Oracle Maximum
Availability Architecture

Identity Management 11.1.2.3 Multi Data Center Active-Active

Best Practices

ORACLE WHITE PAPER | JANUARY 2017
VERSION 1.0

Table of Contents

Executive Overview

Business continuity is a key requirement for many e-business operations. Downtime of mission-critical applications translates directly into reduction in productivity, service quality, and lost revenue. Mission-critical application services require both a local high availability solution and a disaster recovery solution. A local high availability solution provides redundancy in one data center. Additionally, applications need protection from unforeseen disasters, natural calamities, and downtime that can affect an entire data center. An effective disaster that disables an application service is not necessarily one that destroys the whole data center (e.g. flood, fire), but is more likely to disable one particular type of resource. For example, a failure of corporate gateways or ISP network connections, a spread of viruses to all HTTP listener nodes, a misconfiguration, a power outage, or an incorrect patch could all lead to days of complete loss of services. The same applies to planned outages: a network infrastructure update, a firewall upgrade, etc. may have similar downtime effects in a datacenter. In an Identity and Access Management (IAM) architecture multiple corporate systems will depend on the IAM functionality to be available continuously in order to access other systems. As the adoption of these architectures grow, so does the need for failure and downtime protection not only in the scope of a single machine, but also against events that may bring down a group of machines, an entire room or an entire building. Traditional disaster protection systems use a model where one site is running while another site is on standby in prevention of possible failover scenarios (also called Multi Data Center Active-Passive or Active-Passive Disaster Protection). Such approaches usually incur increased operational and administration costs, while the need for continuous use of resources and increased throughput (i.e. avoiding situations where the standby machines are idle) have increased through the years. IT systems' design is increasingly driven by capacity utilization and even distribution of load, which leads to the adoption of disaster protection solutions that use, as much as possible, all resources available (called Multi Data Center Active-Active or Active-Active Disaster Protection).

This paper describes the recommended Active-Active solutions that can be used for protecting an Oracle Fusion Middleware 11*g* IAM system against downtime across multiple locations (referred to as IAM Active-Active Disaster Recovery Solution or IAM Multi Data Center Active-Active Deployment) It provides the required configuration steps for setting up the recommended topologies and guidance about the performance and failover implications of such a configuration.

Introduction

Paradigms for Designing a Multi Data Center Active-Active Deployment for Oracle Fusion Middleware IAM

There are multiple factors that can drive the design of a Multi Data Center Deployment. The following are usually considered:

Availability: RTO and RPO

Disaster Recovery designs need to minimize the Recovery Point Objective (RPO) and the Recovery Time Objective (RTO) metrics. RPO measures the amount of data that can be lost in the event of a failure while RTO measures the time the system will be unavailable should a failure occur.

The main advantage of a Multi Data Center Active-Active Deployment system as compared to traditional Multi Data Center Active-Passive Disaster Recovery design is that in the event of complete middle tier failure in one site (all middle tier servers in one location), the system can fulfill requests because there are middle tiers in the peer site that remain available. In other words, RTO and RPO for Multi Datacenter Active-Active Deployments are null in this type of scenario. For this, the middle tier servers in the alternative location need to be able to sustain the combined load of all locations. The appropriate capacity planning must be done to account for such scenarios. Depending on the design, requests from end clients may need to be throttled when only one site is active. Otherwise, sites must be designed with exceeding power, hence partially defeating the purpose of constant and efficient capacity usage.

Oracle IAM can be divided into two different categories each with different availability requirements.

- Access Management (OAM), which is used to collect credentials and grant access to other system resources.
- Provisioning/Identity Management (OIM), which is the ability to create new accounts and grant access, rights to those accounts.

Many organizations use both products, which they integrate together to provide a complete solution. However, the approach to achieve and Active/Active deployment in each case is different. OAM has had a multi-datacenter solution since PS2, but this has been a standalone solution, which has not had the ability to integrate with OIM.

The two product groups therefore need to be treated differently:

- OAM uses two independent databases and proprietary OAM replication technologies to keep those databases in sync.
- OIM has no dedicated MDC solution and therefore must use a solution similar to that of the traditional DR solution. That is to say that it will use a single database, which is replicated, to the DR site. All writes to the database will go to the active site whichever it is.

When a failure occurs in the OIM database tier, both Multi Data Center Deployment Active-Active and Multi Data Center Active-Passive present similar RTO and RPO since the database is the driver for recovery and in both cases it is active only in one Site 1nd passive in the other. The only advantage of Multi Data Center Active-Active Deployment systems is that an appropriate Data Source configuration can automate the failover of database connections from the middle tiers, reducing RTO (the recovery time is decreased because restart of the middle tiers is not required)[1].

When a failure occurs in the OAM database tier, there is no discernable system impact, as the surviving database will continue to process requests.

Whilst the OAM solution will be active/active the OIM solution may be Active/Passive.

It is worthwhile noting that whilst OAM and OIM can be treated differently, then can also be treated the same if OAM is handled in the same manner as OIM. That is to say that both sites, use a single OAM database with all writes being directed to that database. Having a single integrated solution in this manner is easier to maintain and manage. However, the adoption of the distinct OAM Active/Active solution allows the two data centers to be further apart. Please note however, even in an distributed OAM Active/Active solution, OAM policy changes can only be made on one instance, this instance is designated the “master” instance.

Performance

Besides the common performance paradigms that apply to single-datacenter designs, Oracle Fusion Middleware IAM Multi Data Center Active-Active systems need to minimize the traffic across sites to reduce the effect of latency on the system’s throughput. In a typical Oracle Fusion Middleware Integrated IAM System, besides database access (for dehydration, metadata access, and other database read/write operations that custom services that participate in the system may perform), communication between the different tiers can occur mainly over the following protocols:

1 The Oracle WebLogic Servers may need to be restarted depending on different aspects. Server migration, for example may, trigger a server shutdown. When using database leasing, Oracle WebLogic Servers may shut down if the database remains unavailable (during switchover or failover) for longer periods than their server migration fencing times.

- Incoming HTTP invocations from Load Balancers (LBR) or Oracle Web Servers (OHS/OTD) and HTTP callbacks
- Incoming HTTP invocations between OAM and OIM
- JNDI/RMI and JMS invocations between Oracle WebLogic Servers
- OAP requests between Web Servers and OAM

For improved performance, all of the above should be restrained, as much as possible, to one single site. That is, servers in SiteN ideally should just receive invocations from Oracle Web Servers in SiteN. They should make JMS, RMI and JNDI invocations only to servers in SiteN and should get callbacks generated by servers only in SiteX. Additionally, servers should use storage devices that are local to their site to eliminate contention (latency for NFS writes across sites may cause severe performance degradation). Only if a component in SiteN is available should a request be sent to an alternate site.

There are additional types of invocations that may take place between the different IAM servers that participate in the topology:

» Oracle Coherence notifications: Oracle Coherence notifications need to reach all servers in the system to provide a consistent compoSite 1nd metadata image to all SOA requests, whether served by one site or the other.
» HTTP session replications: some Oracle Fusion Middleware IAM components use stateful web applications that may rely on session replication to enable transparent failover of sessions across servers. Depending on the usage patterns and number of users this may generate a considerable amount of replication data. Replication and failover requirements have to be analyzed for each business case, but ideally session replication traffic should be reduced across sites as much as possible.
» LDAP/policy/identity store access: Access to policy and identity stores is performed by Oracle WebLogic Server infrastructure and IAM components for authorization and authentication purposes. In order to enable seamless access to users from either site, a common policy or identity store view needs to be used. Ideally each site should have an independent identity and policy store that is synchronized regularly to minimize invocations from one site to the other.

Administration

Another key aspect of the design and deployment of an Oracle Fusion Middleware IAM Multi Data Center Deployment is the **administration overhead** introduced by the solution. In order to keep a consistent reply to requests, the sites involved should use a configuration such that the functional behavior of the system is the same irrespective of which site is processing those requests. Oracle

Fusion Middleware IAM keeps its configuration and metadata in the Oracle database. It is for this reason that Multi Data Center Active-Active Deployments with a unique active database guarantee consistent behavior at the composite and metadata level (there is a single source of truth for the involved artifacts). The Oracle WebLogic Server configuration, however, is kept synchronized across multiple nodes in the same domain by the Oracle WebLogic Server infrastructure. Most of this configuration usually resides under the Administration Server's domain directory. This configuration is propagated automatically to the other nodes in the same domain that contain Oracle WebLogic Servers. Based on this, the administration overhead of a Multi Data Center Active-Active Deployment system is very small as compared to any active-passive approach where constant replication of configuration changes is required.

Fusion middleware binaries across all sites must be the same, the same location and the same patches applied. This can be achieved by independent installation or by by disk mirroring. If using disk mirroring be sure to ensure that at least two different versions are available. That way a corrupt patch will only impact half of the deployment (in a site) and that the binary corruption is not replicated to the DR site. For example:

- FMW binary set 1 – SiteAHost1, SiteAHost3, SiteAHost5
- FMW binary set 2 – SiteAHost2, SiteAHost4, SiteAHost6

FMW binary set 1 and 2 replicated to Site 2 and mounted to

- FMW binary set 1 (copy) – SiteBHost1, SiteBHost3, SiteBHost5
- FMW binary set 2 (copy) – SiteBHost2, SiteBHost4, SiteBHost6

Latency, Jitter, Packet Loss and Bandwidth Across Sites

The overall network throughput of an Oracle Fusion Middleware OIM Multi Data Center Active-Active Deployment system is primarily driven by two factors: the length of the route that the requests have to take between the different sites (mainly for database access) and the interaction between the TCP reliability and congestion control protocols. Regardless of the speed of the processors where Oracle Fusion Middleware OIM runs or the efficiency of the software, it takes a finite amount of time to manipulate and "present" data from one site to the other. Two important measurements of time intervals in network transmission systems are referred to as **latency** and **jitter**. Network latency is the

amount of time it takes for a packet to be transmitted end-to-end across a network, and it is composed of multiple variables (the type and number of switches between sites, the type of cabling, etc.) Latency in a network is measured either one-way (the time from the source sending a packet to the destination receiving it), or round-trip (the one-way latency from source to destination plus the one-way latency from the destination back to the source). Round-trip-time (RTT) latency is used more frequently because it provides a more realistic figure of the delay (accounts for traffic in both directions) and can be measured with the *ping* utility in most systems. Jitter is a term that refers to the variance in the arrival rate of packets from the same data flow. Both latency and jitter have a negative impact on applications with communications across sites. They are critical for the appropriate behavior of an Oracle Fusion Middleware OIM Multi Data Center Active-Active Deployment. Jitter, however, is typically more relevant in systems with extremely low latency. Thus, latency is effectively the main aspect that must be controlled in a Multi Data Center Active-Active Deployment. The main causes of latency are:

- propagation/distance delay
- serialization
- data protocols
- routing and switching
- queuing and buffering

Of all of the above causes, distance delay is typically the most relevant one. Distance delay is the minimum amount of time that it takes the electrical signals that represent bits to travel on a physical wire. Optical cable sends bits at about ~5.5 µs/km, copper cable sends it at ~5.606 µs/km, and satellite sends bits at ~3.3 µs/km. Distance delay can have a significant impact on the performance of an Oracle Fusion Middleware OIM Multi Data Center Active-Active Deployment because multiple network round trips (mainly from the Oracle Fusion Middleware OIM/SOA servers to the OIM database) are required to complete each composite instance. Tests conducted have shown that an Oracle Fusion Middleware OIM Multi Data Center Active-Active Deployment's performance (where Oracle WebLogic Server OIM/SOA servers use a database in a different site) degrades considerably when latency exceeds 5-10 milliseconds. This also applies if OAM is using the same single database solution as OIM. If the distributed OAM solution is used then network latency can be higher (for the OAM component).

