON MARKETPLACE DISASTER RECOVERY SETUP

In this document it is assumed that, as starting point, the primary site, that consists of an OCI DB system, a SOA Suite on Oracle Cloud Marketplace system, and the Load Balancer (OCI LBR), is live. The secondary DR configuration, that resides in a geographically remote site, will be created for the existing primary system.

Since the primary system may already be running in production, the DR configuration process is designed to cause minimum downtime (only the modification of the front-end address requires WebLogic server restarts).

The following flow chart describes the steps of the Disaster Recovery setup process for the three models described in this paper:

![Flow chart of the DR setup steps for all the models described in this whitepaper](image)

Figure 6 Flow chart of the DR setup steps for all the models described in this whitepaper
As described in the Topology Description point, depending on how the WebLogic domain configuration is replicated to the secondary site, there are three different DR models: **DBFS, FSS with RSYNC, and Block Volume Cross-Region replica**.

The next points in this section explain how to perform the DR setup for the DBFS and the FSS with RSYNC models. This is a summary of the steps for the set-up process:

1. **Choose a virtual front-end name**
2. **Prepare primary mid-tier for the virtual front-end**
3. **Setup the Database in Secondary Site**
4. **Provision SOA Suite on Marketplace in Secondary Site**
5. **Prepare Secondary mid-tier for the virtual front-end**
6. **Create the mid-tier hosts aliases**
7. **Configure the staging mounts for WebLogic domain config replication**
8. **Run the Disaster Recovery Setup utils (DRS)**

Since the **DR model based in Block Volume Cross-region replication differs significantly from the other two** (it does not use a staging location) it is explained separately in **Appendix D – Disaster Recovery Based On Block Volume Cross-Region Replication**.

1. **Choose a virtual front-end name**

   When an SOA Suite instance is created on Marketplace, the Load Balancer provisioned listens on a specific front-end IP. No front-end Fully-Qualified Domain Name (FQDN) is provided nor configured in the system. The primary site’s LBR listens on one front-end IP address, and second site’s LBR will listen on another front-end IP address. Each WLS cluster “Frontend Host” property is provisioned by default with the corresponding Load Balancer front-end IP.

   In a disaster recovery topology, the clients must access the system with a URL that uses a “Cloud region or data center” agnostic front-end FQDN, usually referred to as "virtual" front-end name or “vanity url”. This virtual front-end name should resolve to the LBR IP address for the current active primary site. You have to **choose a virtual front-end name** for the system within a DNS domain (for example, "soampdrs.mycompany.com") and **make it resolvable externally**. If you already have this configured for access to the primary system, you can reuse it for the DR configuration.

   To externally resolve this virtual front-end name, you must register it any formal public DNS (alternatively, you can add it to the client’s local hosts file). To resolve the virtual front-end name in the scope of the WebLogic hosts locally, the system’s hosts file should be manually configured prior to the DRS tool execution, as explained in next points.

   To determine the public IP address of the LBRs in your system, login into the OCI Console, select the correct region and compartment, navigate to Load Balancers section, click on your LBR, and look for the public IP address that the LBR listens on.

2. **Prepare primary mid-tier for the virtual front-end**

   You need perform some actions in the primary mid-tier to prepare it for the DR configuration.

   a) **Add the virtual front-end name and IP to the /etc/hosts file in all primary mid-tier hosts**.

      Each site should always resolve it’s local LBR regardless of client-facing resolution via DNS. With root user, edit the /etc/hosts file and map the primary LBR public IP to the virtual front-end FQDN. Repeat in all primary WebLogic hosts. Example:

      ```bash
      [oracle@soampdrs-soa-0 ~]$ more /etc/hosts
      127.0.0.1  localhost localhost.localdomain localhost4 localhost4.localdomain4
      ::1      localhost localhost.localdomain localhost6 localhost6.localdomain6
      ...
      # Front-end virtual name
      111.111.111.111 soampdrs.mycompany.com
      ```

      **NOTE:** the /etc/hosts file of the primary soa hosts must not be altered when there is a switchover or failover. Primary soa hosts will always resolve the virtual front-end name with its front-end IP. The dns update that is needed during the switchover and failover procedures is performed in the DNS or host files used by the soa clients.

   b) **Configure the WebLogic SOA cluster front-end name property.**
Login in the WebLogic Console of your instance and:
- Navigate to Environment > Clusters and select the cluster
- Go to Configuration > HTTP.
- Set the Fronted host to the virtual front-end FQDN (example “soampdrs.mycompany.com”).
- Save and activate.
- A cluster restart is required for this change to be effective.

![Figure 7 Cluster front-end host configuration](image)

**c) Update t3/rmi urls (if used) with cluster syntax**

The urls used for RMI invocations in the WebLogic cluster need to be agnostic to the IPs or hostnames used by each site. Instead of using the host:port,host:port JNDI urls syntax, change them to use the cluster syntax. The cluster syntax is as follows: cluster:t3://cluster_name. For example, to modify the JMS adapter factory properties to use this syntax, follow these steps:

a. Log into your Oracle WebLogic Server Administration Console for your SOA instance.
b. Click **Deployments** in the left pane for Domain Structure.
c. Click JmsAdapter under Summary of Deployments on the right pane.
d. Click the Configuration > Outbound Connection Pools tab.
e. Expand oracle.tip.adapter.jms.ImsConnectionFactory to see the configured connection factories.
f. Click the specific instance you are using (for example, eis/wls/Queue). The Outbound Connection Properties for the connection factory opens.
g. Click Lock & Edit.
h. In the FactoryProperties field (click on the corresponding cell under Property value), alter the java.naming.provider.url filed to use the cluster syntax (leave the rest of the fields as they were): java.naming.provider.url=cluster:t3://cluster_name
i. Click Save after you update the properties. The Save Deployment Plan page appears.
j. Enter a location for the deployment plan.
k. Copy the deployment plan from your SOA node1 to your SOA node2 in the exact same directory/location or use the default DBFS mount point present in SOA system as the location to host these deployment plans (all nodes in the SOA cluster can access /u01/soacs/dbfs/share)
l. Click Save and Activate.
m. Click Lock & Edit
n. Click Deployments, select the JMS Adapter and Click Update.

---

7 Using the cluster name syntax in t3/RMI URLs is feasible only for *intra-domain* invocations. T3/rmi clients that are external to the SOA domain will not be able to use this approach and will have to use the appropriate DNS mapping of the host:port list when switching to the secondary site. A TCP load balancer can be used for the JNDI InitialContext retrieval, but subsequent requests from JMS clients will connect to each host:port directly, so the DNS mapping to secondary site hosts ips is required also in this case.
3. Setup the Database in Secondary Site

The database in the secondary site is created as a **Data Guard physical standby of the primary database**. There are 2 options to do this: one is to use the OCI Console to enable Data Guard (referred in this document as “automated Data Guard”), and the other option is to manually create and configure the standby database with dgmrgl commands (referred in this document as “manual Data Guard”).

**The recommended approach is to configure the Data Guard with the OCI Console (option 1).** This way, it is integrated with the OCI Console User Interface and allows you to use the Console to manage Oracle Data Guard associations in your DB system. It also provides out of the box configuration for backups in the Data Guard. Follow the point **Option 1) Configuring Data Guard using OCI Console** to enable the Data Guard.

If for any reason the feature to enable Data Guard is not available for your case (please refer to the DB System documentation to check the availability of the Data Guard across regions feature in each DB Systems flavor/edition), you can still configure the Data Guard manually using scripts provided in this whitepaper. Follow steps described in **Option 2) Configuring Data Guard manually** for this.

When using EXACS database, you need to setup secondary database as described in the Exadata Cloud Service documentation **Using Oracle Data Guard with Exadata Cloud Service**. Go to the next point “3.3 Considerations for EXACS”.

3.1. **Option 1) Configuring Data Guard using OCI Console**

When you enable Data Guard with the OCI Console, the **secondary DB system is automatically provisioned and configured as physical standby** when you click on **Enable Data Guard** in the primary DB System. There are some requirements for this, for example: both DB systems will be in the same compartment, both DB Systems will be the same shape type, if the DB Systems will be in different regions, they must be connected via remote VCN peering, etc. See **Using Oracle Data Guard** in Oracle Cloud Infrastructure Documentation for more details about these requirements.

To enable Data Guard to primary database, login to OCI Console and navigate to the primary DB System and click in the primary database. You can enable Data Guard in the section “Data Guard Associations”. Most of the configuration properties of the secondary DB System (like version, DB name, etc) are predefined because they are inherited from primary, but you need to provide some configuration properties. The following table provides examples and requirements for these properties:

<table>
<thead>
<tr>
<th>DB System Configuration Property</th>
<th>Existing Primary DB System / Example</th>
<th>Secondary DB System / Example</th>
<th>Requirement for AUTOMATED DG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Cloud Tenancy</td>
<td>XXXX / paasmaa</td>
<td>YYYY / paasmaa</td>
<td>must be the same</td>
</tr>
<tr>
<td>Compartent</td>
<td>XXXX / soadr</td>
<td>XXXX / soadr</td>
<td>must be the same</td>
</tr>
<tr>
<td>Region</td>
<td>XXXX / Ashburn</td>
<td>YYYY / Phoenix</td>
<td>must be different (it is recommended to use different region for DR)</td>
</tr>
<tr>
<td>Availability Domain</td>
<td>XXXX / efEXT:US-ASBURN-AD1</td>
<td>YYYY / efEXT:PHX-AD-1</td>
<td>must be different</td>
</tr>
<tr>
<td>Peer DB System Name</td>
<td>XXXX / drdba</td>
<td>YYYY / drdbb</td>
<td>must be different</td>
</tr>
<tr>
<td>Shape</td>
<td>XXXX / VM.Standard2.1</td>
<td>XXXX / VM.Standard2.1</td>
<td>must be the same</td>
</tr>
</tbody>
</table>
Virtual Cloud network | XXXX / soadrvcn1ash | YYYY / soadrvcn1pho | systems are expected to be remote, hence different from primary (connected via remote VCN peering)

Client subnet | XXXX / soadrvcn1ashAD1 | YYYY / soadrvcn1phoAD1 | must be different

Hostname Prefix | XXXX / drdba | YYYY / drdbb | must be different

Administrator password | XXXX / acme1234# | YYYY / acme1234# | must be the same

3.2. Option 2) Configuring Data Guard manually

It is required to manually configure Data Guard manually when it is not possible to use the same cloud tenancy for primary and standby, or when the enable Data Guard option provided by OCI Console is not available for the DB flavor and/or locations involved in the DR configuration. In this case, the secondary database must be provisioned as a regular DB System, and then, Data Guard must be manually configured. For this manual configuration, you can use the Data Guard setup scripts provided in this document, as explained in these steps:

3.2.1. Provisioning Secondary Database

Note: In case that the Data Guard has been enabled using the OCI Console, these steps must be skipped and you can continue with section Provision SOA Suite on Marketplace in Secondary Site.

When you configure Data Guard manually, you first need to provision the secondary database, using the same Database name, PDB name, release, patch level, number of nodes, and edition used in primary. This may require patching the primary system (especially if it has been running for a long time) before you create the standby. Oracle recommends to use the same Compute Shape and Storage Size that are used for primary. Follow the steps in the Cloud DB System documentation to provision the required Database System for the standby data center.

The following table provides examples and requirements for the properties that need to be used in the standby DB System creation process:

<table>
<thead>
<tr>
<th>DB SYSTEM Configuration Property</th>
<th>Existing Primary DB System / Example</th>
<th>Secondary DB System / Example</th>
<th>Requirement for MANUAL DG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Cloud Tenancy</td>
<td>XXXX / paasma</td>
<td>YYYY / paasma</td>
<td>can be different</td>
</tr>
<tr>
<td>Compartment</td>
<td>XXXX / soadr</td>
<td>YYYY / soadr</td>
<td>can be different</td>
</tr>
<tr>
<td>Region</td>
<td>XXXX / Ashburn</td>
<td>YYYY / Phoenix</td>
<td>must be different</td>
</tr>
<tr>
<td>Availability Domain</td>
<td>XXXX / efEXT-US-ASBURN-AD1</td>
<td>YYYY / efXT-PHX-AD-1</td>
<td>must be different</td>
</tr>
<tr>
<td>DB System Name</td>
<td>XXXX / ddrba</td>
<td>YYYY / drrdb</td>
<td>must be different</td>
</tr>
<tr>
<td>Shape</td>
<td>XXXX / VM.Standard2.1</td>
<td>XXXX / VM.Standard2.1</td>
<td>must be the same</td>
</tr>
<tr>
<td>Total node count</td>
<td>N / 1</td>
<td>N / 1</td>
<td>must be the same</td>
</tr>
<tr>
<td>Oracle Database Software edition</td>
<td>EE, EE-HP or EE-EP</td>
<td>EE, EE-HP or EE-EP</td>
<td>must be the same</td>
</tr>
</tbody>
</table>
### Table: Disaster Recovery Configuration

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Primary Site</th>
<th>Secondary Site</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available storage</td>
<td>XXXX / 256</td>
<td>XXXX / 256</td>
<td>must be the same</td>
</tr>
<tr>
<td>License type</td>
<td>LI, BYOL / BYOL</td>
<td>LI, BYOL / BYOL</td>
<td>can be different</td>
</tr>
<tr>
<td>SSH public key</td>
<td>XXXX</td>
<td>YYYY</td>
<td>must be the same (for DRS)</td>
</tr>
<tr>
<td>Virtual Cloud network</td>
<td>XXXX / soadrvcn1ash</td>
<td>YYYY / soadrvcn1pho</td>
<td>must be different</td>
</tr>
<tr>
<td>Client subnet</td>
<td>XXXX / soadrvcn1ashAD1</td>
<td>YYYY / soadrvcn1phoAD1</td>
<td>must be different</td>
</tr>
<tr>
<td>Hostname Prefix</td>
<td>XXXX / drdba</td>
<td>YYYY / drdbb</td>
<td>must be different</td>
</tr>
<tr>
<td>Database Name</td>
<td>XXXX / ORCL</td>
<td>XXXX / ORCL</td>
<td>must be the same</td>
</tr>
<tr>
<td>Database Version</td>
<td>XXXX / 21c</td>
<td>XXXX / 21c</td>
<td>must be the same</td>
</tr>
<tr>
<td>PDB name</td>
<td>XXXX / PDB1</td>
<td>YYYY / PDB1</td>
<td>must be the same</td>
</tr>
<tr>
<td>Administrator password</td>
<td>XXXX / acme1234#</td>
<td>YYYY / acme1234#</td>
<td>must be the same</td>
</tr>
<tr>
<td>Enable automatic backups</td>
<td>X / Checked</td>
<td>Y / unchecked</td>
<td>must be disabled in stby</td>
</tr>
</tbody>
</table>

**NOTE:** The default database instance created on the secondary site will be deleted later as it cannot be used as a Data Guard standby database. It is created with the same name as primary to get the required lifecycle scripts seeded in the system with the same configuration as the primary DB.

Make sure to apply the required patches to the DB in both locations (primary and secondary) so that to both are at the same patch level. More precisely for 12c database version: Data Guard configuration requires a fix for bug 22611167. Verify if the patch for this bug is applied in both the primary and secondary DB systems by checking the opatch output, and apply it in case it is not. Latest OCI 12cR2 DB systems have the patch for this bug pre-installed.

### 3.2.1. Configuring Data Guard between primary and secondary

**Note:** In case that the Data Guard has been enabled with the OCI Console, these steps must be skipped and you can continue with section Provision SOA Suite on Marketplace in Secondary Site.

To configure Data Guard manually between the existing primary and secondary databases follow these steps:

a) In a Data Guard, the primary and standby databases need to communicate each other on the listener port. It is also needed that each database can reach its own IP on the appropriate listener port. Make sure that the appropriate ingress rules are defined in each VCN (primary and standby) to allow these connections. For RAC databases, it is a requirement that primary and standby RAC communicate via Dynamic Routing Gateway, because the scan and VIP IP addresses must be reachable from one site to the other.

You can verify each communication with `nc` command (use the public/private IPs depending on your network topology). For example:

```
[opc@drDBa ~]$ nc -wv 5 -z <secondary_db_vip> 1521
```

Example of correct output:
b) Download the set of scripts for manual Data Guard configuration, and follow the instructions described in the solution playbook Configure a standby database for disaster recovery. This set of scripts is valid to configure Data Guard in single instance databases and in RAC scenarios.

c) After the DG setup is complete, enter the Data Guard Broker CLI from the primary system to check the configuration (redo apply may take some time to catch up):

```
DGMGRL> show configuration
Configuration - ORCL_lhr2bb_ORCL_fra22g
  Protection Mode: MaxPerformance
  Members:
    ORCL_lhr2bb - Primary database
    ORCL_fra22g - Physical standby database
  Fast-Start Failover: Disabled
  Configuration Status: SUCCESS (status updated 33 seconds ago)
```

### 3.3. Considerations for EXACS

When the database is Exadata Cloud Service database, follow these recommendations:

- Setup Data Guard as described in the Exadata Cloud Service documentation Using Oracle Data Guard with Exadata Cloud Service.

- When you provision SOAMP with an EXACS database, you can provide as input parameter the PDB service name that you want to use. That will be the service used by SOAMP to connect to the PDB. The EXACS databases use CRS database services out-of-the-box. There are some services already configured for the CDB and for the PDB, which you can list with the srvctl command. The primary EXACS has a PDB service pre-configured, with name <DBNAME_PDBNAME>.paas.oracle.com. Example:

```
[oracle@primary-exadb-host1 ~]$ srvctl status service -db $ORACLE_UNQNAME
Service exadb_dg is running on instance(s) EXADB1,EXADB2
Service exadb_dg_ro is not running.
Service pdb1_dg is running on instance(s) EXADB1,EXADB2
Service pdb1_dg_ro is not running.
```

However, this service is not configured in the standby EXACS database. By default, the standby database has only the “dg” and “read only” services only. Example:

```
[oracle@standby-exadb-host1 ~]$ srvctl status service -db $ORACLE_UNQNAME
Service exadb_dg is not running.
Service exadb_dg_ro is not running.
Service pdb1_dg is not running.
Service pdb1_dg_ro is not running.
```

Follow these steps to get the appropriate service configuration in EXACS Data Guard for SOAMP DR:

[https://github.com/oracle-samples/maa/raw/main/dg_setup_scripts/dg_setup_scripts.zip](https://github.com/oracle-samples/maa/raw/main/dg_setup_scripts/dg_setup_scripts.zip)
• **Create the PDB service in the standby EXACS database.** For consistency, use the same PDB service name that is used by the primary SOAMP to connect to the primary EXACS PDB. Example:

```sql
[oracle@standby-exadb-host1 ~]$ srvctl add service -db $ORACLE_UNQNAME -service exadb_pdb1.paas.oracle.com -preferred EXADB_PHO1,EXADB_PHO2 -pdb PDB1 -role "PRIMARY,SNAPSHOT_STANDBY,PHYSICAL_STANDBY"
[oracle@standby-exadb-host1 ~]$ srvctl modify service -db $ORACLE_UNQNAME -service exadb_pdb1.paas.oracle.com -r!bgoal SERVICE_TIME -clbgoal SHORT -pbd PDB1
[oracle@standby-exadb-host1 ~]$ srvctl start service -db $ORACLE_UNQNAME -service exadb_pdb1.paas.oracle.com
```

Note that it is important to provide the 3 roles to the “-role” parameter when you create the service, so the service automatically starts when the database is in primary, snapshot or physical standby role (which is needed for the setup and for lifecycle operations).

