Disaster Recovery Guide: Oracle SOA Suite 10g on Oracle WebLogic Server

Oracle Maximum Availability Architecture White Paper
June 2009

Maximum Availability Architecture

Oracle Best Practices For High Availability

ORACLE

Disclaimer

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

Disaster Recovery Guide: Oracle SOA Suite 10g on Oracle WebLogic Server

Table Of Contents

Disaster Recovery Guide: Oracle SOA Suite 10g on Oracle WebLogic Server

Executive Overview

The Maximum Availability Architecture (MAA) is a best practices blueprint for achieving high availability and performance using Oracle technologies. This MAA white paper provides practical configuration and deployment techniques for creating disaster recovery topologies using disk replication for the middle tier and Oracle Data Guard for the databases in the data tier.

The Enterprise Deployment topology for Oracle Service Oriented Architecture (SOA) 10g Release 4 is used as an example in this paper; however the concepts and procedures described here are applicable when creating disaster recovery topologies for other supported Oracle Application Server 10g products deployed on certified versions of the Oracle WebLogic Server. These procedures are also applicable for standalone deployments of Oracle WebLogic Server 9.2 and above.

Introduction

Providing Maximum Availability Architecture is one of the key requirements for any Oracle Application Server Enterprise Deployment. Oracle Application Server includes an extensive set of High Availability features such as: Process Death Detection and Restart, Server Clustering, Load Balancing, Failover, Backup and Recovery, Rolling Upgrades, Rolling Configuration Changes, and Dynamic Discovery, which protect an Enterprise Deployment from unplanned down time and minimize planned downtime.

Additionally, Enterprise Deployments need protection from unforeseen disasters and natural calamities. One protection solution involves setting up a standby site at a geographically different location than the production site. The standby site may have equal or fewer services and resources compared to the production site. Application data, metadata, configuration data, and security data are replicated to the standby site on a periodic basis. The standby site is normally in a passive mode; it is started when the production site is not available. This deployment model is sometimes referred to as an active/passive model. This model is normally adopted when the two sites are connected over a WAN and network latency does not allow clustering across the two sites. A standby site not only gets used for Disaster Recovery but is also used when site wide planned maintenance occurs at the primary site.

Disaster Recovery Guide: Oracle SOA Suite 10g on Oracle WebLogic Server

The Oracle Application Server Disaster Recovery solution uses disk replication technology for disaster protection of Oracle Application Server middle tier components. It supports Hot-Pluggable deployments, and it is compatible with third party vendor recommended solutions.

Disaster protection for Oracle databases that are included in your Oracle Application Server is provided through Oracle Data Guard.

This technical paper describes the steps required to create an Oracle Application Server Disaster Recovery solution for enterprise deployment architecture, using disk replication technology for the middle tier and Oracle Data Guard for the data tier.

The Enterprise Deployment Topology of Oracle Application Server SOA Suite 10.1.3.4 on Oracle WebLogic Server 9.2 is used as the reference architecture in this paper. An enterprise deployment of Oracle Service Oriented Architecture (SOA) Suite is a reference configuration that is designed to support large-scale, mission-critical business software applications using SOA components.

Disaster Recovery Concepts and Terminology

This section provides concepts and defines Disaster Recovery terminology:

Application Server host name

The Application Server host name is the host name of the machine on which Oracle Application Server is installed A host can have only one Application Server host name. This white paper differentiates between the terms Application Server host name and network host name.

Network host name

The network host name is the host name by which a particular host is known within the host's network. A host can have the same network host name and Application Server host name. A host can have only one Application Server host name, but it can have multiple network host names. A network hostname resolves to a specific IP address that is resolved through DNS resolution and it is this IP address that is enabled on the machine with which this network hostname is associated.

Topology:

The production site and standby site hardware and software components that comprise an Oracle Application Server Disaster Recovery solution.

Symmetric topology:

An Oracle Application Server Disaster Recovery configuration that is completely identical across tiers on the production site and standby site. In a symmetric topology, the production site and standby site have the identical number of hosts, load balancers, instances, and applications. The same ports are used for both sites. The systems are configured identically and the applications access the same data. This manual describes how to set up a symmetric Oracle Application Server Disaster Recovery topology for an enterprise configuration.

Asymmetric topology:

A disaster recovery configuration that is different across tiers on the production site and standby site. In an asymmetric topology, the standby site can use less hardware (for example, the production site could include four hosts with four Application Server instances while the standby site includes two hosts with four Application Server instances. Or, in a different asymmetric topology, the standby site can use fewer Application Server instances (for example, the production site could include four Application Server instances while the standby site includes two Application Server instances). Another asymmetric topology might include a different configuration for a database (for example, using a Real Application Clusters database at the production site and a single instance database at the standby site). Appendix C, "Creating an Asymmetric Topology" describes asymmetric topologies.

Disaster Recovery:

Disaster Recovery is defined as the ability to safeguard against natural or unplanned outages at a production site by having a recovery strategy for applications and data to a geographically separate standby site.

Production site setup:

This is defined as the process of creating a production site. To create the production site using the procedure described in this manual, you must plan and create Application Server host names and network host names, create mount points and links on the hosts to the Oracle home directories on the shared storage where the Oracle Application Server instances will be installed, install the binaries and instances, and deploy the applications.

Standby site setup:

This is defined as the process of creating a standby site. To create the standby site using the procedure described in this manual, you must plan and create Application Server host names and network host names, perform a switchover operation (which replicates the Oracle home directories and installations from the production site shared storage to the standby site shared storage), and create mount points and links to the Oracle home directories on the standby shared storage.

Site failover:

The process of making the current standby site the new production site after the production site becomes unexpectedly unavailable (for example, due to a disaster at the production site). This book also uses the term "failover" to refer to a site failover.

Site switchover:

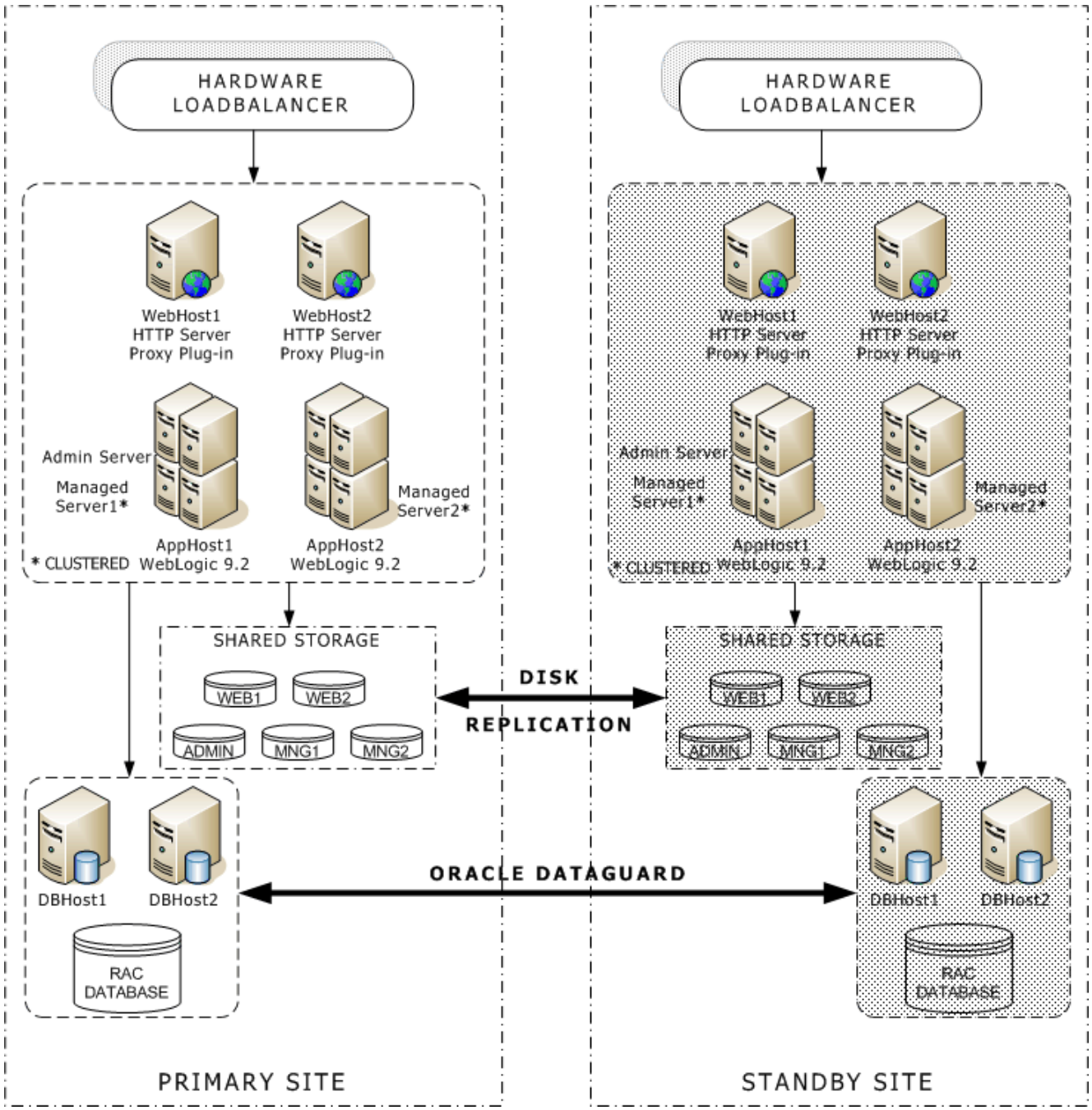
Site switchover is defined as the process of reversing the roles of the production site and standby site. Switchovers are planned operations done for periodic validation or to perform planned maintenance on the current production site. During a switchover, the current standby site becomes the new production site, and the current production site becomes the new standby site. This book also uses the term "switchover" to refer to a site switchover.

Site switchback:

Site switchback is defined as the process of reverting the current production site and the current standby site to their original roles. Switchbacks are planned operations done after switchover operation has been completed. A switchback restores the original roles of each site, the current standby site becomes the production site, and the current production site becomes the standby site. This book also uses the term "switchback" to refer to a site switchback.

Environment Description

The diagram below shows a production site and standby site along with the volumes required on each site. This is an example of a disaster recovery environment.



Disaster Recovery Guide: Oracle SOA Suite 10g on Oracle WebLogic Server

Please refer to the Oracle Fusion Middleware Disaster Recovery Guide for the Identity Management Topology.

Conventions Used

The following naming conventions used in this guide:

Primary Site:

Tier	Network Hostname	Application Server
Web Tier	webhost_p1.mycompany.com webhost_p2.mycompany.com	webhost1 webhost2
Application Tier	apphost_p1.mycompany.com apphost_p2.mycompany.com	apphost1 apphost2
Directory Tier	idmhost_p1.mycompany.com idmhost_p2.mycompany.com	idmhost1 idmhost2
Data Tier	soadbhost_p1.mycompany.com saodbhost_p2.mycompany.com	soadbhost1 soadbhost2
	infradbhost_p1.mycompany.com infradbhost_p2.mycompany.com	infradbhost1 infradbhost2

Metadata Repositories

SOA Metadata Repository	Instance Names Database Name Service Names	psoa1, psoa2 psoa.mycompany.com soa.mycompany.com
IDM Metadata Repository	Instance Names Database Names Service Name	pidm1, pidm2 pidm.mycompany.com idm.mycompany.com

Standby Site:

Tier	Network Hostname	Application Server
Web Tier	webhost_s1.mycompany.com webhost_s2.mycompany.com	webhost1 webhost2
Application Tier	apphost_s1.mycompany.com apphost_s2.mycompany.com	apphost1 apphost2
Directory Tier	idmhost_s1.mycompany.com idmhost_s2.mycompany.com	idmhost1 idmhost2

Data Tier	soadbhost_s1.mycompany.com saodbhost_s2.mycompany.com	soadbhost1 soadbhost2
	infradbhost_s1.mycompany.com infradbhost_s2.mycompany.com	infradbhost1 infradbhost2

Metadata Repositories

SOA Metadata Repository	Instance Names Database Name Service Names	ssoa1, ssoa2 ssoa.mycompany.com soa.mycompany.com
IDM Metadata Repository	Instance Names Database Names Service Name	sidm1, sidm2 sidm.mycompany.com idm.mycompany.com

Software Requirements

This section provides the software requirements for setting up a disaster recovery topology.

- Oracle Database 10*g* release 10.2.0.3 or higher or Oracle Database 11g release 11.1.0.6 or higher
- Oracle Patch 6265268 is required when using Oracle Database 11g
- Oracle WebLogic Server 9.2 MP3
- Oracle Application Server SOA Suite 10*g* Release 3 (10.1.3)
- Oracle Application Server Patch Set 10.1.3.4 (Patch 7272722)
- OPatch Version 10.1.0.0.0 (Patch 6880880)
- Oracle SOA Suite 10.1.3.4 Patch for WebLogic Server (Patch 7490612)
- Hot Pluggability Patch for Oracle SOA Suite on 10.1.3.4 on WebLogic 9.2 (Patch 7337034)
- Apache HTTP Server 2.2.x

High Level Steps

The high level steps required to provision a disaster recovery topology are listed below

- Configure the Shared Storage Device
- Configure the Environment for the Primary Site and Standby Site
- Create and Configure the Primary Site

- Create the Data Tier on the Standby Site
- Create the Application Tier for the Standby Site
- Instantiate the Standby Site
- Test Switchover, Switchback and Failover

Requirements

Storage & Volumes

Create the volumes listed below on your storage device and mount them appropriately on your Application Tier Nodes. These volumes must be created on the primary site and the standby site. Please follow the documentation provided by your storage vendor to create the volumes.

Based on the capabilities of the disk replication technology available with your preferred storage device, you may need to create mount points directories and symbolic links on each of the nodes within a tier. The mount points and symbolic links are set up so that the same directory structure can be used on each Application Server host within a tier.

Note:

Oracle strongly recommends using the same directory structure across all the Application Server hosts within a tier. This is an Oracle Best Practice recommendation and not a requirement.

Based on the capabilities of the disk replication technology available with your preferred storage device, you may need to create mount point directories and symbolic links on each of the nodes within a tier.