Supportability Scope

Oracle Identity and Access Management contains a lot of discrete products. The solution detailed in this paper is restricted to the following IAM components:

- Oracle Access Manager
- Oracle Identity Manager including SOA and BI
- Oracle Unified Directory

The following products are typically deployed alongside the above products. This solution DOES NOT support the usage of the following components.

- Oracle Adaptive Access Manager (OAAM)
- Oracle Mobile Security Suite (OMSS)

Requirements

Topology

The analysis and recommendations included in this paper are based on the topology described in the "Topology Model for an Oracle Fusion Middleware IAM Active-Active Multi Data Center Deployment" section. Each site locally uses a slightly modified version of the Oracle Fusion Middleware IAM Enterprise Deployment Topology. The system requirements are those specified in the Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity and Access Management 11.1.2.3.3

Additionally, the following requisites must be met:

Entry Points

Access and Identity will have different entry points as described in the IAM Enterprise Deployment guide, these entry points will be:

- login.example.com for Access
- prov.example.com for Identity

Database

Separate Databases will exist for OAM and OIM, unless OAM and OIM are using the same active/passive database solution in which case they can be the same.

In order to facilitate a seamless change over when the OIM database changes its role from standby to primary, a separate database service must be used for connecting to the database. This is described in the Creating Database Services section of the Oracle Fusion Middleware Enterprise Deployment Guide for Oracle IAM.

Shared Storage vs. Database for Transaction Logs and Persistent stores

The topology addressed in this paper was tested using database-based persistent stores for Oracle WebLogic Server transactions logs and Oracle WebLogic Server JMS persistent stores. Storing transaction logs and persistent stores in the database provides the replication and high availability benefits inherent from the underlying database system. With JMS, TLOG, and OIM/SOA data in a Data Guard database, cross-site synchronization is simplified and the need for a shared storage sub-system such as a NAS or a SAN is alleviated in the middle tier (they still apply for the Administration Server's failover). Using TLOGs and JMS in the database has a penalty, however, on the system's performance.

As of Oracle FMW 11g, the retry logic in JMS JDBC persistent stores that takes care of failures in the Database is limited to a single retry. If a failure occurs on the second attempt, an exception is propagated up the call stack and a manual restart of the server is required to recover the messages associated with the failed transaction (the server will go into FAILED state due to the persistent store failure). To overcome this, it is recommended to use Test Connections on Reserve for the pertaining DataSources and also configure in-place restart for the pertaining JMS Server and Persistent stores. Refer to Appendix B for details on configuring in-place restart.

Load Balancers

Load balancers from any vendor are supported as long as the load balancer meets the requirements listed in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity and Access Management* chapter 6. The global load balancer should allow rules based on the originating server's IPs (an example is provided for F5 Networks).

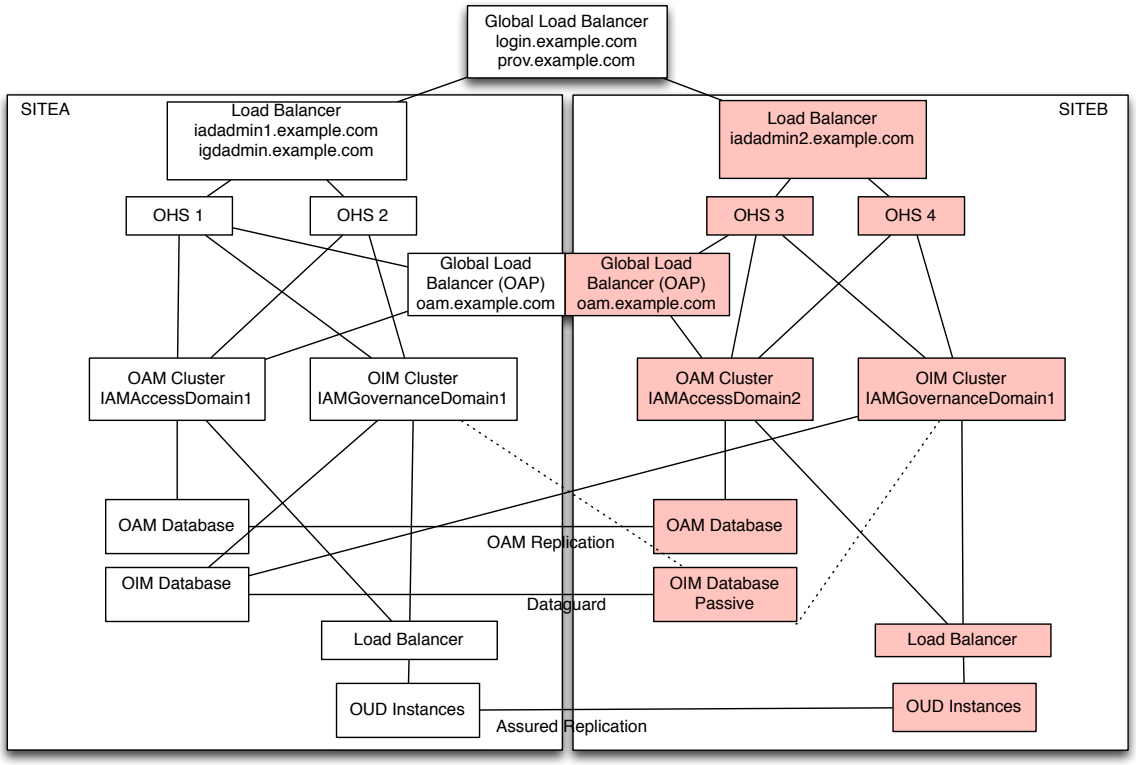
Hardware Resources and Capacity Utilization

A Multi Data Center Active-Active Deployment is usually designed to make effective use of resources available in multiple sites. However, the appropriate capacity planning needs to be done to account for failover scenarios between the two sites. If an entire site loses the middle tiers, the other must be designed to sustain the added load, or the appropriate request throttling and rejection mechanisms must be enabled (typically in the GLBR). Otherwise, cascade failures (where the failover causes such an overhead on the available site that it is rendered unresponsive) may occur. This implies that during normal operation the middle tier nodes must remain underutilized to an extent that will vary depending on the capacity that needs to be available in failover situations.

Topology Model for an IAM Active-Active Multi Data Center Deployment

IAM Multi Datacenter Topology

The following image depicts the main pieces of the Oracle Fusion Middleware IAM Multi Data Center Active-Active Deployment addressed in this white paper.