• Also, **modify the PDB service in primary** to add the standby roles too, because it is normally configured with the “primary” role only.

```sql
[oracle@primary-exadb-host1 ~]$ srvctl modify service -db $ORACLE_UNQNAME -service exadb_pdb1.paas.oracle.com -role "PRIMARY,SNAPSHOT_STANDBY,PHYSICAL_STANDBY"
```

• Pre-load the environment variables file in the oracle user’s profile in the EXACS database primary and standby hosts. The variables ORACLE_HOME, ORACLE_UNQNAME, PATH, etc., that are required to run database commands in the database hosts, are set in a file with name /home/oracle/<DBNAME>.env . However, this file is not pre-loaded by default in the oracle user’s profile. Add it to the oracle’s user profile, by calling it from the .bashrc. For example:

```bash
[oracle@exacus1pho-2t2id1 ~]$ more .bashrc
# User specific aliases and functions
./home/oracle/EXACS.env
```

This is a requirement to run the DRS tool later. DRS tool needs to run some commands on the primary and standby database hosts (e.g. to check the Dataguard status or to perform role conversions) and it requires the database environment variables to be already loaded in the oracle user’s profile.

### 4. Provision SOA Suite on Marketplace in Secondary Site

Make sure that the secondary SOAMP version and patch level provisioned in the secondary location matches the one running in the primary site. SOAMP provisioning offers the 4 latest versions, which means that the customer can provision the same SOAMP version in timeframe 3-6 months. If, at the moment of the secondary provisioning, the PSU or patch level of the primary SOAMP is not available in provisioning menu, the primary SOAMP must be patched to the same level as the newly provisioned secondary site. Check [What’s New in Oracle SOA Suite on Marketplace](https://www.oracle.com/) to see which patches levels are provisioned in each SOAMP version.

Secondary SOA system will be created pointing to the secondary DB system, which must be open in snapshot standby mode. Before you provision the secondary SOA Suite on OCI Marketplace, **convert standby database to snapshot standby.** This will make the standby database to stop applying changes from primary and be opened in read-write, which is needed to allow the secondary SOA creation. To do this, execute the following as oracle user in the primary DB host:

```sql
[oracle@drdba ~]$dgmgrl sys/your_sys_password@primary_db_unqname
DGMGRL> CONVERT DATABASE "secondary_db_unqname" to SNAPSHOT STANDBY;
Converting database "secondary_db_unqname" to a Snapshot Standby database, please wait...
Database "secondary_db_unqname" converted successfully
```

You need to follow the steps in the [SOA Suite on OCI Marketplace documentation](https://www.oracle.com/) to create the secondary site SOA system pointing to the secondary DB System that was converted to snapshot in the previous step.

Note the Stack Name can be different, but you must use the **EXACT same Instance Name Prefix** that you used in your primary location. Oracle recommends that the exact same capacity and compute configuration is used on both primary and standby locations for the ideal failover/switchover behavior.
The following table summarizes the provisioning wizard options for the set up:

<table>
<thead>
<tr>
<th>SOA Suite on Marketplace Property</th>
<th>Value in Primary / Example</th>
<th>Value in Secondary / Example</th>
<th>Requirement for DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region</td>
<td>XXXX / Ashburn</td>
<td>YYYY / Phoenix</td>
<td>Must be different</td>
</tr>
<tr>
<td>Version</td>
<td>XXXX / 12.2.1.4</td>
<td>XXXX / 12.2.1.4</td>
<td>Must be the same</td>
</tr>
<tr>
<td>Stack Name</td>
<td>XXXX / soampdrsPrim</td>
<td>YYYY / soampdrsStby</td>
<td>Can be different</td>
</tr>
<tr>
<td>Instance Name Prefix</td>
<td>XXXX / soampdrs</td>
<td>XXXX / soampdrs</td>
<td>Must be the same</td>
</tr>
<tr>
<td>Service Type</td>
<td>XXXX / SOA with SB &amp; B2B Cluster</td>
<td>XXXX / SOA with SB &amp; B2B Cluster</td>
<td>Must be the same. This document supports “SOA with SB &amp; B2B Cluster” and “MFT Cluster” service types</td>
</tr>
<tr>
<td>Compute Shape</td>
<td>XXXX / VM.Standard2.1</td>
<td>XXXX / VM.Standard2.1</td>
<td>Must be the same</td>
</tr>
<tr>
<td>SSH Public Key</td>
<td>XXXX</td>
<td>YYYY</td>
<td>Must be the same</td>
</tr>
<tr>
<td>Cluster Node Count</td>
<td>N / 2</td>
<td>N / 2</td>
<td>Must be the same</td>
</tr>
<tr>
<td>Administration User Name</td>
<td>XXXX / weblogic</td>
<td>XXXX / weblogic</td>
<td>Must be the same</td>
</tr>
<tr>
<td>Administrator Password</td>
<td>XXXX / acme1234#</td>
<td>YYYY / acme1234#</td>
<td>Must be the same password (although if it is encrypted with KMS the encrypted value can differ)</td>
</tr>
<tr>
<td>Use KMS decryption</td>
<td>X / unchecked</td>
<td>X / unchecked</td>
<td>Can be different. KMS is optional and used for provisioning only.</td>
</tr>
<tr>
<td>Network Compartment</td>
<td>XXXX / soadr</td>
<td>YYYY / soadr</td>
<td>Can be different</td>
</tr>
<tr>
<td>VCN</td>
<td>XXXX / soadrvcn1ash</td>
<td>YYYY / soadrvcn1pho</td>
<td>Must be different</td>
</tr>
<tr>
<td>Subnet</td>
<td>XXXX / soadrvcn1ashAD1</td>
<td>YYYY / soadrvcn1phoAD1</td>
<td>Must be different</td>
</tr>
<tr>
<td>Provision Load Balancer</td>
<td>must be checked</td>
<td>must be checked</td>
<td>Checked in both cases</td>
</tr>
<tr>
<td>Database Strategy</td>
<td>XXX / Database System</td>
<td>XXX / Database System</td>
<td>Must be the same</td>
</tr>
<tr>
<td>DB system</td>
<td>XXXX / drdba</td>
<td>YYYY / drdbb</td>
<td>Must be different</td>
</tr>
<tr>
<td>Database in the DB system</td>
<td>XXXX / ORCL</td>
<td>XXXX / ORCL</td>
<td>Must be the same</td>
</tr>
<tr>
<td>PDB</td>
<td>XXXX / PDB1</td>
<td>XXXX / PDB1</td>
<td>Must be the same</td>
</tr>
<tr>
<td>Database administrator</td>
<td>SYS</td>
<td>SYS</td>
<td>Must be the same</td>
</tr>
</tbody>
</table>
### Database administrator password

| XXXX/ acme1234# | XXXX / acme1234# | Must be the same password (although if it is encrypted with KMS the encrypted value can differ) |

### Specify Custom RCU Schema prefix

| Check/Unchecked | Check/Unchecked | You can specify a custom schema or let the provisioner to create a random schema prefix. |

**(only if “Specify Custom RCU Schema prefix” was checked)**

### Custom schema prefix

| XXXX / PREFIXA | YYYY / PREFIXB | Must be different

If you check to specify a custom rcu schema prefix, you must specify a different RCU schema prefix than primary. This is to prevent from provisioning issues due to already existing schemas.

The secondary schemas will be later discarded: only primary schemas will be used once the DR is setup.

### Specify RCU Schema custom Password

| Check/Unchecked | Check/Unchecked | Can be the same |

**(only if “Specify RCU Schema custom Password” was checked)**

### Custom RCU Schema Password

| XXXX / acme124# | XXXX / acme124# | Can be the same |

### Service Instance Advanced (OPTIONAL custom names and prefixes)

| XXXX | XXXX | Must be the same

If you are not using the default values, make sure that you use the same ports than in primary

### Service Instance Advanced (OPTIONAL ports)

| XXXX | XXXX | It is possible to use this feature to configure custom names for domain, cluster, etc., but the custom names provided must be the same than in primary.

---

**NOTE:** using Key management service during provisioning is optional. In case you check it, KMS service is used only to encrypt and decrypt passwords during the SOA Suite on Marketplace provisioning. It is not used for runtime or lifecycle tasks. The encrypted value of the password that are provided to the provisioning wizard may be different, but the clear password must the same.

---

Once the provisioning process completes, the SOA Suite servers can be sanity verified.
NOTE: Verify that the db connect string in the datasources is compliant with the recommended formats:

If the database is a single instance, the recommended db connect string is:
```java
jdbc:oracle:thin:@/<<db-scan-address>>:<port>/<pdb_service_name>
```

If the database is a RAC, the datasources must be GridLink datasources and the recommended db connect string is:
```java
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=<<db-scan-address>>)(PORT=<port>)))\(CONNECT_DATA=(SERVICE_NAME=<pdb_service_name>)))
```

NOTE: Oracle SOA Suite on Marketplace provisions SOA schemas using a prefix that is specific to each SOA cloud instance. This means that in the initial provisioning, the secondary location servers will use different schemas names than primary. This is critical for systems that are already running because this will prevent the execution of composites/flows by the initial SOA domain in the secondary location. It is needed that only one site has active SOA servers pointing to an available database at any point in time. Otherwise message and callback duplications could occur leading the SOA system to inconsistencies.

Once the secondary location JDBC strings are updated to point to the same schemas as production (once the DR is setup), the SOA servers in the secondary location will see the same data that the production ones were seeing when the snapshot conversion occurred. If any SOA flows, callbacks etc. are pending, the servers in the secondary location will try to complete those. Thus, it is important that instances are drained and completed on the primary site before converting the standby database to snapshot or duplications could occur.

If for any reason, a long time passes since you provision the secondary SOAMP instance until you continue with the DR setup steps, you can stop the WebLogic administration server and managed servers in secondary and convert the standby database to physical standby again. This way, the redo apply gap between standby database and primary does not increase. After you do this, do not try to start the WebLogic administration server and manager servers in secondary site until the DR setup is completed. Because the secondary servers look for the original secondary schemas in the database, and they are not longer there. This is expected because the changes performed to a snapshot database are lost when it is converted to physical standby again. So, in this case, just keep secondary WebLogic admin and managed servers stopped and continue with the DR setup tasks. Note that in this case, you will have to use --skip_checks flag when you run DRS in next steps.

5. Prepare Secondary mid-tier for the virtual front-end

You need to add the front-end name and IP to the `/etc/hosts` file in all secondary SOA mid-tier hosts. With `root` user, edit the `/etc/hosts` file and map the SECONDARY LBR public IP to the virtual front-end name. Repeat in all secondary mid-tier hosts. Example:

```
[oracle@soampdrs-soa-0 ~]$ more /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4 :1 localhost localhost.localdomain localhost6 localhost6.localdomain6
... # Front-end virtual name
222.222.222.222 soampdrs.mycompany.com
```

You do not need to update the front-end address for the secondary WebLogic cluster in the WebLogic Console, because that information will be copied from the primary WebLogic Domain configuration.

NOTE: the `/etc/hosts` file of the secondary soa hosts must not be altered when there is a switchover or failover. Secondary soa hosts will always resolve the virtual front-end name with its front-end IP. The dns update that is needed during the switchover and failover procedures is performed in the DNS or host files used by the soa clients.

6. Create the mid-tier hosts aliases

The WebLogic domain configuration in secondary will be a copy of the primary WebLogic domain once the DR setup is completed. Hence, the hostnames used as listen addresses by the primary WebLogic servers (which are the hostnames of the primary mid-tier hosts) need to be valid in the secondary location but mapping to the secondary IPs.
And the other way around: the hostnames of the secondary servers need to be valid in the primary location but mapping to the primary IPs. This part is not essential, because it is expected that only primary hostnames names are used in the WebLogic configuration. But this is done to avoid errors in primary, in case that any reference to secondary names is added to the config while the secondary site takes the primary role.

To configure the required hostnames mapping, you can use 2 approaches: adding the hostnames as aliases to the /etc/hosts files or adding them to private DNS views in OCI.

6.1. Option 1) Use the /etc/hosts files

The other site’s hostnames are added as aliases to the /etc/hosts files in the mid-tier hosts.

The Disaster Recovery Setup (DRS) utils used in next steps will automatically perform the pertinent modifications in the /etc/hosts of the SOA compute instances:

- In the secondary SOA hosts, DRS will add the primary soa hostnames, as aliases of the secondary names in the /etc/hosts file (hence, they will be resolved with secondary IPs). This is essential because the primary SOA hostnames are used in the WebLogic configuration.
- In the primary SOA hosts, DRS will add the secondary soa hostnames, as aliases of the primary names (hence, they will resolve them with primary IPs). This is not essential, because it is expected that only primary hostnames are used in the WebLogic Configuration. But it is done to avoid errors in primary, in case that any reference to secondary names was added to the config while the secondary site takes the primary role.

You do NOT have to add the hostnames aliases manually. DRS will do it automatically when you run it in the next steps.

This mode is valid in all the scenarios: when the same DNS server is used in primary and secondary sites, and when separated DNS servers are used in primary and secondary. Because the entries in the /etc/hosts file have precedence over the DNS resolution. This precedence is defined in the directive “hosts” of the /etc/nsswitch.conf. By default, it is set to “files”, which means that /etc/hosts resolution takes precedence over the DNS.

A disadvantage of this method is that it requires to manually add the entries to all the SOA hosts. So when you add new nodes in a scale-out operation, the new node is not able to resolve the names until you modify its /etc/hosts file. This requires additional manual steps in the scale-out operations (see Scale-out/in procedures in a SOA Marketplace DR section of this paper).

6.2. Option 2) Use OCI Private DNS views

You can add the required entries to private DNS views of each VCN. Adding the entries to the private DNS views has advantages for the scale-out operations, because the new nodes in secondary site will be able to resolve the primary names out-of-the-box. Scale-out operations are simplified with this approach.

Check the Github repository https://github.com/oracle-samples/maa/tree/main/private_dns_views_for_dr for detailed instructions and terraform scripts.

7. Configure the staging mounts for WebLogic domain config replication

Depending on the method used to replicate the WLS domain configuration (DBFS or FSS with RSYNC), follow the steps described in this section to configure the desired option for your Disaster Recovery topology.

7.1. Option 1) Configure the DBFS mounts for DBFS based method

SOA Suite on OCI Marketplace comes with a Database File System (DBFS) mount already configured and mounted. A DBFS file system is a standard file system interface on top of files and directories that are stored in database table. As it is stored in the database, a DBFS file system can be used as a shared file system accessible by all the mid-tier hosts. Hence, the DBFS filesystem configured in SOA Suite on OCI Marketplace (/u01/soacs/dbfs or /u01/soacs/dbfs_directio for direct-io access) allow sharing files between the mid-tier nodes in the instance (for example, deployment plan xml files).

The Disaster Recovery solution described in this document assumes that this DBFS filesystem is operative in the SOA Suite on OCI Marketplace instance. It is used as an assistance filesystem to sync changes from primary to standby during the initial Disaster Recovery setup, and also to replicate configuration changes during the system’s lifecycle. The DBFS mount is not used for other WebLogic runtime operations related with disaster recovery, so it is not critical for the service nor has a big impact on the performance of the system.

SOAMP hosts are provisioned with a DBFS filesystem configured and mounted by default:
If they are not mounted, you can mount them with the script $DOMAIN_HOME/dbfs/dbfsMount.sh

SOAMP DR will make use of this DBFS mount (/u01/soacs/dbfs) to transfer the primary WebLogic domain configuration to secondary site.

**If you are going to use the DBFS based method topology, you can continue in the step 8. Run the Disaster Recovery Setup utils (DRS).**

### 7.2. Option 2) Configure the FSS mounts for FSS with rsync method

When you use the FSS with rsync method to replicate the WebLogic configuration, two FSS filesystems need to be created: one in the primary site and another in the secondary site. These file systems are mounted by the local hosts only. There are no direct cross-region NFS mounts in the topology for security and performance reasons. These filesystem mounts are used as staging areas for content to be replicated between sites with rsync commands. They store a copy of the domain config. They are not used for runtime.

During initial DR setup, the primary FSS volume needs to be mounted on the primary site admin server host and the secondary FSS volume needs to be mounted on all of the secondary site SOA hosts. The secondary site mount needs to be available on all hosts because it is used as the source for the initial copy of the replicated domain configuration to all the secondary site SOA hosts during the DR setup phase.

---

**NOTE:** Once you have completed the first config sync (which is done during the initial DR setup), the FSS mounts are only required in the primary and standby WLS Administration hosts. You can umount them from the other WLS nodes, unless you use them to store additional artifacts that require them to be mounted in all the nodes.
Follow these steps to configure and mount the OCI FSS mounts:

<table>
<thead>
<tr>
<th>Step to configure and mount FSS</th>
<th>DETAILS</th>
<th>SAMPLE VALUES IN PRIMARY</th>
<th>SAMPLE VALUES IN SECONDARY</th>
</tr>
</thead>
</table>
| 1  Create a mount target in each region (if it does not already exist) | -Connect to OCI Console  
-Select the proper region and compartment (primary or secondary)  
-Go to "File storage" > “Mount target”.  
-Click “Create Mount target”  
-Once created, note down the IP of each one | New Mount Target Name: soampdrPri_mt  
Availability Domain: <same than primary soa>  
Virtual Cloud Network: <same than primary soa>  
Subnet: <same than primary soa> | New Mount Target Name: soampdrStby_mt  
Availability Domain: <same than secondary soa>  
Virtual Cloud Network: <same than secondary soa>  
Subnet: <same than secondary soa> |
| 2  Create a File Storage Service (FSS) file system in each region | -Connect to OCI Console  
-Select the proper region (primary or secondary)  
-Go to "File storage” > “File System”  
-Click “Create File System”  
-Select the proper mount target in each case | Name: soampdrPri_fss  
Availability Domain: <same than primary soa>  
Export: /fssmount  
Mount target: soampdrPri_mt (previously created) | Name: soampdrStby_fss  
Availability Domain: <same than secondary soa>  
Export: /fssmount  
Mount target: soampdrStby_mt (previously created) |
### 3 Validate/set the network security rules required for FSS mount

Some network rules in each subnet are required to allow the NFS traffic between hosts and mount target. Use the instructions in Configuring VCN Security Rules for File Storage to set up security rules correctly for your file systems.

### 4 Mount the file system on the SOA hosts

- **In ALL SOA hosts:**
  - Create the local mount point
    ```bash
    # sudo mkdir /fssmount
    ```
  - With user root, edit `/etc/fstab` and add the mount, with the appropriate region-specific mount target IP (primary mount target IP in primary site and secondary mount target IP in secondary site):
    ```bash
    <mount_target_ip_address>:<export_name>
    <your_local_mount_point> nfs
    defaults,nofail,nosuid,resvport 0 0
    ```
  - Mount the new filesystem:
    ```bash
    # sudo mount
    ```

### 5 Verify mounted file system

- **In all SOA hosts:**
  ```bash
  # df -h | grep fssmount
  # ls -la /fssmount
  ```

### 6 Create the folders that will be used later

- Once the FSS volumes are mounted, create the folder **“domain_config_copy”** with user oracle (run these commands in one of the SOA hosts in both sites):
  ```bash
  # sudo chown oracle:oracle /fssmount
  # sudo su - oracle
  # mkdir -p /fssmount/domain_config_copy
  ```

### 8. Run the Disaster Recovery Setup utils (DRS)

The Disaster Recovery Setup utils (DRS) is a set of scripts that orchestrates and runs the configuration steps for the SOA Suite on Marketplace disaster recovery setup.