If your storage device's disk replication technology guarantees consistent replication across multiple volumes, you can:

- Create one volume per server running on that tier. For example, you can create one volume for the WebLogic Administration Server and another volume for the Managed Servers. Refer to Appendix A1 for examples.
- The volumes are usually accessed for a read/write operation by a single node at a time. In some cases a volume may be mounted on more than one node and can be accessed by both the nodes at once. In such cases a clustered file system may be required depending on the storage subsystem. However, there is no known case of a single file or directory tree being concurrently accessed by processes on different machines.
- Create one consistency group for a tier and have the volumes be members of that consistency group. For example, create a consistency group with the admin server volume and managed server volume as members of that group.

- Mount the volume on the respective node. It is a best practice to have the same directory structure across the application server hosts within a tier. Refer to Appendix A1 for the volume details.

If your storage device's disk replication technology does not guarantee consistent replication across multiple volumes, you can:

- Create a volume for each tier. Refer to Appendix A2 for the volume details.
- The volumes are usually accessed for a read/write operation by a single node at a time. In some cases a volume may be mounted on more than one node and can be accessed by both the nodes at once. In such cases a clustered file system may be required depending on the storage subsystem. However, there is no known case of a single file or directory tree being concurrently accessed by processes on different machines.
- Create a separate directory for each node in that tier.
- Mount the volume on the respective node.
- Create a mount point directory on each node to the directory on the volume.
- Create a symbolic link to the mount point directory. A symbolic link should be created so that the same directory structure can be used across the nodes in a tier.

Oracle Cluster File System (OCFS2) Configuration

High Availability for Persistent Stores

The WebLogic application servers are usually clustered for high-availability. For local site high-availability, a persistent file-based store can be migrated along with its parent server as part of the "server-level" migration feature that provides both automatic and manual migration at the server-level. However, file-based stores must be configured on a shared disk that is available to the migratable target servers in the cluster.

Use High Availability Storage for State Data

The server migration process moves or “migrates” services, but not the state information associated with work in process at the time of failure.

To ensure high availability, it is critical that such state information remains available to the server instance and the services it hosts after migration. Otherwise, data about the work in process at the time of failure may be lost. State information maintained by a migratable server, such as the data contained in transaction logs, should be stored in a shared storage system that is accessible to any potential machine to which a failed migratable server might be migrated. For highest

reliability, use a shared storage solution that is itself highly available—for example, a storage area network (SAN).

A shared storage solution in a SAN as deployed in this whitepaper uses a host based clustered or shared file system technology. Oracle Cluster File System (OCFS2) is recommended, but any high-availability shared file system technology can be used.

Steps to Configure OCFS2

The process to configure OCFS2 is noted below.

- Identify the volumes which will be used at the Primary and Disaster Recovery Site for the WebLogic Cluster JMS and Transaction Logs.
- Populate the */etc/ocfs2/cluster.conf*. If the console was installed, then it can be used to create this file. If not, then the file can be manually created. Please note the file has a very specific structure, which is described in the Oracle Cluster File System (OCFS2) User's Guide. Validate that a copy of this files resides on all nodes in the cluster.
- Edit the */etc/selinux/config* file. Change *SELINUX=enforcing to SELINUX=disabled*. It is mandatory that this change is made; otherwise the clustered file system will not come online. This file must be edited on all nodes in the cluster.
- On all nodes in the cluster, configure, online and load the cluster service, using */etc/init.d/o2cb*.
- On only one node, create the clustered file system.
- Create the required mount points and then edit */etc/fstab* file and add the entries for the mount points, and then the mount file system on all nodes in cluster.
- To ensure the file systems will mount at boot, use */sbin/chkconfig* to add the o2cb and ocfs2 services, and change the startup information for the specified services.
- Modify kernel configuration parameters *kernel.panic_on_oops* and *kernel.panic* in */etc/sysctl.conf* file to ensure oc2cb will function correctly.

Oracle Cluster File System Resources

For more detail please refer to the following documents.

- OCFS2 A Cluster File System for Linux, User's Guide for Release 1.4
- Oracle Cluster File System (OCFS2) Documentation Library

Installation & Configuration

This section provides the steps for installing and configuring the primary and standby

Install and Configure the Primary Site

The primary site should be installed and configured as described in the Enterprise Deployment Guide for Oracle Application Server SOA Suite 10.1.3.4 with a few variations. The steps to install and configure the primary site are listed below and should be followed in the sequence listed.

1. Create volumes on a shared storage device as described in the "Storage & Volumes" section of this Guide.
2. Setup application server hostname and the network hostnames on the primary and standby sites.
3. Install and configure the Oracle Application Server SOA Suite on WebLogic Server as described in the Enterprise Deployment Guide with the following modifications:
    - Install the Oracle Application Server Suite into the volumes created on the shared storage device.
    - Use the hostname aliases instead of the physical hostnames during the creation and configuration of the Weblogic Domain.
    - Setup SSL certificates using the hostname aliases on all the application server hosts for proper Node Manager Communication.
    - Create a separate volume on each site for the JMS stores and Transaction Logs.

Hostname Planning

Setup the network hostnames and application server hostnames on the primary site and the standby site as shown in the table below.

Primary Site:

Tier	Network Hostname	Application Server
Web Tier	webhost_p1.mycompany.com webhost_p2.mycompany.com	webhost1 webhost2
Application Tier	apphost_p1.mycompany.com apphost_p2.mycompany.com	apphost1 apphost2
Directory Tier	oidhost_p1.mycompany.com oidhost_p2.mycompany.com	oidhost1 oidhost2

Data Tier	soadbhost_p1.mycompany.com saodbhost_p2.mycompany.com	soadbhost1 soadbhost2
	infradbhost_p1.mycompany.com infradbhost_p2.mycompany.com	infradbhost1 infradbhost2

Standby Site

Tier	Network Hostname	Application Server
Web Tier	webhost_s1.mycompany.com webhost_s2.mycompany.com	webhost1 webhost2
Application Tier	apphost_s1.mycompany.com apphost_s2.mycompany.com	apphost1 apphost2
Directory Tier	oidhost_s1.mycompany.com oidhost_s2.mycompany.com	oidhost1 oidhost2
Data Tier	soadbhost_s1.mycompany.com saodbhost_s2.mycompany.com	soadbhost1 soadbhost2
	infradbhost_s1.mycompany.com infradbhost_s2.mycompany.com	infradbhost1 infradbhost2

Configure Node Manager Communication

The communication between the Oracle WebLogic Administration Server and Node Manager is done over SSL. For this communication to work correctly on the standby site, SSL certificates must be created using the network hostnames. The configuration process involves the following steps:

1. Generate Self-Signed Certificates
2. Create an Identity Key Store
3. Create a Trust Key Store
4. Configure Node Manager

Generate Self-Signed Certificates:

1. Set your environment using the setWLS.env script located under the $WL_HOME/server/bin directory.
2. Create a user-defined directory for the certificates. For example, create a directory called "*certs*" under the *$BEA_HOME/user_projects/domains/SOADomain* directory.