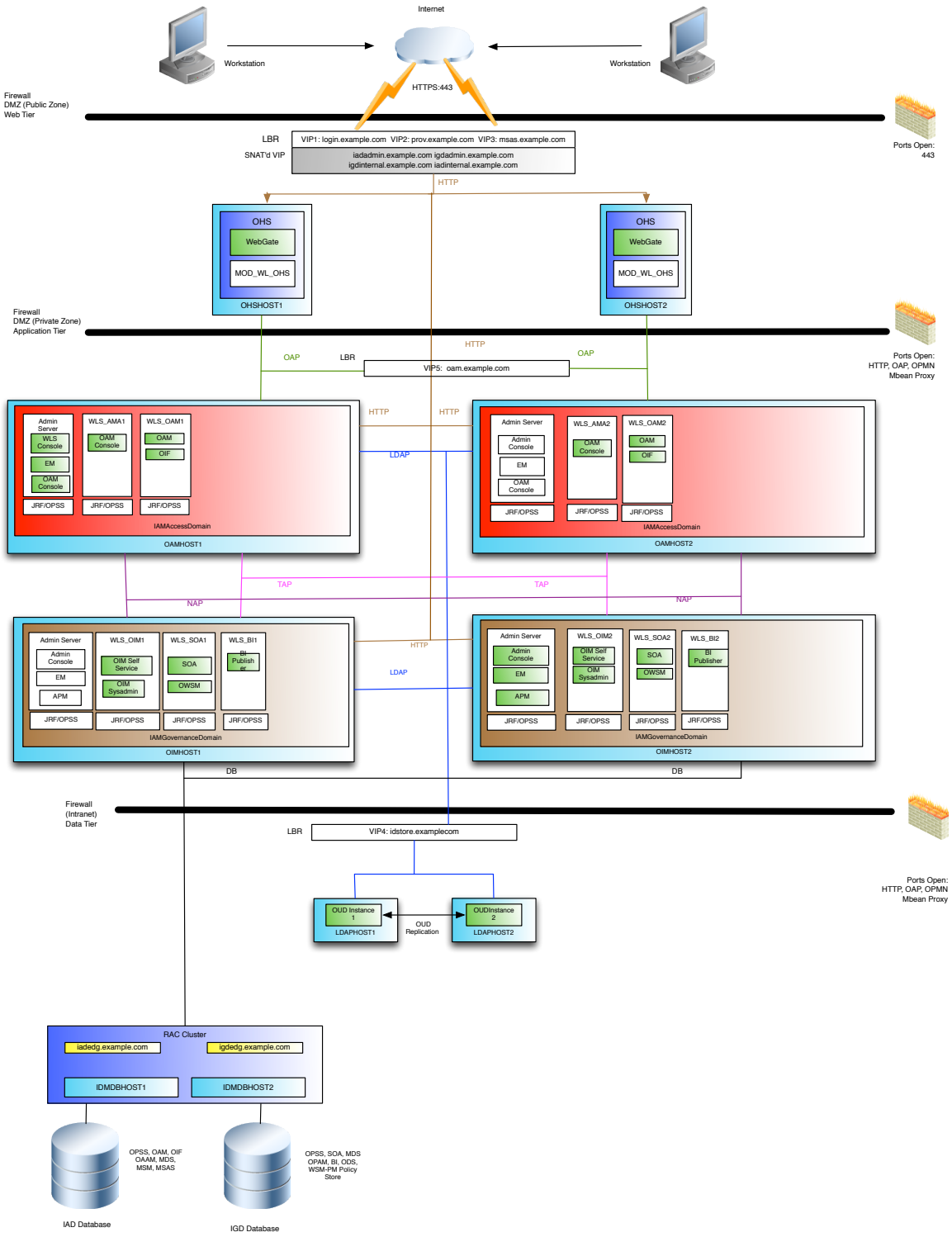


In the Image there are two separate sites (Site1 and Site2 for future reference in this document) that are accessed by one unique access point: a global load balancer which directs traffic to either site (each vendor provides different routing algorithms). Each site has its own local access point – a local load balancer. The local load balancer distributes requests to multiple Oracle HTTP Servers (OHS). Finally, the local HTTP servers allocate requests to specific Oracle WebLogic Servers hosting Oracle Fusion Middleware IAM components.

- Each OAM implementation accesses a local database which is read/write, the databases are kept in sync using OAM replication.
- The OIM implementation shares one unique database that is accessed CONCURRENTLY by servers in both sites. The following sections provide details for each tier.
- Each site has multiple LDAP servers which are kept in sync using OUD replication.
- Each site has a dedicated administration server for the local OAM deployment.
- Site1 has an administration server for the entire OIM deployment which spans both sites.

IAM Enterprise Deployment Topology

The following diagram shows the Identity and Access Management Enterprise Deployment topology featuring an additional load balancer.



This diagram is the same as the diagram depicted in the Oracle Identity and Access Management Enterprise Deployment guide, with the following differences.

A full description of the diagram can be found in the Oracle Identity and Access Management Enterprise Deployment guide.

Mobile Security Manager has been removed from the diagram.

An additional load balancer has been placed between the Oracle HTTP servers and the OAM Managed servers; this load balancer is used for OAP connections only. It provides a level of abstraction that allows OIM/Webgate to communicate with the OAM managed servers without having to know the individual servers in the cluster and where they are located.

Directory Tier

This solution has been tested using Oracle Unified directory (OUD), there is however no reason why the solution will not work with Oracle Internet Directory (OID). OUD is a loosely coupled deployment. Data is replicated between the OUD instances using OUD replication. The second site is just an extension of that principle with the remote OUD instances becoming part of the OUD replication configuration.

If you were using OID then your primary site will be clustered. Your secondary site could extend the cluster but this is likely to have a performance overhead. Another approach is to use OID replication to setup the second site. OID is out of the scope of this document, but OID replication is covered in the Oracle High Availability Guide and the OID documentation. There are however issues when using a replicated OID deployment. OIM uses a process called LDAPSync which is used to keep the entries inside its internal database in sync with the entries in LDAP. To prevent the process having to process every entry in LDAP each time, it uses a changelog which allocates a change number to every transaction in the directory. If you have an OID replicated environment, then the change numbers in each OID cluster are different. This causes issues for failover. You can alleviate this by:

1. Ensuring LDAPSync (OIM reconciliation jobs) only run against a single OID cluster.
2. If you do need to failover to a second OID cluster, then you will need to:
    a. Disable the incremental OIM reconciliation tasks.
    b. Run a full Reconciliation against the new OID cluster.
    c. Update the OIM change number to reflect that of the new OID cluster.
    d. Re-enable incremental OIM reconciliation tasks.

This is not necessary for OUD based solutions which use a cookie based changelog.

Database Tier

Two different databases are required to support Oracle Fusion Middleware IAM Suite. This is due to the way that changes are propagated between the sites.

The synchronization of OAM data is performed using proprietary OAM technology. This technology effectively unloads data from the primary site transfers it to the secondary Site and applies it to the database on that site, using sql commands. The databases on the primary and secondary sites are independent and are both open read/write.

The synchronicity requirements and data types used by the different OIM components limit the possible approaches for the Oracle Fusion Middleware OIM database in a Multi Data Center deployment. This document addresses only a solution where the Oracle Fusion Middleware database used for OIM uses Data Guard to synchronize an active database in Site1 with a passive database in Site2. Although other approaches may work they have not been tested and certified by Oracle and are out of the scope of this document. In this configuration we assume that both sites

where Oracle Fusion Middleware OIM is deployed access the same database (as well as the same schemas within that database), and the database is set up in a Data Guard configuration. Data Guard provides a comprehensive data protection solution for the database. It consists of a standby Site 1 at geographically different location than the production site. The standby database is normally in passive mode; it is started when the production site (called "production" from the database activity point of view) is not available[2].

The Oracle Databases configured in each Site 1 are in an Oracle Real Application Cluster (RAC). Oracle RAC enables an Oracle database to run across a cluster of servers in the same data center, providing fault tolerance, performance, and scalability with no application changes necessary.

In order to facilitate the smooth transformation of database transactions from one site to the other, a role based database service is created on both the primary and standby database sites. A role based service is only available when the database is running in the primary role, that is to say that the database is open read/write. When a standby database becomes a primary then the service is automatically enabled on that side.

By configuring the WebLogic data sources to use this role based service and making those data sources aware of both sites, then no WebLogic reconfiguration is required when the primary database moves between sites.

Oracle Fusion Middleware Tier

Load Balancers and Web Servers

The Global Load Balancer (GLBR) is a load balancer configured to be accessible as an address by users of all of the sites and external locations. The device provides a virtual server which is mapped to a DNS name that is accessible to any client regardless of the site they will be connecting to. The GLBR directs traffic to either Site based on configured criteria and rules. These criteria can be based on the client's IP for example. This should be used to create a Persistence Profile which allows the GLBR to map users to the same site upon initial and subsequent requests. The GLBR maintains a pool, which consists of the addresses of all the local load balancers. In the event of failure of one of the sites, users are automatically redirected to the surviving active site.

At each site, a Local Load Balancer receives the request from the GLBR and directs requests to the appropriate HTTP server. In either case, the Local Load Balancer is configured with a persistence method such as Active Insert of a cookie in order to maintain affinity and ensure that clients are directed appropriately. To eliminate undesired routings and costly re-hydrations, the GLBR is also configured with specific rules that route callbacks only to the LBR that is local to the servers that generated them. This is useful also for internal consumers of IAM services. These GLBR rules can be summarized as follows:

- If requests come from Site1 (callbacks from the IAM servers in Site1 or endpoint invocations from consumers in Site1) the GLBR routes to the LBR in Site1.
- If requests come from Site2 (callbacks from the IAM servers in Site2 or endpoint invocations from consumers in Site2) the GLBR routes to the LBR in Site2.
- If requests come from any other address (client invocations) the GLBR load balances the connections to both LBRs.

2 The Oracle Active Data Guard Option available with Oracle Database 11*g*+ Enterprise Edition enables you to open a physical standby database for read-only access for reporting, for simple or complex queries, or sorting while Redo Apply continues to apply changes from the production database. Oracle Fusion Middleware IAM does not support Oracle Active Data Guard because the IAM components execute and update information regarding IAM composite instances in the database as soon as they are started.

» Additional routing rules may be defined in the GLBR to route specific clients to specific sites (for example, the two sites may provide difference response time based on the hardware resources in each case).

Application Layer

Each site runs from an Oracle Fusion Middleware IAM installation that is "local" to that site (that is, in a file system located nearby the servers) Each local topology uses an Oracle Fusion Middleware IAM Enterprise Deployment Topology for maximum availability and security. Each site must use the same software version including applied patches. Other topologies based on the required high availability principles are allowed. The Oracle WebLogic Server Domain model used in this paper uses one single domain (per site) for Access and One single domain for Governance (OIM).

The Governance domain uses one single cluster for Oracle Fusion Middleware OIM Suite components. This model is also known as a Stretched Cluster. In this topology, all servers (OIM, BI and SOA) are part of a unique Oracle WebLogic Server Domain. They are managed with a single Administration Server that resides in one of the two sites. A database is used as persistent store. A unique Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control provide a central administration point for all the servers. The SOA servers in both sites are part of a unique cluster (SOA_Cluster) and so are the OIM (OIM_Cluster) and BI (BI_Cluster) ones. The Coherence cluster used for composite deployments and MDS updates is also the same one for the two sites. A single database is used for OIM and all Oracle WebLogic Servers point to the same OIM/SOA/BI and MDS schemas. Image 8 describes the topology.

There are separate Access Domains in each site. The two domains are completely independent with content from one updated to the other via OAM replication. Each domain has its own WebLogic and Enterprise Manager FMW control console. Note, whilst there are multiple domains one of those domains will be assigned the "Master Role", policy updates will be applied to that site only.

Characteristics of the Design

By separating the domains it is possible to treat the multi-site implementations of Identity and Access differently. Access can use the proprietary Multi site technologies built into the product. Governance can use either the Active/Passive approach where network latency is high, or if that network latency is low then be active/active as well. If running Active/Passive the OIM components in site 2 will be shutdown until required. The GLBR will be configured to send traffic only to the active site.

Each OAM domain will have a distinct entry point for administrative functions.

Failover of the OIM administration server can be accomplished using disk based replication of the *IGD_ASERVER_HOME* directory and a Virtual IP address (VIP) which can be moved between sites.

Availability (Web Tier): The OHS configuration is based on a fixed list of servers in each site (instead of the "dynamic" list, provided by the OHS plug-in and used in typical single-location deployments). This is done to eliminate undesired routing from one site to another. This has the disadvantage of slower reaction times to failures in the Oracle WebLogic Servers.