#### a) Review the required cross-site connectivity

The DRS tool currently requires the following communication between sites:

- **From Secondary midtier hosts to primary DB IP, port 1521** (and to primary scan IPs if a RAC database is used).
If primary and secondary databases connect using OCI-internal network interconnects via remote peering and Dynamic Routing Gateway, then secondary midtier hosts will connect to primary DB host private IP (and to primary scan IPs if a RAC database is used). This is the **recommended approach**.

If primary and secondary databases connect via their public IPs (because no remote peering/DRG is used between sites), secondary midtier host will connect to primary DB host public IP. This is **not a recommended approach** in general, and not suitable for RAC DG.

A quick check can be run on all the secondary midtiers with user oracle to verify the connectivity to private/public primary database IPs before you run DRS, depending on the network scenario:

```
java -classpath /u01/app/oracle/middleware/wlserver/server/lib/weblogic.jar utils.dbping ORACLE_THIN system <system_password> <primary_ip_to_check>:1521/<primary_db_service>
```

- **From Primary WLS Administration host to secondary WLS Administration host IP, port 22**
  This is required **only in the FSS with rsync approach**, for the WebLogic Domain rsync copy from primary to secondary.

If primary and secondary sites connect using OCI-internal network interconnects via remote peering and Dynamic Routing Gateway, then primary WLS Administration host will connect to secondary WLS Administration host private IP. This is the **recommended approach**.

If primary and secondary sites connect using their public IPs (because no remote peering/DRG is used between sites), then primary WLS Administration host will connect to secondary WLS Administration host public IP. This is **not a recommended approach**.

See the Appendix B – Summary of networking requirements for DR Setup in this document for more details on specific networking requirements.

**b) Choose a host to run DRS**

The tool can be run from any host (with operating system OEL 7 or OEL8) that has SSH connectivity to the SOA and DB hosts involved in the DR topology across both sites. It also requires connectivity to internet, to download some python packages required by DRS. You can either:

- Run DRS from one of the SOA nodes.
- Run DRS from another compute instance (OEL 7 or OEL8) in your cloud tenancy. This compute instance can be used to run the DRS tool and removed later, once the DR configuration is done and DRS is not needed anymore.

Consider the DRS SSH access requirements when you choose the host that will run DRS:

- If public networks are used by the SOAMP midtier and db hosts, and the hosts are SSH reachable via their public IP addresses, **DRS can run in any host that can connect via ssh to these public IP addresses.** When you configure the DRS property file prior to run DRS, provide the host’s public IPs.
- If private-only networks are used, so the hosts do not have public IPs, **the host that run DRS needs to be collocated in the same network infrastructure, so it can reach to all the hosts privately** using the cross-site connectivity already configured for the communication between sites. When you configure the DRS property file prior to run DRS, provide the host’s private IPs.

Note that the use of public or private networks for SSH access during DRS execution is a separate consideration from how the primary and secondary sites communicate for DR purposes.

**c) Download and run DRS**

Steps to run the DRS util:

- Download the DRS from [DRS.tar](#), and upload it to the host where the tool will run.
- Extract the contents of this file with the command `tar -xzf drs-mp.tar.gz` and navigate to the ‘drs_mp_soa’ directory it creates.
- Open and review [README.md](#) for instructions and recommendations. Note: it is critical that all specifications are met for successful execution to properly configure your environments.
• Configure the `drs_user_config.yaml` file properties.
• Execute the DRS tool with appropriate parameters.

The DRS tool will automatically perform the required steps to configure secondary SOAMP as standby SOAMP DR site, summarized here:

• It performs initial checks to verify that the environment is prepared for the DR setup.
• It adds the required host alias configurations in the /etc/hosts files on the primary and secondary SOA servers: secondary midtier host names will be added as aliases to primary midtier’s /etc/hosts, and the primary midtier host names will be added as aliases to the secondary midtier’s /etc/hosts file.
• When you use DBFS based method, it recreates the dbfs artifacts in DOMAIN_HOME/dbfs folder, in primary and secondary midtier hosts: it adds the required aliases in the tsnames.ora file (aliases to remote and local CDBs will be used for future domain configuration replication), it recreates the dbfs wallet, and updates the dbfsMount.sh script accordingly. The dbfs mounts are remounted in primary admin node and secondary nodes during the DR setup process.
• If performs a backup of the secondary domain configuration before it modifies it (i.e. /u01/data/domains/soampdrs_domain_backup_<timestamp>.
• It copies the primary domain configuration copy to secondary site: it copies the primary domain configuration to the staging mount (DBFS or FSS), and then, in secondary site, from the staging mount to the domain folder in secondary hosts.
• It performs the replacement of the database connection string after copying the domain configuration from primary to secondary domain: primary database connection string is replaced by secondary database connection string in the secondary domain configuration. Note that only the db connection string is different between primary and secondary domain, because once DR is configured, the secondary domain will point to the same schema names than primary.
• It verifies that the secondary domain is correctly configured for DR. It starts the secondary managed servers in a rolling manner after the DR configuration, using the database in snapshot mode. It checks the connection to the secondary front-end soa-infra url. This verification can be optionally skipped by providing the flag "--do_not_start" when you run DRS.
• During the process, the tool performs some database role conversions in the secondary database (conversions to snapshot standby and back to physical standby).

During execution, the DRS logs to a log file named "logfile_<date-time-stamp>.log". You can monitor setup progress with this file and with the standard output of the process. Once it finishes, it leaves the secondary database in physical standby role and the secondary admin and managed servers stopped.

Your secondary site is now ready for DR!

IMPORTANT: Up to this point, the SOA servers in the secondary location have been pointing to “empty” SOAINFRA schemas with no composites deployed, no policies and no flows pending of execution. Once the secondary location JDBC strings have been updated to point to the same schemas as production per the above steps, the SOA servers in the secondary location will see the same data that the production ones are seeing. If any flows, callbacks, etc. were pending to be executed; the secondary location servers will try to complete those at this point if started. Thus, it is important that instances are drained and completed on the primary site before you convert to snapshot the standby database as already indicated above.

SOA SUITE ON MARKETPLACE DISASTER RECOVERY LIFECYCLE PROCEDURES

Configuration Replication

Any data that resides in the database is automatically replicated to the standby site via the Data Guard: SOA composite deployments, domain and WSM policies, MDS data, SOA runtime data, JMS and TLOGs (as long as they use JDBC persistent stores), and customer data. But most of the configuration of a WebLogic domain resides in the WebLogic domain folder files. When a configuration change is done in the primary WebLogic domain (for example: a new deployment, a new
datasource, a change in a deployment plan, etc.), the change must be replicated to the secondary domain in some way. Two main approaches can be used to maintain matching WebLogic domain configurations in both locations. The applicability of each depends on how frequently this "file-system-resident" configuration is modified:

a) For cases where the SOA domain configuration is **infrequently** altered it is recommended to simply apply the configuration changes manually twice, once in the primary site's domain and once in standby. This requires additional maintenance steps to temporarily convert the secondary database to snapshot standby, start the administration server, make the required change, then shut down the admin server and convert the secondary database back to a physical standby. See details in the table below.

b) For cases where the domain configuration is **modified regularly**, a script that replicates the WLS domain configuration from primary to secondary can be used. The `config_replica.sh` script is provided for this. Depending on the method used during the DR Setup (DBFS or FSS with rsync method), the script will replicate the WLS domain config with the selected method. If the DR setup was done with the DBFS approach, Oracle Database File System (DBFS) will be used to synchronize the configuration using Data Guard. Otherwise, if the FSS with rsync method was used, rsync to remote site will be used to replicate the configuration.

Both approaches described in detail below:

**a) Apply domain configuration changes in both sites**
To maintain the file system configuration synchronized by repeating the config change in the secondary site, follow these steps:

<table>
<thead>
<tr>
<th>STEP</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Apply the configuration change normally in the primary site</td>
</tr>
</tbody>
</table>
| 2 | Convert the standby database to a snapshot standby | Execute these steps as **oracle** user in the primary Database host:  

```
[oracle@drdb a] $ dgmgrl sys/your Sys_password@primary_db_unqname  
DGMGRL> CONVERT DATABASE secondary_db_unqname to SNAPSHOT STANDBY;  
Converting database " secondary_db_unqname" to a Snapshot Standby database, please wait...  
Database " secondary_db_unqname" converted successfully
```
| 3 | Start (if it wasn't started) the WebLogic Administration Server on the secondary site⁹ | Follow the steps in the Oracle Cloud documentation to start the administration server. It is important that ONLY the administration server and not the managed servers is started on the secondary location |
| 4 | Repeat the configuration change in the secondary site | Use the WLS Administration Console in the secondary location to apply the configuration change. Activate the change and verify that the change is working as expected. |
| 5 | Revert the database to physical standby | Execute this steps as **oracle** user in the primary Database host:

```
[oracle@drdb a] $ dgmgrl sys/your Sys_password@primary_db_unqname  
DGMGRL> CONVERT DATABASE secondary_db_unqname to PHYSICAL STANDBY;  
Converting database " secondary_db_unqname" to a Physical Standby database, please wait...  
Oracle Clusterware is restarting database "orclb" ...
```

⁹ Changes to a reduced number of configuration artifacts in SOA and OSB may require the servers to be up in order to be applied; in these cases a start of the managed servers will be needed. Refer to the specific product documentation to identify these artifacts. In this case and if there are pending messages in the database those could be re-executed in the standby location. In such scenarios, Oracle recommend draining/truncating the SOA database schemas in the snapshot database following the SOA standard procedures BEFORE starting the SOA WLS servers.
b) **Using a script to propagate configuration changes**

When the system’s lifecycle involves frequent updates to the domain file system, the process can be automated with a script (see the provided `config_replica.sh` script) that replicates changes via DBFS or FSS with rsync, depending on the method chosen for the DR topology.

**Option 1) DBFS method**

In this approach, a DBFS file system is used as an **assistance file system** where there is a copy of the primary site’s domain configuration. The DBFS filesystem that is automatically configured by default in each SOA MP instance (/u01/soacs/dbfs) is used.

---

**NOTE:** The WebLogic Server domain configuration cannot reside directly on the DBFS mount because that would make the middle tier dependent on the DBFS infrastructure in order to come up (the dependency is not only on the database but also on FUSE libraries, mount points, etc.).

---

The information in this filesystem is automatically replicated to standby location via Data Guard. In the standby site, the DBFS file system can be also mounted, although it is not available unless the standby database is open in read-only mode (when Active Data Guard is used), or when the database is converted to a snapshot standby.

Notice also that the WebLogic Server domain configuration cannot be copied “as is” in this paper’s design since each site in the domain has references to the local DB service in the JDBC connect strings. The configuration has to be modified after it is copied to each site.

The steps of this procedure are as follows:
- The primary WebLogic domain configuration directory contents are copied to the local DBFS file system. Files and folders that are irrelevant or not required (i.e: tmp folders) are excluded.
- The files copied into the DBFS are stored in the database and automatically transferred to the standby database via Data Guard.
- In the standby site, the current production WLS domain configuration files are copied from the DBFS mount to the standby domain folder. Some file modifications are performed during this step (updates the db url to point to the secondary db).

![Figure 9 Replicate domain configuration changes to standby SOA in DBFS method](image)

The advantage of this procedure is that it relies on the robustness of the Data Guard replication to make the config updates available in the standby site. The replica direction is totally consistent with the roles of the sites and it automatically changes when there is a switchover or failover.
You can make a quick and simple validation of this replica method with these steps:

- Verify that the DBFS mount is available in primary soa node1
  ```
  [oracle@soampdrsoa-0 ~]$ df -h | grep dbfs
  dbfs@PDB1:/ 244G 388M 243G 1% /u01/soacs/dbfs_directio
  dbfs@PDB1:/ 244G 388M 243G 1% /u01/soacs/dbfs
  ```
- Write a sample file in primary soa node1 mount
  ```
  [oracle@soampdrsoa-0 ~]$ echo "test" > /u01/soacs/dbfs/share/test.txt
  ```
- Verify that the DBFS mount is available in secondary. This requires that, either the standby database is open in read-only (possible when Active Data Guard is used), or by converting it to a snapshot standby. If DBFS filesystem is not present in secondary once the DB is in read-only or in snapshot mode, you can mount it with the script dbfsMount.sh:
  ```
  [oracle@soampdrsoa-0 ~]$ cd $DOMAIN_HOME/dbfs
  [oracle@soampdrsoa-0 dbfs]$/dbfsMount.sh
  ```
- See if the file appears in secondary site
  ```
  [oracle@soampdrsoa-0 ~]$ ls -la /u01/soacs/dbfs/share/test.txt
  -rw-rw-r-- 1 oracle oracle 5 Mar 27 16:09 /u01/soacs/dbfs/share/test.txt
  ```

**NOTE:** The midtier mounts the dbfs mount by connecting to the local pdb database with a tns alias. This alias is in the $DOMAIN_HOME/dbfs/tnsnames.ora file. This alias is created with a retry parametrization, so in case that there is an issue in connecting to the database during the copy from or to the dbfs mount, these retries will help. The values configured by default (total time of 10 mins, to support a minimum db host reboot) can be adjusted or reduced to meet your specific requirements, if needed. Note that operating system commands that retrieve info from the dbfs filesystem (like “df -h”, or an “ls” in the dbfs mount folders) may take long periods of time to return due to the retries, if the PDB’s service is not reachable.

**Option 2) FSS with rsync method**
The rsync command can be used to replicate the primary site WLS Domain configuration to the secondary site on a regular basis. The steps of this procedure are as follows:

- On the primary site, the domain configuration is synchronized to the local FSS filesystem, then when complete, the local FSS filesystem is synchronized to the remote site’s FSS filesystem.
- On the secondary site, the domain configuration is synchronized from the local FSS filesystem to the WLS domain directory, then environment-specific configuration values are updated (the db url to point to the secondary db).

![Figure 10 Replicate WebLogic domain configuration in FSS with rsync method](image-url)
This procedure can be automated with the provided config_replica.sh script both for the DBFS and FSS with rsync methods. The same script is used in primary and standby, and it is valid for both approaches. It checks the current role of the site and performs the required actions, depending on whether it is the primary or the secondary site and if the method is DBFS or FSS with rsync method.

Follow these steps to use the config_replica.sh script to replicate the WebLogic configuration:

1. The config_replica.sh script has the following communication requirements:
   a. In DBFS method: it requires access from each WebLogic Administration host to the remote Database listener port to perform db role changes (when the script runs in standby role, it needs to convert standby database to snapshot in order to mount the dbfs mount).
   b. In RSYNC method: it requires ssh access from each WebLogic Administration host to the remote WebLogic Administration host (ssh port). It needs to connect to the remote admin host to perform the remote rsync copy.

Make sure you create the appropriate rules to allow this communication. This communication can be done through public IPs (in case that Internet Gateway is used for the connectivity between the sites), or through internal IPs (in case that the sites are connected via Dynamic Routing Gateway, which is the recommended approach).

2. Download the config_replica.sh script and upload it to the primary admin node and to the secondary admin node.

3. Execute the script first in the primary site’s WebLogic Administration host (with oracle user). Monitor the execution and watch for any errors. The script will verify the DG status and will copy the domain configuration from the primary WebLogic domain to the secondary site (via DBFS or via FSS with rsync method).

Syntax is:
```
./config_replica.sh <DR_METHOD> [REMOTE_ADMIN_NODE_IP] [REMOTE_KEYFILE]
```

Where the parameters are:

<table>
<thead>
<tr>
<th>Input Parameter</th>
<th>Details</th>
</tr>
</thead>
</table>
| DR_METHOD         | The method used for the config replica. Valid values are DBFS or RSYNC:
|                   | DBFS: use this value when you use DBFS based method                     |
|                   | RSYNC: use this value when you FSS with rsync method                    |
| REMOTE_ADMIN_NODE_IP | Required if DR_METHOD is RSYNC, omit if DR_METHOD is DBFS.             |
|                   | Specify the IP address of the remote site’s WebLogic Administration host.
|                   | I.e: when you run the script in primary, provide the secondary WLS Administration host’s IP, and when you run the script in secondary, provide the primary WLS Administration host’s IP. |
|                   | Hostname/FQDN must NOT be provided here.                                |
| Remote KEYFILE    | Required if DR_METHOD is RSYNC, omit if DR_METHOD is DBFS.             |
|                   | Is the complete path to the private keyfile required to ssh to remote site’s WebLogic Administration host. Make sure that the file has been uploaded to this host and is readable by oracle user only. |

The script will prompt for the database sys password.

Example to run in the DBFS Method:
```
./config_replica.sh DBFS
```

Example to run in the FSS with rsync method:
```
./config_replica.sh 'RSYNC' '10.1.2.43' '/u01/install/MyKeyWithoutPassPhrase.priv
```
NOTE: You can hardcode these values in the script, there is a section in the script for that. In that case, the sys password value must be set encrypted. To obtain an encrypted value of the sys password, run the following in the primary WLS Administration host (with oracle user):

```bash
cd $DOMAIN_HOME/bin
./setDomainEnv.sh
java -Dweblogic.RootDirectory=<domain_home> weblogic.security.Encrypt
```

Provide the sys password in clear, and it will return the encrypted value. Use the encrypted value when setting the password in the script.

4. Once it completes, execute the script in the secondary site’s WebLogic Administration Server host (with oracle user). Provide the input parameters according to the method used and the site. Monitor the execution and watch for any errors. The script will verify the DG status. As it is the standby, it will copy the domain configuration from the secondary staging filesystem to the secondary WebLogic domain. Then it will make the replacements in the configuration that are required in the standby (it will update the db connect string in datasources).

Normally the secondary WebLogic Administration server is stopped when the changes where replicated. So the changes will take effect next time it is started (during the switchover, failover or open secondary site for validation operations).

In case the WebLogic Administration server is up in secondary location, you need to restart it for the changes to take effect. Note that to start the secondary WebLogic Admin server, it is required to have the secondary DB in snapshot standby mode or use Active Data Guard. Once it has been started, it remains up even if the standby database is converted again to physical standby again (note that standby database must be in physical standby status in normal operation, in order to receive and apply database redo log from primary database via Data Guard).

NOTE: The configuration under `<domain_home>/config` is automatically copied over to all other nodes that are part of the WebLogic domain when the managed servers are restarted and connect to the Administration Server. Any other configuration that resides out of the domain_home/config directory will be copied ONLY to the first node and will have to be manually replicated to each of the managed servers nodes. This includes any customizations to start scripts under domain_home/bin domain_home/security etc.

Furthermore, the script only transfer changes for files under the domain. Any data or files that are created OUTSIDE the domain directory in the Weblogic Administration Server node, are not taken care of by the config_replica.sh script and need to be synchronized separately.

NOTE: For application deployment operations, Oracle recommended to use the WebLogic deployment “Upload your files” option in the WebLogic Administration Console so that the deployed files are placed under the upload directory of the Administration Server (under domain directory/servers/admin_server_name/upload). That way these files will be synced to standby by the DBFS copy script.