3. Run the "*utils.CertGen*" tool from the user-defined directory to create the certificates for both apphost_p1 and apphost_p2.

Syntax:

java utils.CertGen <key_passphrase> <cert_file_name> <key_file_name> [export|domestic] [hostname].

For example:

java utils.CertGen welcome1 apphost1_cert apphost1_key domestic apphost1

java utils.CertGen welcome1 apphost2_cert apphost2_key domestic apphost2

Create an Identity Key Store

1. Create a new identity keystore called "*appIdentityKeyStore*" using the "*utils.ImportPrivateKey*" utility.
2. Create this keystore under the same directory as the certificates (i.e. *$BEA_HOME/user_projects/domains/j2eeDomain/certs*).

   Please note that the Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the "*utils.ImportPrivateKey*" utility.
3. Import the certificate and private key for both apphost_p1 & apphost_p2 into the Identity Store; also make sure to use a different alias for each of the certificate/key pair imported.

Syntax:

java utils.ImportPrivateKey <keystore_file> <keystore_password> <certificate_alias_to_use> <private_key_passphrase> <certificate_file> <private_key_file> [<keystore_type>]

For example:

java utils.ImportPrivateKey appIdentityKeystore.jks welcome1 appIdentity1 welcome1

$BEA_HOME/user_projects/domains/SOADomain/certs/apphost1_cert.pem

$BEA_HOME/user_projects/domains/SOADomain/certs/apphost1_key.pem

java utils.ImportPrivateKey appIdentityKeystore.jks welcome1 appIdentity2 welcome1

$BEA_HOME/user_projects/domains/SOADomain/certs/apphost2_cert.pem

$BEA_HOME/user_projects/domains/SOADomain/certs/apphost2_key.pem

Create a Trust Key Store

- Create a new trust keystore called "*appTrustKeyStore*" using the "*keytool*" utility.

- Use the standard java keystore to create the new trust keystore since it already contains most of the root CA certificates needed. It is recommended not to modify the standard Java trust key store directly.
- Copy the standard Java keystore "*cacerts*" located under the *$WL_HOME/server/lib* directory to the same directory as the certificates.

For example:

cp $WL_HOME/server/lib/cacerts $BEA_HOME/user_projects/domains/SOADomain/certs/appTrustKeystore.jks

- The default password for the standard Java keystore is "*changeit*" and it is always recommended to change the default password. Use the keytool utility to do this

Syntax:

keytool -storepasswd -new <NewPassword> -keystore <TrustKeyStore> -storepass <Original Password>

For example:

keytool -storepasswd -new welcome1 -keystore appTrustKeyStore.jks -storepass changeit

- The CA certificate, "*CertGenCA.der*" is used to sign all certificates generated by utils.CertGen tool and is located at *$WL_HOME/server/lib* directory. This CA certificate needs to be imported into the *appTrustKeyStore* using the "*keytool*" utility.

Syntax:

keytool -import -v -noprompt -trustcacerts -alias <AliasName> -file <CAFileLocation> -keystore <KeyStoreLocation> -storepass <KeyStore Password>

For example:

keytool -import -v -noprompt -trustcacerts -alias clientCACert -file $WL_HOME/server/lib/CertGenCA.der -keystore appTrust.jks -storepass welcome1

Configure Node Manager

Follow the step below to enable the node manager to use the newly created key stores.

- Edit the *nodemanager.propeties* file located under the *$WL_HOME/common/nodemanager/* directory and the following lines at the end of the file.

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=<Identity KeyStore>
CustomIdentityKeyStorePassPhrase=<Identity Keystore Passwd>
CustomIdentityAlias=<Identity Key Store Alias>
CustomIdentityPrivateKeyPassPhrase=<Private Key used when creating Certificate>
CustomTrustKeyStoreFileName=<Trust Keystore>
CustomTrustKeyStorePassPhrase=<Trust Keystore Passwd>
```

For example on Node 1:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=$BEA_HOME/user_projects/domains/SOADomain/certs /appIdentityKeystore.jks
CustomIdentityKeyStorePassPhrase=welcome1
CustomIdentityAlias=appIdentity1
CustomIdentityPrivateKeyPassPhrase=welcome1
CustomTrustKeyStoreFileName=$BEA_HOME/user_projects/domains/SOADomain/certs/appTrust.jks
CustomTrustKeyStorePassPhrase=welcome1
```

For example on Node2:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=$BEA_HOME/user_projects/domains/SOADomain/certs /appIdentityKeystore.jks
CustomIdentityKeyStorePassPhrase=welcome1
CustomIdentityAlias=appIdentity2
CustomIdentityPrivateKeyPassPhrase=welcome1
CustomTrustKeyStoreFileName=$BEA_HOME/user_projects/domains/SOADomain/certs/appTrust.jks
CustomTrustKeyStorePassPhrase=welcome1
```

Configure Disk Replication Software

Configure the disk replication software in your environment using the documentation provided by your vendor and then follow the steps below to complete your setup.

- On the standby site, make sure the same Application Server host names are used for middle tier hosts as were used for the middle tier hosts at the production site.
- On the shared storage at the standby site, create the same volumes as were created on the shared storage at the production site.

- On the standby site, create the same mount points and symbolic links that you created at the production site.

Note:

It is not necessary to install the same Oracle Application Server instances at the standby site as were installed at the production site. When the production site storage is replicated to the standby site storage, the Oracle software installed on the production site volumes will be replicated at the standby site volumes.

- Perform any other necessary configuration required by the shared storage vendor to enable disk replication between the production site shared storage and the standby site shared storage.
- Create the baseline snapshot copy of the production site shared storage that sets up the replication between the production site and standby site shared storage. Create the initial baseline copy and subsequent snapshot copies using asynchronous replication mode. After the baseline snapshot copy is performed, validate that all the directories inside the standby site volumes have the same contents as the directories inside the production site volumes.
- Set up the frequency of subsequent copies of the production site shared storage, which will be replicated at the standby site. When asynchronous replication mode is used, then at the requested frequency the changed data blocks at the production site shared storage (based on comparison to the previous snapshot copy) become the new snapshot copy, and the snapshot copy is transferred to the standby site shared storage.
    - Make sure that disaster protection for the metadata repositories is provided by Oracle Data Guard, as described in the section below "Configuring Oracle Data Guard." Do not use disk replication technology to provide disaster protection for Oracle databases.
    - Make sure that your databases are running under the “Maximum Availability Mode”.
    - The standby site shared storage receives snapshots transferred on a periodic basis from the production site shared storage. After the snapshots are applied, the standby site shared storage will include all the data up to and including the data contained in the last snapshot transferred from the production site before the failover or switchover.
    - You should manually force a synchronization operation whenever a change is made to the middle tier at the production site (for example, when a new application is deployed at the production site). Follow the vendor-specific instructions for forcing a synchronization using disk replication technology.

- Oracle Data Guard provides disaster protection and synchronizes the Oracle Databases in the topology.

Configuring Oracle Data Guard

Oracle Data Guard is the recommended disaster protection technology for the Oracle Fusion Middleware Metadata Repository databases on the primary site and standby site. The databases on the standby site should be setup as Physical Standby Databases, The section below detail the setup and configuration of the data tier on the standby site.

For more information regarding Oracle Data Guard and Administration, please refer to the Oracle Database High Availability Guide for your database version.

This section covers the following topics:

Prerequisites & Assumptions

The Data Guard setup and configuration steps below assume that the following conditions are met:

- The RAC cluster and ASM instances on the standby site have been created.
- The RAC databases on the standby site and the primary site are using a Flash Recovery Area.
- The database hosts on the standby site already have Oracle Software installed.
- The physical path for the DB_HOME on the standby site matches that of the primary site.

Environment Description

The steps below use the environment variables shown in the table below:

Primary Site:

	Parameter	Value
SOA Metadata Repository	Hostname	soadbhost1.mycompany.com soadbhost2.mycompany.com
	ORACLE_HOME	/opt/maa/oracle/product/10.2.0/db_1
	DB NAME	PSOA
	DB_UNIQUE_NAME	PSOA
	DB_INSTANCE_NAMES	PSOA1, PSOA2
	SERVICE NAME	soa.mycompany.com
	Hostname	idmdbhost1.mycompany.com idmdbhost2.mycompany.com

IDM Metadata Repository	ORACLE_HOME	/opt/maa/oracle/product/10.2.0/db_1
	DB NAME	PIDM
	DB_UNIQUE_NAME	PIDM
	DB_INSTANCE_NAMES	PIDM1, PIDM2
	SERVICE NAME	idm.mycompany.com

Standby Site:

	Parameter	Value
SOA Metadata Repository	Hostname	soadbhost1.mycompany.com soadbhost2.mycompany.com
	ORACLE_HOME	/opt/maa/oracle/product/10.2.0/db_1
	DB NAME	SSOA
	DB_UNIQUE_NAME	SSOA
	DB_INSTANCE_NAMES	SSOA1,SSOA2
	SERVICE NAME	soa.mycompany.com
IDM Metadata Repository	Hostname	idmdbhost1.mycompany.com idmdbhost2.mycompany.com
	ORACLE_HOME	/opt/maa/oracle/product/10.2.0/db_1
	DB NAME	SIDM
	DB_UNIQUE_NAME	SIDM
	DB_INSTANCE_NAMES	SIDM1, SIDM2
	SERVICE NAME	idm.mycompany.com

Data Guard Setup: High Level Tasks

1. Gather Files and Perform Backup
2. Configure Oracle Net Services on the Standby Site
3. Create Instances and Databases on the Standby site
4. Configure the Primary Database for Data Guard
5. Verify Data Guard Configuration

Gather Files and Perform Backup

1. On the SOADBHOST1 of the primary site, create a directory for staging purposes. For example:

```
$ mkdir –p /opt/maa/stage/psoa
```

2. Create the exact path on SOADBHOST1 of the standby site. Follow the example shown in step1.
3. On the SOADBHOST1 of the primary site, connect to the database instance "*psoa1*" and create a *pfile* from the *spfile*. For example:

```
SQL > create pfile='/opt/maa/stage/psoa/initpsoa.ora' from spfile;
```

4. On the SOADBHOST1 of the primary site, connect to RMAN, perform a backup of the database, and place the backup files in the stage directory. For example:

```
$ $ORACLE_HOME/bin/rman target /

RMAN> backup device type disk format '/opt/maa/stage/psoa/%U'
database plus archivelog;

RMAN> backup device type disk format '/opt/maa/stage/psoa/%U'
current controlfile for standby;
```

5. Verify that the backups created by RMAN are valid. Follow the steps below to validate the backups. See examples below:
    a. Connect to RMAN on SOADBHOST1 of the primary site.
    b. List the backup summary.
    c. Validate the backup sets created by RMAN in Step 4.

```
RMAN> list backup summary;
using target database control file instead of recovery catalog
List of Backups
===============
Key     TY LV S Device Type Completion Time #Pieces #Copies Compressed Tag
------- -- -- - ----------- --------------- ------- ------- ---------- ---
93      B  A  A DISK        14-MAY-07       1       1       NO         TAG20070514T122312
94      B  F  A DISK        14-MAY-07       1       1       NO         TAG20070514T122315
95      B  F  A DISK        14-MAY-07       1       1       NO         TAG20070514T122315
96      B  A  A DISK        14-MAY-07       1       1       NO         TAG20070514T122629
97      B  F  A DISK        14-MAY-07       1       1       NO         TAG20070514T123220
RMAN> validate backupset 93;
allocated channel: ORA_DISK_1
channel ORA_DISK_1: sid=451 instance=psoa1 devtype=DISK
channel ORA_DISK_1: starting validation of archive log backupset
channel ORA_DISK_1: reading from backup piece /opt/maa/stage/psoa/34ihmtdg_1_1
channel ORA_DISK_1: restored backup piece 1
piece handle=/opt/maa/stage/psoa/34ihmtdg_1_1 tag=TAG20070514T122312
channel ORA_DISK_1: validation complete, elapsed time: 00:00:02
```

6. On SOADBHOST1 of the primary site, copy the listener.ora, sqlnet.ora, and tnsnames.ora files from the *$ORACLE_HOME/network/admin* directory to the staging directory.
7. Using OS utilities, copy the contents of staging directory on SOADBHOST1of the primary site to the staging directory on SOADBHOST1 of the standby site.

Data Guard Setup: Configure Oracle Net Services on the Standby Site

1. Copy the listener.ora, sqlnet.ora, and tnsnames.ora files from the staging directory on SOADBHOST1 on the primary site to the *$ORACLE_HOME/network/admin* directory on all the nodes of the standby site.
2. Modify the listener.ora file on each of the standby host to contain the VIP of that host.
3. Modify the tnsnames.ora file on each node, including the primary RAC nodes and standby RAC nodes, to contain all primary and standby net service names.
4. Modify the Oracle Net aliases that are used for the *local_listener* and *remote_listener* parameters to point to the listener on each standby host. The example below shows excerpts from the *tnsnames.ora* file.

```
PSOA =
(DESCRIPTION =
(ADDRESS =
(PROTOCOL = TCP)
(HOST = soadbhost1-vip)
(HOST = soadbhost2-vip)
(PORT = 1521))
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = psoa)
)
)
SSOA =
(DESCRIPTION =
(ADDRESS =
(PROTOCOL = TCP)
(HOST = soadbhost1-vip)
(HOST = soadbhost2-vip)
(PORT = 1521))
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = ssoa)
)
)
```

5. Start the listeners on the standby database hosts.

Data Guard Setup: Create Instances and Database on the Standby Site

1. To enable secure transmission of redo data, make sure the databases on the primary and standby sites use a password file, and make sure the password for the SYS user is identical on every system. Create a password file on both the nodes of the standby databases. For example:

```
On SOADBHOST1 of the standby site

$ cd $ORACLE_HOME/dbs

$ orapwd file=orapwpsoa1 password=welcome1

On SOADBHOST2 of the standby site

$ cd $ORACLE_HOME/dbs
```

```
$ orapwd file=orapwpsoa2 password=welcome1
```

2. Copy and rename the pfile from the staging area to the *$ORACLE_HOME/dbs* directory on SOADBHOST1 of the standby site. For example:

```
$ cp /opt/maa/stage/psoa/initpsoa.ora
$ORACLE_HOME/dbs/initpsoa1.ora
```

3. Modify the standby initialization parameter file copied from the primary node to include Data Guard parameters as illustrated in the table below.

RAC Parameters	*.cluster_database=true PSOA1.instance_name=PSOA1 PSOA2.instance_name=PSOA2 PSOA1.instance_number=1 PSOA2.instance_number=2 PSOA1.thread=1 PSOA2.thread=2 PSOA1.undo_tablespace=UNDOTBS1 PSOA2.undo_tablespace=UNDOTBS2 *.remote_listener=LISTENERS_PSOA
Data Guard Parameters	*.db_unique_name=SSOA *.log_archive_config='dg_config=(SSOA,PSOA)' *.log_archive_dest_2='service=PSOA valid_for=(online_logfiles,primary_role) db_unique_name=PSOA' *.db_file_name_convert='+DATA/PSOA/','+DATA/SSOA/','+RECO/PSOA', '+RECO/SSOA' *.log_file_name_convert='+DATA/PSOA/','+DATA/SSOA/','+RECO/PSOA', '+RECO/SSOA' *.standby_file_management=auto *.fal_server='PSOA' *.fal_client='SSOA'
Miscellaneous Parameters	*.background_dump_dest= /opt/oracle/admin/PSOA/bdump *.core_dump_dest= /opt/oracle/admin/PSOA/cdump *.user_dump_dest= /opt/oracle/admin/PSOA/udump *.audit_file_dest= /opt/oracle/admin/PSOA/adump *.db_recovery_dest='+RECO' *.log_archive_dest_3= 'LOCATION=USE_DB_RECOVERY_FILE_DEST' *.dispatchers=PSOAXDB

4. Connect to the ASM instance on SOADBHOST1 of the standby site, and create a directory within the DATA disk group that has the same name as the DB_UNIQUE_NAME of the standby database. For example:

   ```
   SQL> alter diskgroup data add directory '+DATA/SSOA';
   ```

5. Connect to the standby database on SOADBHOST1 of the standby site, with the standby database in the IDLE state, and create an SPFILE in the standby DATA disk group. For example:

   ```
   SQL> CREATE SPFILE='+DATA/SSOA/spfilepsoa.ora' FROM
   PFILE='?/dbs/initpsoa1.ora';
   ```

6. In the *$ORACLE_HOME/dbs* directory on SOADBHOST1 and SOADBHOST2 of the standby site, create a PFILE that contains a pointer to the SPFILE. The PFILE should follow the naming convention "*init<OracleSID>.ora*". For example:

   ```
   On SOADBHOST1

   $ cd $ORACLE_HOME/dbs

   $ echo "SPFILE='+DATA/SSOA/spfilepsoa.ora'" > initpsoa1.ora


   On SOADBHOST2

   $ cd $ORACLE_HOME/dbs

   $ echo "SPFILE='+DATA/SSOA/spfilepsoa.ora'" > initpsoa2.ora
   ```

7. Create the dump directories on all standby hosts as referenced in the standby initialization parameter file. For example:

   ```
   $ mkdir -p $ORACLE_BASE/admin/psoa/bdump

   $ mkdir -p $ORACLE_BASE/admin/psoa/cdump

   $ mkdir -p $ORACLE_BASE/admin/psoa/udump

   $ mkdir -p $ORACLE_BASE/admin/psoa/adump
   ```

8. On SOADBHOST1 of the standby site, set the ORACLE_HOME, PATH, ORACLE_SID and startup the standby database without mounting the control file. This host should have the staging directory. For example:

   ```
   SQL > startup nomount
   ```

9. From SOADBHOST1 of the primary site where the standby instance was just started, duplicate the primary database as a standby into the ASM disk group by using RMAN. For example:

   ```
   $ rman target sys/oracle@psoa auxiliary /
   ```

RMAN> duplicate target database for standby;

10. Use *sqlplus* to login to the newly created database to validate that it was created correctly. For example:

```
$ sqlplus '/as sysdba'
```

11. Connect to the standby database on SOADBHOST1 of the standby site, and create the standby redo logs to support the standby role. The redo log files should be sized according to the requirements in your environment. The sizes shown below are examples. For example:

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 1

GROUP 5 SIZE 300M,

GROUP 6 SIZE 300M,

GROUP 7 SIZE 300M;


SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 2

GROUP 8 SIZE 300M,

GROUP 9 SIZE 300M,

GROUP 10 SIZE 300M;
```

12. On SOADBHOST1 of the standby site, start managed recovery and real-time apply on the standby database. For example:

```
SQL> ALTER DATABASE recover managed standby database using current
logfile disconnect;
```

13. On SOADBHOST1 and SOADBHOST2 of the standby site, register the standby database and the database instances with the Oracle Cluster Registry (OCR) using the Server Control (SRVCTL) utility. For example:

```
$ srvctl add database -d psoa -o
/opt/maa/oracle/product/10.2.0/db_1

$ srvctl add instance -d psoa -i psoa1 -n soadbhost1

$ srvctl add instance -d psoa -i psoa2 -n soadbhost2
```

14. Establish a dependency between the database and the ASM instance. For example:

```
$ srvctl modify instance -d psoa -i psoa1 -s +ASM1

$ srvctl modify instance -d psoa -i psoa2 -s +ASM2
```

```
$ srvctl enable asm -n stbdd03 -i +ASM1

$ srvctl enable asm -n stbdd04 -i +ASM2
```

15. Configure the Primary database for Data Guard by modifying/adding the Data Guard parameters shown in the table.

```
*.log_archive_config='dg_config=(SSOA,PSOA)'
*.log_archive_dest_2='service=SSOA valid_for=(online_logfiles,primary_role) db_unique_name=SSOA'
*.db_file_name_convert='+DATA/SSOA/','+DATA/PSOA/','+RECO/SSOA','+RECO/PSOA'
*.log_file_name_convert='+DATA/SSOA/','+DATA/PSOA/','+RECO/SSOA','+RECO/PSOA'
*.standby_file_management=auto
*.fal_server='PSOA'
*.fal_client='PSOA'
```

16. Restart the primary database after modifying the parameters.
17. Create the standby redo logs on the primary database to support the standby role. For example:

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 1

GROUP 5 SIZE 300M,

GROUP 6 SIZE 300M,

GROUP 7 SIZE 300M;


SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 2

GROUP 8 SIZE 300M,

GROUP 9 SIZE 300M,

GROUP 10 SIZE 300M;
```

18. Verify the Data Guard configuration by querying the V$ARCHIVED_LOG view to identify existing files in the archived redo log. For example:

```
SQL> select sequence#, first_time, next_time from v$archived_log
order by sequence#;
```

19. On the primary database, issue the following SQL statement to force a log switch and archive the current online redo log file group.