If you are using Oracle Exalogic then the web functionality will be provided by Oracle Traffic Director, this will be configured in the same way as OHS using a fixed list of servers. If you are deploying on Exalogic then all components in the topology need to be configured using the EoIB network. This is because components in site 1 will be talking directly to components in site 2. This would not be feasible if the IPoIB network were to be used.

Note: It is possible to have site 1 using OTD on Exalogic and site 2 using OHS on commodity hardware.

Characteristics of the OAM design

Availability: Because each site is independent there is little impact on operations. In an OAM MDC one site it she nominated master, if that site fails then the master role has to be passed on to site 2. This will effect the creation of policies only. Runtime will not be effected. There are however design considerations to be made for runtime.

If a request is authorized at site 1 and site 1 becomes unavailable, then you have the option of forcing the user to re-authenticate at site 2 or for site 2 to just accept the authentication that has already occurred.

Administration: In a Multi Data Center Active-Active Deployment each site is independent of the other, the OAM replication mechanism will take care of the replication of OAM data, but WebLogic/application configuration will need to be performed at each site independently. OAM does not use runtime artifacts (with the exception of the authentication cookie) so this simplifies the process, however having multiple independent configurations does increase the administration overhead.

Performance: If the appropriate load balancing and traffic restrictions are configured (see following sections) the performance of OAM across sites should be similar to that of a cluster with the same number of servers residing in one single site.

Load Balancing: In the OAM deployment, the load balancer virtual hosts are as described in *the Enterprise Deployment Guide* with the following differences:

- login.example.com is configured both locally and at the Global Load Balancer level with location affinity.
- iadadmin.example.com is unique to each site, that is to say there is iadadminsite1.example.com and iadadminsite2.com.
- oam.example.com this is an additional load balancer entry point which is resolvable in each site. This virtual host routes requests to the OAM Proxy port on the OAM Managed servers for example 5575. This load balancer entry point is also configured at the GLBR level and distribute requests to each of the OAM Managed servers in both MDC domains with location affinity. The advantage of configuring it at the GLBR level is that if the managed servers in site 1 become unavailable but the web tier is available then authentications can still happen. The down side to this approach is that it will generate a lot of cross site traffic. As each data center is HA in its own right, it is unlikely that just the two hosts are effected by an outage. It makes more sense if both managed servers are down to redirect all traffic to the second site. To do this you will need to configure the login.example.com monitoring service to monitor not just the web servers but the availability of the OAM service.

  Note: Because OAP requests are being handled by the load balancer a delay can result whilst the load balancer detects that an OAM server is not available.

Characteristics of the OIM design

Availability: The database connection failover behavior and the JMS and RMI failover behaviors are similar to those that take place in a standard Enterprise Deployment Topology. There is, at all times, one single CLUSTER_MASTER server, that is, just one server among all the available servers in the Multi Data Center Active-Active Deployment is able to perform automatic recovery. Instances can be recovered equally from Site1 and Site2 should a failure occur on the partner site.

1. From Site1 when Site2 is up if the CLUSTER MASTER resides in Site1
2. From Site2 when Site1 is up if the CLUSTER MASTER resides in Site2
3. From Site1 when Site2 is down
4. From Site2 when Site1 is down

Should a failure occur in Site1 that affects all of the middle tiers, recovery of the Administration Server is required to resume the Oracle Enterprise Manager Fusion Middleware Control and the Oracle WebLogic Server Administration Console.

Those servers that are remote to the Administration Server take longer to restart than in a regular Enterprise Deployment Topology. The reason is that all the communications with the Administration Server (for retrieving the domain configuration upon start) and initial connection pool creation and database access is affected by the latency across sites.

From the RPO perspective, transactions that were halted by a site failure can be resumed in the site that remains available by manually starting the failed servers in it. Automated server migration across sites is not recommended unless a database is used for JMS and TLOG persistence, otherwise a constant replica of the appropriate persistent stores needs to be set up between the sites. It is also unlikely (depending on the customer's infrastructure) that the Virtual IPs used in one Site are valid for migration to the other. It usually requires additional intervention to enable a listen address initially available in Site1 in Site2 and vice versa. This intervention can be automated in pre-migration scripts, but in general the RTO will increase compared to a standard automated server migration (taking place in the scope of single data center).

Performance: If the appropriate load balancing and traffic restrictions are configured (see following sections) the performance of a stretched cluster with low latency across sites should be similar to that of a cluster with the same number of servers residing in one single site. The configuration steps provided in the following sections are intended to constrain the traffic inside each site for the most common and normal operations. This isolation, however, is non-deterministic (for example, there is room for failover scenarios where a JMS invocation could take place across the two sites). That said, most of the traffic takes place between the Oracle Fusion Middleware OIM Servers and the database. This will be the key to the performance of the Multi Data Center running in an Active-Active scenario.

If the sites are separated by a high latency network, then OIM should be run Active/Passive to avoid significant performance degradation.

Administration: In a Multi Data Center Active-Active Deployment the Oracle WebLogic Server infrastructure is responsible for copying configuration changes to all the different domain directories used in the domain. The Coherence cluster configured is in charge of updating all of the servers in the cluster when composites or metadata are updated[3]. Except for the replication requirement for runtime artifacts across file systems a Multi Data Center Active-Active Deployment is administrated like a standard cluster. This makes its administration overhead very low.

Configuring the Oracle Fusion Middleware IAM Active-Active Topology

The following sections provide the steps for configuring an Oracle Fusion Middleware IAM Multi Data Center Active-Active Deployment. Basic understanding of the common Oracle WebLogic Server administration tasks as well as familiarity with the procedures and configuration steps included in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity and Access Management* (EDG) is assumed. The steps are very similar to those described in the guide, but specific configuration changes are applied in different sections of the EDG to minimize traffic across sites.

NOTE: The EDG provides steps to build the deployment manually or via automated provisioning (IDMLCM), you need to follow the manual steps with the changes below.

In summary the steps are:

3 See the sections related to Composite Deployment and MDS Updates for details on the possible effects of latency in the system from the administration perspective.

1. Configure GLBRs and LBRs as per the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle IAM* but with the appropriate rules for local routing.
2. Configure an additional Load Balancer entry point for OAM OAP calls.
3. Configure OHS as per the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle IAM Suite* but with routing restricted to each site.
4. Configure the application tier with special steps for the following:
   - OAM and OIM must use separate entry points.
   - Shared storage/directory configuration
   - Server migration configuration
   - JMS/TLOG Database configuration
   - Data Source configuration
   - Integrate OIM with OAM using the new OAM LBR entry point.
   - Depending on whether the latency between sites is approaching the 10 msecs limit, adjust Oracle Coherence settings, Oracle Net settings, and JTA/Timeout settings.

The sections that follow detail each of these aspects.

Configuring Load Balancers and Global Load Balancers

As indicated in previous sections, the Global Load Balancer (GLBR) is responsible for performing smart routing of requests between multiple Local Load Balancers. This smart routing is usually done based on the originating request. In an Oracle Fusion Middleware IAM Multi Data Center Active-Active Deployment it is recommended that you restrain callbacks and invocations that come from servers in a specific site to the same site again. Because the GLBR is typically located in one of the two sites (physically) this also makes the invocations to such a site more efficient. The following procedures provide an example of configuration for F5's products.

Configuring the Local Load Balancer

The Local Load Balancers (LBR) receive requests from the Global Load Balancer and send requests to the Oracle HTTP Servers. Each LBR should be configured as indicated in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity and Access Management chapter 6*.

Configuring the Global Load Balancer

The following procedure is specific to F5 BIG-IP Global Traffic Manager (GTM) and LBR. The procedure is provided as an example of the configuration required. Refer to the F5 knowledge base or to your GTM's specific documentation for details[4].

1. It is assumed that the appropriate listener already exists in the GTM
2. In the DNS - GSLB – Datacenter list, menu of the F5 Administration Console, create two Data Centers, one for each site participating in the Multi Data Center Deployment configuration, a data center defines the servers (Local Load balancers) and links that share the same subnet on the network) The defaults are appropriate.
3. In the DNS – GSLB – Data Center – Servers Menu of the F5 Administration Console create a server for each site (assuming one LBR per site) and assign it to the appropriate site (a server defines a specific physical Load balancer system on the network) as follows:

   Name: Name of the server for example Datacenter1_LBR
   For Product, select BIG-IP System (single).
   Use the address of the first site's Local LBR for this server.
   Data center use the data center you created in Step 2.
   Use the appropriate health monitor for the Server (this may be a TCP monitor or a combination of multiple monitors, depending on the services the Local LBR is running).

   Note: if the latency across sites is high, you may want to use a different monitor depending on the site (a more permissive probe may be needed for high latencies).

   In the virtual server list, list the names of the virtual servers that are defined at the target datacenter. For example VS1_prov_example_com. These are the names of the load balancer virtual servers you created when you configured the load balancer for the Enterprise Deployment.

4. In the DNS – GSLB - Pools Menu of the F5 Administration Console create a new pool for each data center (for future reference we will call it the MDCPool1 and MDCPool2). A pool represents one or more virtual servers that share a common role on the network. A virtual server, in the context of GTM, is a combination of IP address and port number that points to a specific resource on the network.

   You need to create one pool for each load balancer entry point/Datacenter for example:

   Mdc1_login_example_com

   Mdc2_login_example_com

   Use the appropriate health monitor for the server (HTTP or HTTPS exists with the device's factory configuration) according to your system's protocol in the Local LBR (this typically would be HTTP).

   Assign as members the virtual servers created in the previous steps. This monitor should be the most permissive one of the two monitors used for the sites.

   For example:

4 The redundancy and DNS server configuration required for providing redundancy for GTM servers is out of the scope of this paper.

Mdc1_logic_example_com members: MDCPool1

Mdc2_logic_example_com members: MDCPool2

5. In the DNS GSLB –Wide IP of the F5 Administration Console create a new Wide IP. A Wide IP maps a fully-qualified domain name (FQDN) to a set of virtual servers that host the domains content. For example create a wide IP for login.example.com as follows:
   Name Use the FQDN that will be used to access the IAM Multi Data Center Deployment system (login.example.com)
   Type: A
   Add the pools previously created to the Wide IP MDCPool1 and MDCPool2
   Enable persistence for the Pool.
   Use Topology as the load balancing method.