Once this initial execution in primary and secondary is complete, the scripts can be added to the cron list in the system (or to any other scheduling tool used by the customer) so that they are executed regularly and/or after a configuration change in primary system. The script must always be run both in primary and standby, first in the primary WebLogic admin host (to copy the domain config to the staging folder) and then in the standby WebLogic admin host (to copy the domain config copy from the staging to the domain folder). Notice that when there is a role change, the script automatically adapts the execution to the new role, because it checks the actual role of the site in order to take one action or other.

Notice that “croning” the copy script automates synchronization but also has the following implications:

- Synchronization may incur in latency as high as the frequency of the cron jobs in both locations added up. i.e. if the cron jobs are set to execute every 30 minutes each, the changes may take 60 minutes to be
available if the window in primary overlaps with the one on the secondary location. Before you perform a switchover, make sure that this amount of time has passed by after the last configuration change. Otherwise, you could switchover before the change is present on standby and overwrite the changes originally applied with the role switch.

- The cron frequency should be set at minimum to the largest amount of time a deployment or configuration change may take to be copied from the domain directory to the dbfs stage directory. Otherwise, copy jobs may overlap.
## Switchover

A switchover is a planned operation where an administrator reverts the roles of the two sites. The roles change from the primary to the standby as well as from standby to primary.

To perform a **manual switchover** in a SOA Suite on OCI Marketplace DR configuration follow these steps:

<table>
<thead>
<tr>
<th>SWITTOVER STEP</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Propagate any pending configuration changes</td>
</tr>
<tr>
<td>2</td>
<td>Stop servers in primary Site</td>
</tr>
<tr>
<td>3</td>
<td>Switchover DNS name</td>
</tr>
</tbody>
</table>
| 4              | Switchover Database | Use DG broker in primary db host to perform the switchover. As user oracle:  
[oracle@drdbwlmpl1 ~]$ dgmgrl sys/your_sys_password@primary_db_unqname  
DGMGR> switchover to “secondary_db_unqname” |
| 5              | Start the servers in secondary site (new primary) | Start the secondary Admin Server (or restart if it was already started, so the configuration changes that were replicated while this was standby take effect.)  
Start secondary managed servers (use the WebLogic Console or scripts) |
Figure 11 SOA Suite on Marketplace disaster recovery AFTER a switchover (DBFS based method)

Figure 12 SOA Suite on Marketplace disaster recovery AFTER a switchover (FSS with RSYNC method)
Failover

A failover operation is performed when the primary site becomes unavailable, and it is commonly an unplanned operation. You can role-transition a standby database to a primary database when the original primary database fails and there is no possibility of recovering the primary database in a timely manner. There may or may not be data loss, depending upon whether your primary and target standby databases were consistent at the time of the primary database failure.

To perform a manual failover in a SOA Suite on OCI Marketplace DR configuration follow these steps:

<table>
<thead>
<tr>
<th>FAILOVER STEP</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Switchover DNS name</td>
<td>Perform the required DNS push in the DNS server that hosts the names used by the system or alter the file host resolution in clients to point the front-end address of the system to the public IP used by LBR in site2. For scenarios where DNS is used for the external front-end resolution (OCI DNS, commercial DNS, etc.), appropriate API can be used to push the change. An example that push this change in an OCI DNS can be found <a href="#">here</a>.</td>
</tr>
</tbody>
</table>
| 2 Failover Database | Use DG broker in secondary db host to perform the failover. As user oracle:  
  `dgmgrl sys/your_sys_password@secondary_db_unqname`  
  DGMGRL> failover to “secondary_db_unqname” |
| 3 Start the servers in secondary site | Start the secondary Admin Server (or restart if it was already started, so the configuration changes that were replicated while this was standby take effect.)  
  Start secondary managed servers (use the WebLogic Console or scripts) |

See [Appendix C – Additional Lifecycle Operations](#) for additional lifecycle operations.
RTO AND RPO OVERVIEW

NOTE: The following values are typical values provided for reference purpose only, and they must NOT be taken as contractual values. These times can be different in the customer’s system, depending on many factors (the application, the connection pool configuration, the host shapes, the load, the tuning, etc.). Notice that there are formal SLA/SLO values in the Oracle Cloud Pillar documents which are the real contractual obligations in terms of availability by Oracle. You can check those here: https://www.oracle.com/assets/paas-iaas-pub-old-srvs-pillar-4021422.pdf

Expected RTO

The Recovery Time Objective (RTO) describes the maximum acceptable downtime should an outage occur for a particular system. The switchover and failover are events that require a downtime, hence, they have an impact on the RTO of the system. The downtime caused by a failover depends on multiple “uncontrollable” factors, because it is normally an unplanned event caused by a critical issue that affects to the system. But it is possible to measure the required downtime for a planned switchover event.

The following table shows typical times taken by each switchover step in sample SOAMP and MFTMP systems. These particular systems taken as examples use VM.Standard2.1 shapes in the SOA/MFT hosts, 8G heap memory size for wls servers in the SOAMP case and 1G heap in the MFTMP case. They use out-of-the-box configuration in the connection pools of the WebLogic servers. The SOAMP system has the “Fusion Order Demo” application deployed (3 composites + 3 applications), and the MFTMP system has over 20 transfer instances deployed.

<table>
<thead>
<tr>
<th>SWITCHOVER STEP</th>
<th>SAMPLE TIMES IN SOAMP</th>
<th>SAMPLE TIMES IN MFTMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Propagate any pending configuration changes</td>
<td>This does not cause downtime</td>
<td></td>
</tr>
<tr>
<td>Downtime starts….</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Stop servers in primary Site</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.1 Stop managed servers</td>
<td>~ 30 sec (Force) / ~2 min (Graceful)</td>
<td></td>
</tr>
<tr>
<td>2.2 Stop Admin server</td>
<td>~ 8 sec (Force) / ~2 min (Graceful)</td>
<td></td>
</tr>
<tr>
<td>3 Switchover DNS name</td>
<td>This is customer specific. For example, if you use OCI DNS it can be as low as 30 sec, but it could take hours depending on the DNS provider used. This can be done in parallel with the rest of the steps.</td>
<td></td>
</tr>
<tr>
<td>4 Switchover Database</td>
<td>~3 min</td>
<td></td>
</tr>
<tr>
<td>5 Start the servers in secondary site</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.1 Start Admin</td>
<td>~3 min</td>
<td>~ 2 min</td>
</tr>
<tr>
<td>5.2 Start managed servers (in parallel)</td>
<td>~5 min</td>
<td>~ 3 min</td>
</tr>
</tbody>
</table>

... Downtime ends

Natural delays between a step and another, or any other additional validation, are not included in the above times, because it depends on how the switchover steps are executed (e.g.: manually, automated with custom scripts, with orchestration custom tools, with Oracle Site Guard, etc). So obviously, some additional time must be considered for the total time, not just the arithmetic sum of the times. The time for DNS switchover is also excluded because it is customer specific.

Normally, the total switchover time is expected to be in the 15-30 min range. Here is a list of tips to minimize the downtime during the switchover operation:
• Perform any switchover related activity that does not require downtime before you stop the primary servers. For example, the WLS configuration replication based on config_replica.sh script does not require downtime, you can perform it while the primary system is up and running. Other example is to start any shutown host in the standby site.
• If possible, stop the managed servers and admin server in parallel.
• If applications/business allows it, use force shutdown to stop the WebLogic servers.
• The max time taken by the WLS servers to shutdown is limited by the parameters "server lifecycle timeout" (normally set to 30 secs) and "graceful shutdown" (normally set to 120 secs). Make sure that these parameters are configured, in order to limit the maximum shutdown time.
• The front-end update in DNS is customer dependant. Use a low TTL value in the appropriate DNS entry (at least during the switchover operation) to reduce the time for update. Once the switchover finished, the TTL can be reverted to its original value.
• Using Data Guard Broker commands (dgmgrl) to switchover the database is faster than using the OCI Console. The RTO can be as low as two (2) minutes. However, the roles of each DB System in the OCI Console UI are not be refreshed automatically. The database switchover with OCI Console automatically refreshes the roles in the OCI Console, but the DB switchover takes longer when performed with the OCI Console.
• The OCI LBR takes some time also to realize that the servers are up and to start sending requests to them. It is usually some seconds, depending on the frequency of the OCI LBR health checks. Lower the interval used for the checks is, faster it realizes that the servers are up. However, be cautious when you use too low intervals: if the healthcheck is a heavy check, it could overload the backend.

Expected RPO

The Recovery Point Objective (RPO) describes the maximum amount of data loss that can be tolerated. In SOA’s case this is especially related to transaction logs, JMS messages and SOA instance information which all resides in the same database. Given that the database and the WebLogic configuration are replicated with different mechanisms, we can differentiate between the RPO for the runtime data and the RPO for the WebLogic configuration.

The actual achievable RPO for the runtime data relies upon the RPO of the database, because the runtime data (composite instances, JMS messages, TLogs, customer data, etc.) are stored in the database. In some cases, there can be runtime artifacts stored in the file systems too (like files consumed by MFT). So the RPO for the runtime data depends upon the following:

• The available network bandwidth and network reliability between primary and standby. When Dynamic Routing Gateway and Remote VCN peering are used to interconnect primary and standby, the Oracle Cloud Infrastructure backbone network is used. The OCI backbone network provides privately routed inter-region connectivity with consistent performance for bandwidth, latency, and jitter when compared to the public Internet (for more information about the network latency between regions, check Inter-Region Latency Dashboard in the console). Using the OCI backbone When DB systems Data Guard is enabled and the OCI network backbone is used, the RPO is up to five (5) minutes. For an optimum behavior, manual configuration of Fast-Start Failover Observer may be required. Refer to the Oracle DB System documentation to configure Observer.

• The Data Guard protection mode used: either Maximum Availability, Maximum Protection or Maximum Performance (default).
  • Maximum Availability mode ensures zero data loss except in the case of certain double faults, such as failure of a primary database after failure of the standby database.
  • Maximum Performance mode offers slightly less data protection than maximum availability mode and has minimal impact on primary database performance.

---

10 Open a Service Request in My Oracle Support to get the DB Systems roles updated in the OCI Console in case they are not automatically refreshed after switching over with dgmgrl commands.
• **Maximum Protection** mode ensures that no data loss occurs if the primary database fails. To ensure that data loss cannot occur, the primary database shuts down, rather than continue processing transactions, if it cannot write its redo stream to at least one synchronized standby database. The best data guard protection mode for a system depends on the business requirements. In some situations, a business cannot afford to lose data regardless of the circumstances. In other situations, the availability of the database may be more important than any potential data loss in the unlikely event of a multiple failure. Finally, some applications require maximum database performance at all times, and can therefore tolerate a small amount of data loss if any component fails. For more information, see the Oracle Data Guard Protection Modes in the Oracle DataGuard documentation.

• If, additionally, there are runtime artifacts stored in file systems that are not located in the database (e.g. files stored in custom File Storage Services, which are consumed or generated by MFT or by File/FTP adapters), the RPO of them depends upon how frequently they are synchronized to the secondary location. What, how and when should this content be synchronized is determined by the business needs. For example: if these runtime files are very volatile (created/consumed fast), syncing it maybe an unnecessary and an overkill. But if the content is more static, and it is required to be have it in secondary in case of a DR event, the frequency to copy it should be according to the expected RPO of the system: the RPO will be the amount of data generated between the replications of this content. Alternatively, these runtime files can be located in a DBFS file system (e.g. the MFT runtime files stored in /u01/soacs/dbfs/share/mft). In that case, they are replicated to standby via the underlying Data Guard replica, so the RPO is provided by the Data Guard protection mode.

The actual achievable **RPO for the WebLogic configuration** depends upon:

• **How frequently** the WebLogic configuration is modified. The WebLogic configuration does not change as dynamically as the runtime data. Despite the initial stages of a system, it is not common to have configuration changes continuously. The more frequently the configuration is modified, the higher amount of config changes could be lost in a disaster event.

• **How frequently** the WebLogic configuration is synchronized to the standby. As described in this document, the WebLogic configuration can be replicated manually or automatically with the config_replica.sh script. One approach is to replicate the configuration after every configuration change that is performed in primary. This ensures that secondary WebLogic configuration is always up-to-date with primary, but requires to include the replication process in every change performed to primary. Another approach is to schedule the replication on a regular basis (e.g. every night). In this case, under a DR unplanned event, the configuration changes performed in primary since the last replication will be lost.

• The **reliability of the procedure** used for the WebLogic configuration replication. Both DBFS and FSS with rsync methods are reliable, but obviously, any failure in the underlying infrastructure (e.g. unavailability of the staging folder, connectivity outages, etc.) can impact on the RPO. Thus, it is recommended to verify the proper functioning of the replication procedure, and to perform regular validations of the secondary site.
BEST PRACTICES

During the lifecycle of a Disaster Recovery topology, Oracle recommends some best practices:

- Use JDBC persistent stores. By default, the JMS persistent stores used by the SOA servers are JDBC stores. In case you create custom persistent stores, be sure that you create them as JDBC persistent stores as well. This way, the JMS messages will be stored in database tables, so this information will be replicated to the standby site via Data Guard.
- Maintain the same patch level in primary and standby sites. The software is not replicated automatically to the secondary site in any tier. If you install a patch in primary, you have to install the same patch in the standby location. When you patch the database, check the specific patch’s documentation on how to apply the patch in a Data Guard topology.
- Maintain the same configuration in primary and standby sites: any changes applied to the primary system that is not part of the WebLogic Configuration (thus, is not replicated with config_replica.sh script) must be performed in the secondary system too, so both primary and secondary systems have the same configuration. For example: a modification in the primary Load Balancer, any modifications to the operating system, etc.
- Perform regular switchovers to verify the health of the secondary site. You can alternatively open the secondary site for validation without performing a complete switchover, as explained in “Appendix C Open Secondary Site for validation”.
- For application deployment operations, Oracle recommends to use the WebLogic deployment “Upload your files” option in the WebLogic Administration Console so that the deployed files are placed under the upload directory of the Administration Server (under domain directory/servers/admin_server_name/upload). That way these files will be synced to standby by the configuration replication script.
- By default, the WLS admin server and managed servers are “auto” started when the soa hosts are rebooted. However, this is not desirable in the standby site. A good practice in the standby site is to use the feature described in https://docs.oracle.com/en/cloud/paas/soa-cloud/soa-marketplace/soamp-disable-server-restart-instance-reboot.html to disable this auto restart. When you set “start_server_on_reboot” to false (in all soa standby hosts), only the nodemanager will be started on the machines boot. The file to set that property (soampRebootEnv.sh) is not overridden during the config replication, so you can have different values in the primary system (expected to have it to true) and in the standby system (you can set it to false). Then, if case you perform a switchover and you plan to use the secondary site as primary for a long time, you can change the values in each site accordingly.
- Perform regular block volume backups, or configure automatic block volume backup, in the block volumes used by the SOAMP hosts, both in primary and standby. See Back Up a Block Volume in the SOA Marketplace documentation for more information.

CONCLUSION

Disaster recovery in an SOA Suite on OCI Marketplace configuration consists of a production database and a standby database synchronized by Oracle Data Guard, two middle tier configurations pointing to their local database, and scripted solution options to manage the minimally necessary file replication. With this Disaster Recovery solution, Oracle Cloud eliminates the costs and complexity of owning and managing a standby hardware, software, and remote data center – while achieving industry-leading Recovery Time Objective and Recovery Point Objective.

The use of Oracle Data Guard for disaster recovery provides better RTO and RPO than restoring a remote backup; production is quickly failed over to an already running and synchronized copy of your production database on the Oracle Cloud. The standby database in the cloud not only provides disaster recovery, it can also be used to seed clone databases for development and test.

The use of middle tiers with a streamlined configuration replication facilitates maintenance and reduces the overhead caused by continuous configuration approaches. However, an appropriate methodology and regular standby verifications are needed to guarantee a consistent recovery. Depending on each system’s lifecycle, different configuration synchronization approaches may be used for optimum behavior.
APPENDIX A – DB SYSTEM BACKUPS ON MANUALLY CONFIGURED DATA GUARD

The back up of the DB System is a key aspect of any Oracle database environment. Oracle Cloud offer various approaches. You can: store backups in local or cloud storage; the backup can be automatic, custom rman, or dbcli. In a DR scenario, there are some special considerations because the databases are configured with Data Guard.

When the Data Guard was configured manually (Option 2 Configuring Data Guard manually) the backup needs to be configured manually in order to get the optimal configuration in a Data Guard environment. You need to perform the backups in one of the databases (primary or standby) and control the archivelog growth in the other one.

To configure manual backups in the primary DB System:

- If the automatic backup was enabled in OCI Console for this system, the backup module should be already configured by the automatic backups. In that case, disable automatic backup so you can customize it. If automatic backup have never been enabled before, you can follow the steps described in Backing Up a Database to Object Storage Using RMAN to install and configure the backup module in the Primary DB.

- Configure rman settings as recommended in the link. In addition to that, ensure that you also include the archivelog deletion policy recommended for Data Guards:

  RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP 1 TIMES TO ‘SBT_TAPE’ APPLIED ON ALL STANDBY;

- Create your rman backup scripts as per your backup requirements and include it in the crontab. This is an example to run a full backup:

  # Run RMAN
  export ORACLE_HOME=/u01/app/oracle/product/18.0.0.0/dbhome_1
  export ORACLE_SID=ORCL
  $ORACLE_HOME/bin/rman <<RMAN
  connect target /
  SET ENCRYPTION ON;
  BACKUP DATABASE PLUS ARCHIVELOG TAG "FULL_BACKUP";
  exit;
  RMAN
  echo "Completed full backup for" $ORACLE_SID

To control the archivelog growth in the standby:

- Disable automatic backup if it was enabled for this system, and then configure the proper archivelog deletion policy so archivelog are not deleted if they are not yet applied to standby with the following command.

  RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY;

- Although setting the correct archivelog deletion policy should be enough to control the archivelog growth in the FRA, you can also create a cleanup script to delete old archive logs. This is an example to clean old archive logs that uses an archivelog deletion policy to prevent undesired archivelog deletion:

  # Use this script to clean old archive logs from disk
  # when the database is in STANDBY role and no backups are performed
  # Run RMAN
  export ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1
  export ORACLE_SID=ORCL
  $ORACLE_HOME/bin/rman <<RMAN
  connect target /
  # To prevent undesired archivelog deletion if this DB takes primary role
  CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY;
  # Delete archivelog older than 20 days
  delete noprompt archivelog all completed before 'SYSDATE-20';
  exit;
  RMAN
  echo "deleted applied old archivelogs on $ORACLE_SID"
When the Data Guard was configured with Cloud Console UI, you can enable automatic backups in the primary database with the Cloud UI Console, and this is a good approach. The default rman configuration in those cases should use the recommended archivelog deletion policy for the Data Guard scenario. However, you can control the archivelog growth in the secondary database as well as explained before.

NOTE: The Oracle DataGuard configuration in the topology should provide protection for most database failure scenarios. i.e. in most cases, should a failure occur in the primary database, switching over to standby will allow to resume operations. In the extreme case where the primary is failing and a switchover to standby is impossible, the primary may need to be restored from a backup. In such an infrequent scenario, the standby database will have to be recreated as well. To recreate the standby database:

In a manual Data Guard, you can re-run the scripts that are provided in the step Option 2) Configuring the Data Guard manually in order to recreate the standby database and reconfigure the Data Guard again after a restore in the primary database.