```
SQL> alter system archive log current;
```

20. On the standby database, query the V$ARCHIVED_LOG view to verify that the redo data was received and archived on the standby database.

    ```
    SQL> select sequence#, first_time, next_time from v$archived_log
    order by sequence#;
    ```

Data Guard Setup: Test Database Switchover and Switchback

Test that the switchover and switchback operation works correctly between the newly created physical standby database and the primary RAC databases. Follow the steps below to test the switchover and switchback scenarios.

1. Shutdown all but one instance of the RAC database (PSOA) on the primary site. For example, run the command below on SOADBHOST1 of the primary site:

   ```
   $ srvctl stop instance –d psoa –i psoa2
   ```

2. Initiate the role transition to the physical standby on the current primary database. For example, run the command below on SOADBHOST1 of the primary site:

   ```
   SQL > ALTER DATABASE COMMIT TO SWITCHOVER TO PHYSICAL STANDBY WITH
   SESSION SHUTDOWN;
   ```

3. Shutdown the primary instance and mount the primary instance. For example, run the command below on SOADBHOST1 of the primary site:

   ```
   SQL > shutdown immediate

   SQL > startup mount
   ```

4. At this point, we have both the databases in "Physical Standby". To verify that both the databases are in the "Physical Standby" mode, run the SQL query on both the databases:

   ```
   SQL> select database_role from v$database;

   DATABASE_ROLE

   ----------------

   PHYSICAL_STANDBY
   ```

5. Switch the physical standby database role to the primary role. For example, run the command below on SOADBHOST1 of the standby site:

   ```
   SQL> ALTER DATABASE COMMIT TO SWITCHOVER TO PRIMARY WITH SESSION
   SHUTDOWN;
   ```

6. Now the physical standby database is the new primary.

7. Shutdown the new primary database and startup both the RAC nodes using srvctl. For example, run the following command on the SOADBHOST1 of the standby site:

```
$ srvctl start database –d psoa
```

8. On the new physical standby database (the old primary), start the managed recovery of the database. For example, run the command below on SOADBHOST1 of the primary site:

   ```
   SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT
   FROM SESSION;
   ```

9. Start sending the redo data to the new physical standby database. For example, run the command below on SOADBHOST1 of the standby site

   ```
   SQL> ALTER SYSTEM SWITCH LOGFILE;
   ```

10. Check the new physical standby database to see if it is receiving the archive log files by querying the V$ARCHIVED_LOG view.

Data Guard Setup: Physical Standby for IDM Repository

Set up a physical standby database for the IDM Repository by following the steps detailed in the section above. The Service Name for the IDM database is "idm.mycompany.com". Use the appropriate SID "idmdb" to create the physical standby database for the IDM Repository.

Standby Site Creation

Follow the steps below to set up the middle tier hosts on the standby site

Prerequisite Check

1. On the standby site, make sure the same host names are used for middle tier hosts as were used for the middle tier hosts at the primary site.
2. On the shared storage at the standby site, create the same volumes as were created on the shared storage at the primary site.
3. On the standby site, create the same mount points and symbolic links that you created at the primary site.

Setup

The middle tier hosts on the standby site do not require the installation of any Oracle Application Server or WebLogic Server software. When the primary site storage is replicated to the standby site storage, the software installed on the primary site volumes will be replicated at the standby site volumes. Follow the steps below to setup the middle tier hosts on the standby site:

On the Primary Site:

1. Perform all necessary configurations required by the storage device to enable disk replication.
2. Create a baseline snapshot copy of shared storage on the primary site that sets up the replication between the storage devices. Create the initial baseline copy and subsequent snapshot copies using asynchronous replication mode.
3. Synchronize the shared storage at the primary site with the shared storage at the standby site. This will transfer the initial baseline snapshot from the primary site to the standby site.
4. Set up the frequency of subsequent copies of the production site shared storage, which will be replicated at the standby site. When asynchronous replication mode is used, then at the requested frequency the changed data blocks at the production site shared storage (based on comparison to the previous snapshot copy) become the new snapshot copy, and the snapshot copy is transferred to the standby site shared storage.

On the Standby Site:

1. If not already completed, perform all necessary configurations required by the storage device to enable disk replication.
2. After the baseline snapshot copy is performed, validate that all the directories inside the standby site volumes have the same contents as the directories inside the primary site volumes.
3. Validate the standby site by following the steps in Section 4.7.

Validate Standby Site

Validate the standby site by following the steps below:

1. Shut down any processes still running on the primary site.
2. Break the replication between the primary site shared storage and the standby site shared storage.
3. Use Oracle Data Guard to fail over the databases.
4. On the standby site hosts, manually start up the processes for the Application Server instances.
5. Use a browser client to perform post-failover testing to confirm that requests are being resolved and redirected to the standby site.
6. After validating that the standby site is working, make sure to switch back to make the primary site the primary site.

Site Level Operations

Switchover

Follow the steps below to switchover your primary site to the standby site:

1. Shut down any processes still running on the primary site.
2. Break the replication between the primary site shared storage and the standby site shared storage.
3. Use Oracle Data Guard to switchover over the databases.
4. On the standby site hosts, manually start up the processes for the Application Server instances.
5. Ensure that all user requests are routed to the standby site. This can be achieved through a global DNS push or something similar.
6. Use a browser client to perform post-switchover testing to confirm that requests are being resolved and redirected to the standby site.
7. The standby site is now the new primary site and the primary site is now the new standby site.
8. Reestablish the replication between the two sites, but configure the replication so that the snapshot copies go in the opposite direction (from the current primary site to the current standby site).

Switchback

Follow the steps below to switchback the new primary site (old standby) to the standby site:

1. Shut down any processes still running on the new primary site.
2. Break the replication between the new primary site shared storage and the standby site shared storage.
3. Use Oracle Data Guard to switchback the databases.
4. On the primary site hosts, manually start up the processes for the Application Server instances.
5. Ensure that all user requests are routed to the primary site. This can be achieved through a global DNS push or something similar.
6. Use a browser client to perform post-switchback testing to confirm that requests are being resolved and redirected to the primary site.
7. The standby site is now the new primary site and the primary site is now the new standby site.

8. Reestablish the replication between the two sites, but configure the replication so that the snapshot copies go in the opposite direction (from the current primary site to the current standby site).

Failover

Follow the steps below to failover the primary site to the standby site:

1. Break the replication between the primary site shared storage and the standby site shared storage.
2. From the standby site, use Oracle Data Guard to failover the databases.
3. On the standby site hosts, manually start up the processes for the Application Server instances.
4. Ensure that all user requests are routed to the standby site by performing a global DNS push.
5. Use a browser client to perform post-failover testing to confirm that requests are being resolved and redirected to the primary site.
6. The standby site is now the new primary site and the primary site is the new standby site.

Best Practices

JMS Stores and Transaction Logs

The Oracle WebLogic Server has several subsystems/services; however the focus of this paper is around JMS messages and JTA Transaction Logs (TLOG).

The JMS Messages consist of persistent messages and durable subscribers. The TLOG's consist of information about committed transactions coordinated by the server which may not have been completed.

There are two different types of persistent stores. There are file stores, which are groups of files maintained on a file system. The other is JDBC stores. These stores are maintained in the database.

Either the file store or the JDBC store can survive a process crash or hardware power failure without losing any committed updates. Uncommitted updates may be retained or lost, but in no case will a transaction be left partially complete after a crash.

When configuring the persistent stores, the user has the option of choosing default or custom store. It is important to understand, the similarities and differences between the two configurations of stores.

The file store generally offers better throughput than a JDBC store. In order to obtain better performance with the JDBC store, the database's underlying storage must be high-end, fast, tier one, and the subsequent storage for the WebLogic Server executing on slower hardware. File stores generate no network traffic, but JDBC stores will generate network traffic if the database resides on a different tier other than the WebLogic Server. In production it is commonplace for the database to reside on its own servers behind a firewall.

For high-availability, a persistent file-based store is configured on a shared disk which is available to the migratable target servers in the cluster. It is also recommended, the persistent file-based store utilizes a clustered file system. For this implementation, the clustered file system OCFS2 was chosen. Please see configuration details for OCFS2 in previous section of this document.

There are two methods for configuring a synchronous write policy. They are the Cache-Flush and the Direct-Write policies. The Cache-Flush policy improves performance, but the downside is possibly losing sent messages or generating duplicate messages in the event of an operating system crash or hardware failure. This is due to the fact; transactions are complete as soon as the writes are cached in memory, instead of waiting for acknowledgement the writes are written to disk.

The Direct-Write policy is recommended and implemented in this configuration. It should also be noted, the rate of transfer/replication should be higher than the SOA and WebLogic binaries and configuration files. In this implementation the policy is Direct-Write and rate of transfer/replication is higher.

For additional information on WebLogic Persistent Stores please refer to the Oracle WebLogic Server documentation.

Appendix A1

The tables below show the Volume Layout and Mount Points where storage device's disk replication technology guarantees consistent replication across multiple volumes:

Volume Layout:

VOLUME NAME	TIER	MOUNTED ON NODES	MOUNT POINT	NOTES/COMMENTS
VolWeb	Web Tier	webhost	/u01/app/oracle	Volume for Apache Install.
VolAdmin	App Tier	apphost1	/u01/app/oracle/wls/soaDomain/admin	Volume for Admin server Instances
VolWLS1	App Tier	apphost1	/u01/app/oracle/wls/soaDomain/mng1	Volume for Managed Server Instance
VolWLS2	App Tier	apphost2	/u01/app/oracle/wls/soaDomain/mng2	Volume for Managed Server Instance
VolData	App Tier	apphost1, apphost2	/u01/app/oracle/data	Volume for T-Logs and JMS Data
VolOrcl1	App Tier	apphost1	/u01/app/oracle/product	Volume for Binaries, both Oracle and WebLogic.
VolOrcl1	App Tier	apphost2	/u01/app/oracle/product	Volume for Binaries, both Oracle and WebLogic.

Mount Points:

VOLUME NAME	TIER	NODES	NOTES/COMMENTS
VolWeb	Web Tier	WEBHOST1, WEBHOST2	Volume for Apache Install.
VolAdmin	App Tier	APPHOST1, APPHOST2	Volume for Admin Server
VolWLS	App Tier	APPHOST1, APPHOST2	Volume for Managed Server Instances
VolData	App Tier	APPHOST1, APPHOST2	Volume for T-Logs and JMS Data
VolOrcl	App Tier	APPHOST1, APPHOST2	Volume for Binaries, both Oracle and WebLogic.
VolOID	Data Tier	OIDHOST1, OIDHOST2	Volume for OID Install

Appendix A2

This section provides the volume layout where storage device's disk replication technology does not guarantee consistent replication across multiple volumes.

The table below shows the mount points and symbolic links that need to be created on the mid-tier hosts. These mount points and symbolic links are set up so that the same directory structure can be used on each Application Server host. Using the mount points and symbolic links on the host simplifies the installation, management and maintenance of the Application Server instances for the hosts in the weblogic domain

HOSTNAME	MOUNT POINT DIRECTORY ON TO VOLUME	SYMBOLIC LINK	DIRECTORY ON THE VOLUME
Webhost1	/u02/volwebmount/web1	/u01/app/oracle/apache	/vol/volweb/web1
Webhost2	/u02/volwebmount/web2	/u01/app/oracle/apache	/vol/volweb/web2
Apphost1	/u02/volOrclmount/soa /u02/volOrclmount/wls /u02/volAdminmount/admin /u02/volWLSmount/mng1 /u02/voldatamount/JMS /u02/voldatamount/Tlog	/u01/app/oracle/10.1.3/soa /u01/app/oracle/wls /u01/app/oracle/wls/soaDomain/admin /u01/app/oracle/wls/soaDomain/mng1 /u01/app/oracle/wls/data/Jms /u01/app/oracle/wls/data/Tlogs	/u01/volOrcl/soa /u01/volOrcl/wls /u01/volAdmin/admin /u01/volWLS/mng1 /u01/voldata/JMS /u01/voldata/Tlog
apphost2	/u02/volOrclmount/soa /u02/volOrclmount/wls /u02/volWLSmount/mng2 /u02/voldatamount/JMS /u02/voldatamount/Tlog	/u01/app/oracle/10.1.3/soa /u01/app/oracle/wls /u01/app/oracle/wls/soaDomain/mng2 /u01/app/oracle/wls/data/Jms /u01/app/oracle/wls/data/Tlogs	/u01/volOrcl/soa /u01/volOrcl/wls /u01/volWLS/mng2 /u01/voldata/JMS /u01/voldata/Tlog

Steps to create Mount points & Symbolic Links on the Hosts

This section illustrates the steps create mount points and the symbolic links on the nodes in the web-tier. Also refer to the storage documentation for vendor specific mount point options.

1. Log in as root on WEBHOST1 and create the following directories:

   ```
   prompt> mkdir /u01/app/oracle