6. In DNS GSLB – Topology – Records menu of the F5 Administration Console create a set of records specifying the IP Subnet number of the source requests/CIDR and the Destination Data Center that you wish to service that request set the weight of the record to 1.

With these settings the F5 GTM should round robin request to both sites or datacenters.

Use the appropriate IP address ranges and definitions that apply to each datacenter or site. With this the system is enabled to redirect requests to each Local LBR based on the originating request's IP. For additional details refer to the F5 GTM documentation at http://support.f5.com/kb/en-us/products/big-ip_gtm/manuals/product/gtm-concepts-11-2-0.html.

Preparing the Servers/Storage

The servers in both sites should be prepared in exactly the same way this includes:

- Using the same operating system and patch level.
- Installing the same operating system packages.
- Configuring the same Kernel Parameters
- Configuring NTP
- Configuring NIS
- Configuring the same storage structure. However each site uses shared storage local to the site, this provides both fault tolerance and the need for inter-site file access.
- The binary installations, *Shared config*, and runtime will use the same location at both sites.
- In addition to facilitate the failover of the OIM administration server, remote disk replication should be configured for the directory *IGD_ASERVER_HOME*.

Preparing the Databases on Site 2

Creating the Access Database

Create a database to hold the Access Domain content. This database will be created on site 2 using the same procedure as used to create the Access Database (IADDB) on site 1. This database is an independent database.

Creating the Identity Database

The Identity database is a copy of the Identity database on site1. This database should be created as an Active Dataguard database. The instructions to do this can be found in the Dataguard documentation appropriate to the version of your database.

For example: Oracle Data Guard Concepts and Administration (12c)

Create a Role based database service

Beginning with Data Guard 11g Release 2, you can automatically control the startup of database services on primary and standby database by assigning a database role to each service, this service is in addition to the default service created when the database was commissioned. A role based database service will automatically start upon database startup if the management policy of the service is AUTOMATIC and if one of the roles assigned to that service matches the current role of the database, for example if the database is running as a primary.

Creating a database service in this way means that the service is started whenever the database with the role "primary" is started. The service will move between sites as the underlying databases roles are moved through switchover or failover.

Services must be configured with the Server Control (SRVCTL) utility identically on all databases in a Data Guard configuration. In the following example, a service named oim.example.com is configured to be active when the database in site1 is in the primary role (-role PRIMARY). The same service is also configured on the standby database site 2 so that it is started whenever Site 2 functions in the primary role.

On the primary and standby hosts, create the read/write workload service (oim.example.com) that the WLS data source will use to connect to the database. The service should be created such that it is associated with and runs on the database when it is in the 'PRIMARY' database role:

The following example is for an Oracle 12c administrator managed database

Primary cluster:

```
srvctl add service -db iagdb -service oim.example.com -preferred  iagdb1,iagdb2 -role
PRIMARY -tafpolicy NONE -failovermethod NONE -failoverdelay 0 -notification TRUE -clbgoal
SHORT -policy AUTOMATIC
```

Standby cluster:

```
srvctl add service -db iagdbdg -service oim.example.com -preferred  iagdb1,iagdb2 -role
PRIMARY -tafpolicy NONE -failovermethod NONE -failoverdelay 0 -notification TRUE -clbgoal
SHORT -policy AUTOMATIC
```

To start the service use the command

```
srvctl start service -db iagdb -service oim.us.oracle.com
```

Modify the service for the appropriate service goals.

Run-time connection load balancing requires configuring Oracle RAC Load Balancing Advisory with service-level goals for each service for which load balancing is enabled. The Oracle RAC Load Balancing Advisory may be configured for SERVICE_TIME or THROUGHPUT. The connection load balancing goal should be set to SHORT.

Oracle RAC Load Balancing Advisory configured for SERVICE_TIME:

```
srvctl modify service -db <database name> -service <service name> -B SERVICE_TIME -j SHORT
```

Oracle RAC Load Balancing Advisory configured for THROUGHPUT:

```
srvctl modify service -db <database name> -service <service name> -B THROUGHPUT -j SHORT
```

Installing Software

Install the software on the hosts in the both sites using the steps in chapter 11 Installing Oracle Fusion Middleware in Preparation for an Enterprise Deployment. It is important that the directory structure in both sites is identical, and that the same software versions/patch sets are applied to both sites.

An alternative approach to installing the software manually is to clone the installation using the Oracle T2P software suite.

Configuring Oracle Unified Directory

Configuring Oracle Unified Directory on Site 1

Configure OUD on site 1 as described in the Enterprise Deployment Guide in Chapters 12.1 Configuring Oracle Unified Directory and Chapter 13 Preparing the Identity Store.

Configuring Oracle Unified Directory on Site 2

Configuring OUD is the same as configuring OUD on LDAPHOST2. Follow the steps in section 12.1.2.3 Configuring Oracle Unified Directory Instance on LDAPHOST2. Change the instance names to oud3 and oud4.

After configuring each of the new instances you also need to perform the following steps:

1. 12.1.2.6 Relaxing Oracle Unified Directory Creation Restrictions
2. 13.5.3 Grant OUD Changelog access
3. 13.5.4 Update Oracle Unified Directory ACI's for LDAP Synchronization.
4. 13.5.4 Creating OUD indexes

Configuring Web Tier

Configuring Web Tier on Site 1

Configure Oracle HTTP Server/Oracle Traffic Director on site 1 as described in the Enterprise Deployment Guide in Chapter 14 Configuring the Web Tier.

Configuring Web Tier on Site 2

Configure Oracle HTTP Server/Oracle Traffic Director on site 2 as described in the Enterprise Deployment Guide in Chapter 14 Configuring the Web Tier. Changing the target managed servers to those on Site 2.

Adding OAM Directives

Each OAM Domain uses inbuilt Multi Datacenter functionality to keep the two OAM domains in sync. This is achieved by invoking REST API's which are located in the administration server of each domain. The Enterprise Deployment Guide does not expose these REST API's. In order to gain access to these API's you must add an extra directive to the IADMADMIN1/IADADMIN2 virtual hosts. You do this by creating an entry similar to the following in the files iadadmin1_vhn.conf and iamadmin2_vhn.conf, you will have created these files as part of the Web Tier configuration.

```
<Location /oam/services>
   WLSRequest ON
   WebLogicHost iadadmin1vhn.example.com
   WeblogicPort 7001
</Location>
```

Be sure to set the WebLogicHost to be the Virtual Host Name for the site you are configuring.

Note: No changes are required if you are using OTD.

Disable Dynamic Cluster Notifications

When an application is deployed across sites it is good practice to keep traffic within a single site as much as possible. This can be achieved at the web tier level by turning off the dynamic server directive. This will ensure that requests from the webtier are directed to managed servers sited in the same site. That is to say only those managed servers that are explicitly listed. This is especially important in OIM where a single stretched cluster spans the two sites.

The following excerpts from the mod_wl_ohs.conf files in the OHS provide an example of the required configuration for routing to the oam web application.

These changes are only required for the prov.example.com virtual host.

Site1:

```
# OIM Self Service
<Location /identity>
    SetHandler weblogic-handler
    WebLogicCluster oimhost1vhn1.example.com:14000, oimhost1vhn1.example.com:14000
    DynamicServerList OFF
</Location>
```

Site2:

```
# OIM Self Service
<Location /identity>
    SetHandler weblogic-handler
    WebLogicCluster oimhost3vhn1.example.com:14000, oimhost4vhn1.example.com:14000
   DynamicServerList OFF
</Location>
```

Configuring Oracle Access Manager

Configuring OAM on Site1

OAM on site 1 is a self-contained installation. You create the OAM domain and Configure that domain using the steps listed in the Enterprise Deployment Guide for Identity and Access Management, specifically:

Chapter 15 Creating Domains for an Enterprise Deployment

Chapter 16 Setting Up Node Manager for an Enterprise Deployment

Chapter 17 Configuring Oracle Access Management

Follow these instructions with the following changes:

Deviations from the Standard Enterprise Deployment

Creating the Configuration File

When creating the configuration file in section 17.2.3.2.1 Create a Configuration File make the following changes:

Set PRIMARY_OAM_SERVERS to the OAM load balancer name for example oam.example.com:5575

A sample configuration file for OUD is below:

```
WLSHOST: iadadminvhn.example.com
WLSPORT: 7001
WLSADMIN: weblogic
WLSPASSWD: Manager1
IDSTORE_HOST: idstore.example.com
IDSTORE_PORT: 1389
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_PASSWD: Manager1
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
OAM11G_SERVER_LOGIN_ATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=example,dc=com
IDSTORE_SEARCHBASE: dc=example,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=example,dc=com
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=example,dc=com
IDSTORE_OAMSOFTWAREUSER: oamLDAP
IDSTORE_PWD_OAMSOFTWAREUSER: Manager1
IDSTORE_OAMADMINUSER: oamadmin
IDSTORE_PWD_OAMADMINUSER: Manager1
IDSTORE_DIRECTORYTYPE: OUD
PRIMARY_OAM_SERVERS: oam.example.com:5575
WEBGATE_TYPE: ohsWebgate11g
ACCESS_GATE_ID: Webgate_IDM
OAM11G_OIM_WEBGATE_PASSWD:Manager1
COOKIE_DOMAIN: .example.com
OAM11G_WG_DENY_ON_NOT_PROTECTED: true
OAM11G_IDM_DOMAIN_OHS_HOST:login.example.com
OAM11G_IDM_DOMAIN_OHS_PORT :443
OAM11G_IDM_DOMAIN_OHS_PROTOCOL:https
OAM11G_SERVER_LBR_HOST:login.example.com
OAM11G_SERVER_LBR_PORT:443
OAM11G_SERVER_LBR_PROTOCOL:https
OAM11G_OAM_SERVER_TRANSFER_MODE: simple
OAM_TRANSFER_MODE: simple
OAM11G_IDM_DOMIN_LOGOUT_URLS:/console/jsp/common/logout.jsp,/em/targetauth/emaslogout.jsp
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
OAM11G_SSO_ONLY_FLAG: true
COOKIE_EXPIRY_INTERVAL: 120
OAM11G_IMPERSONATION_FLAG:false
```

```
OAM11G_OIM_INTEGRATION_REQ:true
OAM11G_OIM_OHS_URL:https://prov.example.com:443/
SPLIT_DOMAIN:true
OAM11G_IDSTORE_NAME:LDAPStore
```

Update the Webgate Agents to use the OAM load balancer

Due to a bug the above script will create a webgate agent, which will point to each of the OAM servers in the topology. The newly created agents plus all existing agents need to be modified to point to the OAM load balancer. You achieve this by performing the following steps:

Login to the OAM console using the URL http://iadadmin.us.oracle.com/console log in using the oamadmin user you have created in the EDG.