In an automated Data Guard, however, the OCI Console does not yet provide a feature to recreate the standby database from the UI Console. To restore primary database from a backup, it is required to remove the Data Guard association (which is done by terminating the standby DB System) and re-enable it again once the primary database has been restored. This will create a new standby DB System. Some properties need to be updated in the SOA Suite midtier to reassemble them with this new standby DB system. See the point “Recreating the standby DB System” in the Appendix C – Additional Lifecycle Operations for more details on this.

APPENDIX B – SUMMARY OF NETWORKING REQUIREMENTS FOR DR SETUP

Specific network requirements for SOA Marketplace DR are listed in the following table:

<table>
<thead>
<tr>
<th>ACTION</th>
<th>SSH</th>
<th>SQLNET (1521)</th>
<th>HTTPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DR setup (with DRS)</td>
<td>From the host that runs DRS to all db and midtier hosts, to the IPs set in yaml config file (normally public IPs, but they could be set to private ips when DRS can connect through internal subnets to the nodes). If you the FSS with rsync method for file replication, this also requires connectivity from primary site’s WLS Admin server host to the secondary site’s WLS Admin server host (to private IP if they communicate via Dynamic Routing Gateway, or to public IP if they communicate via Internet. ¹¹</td>
<td>From all secondary site midtier hosts to primary site DB private IP (and scan IPs in case of RAC), when primary and secondary regions communicate via Dynamic Routing Gateway, or From all secondary site midtier hosts to primary site DB public IP (when primary and secondary regions communicate via Internet).¹¹</td>
<td>From the host that runs DRS to the primary site front-end IP. From the host that runs DRS to the secondary site front-end IP. From the host that runs DRS to Internet.</td>
</tr>
<tr>
<td>WLS domain configuration replication via config_replica.sh (FSS with RSYNC method)</td>
<td>From each site’s WLS Admin server host to the other site’s WLS Admin server host (to private IPs if they communicate via Dynamic Routing Gateway, or public IPs if they communicate via Internet. ¹¹</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

¹¹ Cross-site communication via Internet discouraged in latest versions, since OCI allows Dynamic Routing Gateway for private traffic between VCN networks located in different regions.
<table>
<thead>
<tr>
<th>WLS domain configuration replication via <code>config_replica.sh</code> (DBFS based method)</th>
<th>From each site's WLS Admin server host to remote DB private IP (and scan IPs in case of RAC) when primary and secondary regions communicated via Dynamic Routing Gateway. or From each site's WLS Admin server host to remote DB public IP (when primary and secondary regions communicated via Internet).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal runtime</td>
<td>Between primary and secondary site's databases (this is a requirement for Data Guard).</td>
</tr>
</tbody>
</table>
APPENDIX C – ADDITIONAL LIFECYCLE OPERATIONS

Following additional lifecycle operations are explained in this section:

- **Open Secondary Site for validation**
- **Patching the SOAMP DR**
- **Recreate the dbfs wallet**
- **About having compute instances stopped in standby site**
- **Scale-out/in procedures in a SOA Marketplace DR**
- **Reassemble the SOAMP DR after recreating the standby DB System**

Open Secondary Site for validation

It is possible to validate the standby site without performing a complete switchover by converting the standby database to snapshot standby. This allows the secondary SOA servers to be started in the standby site and verify the secondary system. Any change performed in the standby site database while it is in snapshot standby mode will be discarded once it is converted to physical standby again, so primary data will not be affected by secondary site validations.

This operation must be done with caution; however, if there are pending messages or composites in the database when it is converted into snapshot, the standby site’s SOA servers will process them when they start. Check that there are no pending actions in primary database when you convert to snapshot standby, otherwise, remove records from runtime SOA tables in the standby database after you convert it to snapshot standby database and before you start the secondary site’s SOA servers (see Removing Records from the Runtime Tables Without Dropping the Tables).

The steps to validate the standby site without performing a switchover are the following:

<table>
<thead>
<tr>
<th>STEPS TO OPEN THE STANDBY SITE FOR VALIDATIONS</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Convert the standby DB into snapshot standby</td>
<td>Use DG broker in primary db host and convert the secondary to snapshot standby. As user oracle: <code>[oracle@drdbA ~]$ dgmgrl sys/your_sys_password@primary_db_unqname</code> &lt;br&gt;<code>DGMGRL&gt; convert database &quot;secondary_db_unqname&quot; to snapshot standby</code>&lt;br&gt;Use “show configuration” to verify that the conversion has been correctly performed.</td>
</tr>
<tr>
<td>2 Verify that there are no pending actions in the secondary environment</td>
<td>If there were pending actions (transactions, messages) in the primary DB when the standby is converted to snapshot, the secondary soa servers will try process them when they start. &lt;br&gt; You can use the soa truncate script to remove the records from the SOA runtime tables in secondary database to clean the runtime data before you start the secondary servers. See Removing Records from the Runtime Tables Without Dropping the Tables. &lt;br&gt;Run this action with caution, do not truncate tables in primary DB.</td>
</tr>
<tr>
<td>3 Start the servers in the secondary site</td>
<td>Start the secondary admin server. Example &lt;br&gt;<code>$ cd /u01/app/oracle/middleware/oracle_common/common/bin</code>&lt;br&gt;<code>$ ./wlst.sh</code>&lt;br&gt;<code>wlst&gt; nmConnect ('weblogic','acme1234#','soampdr-dsoa-0',&quot;5556&quot;,&quot;soampdrss_domain&quot;,&amp;u01/data/domains/soampdrss_domain&quot;,&quot;SSL&quot;)</code>&lt;br&gt;<code>wlst&gt; nmStart('soampdrss_adminserver')</code>&lt;br&gt;Start secondary managed servers (use the secondary WebLogic Console or scripts)</td>
</tr>
<tr>
<td>4 Validate</td>
<td>Note: As this is not a swichover and the primary site is still active, the virutal front-end name will resolve to the primary site’s LBR IP address, so any browser access will, by default, be redirected to the active primary site.</td>
</tr>
</tbody>
</table>
In order to directly access the secondary site’s SOA services, you must update the /etc/hosts file in a controlled client (laptop, etc.) and set the virtual front-end name to resolve to the secondary site’s front-end LBR IP address, and run any validation from this client.

NOTE: verify that the client used for validations does not access the SOAMP system via an HTTP proxy, because the HTTP proxy may continue to resolve the virtual front-end name with the primary site’s LBR IP address regardless of which name is in the /etc/hosts of the client.

NOTE: Non-linux clients may require a reset of their local DNS cache before a browser will resolve the IP address with the customized host file entry.

NOTE:
ORA-01403: no data found ORA-06512 errors. While you validate the secondary site as described here (without performing a complete switchover, i.e. just opening standby in snapshot standby mode) “ORA-01403: no data found ORA-06512” errors may show up in the logs of the standby soa servers. These error are related to the SOA auto purge job. These errors arise because jobs in the database may have db role dependencies (they are defined to be enabled only when the database is in primary role). This is an expected and desired behavior that prevents jobs from being executed twice (once in primary and once in standby). The soa auto purge job is defined with primary role, so it is not shown in DBA_SCHEDULER_JOBS view when the database is in snapshot standby mode. The database_role defined for each job can be seen in the view DBA_SCHEDULER_JOB_ROLE. In summary, these errors can be ignored as long as they appear in the standby system. The scheduler job for SOA auto purge will be executed on the DB if and only if the instance changes its role to PRIMARY.

Once the secondary site has been validated, follow these steps to revert it back to standby role again:

<table>
<thead>
<tr>
<th>STEPS TO REVERT BACK STANDBY TO STANDBY ROLE</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Stop managed servers and admin servers in secondary</td>
<td>You can connect to secondary WebLogic Console and shutdown managed servers and Administration server in secondary site.</td>
</tr>
<tr>
<td>2: Convert the standby DB into a physical standby again</td>
<td>Use DG broker in primary db host and convert the secondary to physical standby again. As user oracle: [oracle@drdbA ~]$ dgmgrl sys/your_sys_password@primary_db_unqname DGMGRL&gt; convert database “secondary_db_unqname” to physical standby Use “show configuration” to verify that the conversion has been correctly performed.</td>
</tr>
<tr>
<td>3: Revert any updated client’s /etc/hosts</td>
<td>If you updated any /etc/hosts file in a client in order to point to secondary site, revert it back so the virtual front-end name points to primary front-end IP again.</td>
</tr>
</tbody>
</table>
Patching the SOAMP DR environment

These are the guidelines to apply patches to the Oracle software in a SOAMP DR system. A Disaster Recovery topology helps (in some cases) to reduce the patching downtime:

- **Database patches**
  SOAMP DR topology uses Data Guard. The advantage Data Guard instead of only a primary database, is that you can first patch one site and then the other. But not all the database patches allow this approach. The downtime and procedure to patch the database depends on each patch. The database patches can be:
  - Data Guard Standby-First. These can be applied first in standby and then in primary. Various options possible. See "Oracle Patch Assurance - Data Guard Standby-First Patch Apply (Doc ID 1265700.1)"
  - Non Data Guard Standby-First. These kinds of patches require to be applied on both primary and standby databases at the same time and require shutdown.

So, if the patch is standby first applicable, the downtime can be minimized or reduced to a switchover. If not, it requires shutdown of primary and standby and apply in both.

- **Midtier only patches (that patch only midtier bits)**
  Some of these can be marked as FMW_ROLLING_ORACLE_HOME in the readme. In that case, they do not require downtime, regardless of you have DR or not. However, normally they are not FMW_ROLLING_ORACLE_HOME and require midtier shutdown. For those cases, a Disaster Recovery topology helps, you can:
  1. Convert secondary database to snapshot standby.
  2. Patch the secondary midtier domain first.
  3. Test the secondary domain with the patch.
  4. Once everything is validated on secondary, convert secondary database back to physical standby.
  5. Switchover to secondary (at this point secondary region becomes your primary and runs the business).
  6. Convert old primary database to snapshot.
  7. Patch old primary and test it.
  8. Convert database back to physical standby.
  9. Then switchback to original site.

**So, the downtime is only the time of the switchover procedure time.**

If you didn't have DR, your downtime would include the patching time, plus the time to stop and start the system.

- **Midtier patches that include db schema changes**
  Same than before, if these are not FMW_ROLLING_ORACLE_HOME and due to db schema changes require to patch midtier and db at the same time, the approach is a bit different than before to not lose db changes. With DR you can:
  1. Convert secondary database to snapshot standby.
  2. Patch the secondary midtier domain first.
  3. Test the secondary domain with the patch.
  4. Once everything is validated on secondary, convert secondary database back to physical standby. At this point, secondary WebLogic domain is misaligned: the midtier has one version but the schemas are in the older version.
  5. Patch primary.

So, the downtime is the same than without DR, but with the advantage that you can verify the patching and the procedure (i.e., identify issues and verify the patching procedure itself) in standby.
Recreate the dbfs wallet

NOTE: This applies to DBFS based method only.

The dbfs wallet, tnsnames.ora and dbfsMount.sh (in <domain_home>/dbfs folder) of the midtier hosts are updated during the DR setup12. In case you need to update the wallet because the password of the SchemaPrefix_DBFS user has been changed, you can follow the steps described in the section “Update the Wallet Password Manually” in Change the Database Schema and Wallet Passwords in SOA Marketplace documentation, but taking into account that the alias used to mount dbfs in a DR environment is the PDB name (instead of the default “ORCL” alias). Hence, in the commands to generate the wallet, you should use the PDB name. Example:

```
$middleware_home/oracle_common/bin/mkstore-wrl /u01/data/domains/domain_name/dbfs/wallet -create < /var/tmp/dbfsp
$middleware_home/oracle_common/bin/mkstore-wrl /u01/data/domains/domain_name/dbfs/wallet -createCredential <PDB_NAME>
```

Make sure that the dbfsMount.sh script in <domain_home>/dbfs folder uses the <PDB_NAME> alias in the dbfs client mounts commands. Example:

```
$ORACLE_HOME/bin/dbfs_client -o wallet /@<PDB_NAME> -o direct_io $MOUNT_PATH_DIRECTIO &
$ORACLE_HOME/bin/dbfs_client -o wallet /@<PDB_NAME> $MOUNT_PATH &>dbfs.log &
```

Make sure also that the alias <PDB_NAME> exists in the <domain_home>/dbfs/tnsnames.ora pointing to the local PDB.

To mount the dbfs mounts, it is recommended to use the script <domain_home>/dbfs/dbfsMount.sh instead of using dbfs_client commands directly. In case you use dbfs_client commands, make sure you use the correct alias.

Note that the wallet recreation after a password change in SchemaPrefix_DBFS user is required to be done both in primary and standby host midtiers, because the folder <domain_home>/dbfs is specific to each domain (and it should not be replicated from primary to standby).

NOTE: in order to update the rest of the schema passwords (SOAINFRA, STB, etc), you can simply follow the steps described in Change the Database Schema Password Manually in primary domain, and then use config_replica.sh to replicate changes to secondary domain. Any password change in the datasources and other files under the domain configuration will be replicated to secondary.

About having compute instances stopped in standby site

The standby database should not be shutdown during normal business operation, because it will not receive updates from primary and it will become out-of-sync. This can result in a data loss in case a switchover needs to be performed. Furthermore, unresolvable gaps in redo between the primary and secondary database may require a full reinstantiation and configuration of the physical standby. It is hence recommended to avoid long periods of disconnection between the primary and standby database. This includes scenarios where the secondary is stopped or problems at the network level that could prevent the communication between the two sites during normal business operations.

The standby midtier compute instances can be stopped without affecting primary, but it has the following implications on the disaster recovery:

- Impact in RPO: the domain configuration changes that are replicated from the primary site will not be pushed to the standby domain configuration if the standby admin server host is stopped. In case of a failover, the standby domain can be out-of-sync from the primary configuration.

---

12 Note that these updates are performed during DR setup in the primary admin node host and in all the secondary midtier hosts. The rests of the primary midtier hosts keep the original dbfs wallet, tnsnames.ora and dbfsMount.sh in <domain_home>/dbfs folder, because only primary admin node is used to copy the primary configuration to dbfs mount. Once the DR setup is done, you can homogenize this by copying these files from primary admin node host to the rests of the primary managed server hosts.
• Impact in RTO: the recovery time is increased if the secondary midtier hosts are stopped and need to be started. Recovery time is further increased as the domain configuration synchronization would then need to be executed in the secondary domain to apply any primary site domain configuration changes before a switchover or failover.

To minimize these implications, maintenance operations time and effort when want to have some secondary SOA compute instances stopped, minimally keep the secondary site’s WLS Administration compute instance up and shut down only the other WLS managed server compute instances.

NOTE:
Customer billing conditions are out-of-scope of this document. To confirm the impact on your billing of having some servers stopped, contact your Oracle license team in order to get confirmation about your billing conditions.

Note that in all the cases, stopping an instance using the instance’s OS does not stop billing for that instance. If you stop an instance this way, be sure to also stop it from the Console or API. The billing of the stopped compute instances will normally follow the OCI compute model. In SOAMP, all the compute shapes are supported, so the billing of stopped instances depend on the compute shape.

Scale-out/in procedures in a SOA Marketplace DR

It is possible to scale-out and scale-in a SOA Marketplace system following the steps described in the SOA Marketplace documentation Scale an Oracle SOA Suite Instance Cluster Out or In.

When you perform a scale-out or scale-in a SOA Marketplace DR environment, there are some characteristics specific to a DR environment that must be considered: there are 2 SOA Marketplace instances (primary and secondary) and the domain configuration in secondary is a copy of the primary configuration, so it uses primary hostnames as listen-addresses.

When the listen-address hostnames are added as aliases in the midtier’s /etc/hosts, the new nodes provisioned during a scale-out operation do not include these aliases in its /etc/hosts file by default. This can cause the scale-out procedure to fail in the secondary location, because the new nodes cannot connect to WLS administration server. To avoid this problem during scale-out of the WLS for OCI DR environment, required steps are documented in this point.

When you added the primary hostnames entries to a DNS private view in secondary, as described in https://github.com/oracle-samples/maa/tree/main/private_dns_views_for_dr, the scale-out procedures are simplified, because any new node is able to resolve the primary hostnames as soon as it is created.

See the following points for detailed steps.

Scale-out

The recommended procedure to SCALE-OUT a SOA MP DR environment is as follows:

a) **Scale-out primary** SOA Marketplace instance:
   1. Stop any periodic scheduled execution of the config_replica.sh.
      IMPORTANT: DO NOT run config_replica.sh replication to secondary until the secondary it is scaled-out also. If secondary SOA system configuration has a weblogic server node that is not recognized by secondary servers (secondary SOA will not have an equivalent node until it is scaled-out), the startup of the soa-infra will fail in secondary. See “About having different number of nodes in primary and standby”.
   2. Follow the steps described in Scale Out an Oracle SOA Suite Instance Cluster in primary stack.
   3. Once the scale-out has finished correctly, connect with ssh to the new node and:
      a. Edit /etc/hosts to add the front-end FQDN with primary front-end LBR IP address. Example:

         ```
         # Front-end virtual host for DR, poiting to primary front-end IP
         111.111.111.111 soampdrs.mycompany.com
         ```
      b. **(Not needed if you are using the DNS private view approach for hostname aliases)**
         Edit /etc/hosts in the new node and add the aliases that already exist in the rest of primary nodes, that include secondary names. Example:

         ```
         10.0.0.82 <prim_midtier1_fqdn> <prim_midtier1_hostname> <sec_midtier1_fqdn> <sec_midtier1_hostname>
         10.0.0.81 <prim_midtier2_fqdn> <prim_midtier2_hostname> <sec_midtier2_fqdn> <sec_midtier2_hostname>
         ```
      c. **(Not needed if you are using the DNS private view approach for hostname aliases)**
         Edit the /etc/oci-hostname.conf and set PRESERVE_HOSTINFO to 3 so these changes are persisted across reboots.
   4. Restart the new managed server
b) **Scale-out secondary** SOA Marketplace instance:

Scaling-out the secondary requires intervention before the scale-out. Remember that the WebLogic domain configuration in the standby is a copy from primary and it uses the primary hostnames as listen addresses for the servers. The new node that is added to secondary when scaling-out is not aware of them (aliases of the primary names are not included by default in the /etc/hosts file of the new node). To allow the scale-out in the secondary to finish successfully, previous to proceed with the scale-out, modify the listen-addresses in the secondary domain and set there the secondary soa hostnames. This makes that the scale-out procedure run without issues. Later, once the scale-out is finished, the aliases will be added too as in option 1. Detailed steps explained here:

1. **Convert the standby database into snapshot standby.**
2. **(Not needed if you are using the DNS private view approach for hostname aliases)** Change the listen address of the weblogic servers in the secondary domain to use the secondary instance's fully-qualified domain names instead the primary instance fully-qualified domain names. This change will be reverted later, it is needed because the new added node will not have in the /etc/hosts the aliases for the primary servers that are set in the config.xml. For this:
   - **Identify primary soa hosts FQDN** (the existing nodes previous to the scale-out). Example:
     
     soampdr6-soa-0.mysubnet1.myregion1vcn.oraclevcn.com
     soampdr6-soa-1.mysubnet1.myregion1vcn.oraclevcn.com

     Primary midtier1 fqdn is soampdr6-soa-0.mysubnet1.myregion1vcn.oraclevcn.com, and its hostname is soampdr6-soa-0.
     Primary midtier2 fqdn is soampdr6-soa-1.mysubnet1.myregion1vcn.oraclevcn.com and its hostname is soampdr6-soa-1.