   prompt> mkdir /u02/volwebmount
   ```

2. On WEBHOST1, mount the /u02/volwebmount directory to the /vol/volweb volume on the storage, and set up the mount point permissions and ownership as necessary. Refer to vendor-specific information for the shared storage to perform this step.

3. While logged in as root on WEBHOST1, mount the storage volume:

```
prompt> mount /u02/voloidmount
```

4. On WEBHOST1, create the Oracle home directory for the Application Server instance in the storage:

   ```
   prompt> cd /u02/volwebmount

   prompt> mkdir web1
   ```

5. On WEBHOST1, create a symbolic link named apache in the /u01/app/oracle directory to the "/u02/volwebmount/web1" directory on the storage:"

   ```
   prompt> cd /u01/app/oracle

   prompt> ln -s /u02/volwebmount/web1 apache
   ```

6. On WEBHOST1, the following command changes the working directory to the /vol/volweb/web1 directory on the storage.

   ```
   prompt> cd /u01/app/oracle/web1
   ```

7. Follow the steps shown above to create the mountpoints and symbolic links on all the other mid tier nodes by listed in the table.

References

1. Oracle Maximum Availability Architecture web site
2. Oracle Database High Availability Overview (Part #B14210)
3. Oracle Database High Availability Best Practices (Part B25159)
4. Oracle Application Server High Availability Guide 10g Release3
5. Oracle Application Server Enterprise Deployment Guide 10g Release4
6. Oracle Application Server Disaster Recovery Guide 10g Release 3

White Paper Title
June 2009
Author: Bharath K Reddy
Contributing Authors: Pradeep Bhat, Shailesh Dwivedi, Susan Kornberg

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

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

0109