From the Launch pad click on Agents – The Agent search screen will be displayed.

Click Search a list of existing agents will be displayed. Perform the following for each of the agents.

a) Click on the agent name

b) In the primary server box click on the Add button to create a new agent, and enter the following information:

   a. Access Server: Other

   b. Hostname: The loadbalancer name for example, oam.example.com

   c. Host Port: The loadbancer port you have configured for OAP for example 5575

c) Delete each of the other entries in the Primary Server List

d) Click Apply

e) Repeat for each of the agents.

Configuring Node Manager on Site 1 and Site 2

Configuring Site 2 consists of cloning the OAM configuration on site 1 to site 2. When you built site 1 you will create custom keystores, before starting the configuration on Site 2 you must create appropriate keystores on site 2 so that the cloning process can start the managed servers on site 2.

To do this you need to perform the following steps from the IAM Enterprise Deployment Guide on site 2.

16.5.1 Generating Self Signed Certificates using the utils.CertGen Utility

16.5.2 Creating an Identity Keystore Using the utilsImportPrivateKey Utility

16.5.3 Creating a Trust Keystore Using the Keytool Utility

16.5.4 Adding a Load Balancer Certificate to the Trust Store.

Configuring OAM on Site 2

Step 1 – Create a working Directory

Create a working directory on your system which we will refer to henceforth as MDC_HOME

mkdir –p *MDC_HOME*

Step 2 – Create a common environment File

Create a file containing your environment details in MDC_HOME called cConfig.sh put the following information in the file:

```
export JAVA_HOME=/u01/oracle/products/access/jdk
export MW_HOME=/u01/oracle/products/access
export ORACLE_HOME=$MW_HOME/iam
export T2P_HOME=$MW_HOME/oracle_common/bin
export WL_DOMAIN_HOME=/u01/oracle/config/domains/IAMAccessDomain
```

Where:

JAVA_HOME points to the location of your JAVA installation.
MW_HOME points to the location of your Access MW_HOME in the case of the EDG this will be IAD_MW_HOME.
ORACLE_HOME points to the location of the Oracle Identity and Access management software within the MW_HOME
T2P_HOME is the location of the T2P software
WL_DOMAIN_HOME points to the location of the IAD_ASERVER_HOME directory.

Create this file on OAMHOST1.

Step 3 – Place OAM into Multi-datacenter mode on Site 1

Enable OAM Multi Data Centre on Site 1.

Create a file in MDC_HOME called OAMMDC.properties with the following information on OAMHOST1:

```
SessionMustBeAnchoredToDataCenterServicingUser=false
SessionDataRetrievalOnDemand=true
Reauthenticate=false
SessionDataRetrievalOnDemandMax_retry_attempts=3
SessionDataRetrievalOnDemandMax_conn_wait_time=80
SessionContinuationOnSyncFailure=true
MDCGitoCookieDomain=.example.com
```

Note: MDCGitoCookieDomain should be the same as the cookie domain you created in creating the configuration file.

Use wlst to apply these configuration settings to your OAM Domain, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file called enable_mdc_site1.py with the following information

```
connect('weblogic','password','t3://iadadminvhn.example.com:7001')
domainRuntime()
enableMultiDataCentreMode(propfile="MDC_HOME/OAMMDC.properties")
setMultiDataCentreClusterName(clusterName="EDGMDC1")
setMultiDataCenterWrite(WriteEnabledFlag="true")
validateMDCConfig()

exit()
```

save the file.

Execute the file using the following commands:

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/enable_mdc_site1.py
```

Step 4 – Copy Site 1 Config to Site 2

Before executing the steps in this section, ensure that you have started the following servers in the domain: AdminServer, wls_oam1, wls_ama1, wls_msm1. If these servers are not started T2P will fail, this is especially important of wls_msm1 which needs to be started regardless of whether Mobile Security Suite is configured or not.

Create a file called domain_password in *MDC_HOME* this file will have one line which is the password for the weblogic administrator account in the IAMAccessDomain.

Create a file called copyConfigSite1.sh with the following information on OAMHOST1:

```
.  MDC_HOME/cConfig.sh
cd $T2P_HOME

./copyConfig.sh -javaHome $JAVA_HOME \
-archiveLoc $MDC_HOME/oamt2pConfig.jar \
-sourceDomainLoc $WL_DOMAIN_HOME \
-sourceMWHomeLoc $MW_HOME \
-domainHostName iadadminvhn.example.com \
-domainPortNum 7001 \
-domainAdminUserName weblogic \
-domainAdminPasswordFile $MDC_HOME/domain_password.txt \
-silent true -ldl $T2P_HOME/oam_cln_log_config \
-opssDataExport true -debug true
```

Save and execute the file, this will create a file called $MDC_HOME/oamt2pConfig.jar. Copy this file to a suitable location (we will use the same location MDC_HOME) on site 2.

Step 5 – Create a Moveplan

On Site2 – OAMHOST3 create a T2P movefile. To do this perform the following steps.

Create a directory for *MDC_HOME* and a subdirectory *MDC_HOME*/moveplan

Copy the following files from MDC_HOME on site 1

cConfig.sh
oamt2pConfig.jar

Create a file called extract_moveplan.sh with the following contents

```
. $MDC_HOME/cConfig.sh
```

```
cd $MW_HOME/oracle_common/bin

./extractMovePlan.sh -javaHome $JAVA_HOME -al $MDC_HOME/oamt2pConfig.jar -planDirLoc
$MDC_HOME/moveplan/
```

Save and Execute the file.

This will create a file called moveplan.xml in the directory *MDC_HOME*/moveplan

Step 6 – Modify the Moveplan

This moveplan file determines how the cloned domain will be created. You need to edit this file to change things like hostnames, passwords and database connection information to reflect the second site and its database. This file will require passwords from your environment, instead of entering them into the moveplan directly you must create text files with these passwords in and then reference those files in the move plan. You need to create the following files:

- Create a file in *MDC_HOME* called iad_password.txt and provide the password you wish to use for the IAMAccessDomain on Site2.
- Create a file in *MDC_HOME* called iad_passphrase.txt and provide the password you used when creating the keystores on Site 2.
- Create a file in *MDC_HOME* called iad_dbpassword.txt and provide the password you used when creating the RCU schemas on Site 2.
- Create a file in *MDC_HOME* called iad_ldappassword.txt and provide the OUD administration password (cn=oudadmin)

Edit the moveplan.xml file and make the following changes – Note change every occurrence:

- Change iadadminvhn.example.com to iadadmin2vhn.example.com
- Change oamhost1.example.com to oamhost3.example.com
- Change oamhost2.example.com to oamhost4.example.com
- Change the database service name from that on site1 to that on site 2. For example, iaddbS1.example.com to iaddbS2.example.com
- Change the database scan address used on Site 1 to that used on site2
- If the database schema prefix is different on Site 1 to that of Site 2 change that.
- Locate every entry in the moveplan which has the name Password File. Immediately under this entry add an entry which looks like: <value>/u01/mdc/iad_password.txt</value> Ensure that there are no spaces in the construct.

  After editing the entry should look something like:

```
<configProperty>
    <name>Password File</name>
    <value>/u01/mdc/iad_dbpassword.txt</value>
    <itemMetadata>
       <dataType>STRING</dataType>
       <password>true</password>
       <scope>READ_WRITE</scope>
```

```
    </itemMetadata>
</configProperty>
```

Note: You need to check the context of the password file entry. If the password file relates to the database connection, then the file specified needs to be the MDC_HOME/iad_dbpassword.txt file.

If the password file relates to the LDAP connection, then the file specified needs to be the MDC_HOME/iad_ldappassword.txt file.

» Locate every entry in the moveplan which has the name Passphrase File. Immediately under this entry add an entry which looks like: <value>/u01//mdc/db_passphrase.txt</value> Ensure that there are no spaces in the construct.

After editing the entry should look something like:

```
                    <name>Custom Identity Keystore Passphrase File</name>
                     <value>/u01/mdc/domain_password.txt</value>
                     <itemMetadata>
                         <dataType>STRING</dataType>
                         <password>true</password>
                         <scope>READ_WRITE</scope>
                    </itemMetadata>
                </configProperty>
```

» Change the value of the Custom Trust Store file if you have changed it from Site 1.

» Comment out the LIBOVD configuration.

Locate the line which looks like:

<componentType>LIBOVD</componentType>

Change the line before it to be the same as the following:

```
<!--movableComponent>
    <componentType>LIBOVD</componentType>
```

Locate the next occurrence of the line:

</movableComponent>

And change it to

```
</movableComponent-->
```

Save the file.

Step 7 – Clone the OAM configuration on Site 2

Create a file called paste_config.sh with the following contents on oamhost3