   - **Identify secondary soa hosts FQDN** (the current existing nodes). Example:
     
     soampdr6-soa-0.mysubnet2.mysubnet2.myregion2vcn.oraclevcn.com
     soampdr6-soa-1.mysubnet2.mysubnet2.myregion2vcn.oraclevcn.com

     NOTE: hostnames are expected to be the same in primary and secondary soa hosts, only the fqdn values will differ.

   - In the secondary site's admin server node, replace primary instance's FQDN with the secondary instance's FQDN in the `<DOMAIN_HOME>/config/config.xml` file:

     ```
     cd <DOMAIN_HOME>/config/
     cp config.xml config.xml_backup_pre_scale-out
     sed -i 's/primary_midtier1_fqdn/secondary_midtier1_fqdn/g' config.xml
     sed -i 's/primary_midtier2_fqdn/secondary_midtier2_fqdn/g' config.xml
     ```

     Example:

     ```
     sed -i 's/soampdr6-soa-0.mysubnet1.myregion1vcn.oraclevcn.com/soampdr6-soa-0.mysubnet2.myregion2vcn.oraclevcn.com/g' config.xml
     ```

     ```
     sed -i 's/soampdr6-soa-1.mysubnet1.myregion1vcn.oraclevcn.com/soampdr6-soa-1.mysubnet2.myregion2vcn.oraclevcn.com/g' config.xml
     ```

3. **Start admin and managed servers in the secondary site.**
   NOTE that starting secondary managed servers must be done carefully. If there are pending messages, or composites in the standby database, the servers may process them. Check that there are not pending actions in primary database when converting to snapshot standby or remove records from runtime soa tables in the snapshot standby database before starting the secondary servers.13

---

13 Removing Records from the Runtime Tables Without Dropping the Tables
4. Follow the steps described in Scale Out an Oracle SOA Suite Instance Cluster in secondary stack to add a node.
5. Once the scale-out process finishes, add the required aliases in the new added node:
   • Edit /etc/hosts in the new node and add the front-end FQDN for the secondary front-end LBR IP address, as it is in the rest of the secondary nodes.

   ```python
   # Front-end virtual name for DR, poiting to secondary front-end IP
   222.222.222.222 soampdrs.mycompany.com
   ```

   • (Not needed if you are using the DNS private view approach for hostname aliases)
     Edit /etc/hosts in the new node and add the existing aliases that secondary midtier nodes already have, where the primary node FQDN are aliases of the secondary local IP addresses.

   ```python
   10.2.0.12 <sec_midtier1_fqdn> <sec_midtier1_hostname> <prim_midtier1_fqdn> <prim_midtier1_hostname>
   10.2.0.11 <sec_midtier2_fqdn> <sec_midtier2_hostname> <prim_midtier2_fqdn> <prim_midtier2_hostname>
   ```

   Edit the /etc/oci-hostname.conf and set PRESERVE_HOSTINFO to 3 so these changes are persisted across reboots.
6. Stop servers in secondary site (managed servers and admin).
7. Convert the standby database to physical standby.
8. (Not needed if you are using the DNS private view approach for hostname aliases)
   Optionally, you can revert the change done in step 2 and set again the primary FQDN in the listen addresses. Alternative, this will be automatically done later when you replicate the conf from primary using config_replica.sh (step d).

c) Once both primary and standby are scaled out, complete configuration by adding the aliases for the new node to all the midtier hosts (existing and new nodes).

   If you are using the /etc/hosts approach for the hostname aliases:
   1. In primary, add it to ALL the existing primary midtier nodes (and also in the new one). Example (this mus be in a single line):

      ```python
      <primary_newnode_ip> <primary_newnode_fqdn> <primary_newnode_hostname> <secondary_newnode_fqdn>
      <secondary_newnode_hostname>
      ```

   2. In secondary, add it to ALL the existing midtier nodes (and also in the new one). Example (this mus be in a single line):

      ```python
      <secondary_newnode_ip> <secondary_newnode_fqdn> <secondary_newnode_hostname> <primary_newnode_fqdn>
      <primary_newnode_hostname>
      ```

   If you are using the DNS private view approach, you can just add the hostnames of the new nodes to the appropriate DNS private views instead of adding them to the /etc/hosts. I.e.: the name of the new secondary node to the primary private view, and the name of the new primary node to the secondary private view.

d) At this point, run the config_replica.sh immediately (as usually, first in primary and then in secondary) to propagate the configuration from primary to standby. You can now enable any periodic scheduled execution of the config_replica.sh.

## Scale-in

The recommended procedure to SCALE-IN a SOA MP DR environment is the following:

a) Stop any periodic scheduled execution of the config_replica.sh.

   IMPORTANT: DO NOT run config_replica.sh replication to secondary until the secondary it is scaled-in also.
b) Scale-in primary SOA Marketplace instance:
   1. Follow the steps described in Scale In an Oracle SOA Suite Instance Cluster in primary stack.
c) Scale-in secondary SOA Marketplace instance:
   2. Convert the standby database into snapshot standby
   3. Start the admin server only (starting managed servers is not required)
   4. Follow the steps described in Scale In an Oracle SOA Suite Instance Cluster in secondary stack.
   5. Once finished, convert secondary database to physical standby.

d) Remove the aliases of the deleted node from the /etc/hosts in primary and secondary midtier hosts, or from the DNS private views if you are using that approach.
e) (optional) Run the config_replica.sh (first in primary and then in secondary) to propagate the configuration from primary to standby and verify secondary. You can now enable any periodic scheduled execution of the config_replica.sh.
About having different number of managed servers in primary and standby

Oracle strongly recommends to have the exact same resources (number of nodes, memory, etc.) in primary and standby SOA systems, and in case of scaling-out/in primary location, proceed with the same action in secondary as described previously. Having different number of nodes can cause issues at the functional and performance levels. For example, if primary is scaled-out from 2 to 3 nodes, and that configuration is replicated to standby where there are 2 nodes only, the soa-infra will not start in secondary because there is a new node that is unknown for secondary location (not resolvable because it does not exist any equivalent node in secondary site). There can be errors like the following:

```
<May 18, 2020 10:55:48.394 AM GMT> <Error> <Deployer> <BEA-149231> <Unable to set the activation state to true for the application "soa-infra".

weblogic.application.ModuleException: java.net.UnknownHostException: soampdr6-soa-2.mysubnet1.myregion1vcn.oraclevcn.com
```

If you face this scenario due to a human error or a recovery situation, as a work-around you can add a “fake” alias in the secondary soa hosts for the node that exists in primary but not in secondary, so the existing servers can start. The fake alias would point to a non-existing IP address (or the IP address of the secondary db could be used). This would allow the soa-infra to start in the existing secondary servers. Although the “new” node in secondary does not exists and won’t be contacted, the “unknownhostexception” error will not happen, and the soa-infra application will be able to start in the existing nodes. Note that you should not try to scale-out this secondary domain to add a new node in this situation, because it is not consistent status (it has the new server in the configuration but there is no real host for it). The correct way to recover from this inconsistent situation would be to switch over back to original primary and scale-in it to make it consistent with the secondary number of nodes again, and then run the config_replica.sh replication to replicate primary config to secondary that will now have the same number of nodes.

As a summary, **having different number of configured servers in primary and secondary can cause inconsistencies hence it is not recommended.**

Reassemble the SOAMP DR after recreating the standby DB System

There are a few scenarios where the standby DB system may need to be completely recreated. For example, if the primary DB System is restored from a backup, the OCI Console does not yet provide a feature to recreate the standby database from the UI Console. To restore primary database from a backup, it is required to remove the Data Guard association (which is done by terminating the standby DB System) and re-enable it again once the primary database has been restored. This operation will create a new standby DB System.

In SOAMP DR environments, when you re-enable DG in primary DB system to re-create the Standby DB System, **Oracle recommends to provide the same values for the standby DB System that it had before** (same VCN, same subnet, same hostname prefix). This way, minimal changes are required in the SOAMP DR systems in order to use this new DB System as the standby DB.

Follow the steps described below to **reassemble the SOAMP DR with a new standby DB system:**

- **a)** Note down the DB unique name ($ORACLE_UNQNAME), private and public IP, VCN, subnet, and hostname prefix of the original standby DB System that is going to be terminated.
- **b)** Once the standby DB System has been terminated, review the /etc/hosts file in the primary DB System host(s). If there is any entry for the terminated standby DB host(s), delete or comment it. A new entry for the standby DB host(s) will be added automatically when it is created.
- **c)** When you re-enable DG in the primary DB System using OCI Console, make sure you provide the **same VCN, same subnet, same hostname prefix than the previous standby DB System** was using, With this, the only different values in the new standby DB System vs the previous standby DB System will be the DB unique name, the private IP and the public IP.
- **d)** Once the new DB System has been successfully created and the Data Guard configuration is completed in the OCI Console, note down the following values of the new standby DB system: **DB unique name and Private IP.**

- **e)** In the **standby SOAMP** hosts:
  - Edit the file `/u01/data/domains/local_CDB_jdbcurl.nodelete` and update the standby DB uname with the new standby DB unique name.
  (Only if DBFS based method is used):
  - Edit the file `$DOMAIN_HOME/dbfs/localdb.log`. It contains the DB unique name of the original standby System. Replace it with the DB unique name of the new standby DB System.
• Edit the file `$DOMAIN_HOME/dbfs/tnsnames.ora`. It contains a few aliases. One of the aliases is the original standby DB System unique name. Replace the original standby DB unique name with the new standby DB unique name, in the alias and in the service name of the alias.

f) In the primary SOAMP hosts:
(Only if DBFS based method is used):  
• Edit the file `$DOMAIN_HOME/dbfs/tnsnames.ora`. It contains a few entries. One of the aliases is the original standby DB System unique name. Replace the original standby DB unique name with the new standby DB unique name (in the alias and service name), and replace the original standby IP with the new standby IP.  
  Note that the aliases in tnsnames.ora for the standby CDB may be different in primary and standby soa hosts. In primary, the standby IP is used to point to secondary CDB, while in standby soa hosts the standby hostname is used. This is expected behavior, because it is not expected to have DNS resolution cross-regions.
• No need to update the localdb.log in primary soa hosts, as it contains the primary unique name and this has not changed.

g) Also, verify that any existing OCI security rule created for the original Standby DB System specific IPs is updated to use the new Standby DB System IPs (this is only needed if the rules were specific to the IPs instead of to the CIDRs).

SOAMP DR environment is now ready to use the new Standby DB System!

As an example, let us assume the following values:

<table>
<thead>
<tr>
<th>DB unique name ($ORACLE_UNQNAME)</th>
<th>Original Standby DB System</th>
<th>New Standby DB System</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORCL6_phx1kg</td>
<td>ORCL6_phx1c3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DB System private IP</th>
<th>Original content</th>
<th>New content</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2.0.2</td>
<td>10.2.0.5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DB System hostname</th>
<th>Original content</th>
<th>New content</th>
</tr>
</thead>
<tbody>
<tr>
<td>drdb6b.mysubnet.region2vcn.oraclevcn.com</td>
<td>&lt;same value&gt;</td>
<td>&lt;same value&gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DB System scan name</th>
<th>Original content</th>
<th>New content</th>
</tr>
</thead>
<tbody>
<tr>
<td>drdb6b-scan.mysubnet.region2vcn.oraclevcn.com</td>
<td>&lt;same value&gt;</td>
<td>&lt;same value&gt;</td>
</tr>
</tbody>
</table>

• Hence, in the standby SOAMP hosts:

<table>
<thead>
<tr>
<th>File to update</th>
<th>Original content</th>
<th>New content</th>
</tr>
</thead>
<tbody>
<tr>
<td>/u01/data/domains/local_CDB_jdbcurl.nodelete</td>
<td>drdb6b-scan.mysubnet.region2vcn.oraclevcn.com:1521/ORCL_phx1kg.mysubnet.region2vcn.oraclevcn.com</td>
<td>drdb6b-scan.mysubnet.region2vcn.oraclevcn.com:1521/ORCL_phx1c3.mysubnet.region2vcn.oraclevcn.com</td>
</tr>
<tr>
<td>(only if DBFS based method is used)</td>
<td>ORCL6_phx1kg</td>
<td>ORCL6_phx1c3</td>
</tr>
<tr>
<td>$DOMAIN_HOME/dbfs/localdb.log</td>
<td>(DESCRIPTION = (SDU=65536)(RECV_BUF_SIZE=10485760)(SEND_BUF_SIZE=10485760)(ADDRESS = (PROTOCOL = TCP)(HOST = drdb6b-scan.mysubnet.region2vcn.oraclevcn.com)(PORT = 1521)) (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = ORCL6_phx1kg.))</td>
<td>(DESCRIPTION = (SDU=65536)(RECV_BUF_SIZE=10485760)(SEND_BUF_SIZE=10485760)(ADDRESS = (PROTOCOL = TCP)(HOST = drdb6b-scan.mysubnet.region2vcn.oraclevcn.com)(PORT = 1521)) (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = ORCL6_phx1c3.))</td>
</tr>
<tr>
<td>(only if DBFS based method is used)</td>
<td>ORCL6_phx1c3</td>
<td>ORCL6_phx1c3</td>
</tr>
<tr>
<td>$DOMAIN_HOME/dbfs/tnsnames.ora</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
And in the **primary SOAMP hosts**: 

<table>
<thead>
<tr>
<th>File to update</th>
<th>Original content</th>
<th>New content</th>
</tr>
</thead>
<tbody>
<tr>
<td>(only if DBFS based method is used)</td>
<td>... ORCL6_phx1kg = (DESCRIPTION=(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(ADDRESS=(PROTO COL=TCP)(HOST=10.2.0.2)(PORT=1521))(CONNECT_D ATA=(SERVER=DEDICATED)(SERVICE_NAME=ORCL6_phx1kg. mysubnet.region2vcn.oraclevcn.com)(UR=A))) ...</td>
<td>ORCL6_phx1c3 = (DESCRIPTION=(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(ADDRESS=(PROTO COL=TCP)(HOST=10.2.0.5)(PORT=1521))(CONNECT_D ATA=(SERVER=DEDICATED)(SERVICE_NAME=ORCL6_phx1c3. mysubnet.region2vcn.oraclevcn.com)(UR=A))) ...</td>
</tr>
<tr>
<td>$DOMAIN_HOME/dbfs/tnsnames.ora</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
APPENDIX D – DISASTER RECOVERY BASED ON BLOCK VOLUME CROSS-REGION REPlication

Topology Description
The DR solution based on Block Volume Cross-Region replication is, in most of the aspects, the same as described previously in this paper. The topology is the same and only the way in which configuration is replicated varies. The following are the key aspects of the topology.

It is an active-passive model. There is a primary system consisting of a SOA Suite on Marketplace deployment, load balancer, and Oracle Cloud Infrastructure DB system in one region, and a standby system, consisting of a SOA Suite WLS domain, load balancer, and Oracle Cloud Infrastructure DB system in a different region. Different regions must be used for primary and standby to take advantage of the Block Volume Cross-Region replication.

The primary and standby Oracle Cloud Infrastructure DB Systems are configured with Data Guard. Relying on Data Guard features, all the changes applied to primary database are replicated to secondary database (which acts as the “standby” database).

The standby SOA Suite on Marketplace domain is a replica of the primary domain, using the same name, schemas, passwords, etc. but pointing to the secondary database. The listener addresses of the WebLogic Servers are configured with the primary midtier host names, so in secondary midtier hosts the pertinent aliases are created in the hosts file to resolve them with the secondary IPs.

On the front-end, there is a unique name configured to access the applications running in the system. This “virtual” front-end name will point to the IP of the OCI Load Balancer of the primary site. In case of a switchover, this front-end name is updated to point to the IP of the OCI Load Balancer of the secondary site. It always must point to the LBR IP of the site that has the primary role in each time.

In normal business operation, the standby database is a physical standby. It is either in the mount state, or opened in read-only mode when Active Data Guard is used. The standby database receives and applies redo from primary, but cannot be opened in read-write mode. For some actions, during the DR setup and lifecycle steps described in this document, the standby database will be converted from a physical standby to a snapshot standby. A database in snapshot standby mode is fully updateable database. A snapshot standby database receives and archives, but does not apply, the redo data from a primary database. All the changes performed to a snapshot standby database are discarded when it is converted again into a physical standby.

Like in the other approaches, all the information that resides in the database is automatically replicated to the secondary site by Data Guard. This includes: SOA schemas, OPSS information, custom schemas, TLOGs, JDBC persistent stores, etc. The WebLogic Domain configuration, located on the local filesystem in each site, needs to be replicated from the primary site to the secondary as well.
The difference versus the other methods described in this paper, is how the WebLogic Domain configuration is replicated from primary to standby. In this case, it will be transferred using the **Block Volume Cross-region replication** feature. This capability allows you to perform ongoing automatic asynchronous replication of block volumes and boot volumes to other regions. See [Cross-Region Volume Replication](#) in Oracle documentation for more information. This can be used for Disaster Recovery scenarios like the topology described here.

![SOAMP Disaster Topology diagram using Block Volume cross-region replication](image)

**Advantages and Disadvantages of BV Replication DR model**

The **advantages of a Block Volume replication approach as compared to** the other methods described in this whitepaper can be summarized as follows:

- **It is a general-purpose solution:**
  The DR solution based in block volumes replication is more agnostic to the specific system paticularities. It is generally applicable to other systems, apart from SOAMP, and can be used to replicate any other compute instance’s block volumes in the system.

- **It uses a continuous and dissatended replica process:**
  The replicaton of the block volumes is a dissatended and continuous process performed by OCI infrastructure. It is not based on a script that needs to be manually run or scheduled.
• **The replication is not limited to the domain configuration:**
The information of the replicated block volumes is an exact copy from the primary block volumes. Not only the admin host WLS domain folder configuration is replicated, all the info residing in that block volume, and in the block volumes of the rest of the nodes will be replicated. This means that other custom files that are located outside the domain (but in the block volume that is replicated) are also automatically replicated without need additional intervention.

There are, however, some disadvantages as compared to the other methods described in this whitepaper that need to be considered:

• **Higher Complexity:**
Block Volume Cross-region Replication management may be complex when the number of block volumes replicated is high. It requires a good lifecycle management of the block volume and replica. Switchover and failover operations are also more complex than in the other methods, and there are additional pre and post switchover/failover steps.

• **Additional Cost implications:**
After you enable replication for a volume, the volume will be replicated in the specified region and availability domain. Your bill will include storage costs for the volume replica in the destination region. The volume replica in the destination region is billed using the Block Storage Lower Cost option price, regardless of the volume type in the source region.

Your bill will also include any applicable network costs for the replication process between regions. As part of the replication process, all data being updated on the source volume is transferred to the volume replica, so volumes with continual updates incur higher network costs.

See point “Cost Considerations for Cross-Region Replication” in the Oracle documentation[Cross-Region Volume Replication].

• **The amount of data replicated across regions is much higher:**
In the other methods described in this whitepaper (DBFS and FSS with RSYNC), the WebLogic Domain configuration that is replicated is minimal, and only from the admin server domain. In this case, the complete block volume where the WebLogic Domain resides is replicated from primary and secondary, and for all the nodes. This may be considered an advantage when you require to replicate additional artifacts or files. It is a trade-off between the customer needs and the cost.