```
. MDC_HOME/cConfig.sh

cd $T2P_HOME
```

```
./pasteConfig.sh -javaHome $JAVA_HOME \
 -archiveLoc $MDC_HOME/ \
 -targetMWHomeLoc $MW_HOME \
 -targetDomainLoc $WL_DOMAIN_HOME \
 -movePlanLoc $MDC_HOME/ \
 -domainAdminPasswordFile $MDC_HOME/ \
 -ldl $T2P_HOME/ \
 -silent true
```

Save and Execute the file.

This will create the domain on oamhost3 on site 2.

Step 8 – Update Memory Parameters

Cloning does not take in to account any customizations you may have made in the setDomainEnv.sh and startWeblogic.sh scripts. You therefore need to reapply the memory settings to these files. To do this perform the following steps from the IAM Enterprise Deployment Guide on the newly cloned domain.

15.4.2 Forcing the Managed Servers to use IPv4 Networking.

15.4.3 Setting IAMAccessDomain Memory parameters

Step 9 – Setup NodeManager 16.5.1

Now that the domain is created on Site 2 you need to configure site 2 so that WebLogic servers can started/stopped via Node Manager. To do this you need to perform the following steps from the IAM Enterprise Deployment Guide

Section 15.4.5 Perform Initial Node Manager Configuration

Chapter 16 Setting Up Node Manager for an Enterprise Deployment

Note: You do not need to perform steps 16.5.1 Generating Self Signed Certificates using the utils.CertGen Utility

to 16.5.4 Adding a Load Balancer Certificate to the Trust Store as you will have already performed these.

Step 10 – Move Managed Server Directories to Private Storage

Now that the domain is created the managed server directories need to be moved to local disk. To do this follow the steps in sections 15.4.6 Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server and 15.4.7 Propagating Changes to Remote Servers of the Enterprise Deployment Guide.

Because of a bug it will not be possible to log in to the oamconsole. The workaround for this bug is to temporarily set the weblogic identity store back to the embedded LDAP. To do this you need to perform the following steps:

1. Edit the file oam-config.xml which is located in the directory IAD_ASERVER_HOME/config/fmwconfig

   Locate the section in the file for LDAP it will have the entry:

   Set the IsPrimary/isSystem value to false for the entry in <Setting Name="LDAP" Type="htf:map">

   Set the IsPrimary/isSystem value to true under the entry in Name="UserIdentityStore" Type="htf:map">

After editing the file will look like>

```
<Setting Name="LDAP" Type="htf:map">
        <Setting Name="D90CF57F6D7BB620FB" Type="htf:map">
          <Setting Name="GROUP_SEARCH_BASE" Type="xsd:string">cn=Groups,dc=example,
,dc=com</Setting>
          <Setting Name="LDAP_PROVIDER" Type="xsd:string">OUD</Setting>
          <Setting Name="LDAP_URL"
Type="xsd:string">ldap://idstore.example.com:1389</Setting>
          <Setting Name="Name" Type="xsd:string">LDAPStore</Setting>
          <Setting Name="SECURITY_CREDENTIAL"
Type="xsd:string">{AES}E30408A5ABEF493BE21691A8E4B6FAAB</Setting>
          <Setting Name="SECURITY_PRINCIPAL"
Type="xsd:string">cn=oamLDAP,cn=systemids,dc=us,dc=oracle,dc=com</Setting>
          <Setting Name="Type" Type="xsd:string">LDAP</Setting>
          <Setting Name="USER_NAME_ATTRIBUTE" Type="xsd:string">uid</Setting>
          <Setting Name="USER_SEARCH_BASE"
Type="xsd:string">cn=Users,dc=us,dc=oracle,dc=com</Setting>
          <Setting Name="UserIdentityProviderType"
Type="xsd:string">OracleUserRoleAPI</Setting>
          <Setting Name="ENABLE_PASSWORD_POLICY" Type="xsd:boolean">false</Setting>
          <Setting Name="IsPrimary" Type="xsd:boolean">false</Setting>
          <Setting Name="IsSystem" Type="xsd:boolean">false</Setting>
        </Setting>

        <Setting Name="UserIdentityStore" Type="htf:map">
          <Setting Name="GROUP_SEARCH_BASE"
Type="xsd:string">ou=groups,ou=myrealm,dc=base_domain</Setting>
          <Setting Name="LDAP_PROVIDER" Type="xsd:string">EMBEDDED_LDAP</Setting>
          <Setting Name="LDAP_URL" Type="xsd:string">ldap://ldap-host:7001</Setting>
          <Setting Name="Name" Type="xsd:string">UserIdentityStore1</Setting>
          <Setting Name="SECURITY_CREDENTIAL"
Type="xsd:string">{AES}F8E3A9FAD9D662F753D842979423ED3D</Setting>
          <Setting Name="SECURITY_PRINCIPAL" Type="xsd:string">cn=Admin</Setting>
          <Setting Name="Type" Type="xsd:string">LDAP</Setting>
          <Setting Name="USER_NAME_ATTRIBUTE" Type="xsd:string">uid</Setting>
          <Setting Name="USER_SEARCH_BASE"
Type="xsd:string">ou=people,ou=myrealm,dc=base_domain</Setting>
          <Setting Name="UserIdentityProviderType"
Type="xsd:string">OracleUserRoleAPI</Setting>
          <Setting Name="IsPrimary" Type="xsd:boolean">true</Setting>
          <Setting Name="IsSystem" Type="xsd:boolean">true</Setting>
          <Setting Name="RoleMappings" Type="htf:map">
            <Setting Name="Role Application Administrator"
Type="xsd:string">Operators</Setting>
            <Setting Name="Role System Manager" Type="xsd:string">Deployers</Setting>
```

```
            <Setting Name="Role System Monitor" Type="xsd:string">Monitors</Setting>
            <Setting Name="Role Security Admin" Type="htf:map">
              <Setting Name="Groups" Type="xsd:string">Administrators</Setting>
              <Setting Name="Users" Type="xsd:string">weblogic</Setting>
            </Setting>
```

2. Increment the config file version by 1

```
<Setting Name="NGAMConfiguration" Type="htf:map">
    <Setting Name="BundlePatch" Type="xsd:string">12.2.2.0.0</Setting>
    <Setting Name="Version" Type="xsd:integer">45</Setting>
    <Setting Name="DistributorMode" Type="xsd:string">MapStore</Setting>
```

3. Save the file.
4. Restart the Administration Server.
5. Login to the OAM console using the URL http://iadadmin1.example.com:7001/oamconsole the weblogic user.
6. Click on **Configuration** then **User Identity Stores**.
7. Update the Default Store and System Store to your LDAP identity store for example LDAPStore.
8. Click **Add** In the System Administrators Section – The Add System Administrator Roles search box is displayed.
9. Enter OAM in the Name box and click **Search**.
10. Click on **OAMAdminsitrators** from the returned List and click **Add Selected**.
11. Click **Apply** and Acknowledge the Warning Message.
12. You will now be asked to Validate a System Administrator. Enter the oamadmin user and its password and click **validate**.
13. Restart the Administration Server.
14. You should now be able to log in to the OAM Console using the oamadmin user which is in your LDAP directory.

Step 11 – Reassign LDAP Groups to WebLogic Administrators

When the T2P paste process finishes it does not assign the LDAP groups OAMAdministrators and IDM Administrators to the WebLogic administration group. Without these you will not be able to use LDAP users to access the OAM Console, Policy Manager or the WebLogic console.

To enable this functionality, you need to perform the steps in section 17.2.3.3 Add LDAP Groups to WebLogic Administrators of the IAM Enterprise Deployment Guide.

Step 12 – Export OAM Access Store from Data Center 1 to Data Center 2

Immediately after cloning you need to export the Access Store from Site 1 to Site 2 you do this by performing the following steps.

Create a file export_store.py with the following information on OAMHOST1

```
connect('weblogic','password','t3://iadadmin1vhn.example.com:7001')
exportAccessStore(toFile="MDC_HOME/oamaccess.zip",namePath="/")
exit()
```

Execute the script using the following commands.

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/export_store.py
```

This will generate a file oamaccess.zip copy this files to OAMHOST3.

Use wlst to import the data from these files into the OAM Domain on site 2, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file import_store.py with the following information on OAMHOST3

```
connect('weblogic','password','t3://iadadmin2vhn.example.com:7001')
importAccessStore(fromFile="MDC_HOME/oamaccess.zip",namePath="/")
exit()
```

Execute the script using the following commands.

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/import_store.py
```

Step 13 – Place OAM into Multi-datacenter mode on Site 2

Enable OAM Multi Data Centre on Site 2.

Create a file in MDC_HOME called OAMMDC.properties with the following information on OAMHOST3 (This file is identical to the file created on OAMHOST1):

```
SessionMustBeAnchoredToDataCenterServicingUser=false
SessionDataRetrievalOnDemand=true
Reauthenticate=false
SessionDataRetrievalOnDemandMax_retry_attempts=3
SessionDataRetrievalOnDemandMax_conn_wait_time=80
SessionContinuationOnSyncFailure=true
MDCGitoCookieDomain=.example.com
```

Note: MDCGitoCookieDomain should be the same as the cookie domain you created in configOAM ** Add Link **

Use wlst to apply these configuration settings to your cloned OAM Domain, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file called enable_mdc_site2.py with the following information

```
connect('weblogic','password','t3://iadadmin2vhn.example.com:7001')
domainRuntime()
enableMultiDataCentreMode(propfile="MDC_HOME/OAMMDC.properties")
setMultiDataCentreClusterName(clusterName="EDGMDC2")
setMultiDataCenterWrite(WriteEnabledFlag="true")
validateMDCConfig()
exit()
```

Note: The cluster name is different to that of site1.

Save the file.

Execute the file using the following commands:

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/enable_mdc_site2.py
```

Step 14 – Create Webgate Agents for Multi Datacenter

You need to create two webgate agents for each OAM domain to interact with each other. These agents need to be created on Site 1. To do this perform the following steps.

1. Log in to the Access Manager Console using the url http://iadaccess1.example.com:7001/oamconsole as the user oamadmin.
2. From the landing page click the + in the Agents box and select **Create Webgate**.
3. Create the webgate with the following information:
4. Version: 11g
5. Name: MDC-DC1
6. Security: Simple
7. Access Client Password: Choose a password
8. Click **Apply.**
9. The Agent Details screen is displayed, make the following changes:
10. Replace the Primary Server list with a single entry, Type:Other, Host Name: oam.example.com
11. Select Allow Management Operations
12. Click **Apply.**
13. Create a second agent in the same way this time calling it MDC-DC2.

Step 15 – Copy Keystores

In a multi datacenter configuration, each oam domain communicates with the parte oam domain. This is achieved by the webgates created above. In an EDG implementation these webgates will be configured to use the simple security protocol. In order for the communication to happen, each OAM domain must have access to the keystore and truststore of the partner domain. This is achieved by copying the files from one domain to the other.