• **BV replication based DR's RTO is worse:**
When using Block Volume replication for configuration replication the switchover and failover require additional steps (activate replicas, attach block volumes, etc.) that increment the downtime during a switchover or a failover.

For a normal switchover of a 2 nodes cluster, the time is increased in about 15 minutes, and there are post-switchover tasks (detach block volumes, etc) that take about another 15 minutes more (although, this last set of steps can be applied without affecting the recovery time objective).

• **Worse control on RPO** (referring to WebLogic Configuration, because the RPO for the database is exactly the same than in other methods)

The Block Volume replication process is continuous, with the typical Recovery Point Object (RPO) target rate being less than an hour. However, depending on the change rate of data on the source volume, the RPO can vary. For example, the RPO can be greater than an hour for volumes with a large amount of write I/O operations to the volume. In the others methods described in this paper, as the information that is replicated using a script and the amount of information replicated is less, the user can have a finer control on the RPO for the WebLogic Configuration.

**Assumptions for BV Replication DR model**

**Load Balancer**
Same as described in Assumptions > Load Balancer for DBFS and FSS with RSYNC models.

**Database**
Same as described in Assumptions > Database for DBFS and FSS with RSYNC models.

**Block Volumes Replicated**
The BV Replication DR solution assumes that only the Block Volumes will be replicated to the other site, and that the Boot Volumes are not replicated. Each mid-tier host of a SOA Suite on Marketplace has one block volume attached, mounted
This volume is used to store the WebLogic domain configuration in each compute instance. Hence, only the content of that block volume needs to be replicated to the other site.

Any other content outside the /u01/data folder resides in the Boot Volume and will not be replicated. The Operating System and the Oracle software homes are stored in the Boot Volume, hence, they will not be replicated. If the OS or Oracle products are patched in primary mid-tier hosts, the same patching procedure must be performed in the secondary SOAMP compute instances.

Requirements for BV Replication DR model

Front-end address

Same as described in Requirements > Front-end address for DBFS and FSS with RSYNC DR models.

Instance Name Prefix

Same as described in Requirements > Instance Name Prefix for DBFS and FSS with RSYNC models.

Network communication between sites

Same as described in Requirements > Network communication between sites, the primary and standby databases need to communicate with each other over their listener port for redo transport. The communication between primary and secondary sites can use Oracle Cloud’s internal networks, by using Dynamic Routing Gateway, which is the recommended approach (refer to the Dynamic Routing Gateway documentation for additional details on the network configuration). The communication between sites can also happen over an Internet Gateway (Oracle Net’s traffic is encrypted), but this is not recommended for security and reliability reasons.

In the Block Volume Replication Cross-Replication DR model, no additional network communication is needed: cross-region rsync is not used, and connectivity from mid-tiers to remote database is not needed.

Regions with Block Volume Cross-Region Replication

The primary and standby regions used in the DR topology must be different, and the Block Volume Cross-Region Replication must be available between them. Not all the regions are interconnected for Block Volume Cross-Region replica. The source region for the volume to replicate determines the target regions available to select as destination region. Check the table that lists the source region and target regions available for volume replication in Oracle Cloud documentation link Cross-Region Volume Replication > Replication Target Regions.
Setup Process for BV Replication DR model

The setup procedure for the BV Replication DR model is a variation of the setup process described previously in this document for DBFS and FSS with RSYNC methods, as described in the following flow chart:

As in the other models, it is assumed that, as starting point, the primary site, consisting of an OCI DB system, a SOA Suite on Oracle Cloud Marketplace system, and the Load Balancer (OCI LBR) is live. The secondary DR configuration, residing in a geographically remote site, will be created for this existing primary system. Since the primary system may already be running in production, the DR configuration process is designed to cause minimum downtime (only the modification of the front-end address requires WebLogic server restarts).

This is a summary of the steps for the setup process based on BV replica:
1. Choose a virtual front-end name
2. Prepare Primary mid-tier for the virtual front-end
3. Setup the Database in Secondary Site
4. Provision SOA Suite on Marketplace in Secondary Site
5. Prepare Secondary mid-tier for the virtual front-end
6. Convert the standby DB into physical standby
7. Create the mid-tier hosts aliases
8. Configure the Block Volume Cross-region replication
9. Prepare the script that performs the environment specific replacements
10. Verify the DR Setup

From step 1 to step 5, both included, the steps are exactly the same than explained before in SOA Suite on Marketplace Disaster Recovery. The rest of the steps are specific for the BV replication DR model and explained in the following points.

1. Choose a virtual front-end name
   Same than in SOA Suite on Marketplace Disaster Recovery Setup > 1. Choose a virtual front-end name

2. Prepare Primary mid-tier for the virtual front-end
   Same than in SOA Suite on Marketplace Disaster Recovery Setup > 2. Prepare primary mid-tier for the virtual front-end

3. Setup the Database in Secondary Site
   Same than in SOA Suite on Marketplace Disaster Recovery Setup > 3. Setup the Database in Secondary Site

4. Provision SOA Suite on Marketplace in Secondary Site

5. Prepare Secondary mid-tier for the virtual front-end
   Same than in SOA Suite on Marketplace Disaster Recovery Setup > 5. Prepare Secondary mid-tier for the virtual front-end

6. Convert the standby DB into physical standby
   At this point, the WebLogic servers in the secondary must be stopped and the standby Database can be converted in to physical standby again:
   a) Stop Oracle processes in secondary mid-tier hosts
      Stop the WebLogic Managed Servers, the Admin server and the Node Manager processes in all the Secondary mid-tier hosts.
      Also, in case they are mounted, umount the DBFS mounts in all Secondary mid-tiers hosts.
   b) Convert the standby database into physical standby again
      Execute these steps as oracle user in the primary Database host:

      ```
      [oracle@drdbaa ~]$ dgmgrl sys/your_sys_password@primary_db_unqname
      DGMGRL> CONVERT DATABASE secondary_db_unqname to PHYSICAL STANDBY;
      Converting database "secondary_db_unqname" to a Physical Standby database, please wait...
      Oracle Clusterware is restarting database "orclb" ...
      Continuing to convert database "secondary_db_unqname" ...
      Database "secondary_db_unqname" converted successfully
      ```

7. Create the mid-tier hosts aliases
   This step is also required in the other methods explained previously in this document (DBFS and FSS with RSYNC methods). However, in that case this step is automatically performed by the DRS tool. When using the Block Volume replication model, you need to manually add the required aliases in the mid-tier hosts.

   The hostnames that are used as listen addresses for the WLS servers in primary need to be valid in the secondary location, but mapping to the secondary IPs. And the other way around: the hostnames used as listen address for the WLS servers in secondary need to be valid in the primary location but mapping to the primary IPs.
7.1. Option 1) Use the /etc/hosts files

The other site's hostnames are added as aliases to the /etc/hosts files in the mid-tier hosts.

This mode is valid in all the scenarios: when the same DNS server is used in primary and secondary sites, and when separated DNS servers are used in primary and secondary. Because the entries in the /etc/hosts file have precedence over the DNS resolution. This precedence is defined in the directive “hosts” of the /etc/nsswitch.conf. By default, it is set to “files”, which means that /etc/hosts resolution takes precedence over the DNS.

A disadvantage of this method is that it requires to manually add the entries to all the SOA hosts. So when you add new "files", which means that /etc/hosts resolution takes precedence over the DNS. This precedence is defined in the directive "hosts" of the /etc/nsswitch.conf. By default, separated DNS servers are used in primary and secondary. Because the entries in the /etc/hosts file have precedence over

To configure the required alias:

a) **First of all**, in all the mid-tier hosts (primary and standby), edit the file `/etc/oci-hostname.conf` as root user and set `PRESERVE_HOSTINFO=3`, so the changes implemented in next steps in the /etc/hosts are preserved after node reboots.

b) **Identify the hostnames** of each WebLogic hosts in primary and standby where the server listen. To get them, you can look for the listen addresses in the domain configuration file. For example, in primary domain:

```
[oracle@soampdrsoa-0 config] cd $DOMAIN_HOME/config
[oracle@soampdrsoa-0 config] grep listen-address config.xml
<listen-address>soampdrsoa-0.mysubnet1.myregion1vcn.oraclevcn.com</listen-address>
<listen-address>soampdrsoa-0.mysubnet1.myregion1vcn.oraclevcn.com</listen-address>
<listen-address>soampdrsoa-0.mysubnet1.myregion1vcn.oraclevcn.com</listen-address>
...
```

And in the secondary domain:

```
[oracle@soampdrsoa-0 config] cd $DOMAIN_HOME/config
[oracle@soampdrsoa-0 config] grep listen-address config.xml
<listen-address>soampdrsoa-0.mysubnet2.myregion2vcn.oraclevcn.com</listen-address>
<listen-address>soampdrsoa-0.mysubnet2.myregion2vcn.oraclevcn.com</listen-address>
<listen-address>soampdrsoa-0.mysubnet2.myregion2vcn.oraclevcn.com</listen-address>
...
```

You can also use the command “`hostname --fqdn`” in each primary midtier host to get its hostname.

c) **Edit the /etc/hosts** (as root) in all the primary mid-tier nodes and add the hostnames of standby as aliases of the primary hosts. Each host should have entries as the following:

```
<IP_prim_node1> <long_and_short_hostnames_primary_node1> <long_and_short_hostnames_secondary_node1> <long_and_short_hostnames_secondary_node2>
<IP_prim_node2> <long_and_short_hostnames_primary_node2> <long_and_short_hostnames_secondary_node2>
```

(shortnames are expected to be the same in primary and stby)

Example of the resulting entries /etc/hosts in primary mid-tier hosts:

```
[oracle@soampdrsoa-0 config] more /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
:1 localhost localhost.localdomain localhost6 localhost6.localdomain6
# Frontend
111.111.111.111 mysoampdr.mymycompany.com
# Aliases for SOAMP DR in primary site
10.0.2.10 soampdrsoa-0.mysubnet1.myregion1vcn.oraclevcn.com soampdrsoa-0.mysubnet2.myregion2vcn.oraclevcn.com
10.0.2.11 soampdrsoa-1.mysubnet1.myregion1vcn.oraclevcn.com soampdrsoa-1.mysubnet2.myregion2vcn.oraclevcn.com
```

d) In the same way, edit the /etc/hosts (as root) in all the secondary mid-tier nodes and add the hostnames from primary as aliases of the secondary hosts. Each host should have entries as the following:

```
<IP_secondary_node1> <long_and_short_hostnames_secondary_node1> <long_and_short_hostnames_prim_node1>
<IP_secondary_node2> <long_and_short_hostnames_secondary_node2> <long_and_short_hostnames_prim_node2>
```
Example of the resulting `/etc/hosts` in secondary mid-tier hosts:

```bash
[oracle@soampdr-soa-0 ~]$ more /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost6
:1 localhost6 localhost.localdomain6
# Frontend
222.222.222.222 mywsoampdr.mycompany.com
# Aliases for SOAMP DR in secondary site
10.1.2.5 soampdrs-soa-0.mysubnet2.myregion2vcn.oraclevcn.com
soampdrs-soa-0.mysubnet1.myregion1vcn.oraclevcn.com
10.1.2.4 soampdrs-soa-1.mysubnet2.myregion2vcn.oraclevcn.com
soampdrs-soa-1.mysubnet1.myregion1vcn.oraclevcn.com
```

7.2. Option 2) Use OCI Private DNS views

Instead of adding the entries to all `/etc/hosts`, you can add the required entries to private DNS views of each VCN. Adding the entries to the private DNS views has advantages for the scale-out operations, because the new nodes in secondary site will be able to resolve the primary names out-of-the-box. Scale-out operations are simplified with this approach.

Check the Github repository https://github.com/oracle-samples/maa/tree/main/private_dns_views_for_dr for detailed instructions and terraform scripts.

8. Configure the Block Volume Cross-Region replication

To replicate the block volumes of the mid-tier hosts from primary region to secondary region, you must follow these steps:

- **a) Identify the Block Volumes of the primary mid-tier hosts**
  - To identify these block volumes in primary region:
    - Go to OCI Console, select your Primary region
    - Navigate to Storage > Block Volumes
    - Choose the compartment of your primary SOA Marketplace and look for the block volumes.
    - The Block Volumes attached to the SOA Marketplace compute instances are named as `<soamp_prefix>-block-N`, where N is 0,1,2, etc., corresponding with the compute instance `<soamp_prefix>-soa-0`, `<soamp_prefix>-soa-1`, etc. Note down the names, the OCIDs and the AD where they are located. For example:
      - soampdrs-block-0, with OCID ocid1.volume.oc1.eu-london-1.abtheljs...htgbuhmlg2s3avjradeoirz3q
      - soampdrs-block-1, with OCID ocid1.volume.oc1.eu-london-1.abthelj....cdcauywhph6gw2irsri4rnx7q

  These Block Volumes are mounted in each SOA hosts in `/u01/data`. Example:

  ```bash
  [oracle@soampdrs-soa-0 ~]$ df -h
  Filesystem Size Used Avail Use% Mounted on
  devtmpfs 7.2G 0 7.2G 0% /dev
  tmpfs 7.2G 0 7.2G 0% /dev/shm
  tmpfs 7.2G 65M 7.2G 1% /run
  tmpfs 7.2G 0 7.2G 0% /sys/fs/cgroup
  tmpfs 7.2G 39G 9.0G 30G 24% / → this is boot volume, will not be replicated
  tmpfs 200M 8.6M 192M 5% /boot/efi → this is boot volume, will not be replicated
  tmpfs 1.5G 0 1.5G 0% /run/user/0
  tmpfs 1.5G 0 1.5G 0% /run/user/994
  tmpfs 1.5G 0 1.5G 0% /run/user/1000
  tmpfs 49G 1.5G 46G 4% /u01/data → this Block Volume will be replicated
  tmpfs 200G 128K 200G 1% /u01/soacs/dbfs_directio
  tmpfs 200G 128K 200G 1% /u01/soacs/dbfs
  ```

- **b) Identify the Block Volumes of the secondary mid-tier hosts**
  - Repeat the steps described in a) to get the names, OCID and Availability domains of the block volumes of the secondary mid-tier hosts.

- **c) Enable the Cross-Region replication in the Primary Block Volumes**
  - For each Block Volume of the primary SOA, enable the cross-region replica:
    - Edit the Block Volume, set the Cross Region Replication to ON
    - Region: this is the destination for the replica. Select the secondary region.
- **Availability Domain:** select the AD where the equivalent Block Volume is located in secondary region. Example: if soampdhrs-block-0 in secondary region is located in AD-3, select AD-3 as the location for the replica of the primary soampdhrs-block-0.
- **Name:** to facilitate future automation using OCI CLI, Oracle recommends to use the same name for the block volume replica object regardless in which region the replica is. Use for example: "prefix-block-N_replicated_from_remote_site"
- After saving changes, check that the replicas are being created in the **secondary region:** in the OCI Console, select the secondary region and navigate to **Storage > Block Storage > Block Volume Replicas.**

**NOTE:** If your SOA compute instances reside in the same Availability Domain, you can create a Volume Group and activate the replica for the group instead of activating it for each block volume individually.

### d) Detach the original Block Volumes from the Secondary mid-tiers hosts

**NOTE:** Boot Volumes must NOT be unmounted or detached

For each mid-tier host in Secondary, run the following:
- Unmount the block volume, which is mounted in /u01/data:

  ```bash
  [opc@soampdhrs-soa-0 opc]# sudo umount /u01/data
  ```

  Make sure that there are not Oracle processes running. It is expected that they are stopped at this point, but if there is something still running on that folder, the umount will fail.
- Once unmounted, detach the block volume from the OCI Console

  Go to each **Block Volume > Attached Instances > Detach from Instance**

  The OCI Console will ask you to run some iscsicmd commands before completing the detachment.
- With root user, **edit the /etc/fstab file** and remove the entry for /u01/data. This is to prevent it from trying to mount the original BV in next reboot. Example:

  ```bash
  # Remove this entry:
  #UUID=765185db-a10e-4da4-bff0-0437348b3cf6 /u01/data ext4 auto,defaults,_netdev,nofail 0 2
  ```

  Repeat these steps for the rest of the mid-tier nodes in Secondary.

### e) Delete or rename the detached Block Volumes in Secondary

The original block volumes that have been detached from the secondary mid-tier hosts in the previous step are not going to be used anymore. You can delete them or rename.

### 9. Prepare the script for the environment specific replacements

During a switchover or failover operation, after mounting the replicated block volumes in the secondary mid-tier hosts, you need to perform some replacement on the WebLogic Domain configuration. This is because the WebLogic Domain configuration is a copy from primary, so the DB connection url used in the datasources and other files points to primary database. You need to replace it with the secondary database connection url.

To automate these replacements, you can use the provided script **replacement_script_BVmodel.sh.**

a) Download the script from [https://github.com/oracle-samples/maa/raw/main/wls_mp_dr/scripts_BV_replica_model/replacement_script_BVmodel.zip](https://github.com/oracle-samples/maa/raw/main/wls_mp_dr/scripts_BV_replica_model/replacement_script_BVmodel.zip) and upload it to all the mid-tier hosts (primary and secondary).

b) **Store it in a folder** that is NOT in the Block Volume that is replicated. You can store it, for example, in a folder under the oracle user’s home (for example, /home/oracle/scripts).

c) Change the ownership of the file to **oracle user** (this script will be executed by oracle user).

d) Edit the script and customize it in each mid-tier host with the appropriate values, by providing the local and remote values for the database in each site.

**DO not run** the script at this point. The script will be used next time that a switchover or failover is performed.
10. Verify the DR Setup
At this point, the DR environment setup is completed. Oracle recommends to immediately validate that the DR setup is correct, by performing a complete Switchover (see point Switchover in next page) or, alternatively, by opening the secondary site for validation (see point Open Secondary Site for Validation in next page).
Lifecycle Operations for BV Replication DR model

Switchover

A switchover is a planned operation where an administrator reverts the roles of the two sites. The roles change from the primary to the standby as well as from standby to primary. To perform a manual switchover from Site1 to Site2 in a SOA Suite on OCI Marketplace DR configuration based on Block Volume Cross-Region replication, follow these steps:

a) Pre-Switchover tasks:

These steps do not cause downtime and are required only if they were not performed as post-steps in the last switchover.

- If still attached, detach original or previously used block volumes from the mid-tier hosts in Site2 (umount, detach, and comment the entry from the /etc/fstab)
- Delete or rename the detached volumes in Site2 to prevent from mounting them by mistake. They will not be used anymore.

b) Switchover

The actual switchover procedure starts at this point:

<table>
<thead>
<tr>
<th>SWITCHOVER STEP</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Stop servers in primary Site</td>
</tr>
<tr>
<td>2</td>
<td>Switchover Virtual Front-end DNS name</td>
</tr>
</tbody>
</table>
| 3               | Activate the replicas in Site2 | Until this point, the BV are being continuously replicated from Site1 to Site2. In order to mount the replicated BV in Site2, the BV replicas of Site 2 need to be activated. When you activate a BV replica, an “attachable” BV is created as a clone from the replicated BV. Then, these cloned BV can be attached to the compute instance (note that the entities “block volume replica”, which are shown in BLOCK VOLUMES > BLOCK VOLUME REPLICAS, cannot be attached to a compute instance.)
   - To activate the replicas in Site2, connect to OCI Console
     - Go to Site2. Block Volumes > Block Volume Replicas
     - Click in each BV replica and "Activate"
     - In the replica name, Oracle recommends to use the same name regardless the region where they are created. Preferably, use the same name than the original Block Volume. This will facilitate automations using OCI CLI. Example: “prefix-block-N”.
     - Repeat for all the BV replicas in Site2. |
| 4               | Attach the replicated BV to Site2 mid-tier hosts | The attachable Block Volumes created as a result of the activation must be shown in Site2, in OCI Console > Storage > Block Volume
   - For each mid-tier host in Site2:
     - Attach the appropriate block volume to the host. You can do this in OCI Console, in Compute > Instances > (click on the instance) > Attached block Volumes > Attach Block Volume
     - Once attached, click con “iSCSI Commands & Information” of the attached block volume and run the iscsi commands provided in “Commands for connecting” in the mid-tier host.
   - Once iscsi commands are run, get the UUID of the new attached block volume:
```bash
[root@soampdr-soa-0 opc]# sudo blkid /dev/sda3: UUID="974147f5-d731-41de-bba8-56ff78ed1c9c" TYPE="xfs" PARTUUID="4a95c68a-bc70-4be9-bce8-b5e9955c46b" /dev/sda1: SEC_TYPE="msdos" UUID="593B-B893" TYPE="vfat" PARTLABEL="EFI System Partition" PARTUUID="c5ac3089-6a91-40e0-bcc1-212ba0b45418" /dev/sda2: UUID="9ca12da-d7ea-44a2-8680-5bd76488b054" TYPE="swap"
```
<p>|</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Run the replacement script in Site 2 mid-tier hosts</td>
</tr>
<tr>
<td></td>
<td>Run the script <code>replacement_script_BVmodel.sh</code> in ALL the Site2 mid-tier hosts.</td>
</tr>
<tr>
<td>6</td>
<td>Switch database</td>
</tr>
<tr>
<td></td>
<td>Use DG broker in primary db host to perform the switchover. As user oracle:</td>
</tr>
<tr>
<td></td>
<td>[oracle@drdbwlmpla ~]$ dgmgrl sys/your_sys_password@primary_db_unqname</td>
</tr>
<tr>
<td></td>
<td>DGMGRL&gt; switchover to &quot;secondary_db_unqname&quot;</td>
</tr>
<tr>
<td>7</td>
<td>Start the servers in Site2 (new primary)</td>
</tr>
<tr>
<td></td>
<td>Start the nodemanager in all the secondary servers. Example:</td>
</tr>
<tr>
<td></td>
<td>$ cd $DOMAIN_HOME/bin/</td>
</tr>
<tr>
<td></td>
<td>$ nohup ./startNodeManager.sh &gt; $DOMAIN_HOME/nodemanager/nodemanager.out 2&gt;&amp;1 &amp;</td>
</tr>
<tr>
<td></td>
<td>Start the secondary admin server. Example:</td>
</tr>
<tr>
<td></td>
<td>$ cd /u01/app/oracle/middleware/oracle_common/common/bin</td>
</tr>
<tr>
<td></td>
<td>$ ./wlst.sh</td>
</tr>
<tr>
<td></td>
<td>wlst&gt; nmConnect ('weblogic','acme1234#','soampdrsoa-0','5556','soampdrsoa-0:/u01/data/domains/soampdrsoa_domain','SSL')</td>
</tr>
<tr>
<td></td>
<td>wlst&gt; nmStart('soampdrsoa_adminserver')</td>
</tr>
<tr>
<td></td>
<td>Start the managed servers (use the secondary WebLogic Console or scripts)</td>
</tr>
</tbody>
</table>
During the switchover procedure, the Block Volume Replicas are activated in the standby site.

c) **Post-Switchover tasks:**
At this point, the services are active in the Site2 hence no additional downtime is required. However, there are additional tasks needed to complete the switchover procedure and leave the system in the appropriate role-reversed state. Oracle recommends running them immediately as following:

- Enable the Block Volume Replication in the other way (from Site2 to Site1), in the block volumes of the SOA hosts in Site2. Make sure you provide the appropriate Availability Domain of Site 1 for the replicas.
- Disable the replica in the Site1 block volumes.
- Detach the block volumes from Site1 mid-tier hosts to prepare them for the future switchback (umount, comment the entry in fstab, run the iscsi commands to disconnect and detach).
- Delete or rename the detached volumes from the Site1 mid-tier hosts to prevent from mounting them by mistake. They will not be used anymore.
NOTE: When a compute instance is rebooted, it takes a bit more time to mount the new block volumes than the time it took for mounting the original block volumes. This can make that, for the time by when the WLS startup script runs on the compute instance boot (the script `/opt/scripts/restart/restart_12c_servers.sh`), the `/u01/data` mount is not yet available, so the node manager and the wls processes are not automatically started. If that is the case, you can run the WLS startup script manually. As a workaround, you can introduce a delay before the `restart_12c_servers.sh` execution, by adding the command “sleep 90” to the `/etc/rc.local`, before the line that runs the script.

**Failover**

A failover operation is performed when the primary site becomes unavailable, and it is commonly an unplanned operation. You can role-transition a standby database to a primary database when the original primary database fails and there is no possibility of recovering the primary database in a timely manner. There may or may not be data loss depending upon whether your primary and target standby databases were consistent at the time of the primary database failure.

Failover steps are the same than switchover. The only difference is how the Database role change is performed. In a failover, you need to connect to the standby DB and run the failover command instead of the switchover command:

```
[oracle@drdbwlmp1b ]$ dgmgrl sys/your_sys_password@secondary_db_unqname
DGMRGRL> failover to "secondary_db_unqname"
```

The rest of the steps are the same than in a switchover, including pre and post steps.
Open Secondary Site for Validation

In this model, it is also possible to validate the standby site without performing a complete switchover by converting the standby database to snapshot standby. This allows the secondary SOA servers to be started in the standby site and verify the secondary system. Any change performed in the standby site database while it is in snapshot standby mode will be discarded once it is converted to physical standby again, so primary data will not be affected by secondary site validations.

This operation must be done with caution, however; if there are pending messages or composites in the database when it is converted into snapshot, the standby site’s SOA servers will process them when they start. Check that there are no pending actions in primary database when converting to snapshot standby, otherwise, remove records from runtime SOA tables in the standby database after it is converted to snapshot standby database and before starting the secondary site’s SOA servers (see Removing Records from the Runtime Tables Without Dropping the Tables).

The steps to validate the standby site (Site2 in this case) without performing a switchover are the following:

<table>
<thead>
<tr>
<th>STEPS TO OPEN THE STANDBY SITE FOR VALIDATIONS</th>
<th>DETAILS</th>
</tr>
</thead>
</table>
| 1 Convert the standby DB into snapshot standby | Use DG broker in primary db host and convert the secondary to snapshot standby. As user oracle:  
[oracle@drdbA ~]$ dgmgrl sys/your_sys_password@primary_db_unqname  
DGMGRL> convert database “secondary_db_unqname” to snapshot standby  
Use “show configuration” to verify that the conversion has been correctly performed. |
| 2 Verify that there are no pending actions in the secondary environment | If there were pending actions (transactions, messages) in the primary DB when the standby is converted to snapshot standby, the secondary soa servers will try process them when they start.  
You can use the soa truncate script to remove the records from the SOA runtime tables in secondary database to clean the runtime data before starting the secondary servers. See Removing Records from the Runtime Tables Without Dropping the Tables  
Run this action with caution, do not truncate tables in primary DB. |
| 3 Activate the replicas in Site2 | Until this point, the BV are being continuously replicated from of Site1 to Site2. In order to mount the replicated BV in Site2, the BV replicas of Site 2 need to be activated.  
When you activate a BV replica, an “attachable”BV is created as a clon from the replicated BV.  
Then, these cloned BV can be attached to the compute instance (note that the entities “block volume replica”, which are shown in BLOCK VOLUMES > BLOCK VOLUME REPLICAS, cannot be attached to a compute instance.)  
To activate the replicas in Site2, connect to OCI Console  
- Go to Site2. BLOCK VOLUMES > BLOCK VOLUME REPLICAS  
- Click in each BV replica and “Activate”  
- In the replica name, Oracle recommends to use the same name regardless the region where they are created. Preferably, use the same name than the original Block Volume. This will facilitate automations using OCI CLI. Example: “prefix-block-N”.  
- Repeat for all the BV replicas in Site2. |
| 4 Attach the replicated block volumes to mid-tier hosts in Site2 | The attachable Block Volumes created as a result of the activation must be shown in Site2, in OCI Console > Storage > Block Volume  
For each mid-tier host in Site2:  
- Attach the appropriate Block Volume to the host. You can do this in OCI Console, in Compute > Instances > (click on the instance) > Attached block Volumes > Attach Block Volume  
- Once attached, click con “ISCSI Commands & Information” of the attached BV and run the iscsci commands provided in “Commands for connecting” in the mid-tier host.  
- Once iscsci commands are run, get the UUID of the new attached BV:  
[root@soamprd-soa-0 opc]$ sudo blkid  
/dev/sda3: UUID=“974147f5-d751-41de-bba8-56ff78ed1c9c” TYPE=xfs  
PARTUUID=“4a95c68a-bc70-4be9-bce8-b5e995cf4d6”  
/dev/sda1: SEC_TYPE=“msdos” UUID=“593B-B893” TYPE=vfat PARTLABEL=“EFI System Volume” |
Add an entry for the appropriate UUID in `/etc/fstab`, to mount and persist the mount after reboots.

```
UUID=35e72262-979a-4d84-85ce-a6f91e3b1250 /u01/data ext4 auto,defaults,_netdev,nofail
```

NOTE: after the first switchover, this UUID will not change for each BV, so you can just comment/uncomment the entry in `/etc/fstab`, depending on the role of the Site.

- Mount the new attached BV in the `/u01/data` and verify that it is correctly mounted.

```
[root@soampdrsoa-0 opc]# mount -a
[root@soampdrsoa-0 opc]# grep /u01/data
/dev/sdb 49G 1.4G 46G 3% /u01/data
```

Repeat the steps for all the mid-tier hosts in Site 2.

5. Run the script that makes the replacements in Site 2 mid-tier.

Run the script `replacement_script_BVmodel.sh` in all the Site2 mid-tier hosts.

6. Start the servers in the Site2.

Start the nodemanager in all the secondary servers. Example:

```
$ cd $DOMAIN_HOME/bin/
$ nohup ./startNodeManager.sh > $DOMAIN_HOME/nodemanager/nodemanager.out 2>&1
```

Start the secondary admin server. Example:

```
$ cd /u01/app/oracle/middleware/oracle_common/common/bin
$ ./wlst.sh
```

```
wlst> nmConnect ('weblogic','acme1234#','soampdrsoa-0','5556','soampdrsoa_domain','/u01/data/domains/soampdrsoa_domain','SSL')
wlst> nmStart('soampdrsoa_adminserver')
```

Start secondary managed servers (use the secondary WebLogic Console or scripts)

7. Validate.

Note: As this is not a swichover and the primary site is still active, the virtual front-end name will resolve to the primary site’s LBR IP address, so any browser access will, by default, be redirected to the active primary site.

In order to directly access the secondary site’s SOA services, you must update the `/etc/hosts` file in a controlled client (laptop, etc.) and set the virtual front-end name to resolve to the secondary site’s front-end LBR IP address, and run any validation from this client.

NOTE: verify that the client used for validations does not access the SOAMP system via an HTTP proxy, because the HTTP proxy may continue to resolve the virtual front-end name with the primary site’s LBR IP address regardless of which name is in the `/etc/hosts` of the client.

NOTE: Non-linux clients may require a reset of their local DNS cache before a browser will resolve the IP address using the customized host file entry.
NOTE:
ORA-01403: no data found ORA-06512 errors. While validating the secondary site as described here (without performing a complete switchover, i.e. just opening standby in snapshot standby mode) “ORA-01403: no data found ORA-06512” errors may show up in the logs of the standby soa servers. These error are related to the SOA auto purge job. These errors arise because jobs in the database may have db role dependencies (they are defined to be enabled only when the database is in primary role). This is an expected and desired behavior that prevents jobs from being executed twice (once in primary and once in standby). The soa auto purge job is defined with primary role, so it is not shown in DBA_SCHEDULER_JOBS view when the database is in snapshot standby mode. The database_role defined for each job can be seen in the view DBA_SCHEDULER_JOB_ROLE. In summary, these errors can be ignored as long as they appear in the standby system. The scheduler job for SOA auto purge will be executed on the DB if and only if the instance changes its role to PRIMARY.

Once the secondary site has been validated, follow these steps to revert it back to standby role again:

<table>
<thead>
<tr>
<th>STEPS TO REVERT BACK STANDBY TO STANDBY ROLE</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Stop managed servers, admin servers and node managers in secondary Site2</td>
<td>You can connect to secondary WebLogic Console and shutdown managed servers and Admin server in secondary site. Stop the node manager processes too and umount dbfs if they are mounted.</td>
</tr>
</tbody>
</table>
| 2 Convert the standby DB into a physical standby again | Use DG broker in primary db host and convert the secondary to physical standby again. As user oracle:
```
[oracle@drdbA ~]$ dgmgrl sys/your_sys_password@primary_db_unqname
DGMGRL> convert database "secondary_db_unqname" to physical standby
Use "show configuration" to verify that the conversion has been correctly performed. |
| 3 Revert back any updated /etc/hosts in clients | If you updated the virtual front-end name in a client’s /etc/hosts file in order to point to secondary site, revert it back so the virtual front-end name points to primary front-end IP again. |
| 4 Detach | For each mid-tier host in Site2, run the following:
- Unmount the Block Volume, which is mounted in /u01/data:
```
[opc@soampdrs-soa-0 opc]$ sudo umount /u01/data
```
Make sure that there are not Oracle processes running. It is expected that they are stopped at this point, but if there is something still running on that folder, the umount will fail.
- Once unmounted, detach the Block Volume from the OCI Console. Go to each block volume > attached instances > detach from instance
The OCI Console will ask you to run some iscsi commands before completing the detachment.
- Finally, with root user, edit the /etc/fstab file and comment the entry for /u01/data. This is to prevent it from trying to mount the original BV in next reboot. Example:
```
#Remove or comment this entry:
#UUID=765185db-a10e-4da4-bfd0-0437348b3cf6 /u01/data ext4 auto,defaults,_netdev,nofail 0 2
```
Repeat these steps for the rest of the mid-tier nodes in Secondary. |
| 5 Delete/rename the detached volumes in Site2 to prevent from mouting them by mistake. | Using the OCI Console, delete (or rename) the Block Volumes that have been detached from the Site2 mid-tier hosts in the previous step. They will not be used anymore. |
Scale-out and Scale-in

The procedure for scale-out and scale-in operations in the Block Volume Cross-Region DR model has slight differences with the DBFS and RSYNC models.

This is the procedure to scale-out a SOAMP DR environment in Block Volume cross-region replica model:

a) First, follow the steps described in previous point Open Secondary Site for Validation to open the secondary site, but do not convert the standby database to snapshot yet, and do not start the admin and managed servers yet. This is just to mount in secondary hosts a version of the block volumes prior to any scale action, so the scale action can be performed in primary and secondary independently.

b) You can now proceed with the steps described in the point Scale-out of this document to scale-out primary and secondary. Ignore any mentions to the config_replica.sh script; it does not apply to this method.

c) Then, revert the secondary to the standby role, by detaching the volumes, as described in the previous point Open Secondary Site for Validation.

d) For the Block Volume of the new node in primary, enable the Cross-Region replica in the same way as it is already configured for the existing primary nodes.

NOTE: The terraform scripts used by SOAMP to scale-out create redundant/unnecessary block volumes in the existing nodes. These duplicated block volumes have the same names than the existing block volumes, and they are attached but not mounted to the nodes. These duplicated block volumes are NOT needed and it is strongly recommended to detach and delete them immediately after the scale-out to prevent mistakes and mounts pointing to the incorrect BV. Make sure you delete the duplicated block volumes created by the scale-out job.

This is the procedure to scale-in a WLS for OCI DR environment in Block Volume cross-region replica model:

a) First, follow the steps described in previous point Open Secondary Site for Validation to open the secondary site, but do not convert the standby database to snapshot yet, and do not start the admin and managed servers yet. This is just to mount in secondary hosts a version of the block volumes prior to any scale action, so the scale action can be performed in primary and secondary independently.

b) Disable the Cross-Region replica in the block volumes of the primary node that is going to be deleted. The scale-in job will fail to delete a block volume that has the cross-region replica enabled.

c) You can now proceed with the steps described in the point Scale-in of this document to scale-in primary and secondary. Ignore any mentions to the config_replica.sh script; it does not apply to this method.

d) Then, revert back the secondary to the standby role, by detaching the volumes, as described in the previous point Open Secondary Site for Validation.

NOTE: The terraform scripts used by SOAMP to scale-in create redundant/unnecessary block volumes in the existing nodes. These duplicated block volumes have the same names than the existing block volumes, and they are attached but not mounted to the nodes. These duplicated block volumes are NOT needed and it is strongly recommended to detach and delete them immediately after the scale-in to prevent mistakes and mounts pointing to the incorrect BV. Make sure you delete the duplicated block volumes created by the scale-in job.

Using OCI CLI commands for automating Block Volume Replication tasks

Most of the actions related with Block Volumes (enable/disable replication, activation, attach/detach block volumes) can be performed not only using the OCI Console, also using the OCI Commam Line Interface (CLI) commands.

Here is a list of the commands for the most common actions used during the setup and switchover/failover of this DR model. For information about using the CLI, see Oracle Cloud Infrastructure CLI Command Reference and Cross-Region Volume Replication documentation.

- Enable cross-region replication for a Block Volume
  
  `oci bv volume update --volume-id <volumeld> --block-volume-replicas '{"displayName":"<replicaDisplayName>","availabilityDomain":<availabilityDomain>}']`

- Disable cross-region replication for a Block Volume
  
  `oci bv volume update --volume-id <volumeld> --block-volume-replicas []'`
- Activate the Block Volume replication

  `oci bv volume create --source-volume-replica-id <volumeReplicaId> --compartment-id <compartmentID> --availability-domain <AvailabilityDomain> --display-name <DisplayName>`

- Attach a Block Volume (iscsi type) to a compute instance

  `oci compute volume-attachment attach-iscsi-volume --instance-id <instanceId> --volume-id <volumeId>`

- Detach a Block Volume from a compute instance

  `oci compute volume-attachment detach --volume-attachment-id <volumeAttachmentId>`

As shown above, you need the OCIDs of the Block Volumes and volume attachments for running the OCI CLI commands. These OCIDs change after every switchover. But these values can be retrieved using OCI CLI scripts as well. As long as you use consistent "display names" for the block volumes objects, you can use the display names to get the OCID dynamically.

However, not all the steps can be automated with OCI CLI commands. Some actions, like attaching or detaching Block Volumes, require a combination of OCI Console (or OCI CLI) actions plus Operating System commands (edit fstab, mount, run iscsi commands, etc.). You can find an example of a semi-automation script for the switchover in a SOAMP with Block Volume replication DR model in [https://github.com/oracle-samples/maa/raw/main/wls_mp_dr/scripts_BV_replica_model/example_OCICli_BVmodel.zip](https://github.com/oracle-samples/maa/raw/main/wls_mp_dr/scripts_BV_replica_model/example_OCICli_BVmodel.zip)