From OAMHOST1

Copy the files as follows (change the names as described to make them obvious what they are):

Remote Name (OAMHOST3): *IAD_ASERVER_HOME*/output/webgate-ssl/oamclient-truststore.jks

Local Name: *SHARED_CONFIG_DIR*/keystores/EDGMDC2-truststore.jks

Remote Name (OAMHOST3): *IAD_ASERVER_HOME*/output/webgate-ssl/oamclient-keystore.jks

Local Name: *SHARED_CONFIG_DIR*/keystores/EDGMDC2-keystore.jks

From OAMHOST3

Copy the files as follows (change the names as described to make them obvious what they are:

Remote Name (OAMHOST1): *IAD_ASERVER_HOME*/output/webgate-ssl/oamclient-truststore.jks

Local Name: *SHARED_CONFIG_DIR*/keystores/EDGMDC1-truststore.jks

Remote Name (OAMHOST1): *IAD_ASERVER_HOME*/output/webgate-ssl/oamclient-keystore.jks

Local Name: *SHARED_CONFIG_DIR*/keystores/EDGMDC1-keystore.jks

Step 16 – Register MDC Partners

Now the agents are created they have to be registered as partners inside the OAM Multi Datacenter configuration. This is achieved in the following way.

Create file mdc_partner1.properties with the following contents.

```
remoteDataCentreClusterId=EDGMDC1
oamMdcAgentId=MDC-DC1
PrimaryHostPort=oam.example.com:5575
SecondaryHostPort
AccessClientPasswd=AgentPassword
oamMdcSecurityMode=SIMPLE
agentVersion=11g
trustStorePath=SHARED_CONFIG_DIR/keystores/EDGMDC1-truststore.jks
keyStorePath=SHARED_CONFIG_DIR/keystores/EDGMDC1-keystore.jks
globalPassPhrase=GlobalPassphrase
keystorePassword=GlobalPassphrase
RESTEndpoint=http://iadadmin1.example.com
```

Create file mdc_partner2.properties with the following contents.

```
remoteDataCentreClusterId=EDGMDC2
oamMdcAgentId=MDC-DC2
PrimaryHostPort=oam.example.com:5575
SecondaryHostPort
AccessClientPasswd=AgentPassword
oamMdcSecurityMode=SIMPLE
agentVersion=11g
trustStorePath=SHARED_CONFIG_DIR/keystores/EDGMDC2-truststore.jks
```

```
keyStorePath=SHARED_CONFIG_DIR/keystores/EDGMDC2-keystore.jks
globalPassPhrase=GlobalPassphrase
keystorePassword=GlobalPassphrase
RESTEndpoint=http://iadadmin2.example.com
```

Where:

- RemoteDataCentreClusterID is the cluster name in the data center.
- oamMdcAgentId is the name of the webgate agent you created for the data center
- PrimaryHostPort is the load balanced entry point for the OAM managed servers.
- AccessClientPasswd is the password you assigned to the webgate agent when you created it.
- oamMdcSecurityMode this is the security mode OAM is running in, this will generally be SIMPLE for EDG deployments.
- agentVersion this is the version that you assigned to the webgate agent for example 11g.
- trustStorePath this is the absolute path to the local copy of the trust store that you copied from the host to the directory SHARED_CONFIG_DIR/keystores
- keyStorePath this is the absolute path to the local copy of the key store that you copied from the host to the directory SHARED_CONFIG_DIR/keystores
- globalPassPhrase this is the global pass phrase you assigned when creating the EDG environment. If you are unsure of this you can obtain it using the wlst command: displaySimpleModeGlobalPassphrase().
- keystorePassword this is the same as the global passphrase.
- RestEndpoint this is the location of the Rest endpoints, in an Enterprise deployment this should be the value of webtier entry point for example iadadmin.example.com – Note it is different for each domain.

Register the Partners with each data center using the following commands.

From OAMHOST1

Use wlst to apply these configuration settings to your cloned OAM Domain, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file called register_site1_partners.py with the following information

```
connect('weblogic','password','t3://iadadmin1vhn.example.com:7001')
domainRuntime()
addPartnerForMultiDataCentre(propfile="MDC_HOME/mdc_partner1.properties")
addPartnerForMultiDataCentre(propfile="MDC_HOME/mdc_partner2.properties")
setMultiDataCenterType(DataCenterType="Master")
exit()
```

Execute the script using the following commands.

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/register_site1_partners.py
```

From OAMHOST3

Use wlst to apply these configuration settings to your cloned OAM Domain, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file called register_site2_partners.py with the following information

```
connect('weblogic','password','t3://iadadmin2vhn.example.com:7001')
domainRuntime()
addPartnerForMultiDataCentre(propfile="MDC_HOME/mdc_partner1.properties")
addPartnerForMultiDataCentre(propfile="MDC_HOME/mdc_partner2.properties")
setMultiDataCenterType(DataCenterType="Clone")
exit()
```

Execute the script using the following commands.

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/register_site2_partners.py
```

Step 17 – Export OAM policies from Data Center 1 to Data Center 2

Now that the two sites are linked together the policy information in Site 1 needs to be moved across to Site 2.

Use wlst to export the data from the OAM Domain on site 1 to files, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file export_policies.py with the following information on OAMHOST1

```
connect('weblogic','password','t3://iadadmin1vhn.example.com:7001')

exportPartners(pathTempOAMPartnerFile="<oampartner.xml>")
exportPolicy(pathTempOAMPolicyFile="<oampolicy.xml>")
exit()
```

Execute the script using the following commands.

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/export_policies.py
```

This will generate two files oampolicy.xml and oampartner.xml copy these files to OAMHOST3.

Use wlst to import the data from these files into the OAM Domain on site 2, you can run these commands interactively but for reproducibility it is easier to place them into a file.

Create a file import_policies.py with the following information on OAMHOST3

```
connect('weblogic','password','t3://iadadmin2vhn.example.com:7001')

importPolicy(pathTempOAMPolicyFile="MDC_HOME/oampolicy.xml")
importPartners(pathTempOAMPartnerFile=""MDC_HOME/oampartner.xml/")
exit()
```

Execute the script using the following commands.

```
. MDC_HOME/cConfig.sh
cd $ORACLE_HOME/common/bin
echo $PWD
./wlst.sh MDC_HOME/import_policies.py
```

Enable Automated Policy Synchronization

Now that the Multi Datacenter configuration is configured, OAM can be configured to automatically replicate policies between the master site and the cloned sites. This is achieved by issuing the following commands:

Create a transformation file

When data is replicated from the primary site to the clone site(s), some of the data may need to be changed, for example the primary server list may be different on site 1 than that on site 2. This is achieved by transformation rules. A default set is enabled out of the box, unfortunately this default set can change the name of our GLBR entry point between the sites. We must therefore create our own transformation rules to prevent this happening.

Create a directory in *SHARED_CONFIG* called mdc to hold the transformations file.

Create a file called *SHARED_CONFIG*/mdc/transformation.xml with the following content.

```
<?xml version="1.0" encoding="UTF-8"?>
<mdc-transform-rule>
  <changes-to-include entity-path="/policy"/>
  <changes-to-include entity-
path="/config/NGAMConfiguration/DeployedComponent/Agent/WebGate/Instance">
  </changes-to-include>
  <changes-to-include entity-
path="/config/NGAMConfiguration/DeployedComponent/Server/NGAMServer/Profile/Authentication
Modules"/>
  <changes-to-include entity-
path="/config/NGAMConfiguration/DeployedComponent/Server/NGAMServer/Profile/oamproxy"/>
  <changes-to-include entity-
path="/config/NGAMConfiguration/DeployedComponent/Server/NGAMServer/Profile/Sme/SessionCon
figurations"/>
  <changes-to-include entity-
path="/config/NGAMConfiguration/DeployedComponent/Server/NGAMServer/Profile/OAMServerProfi
le/OAMSERVER">
    <ignore attribute-match="/serverprotocol"/>
    <ignore attribute-match="/serverhost"/>
```

```
    <ignore attribute-match="/serverport"/>
  </changes-to-include>
  <changes-to-include entity-
path="/config/NGAMConfiguration/DataCenterConfiguration/Cluster">
    <ignore attribute-match="/DataCenterType"/>
    <ignore attribute-match="/ClusterId"/>
    <ignore attribute-match="/WriteEnabledFlag"/>
  </changes-to-include>
</mdc-transform-rule>
```

Save the file.

Make sure this file is available in the same location on both sites.

Add Server Start Parameters to Admin Servers

Automated Policy Synchronization (APS) is enabled by adding a Java Parameter to the Admin Server when it starts. To do this you must edit the file setDomainEnv.sh which is located in the directory *IAD_ASERVER_HOME*/bin

Locate the line beginning: EXTRA_JAVA_PROPERTIES="…"

Add the following at the beginning:

```
-Doracle.oam.EnableMDCReplication=true  -Doracle.oam.MDCRuleFile=Path to transformation
file
```

For example:

```
EXTRA_JAVA_PROPERTIES="-Doracle.oam.EnableMDCReplication=true -
Doracle.oam.MDCRuleFile=/u01/oracle.config/mdc/transformation.xml -DCONFIG……"
```

Save the file.

Restart the Administration Server and the wls_oam1 managed server.

Perform this action on both the Master Domain and the Clone.

Validate the REST services

This will enable REST services on the Admin Server. Verify that you can access these REST services via the HTTP servers using the commands:

```
curl -u oamadmin 'http://iadadmin1.example.com/oam/services/rest/_replication/hello'
```

and

```
curl -u oamadmin 'http://iadadmin2.example.com/oam/services/rest/_replication/hello'
```

You should see a response similar to RESPONSE: {"ok":"true"}

These commands should be executed on OAMHOST1.

Encode the oamadmin User

The following commands will setup a replication agreement between the master site and the clone site. When you setup the replication agreement you need to do so using the oamadmin account which resides in your LDAP directory. Setting up the replication agreement is achieved using curl commands. As part of the curl command you

need to supply the oamadmin user/password. This password must be encoded use base64. There are many utilities on the web that allow you to do this, one example is:

http://www.motobit.com/util/base64-decoder-encoder.asp

Use one of these tools to encode oamadmin:password into base 64. For example oamadmin:Password1 would be encoded as: b2FtYWRtaW46UGFzc3dvcmQxIA0K (note enter the user/password in the format username:password).

Make a note of this value.

Setup a Replication Agreement

Issue the following command to setup a replication agreement:

```
curl -u oamadmin -H 'Content-Type: application/json' -X POST
'http://iadadmin1.example.com/oam/services/rest/_replication/setup' -d
'{"name":"AgreementName", "source":"ClusterID1","target":"
ClusterID2","documentType":"ENTITY","config":{"entry":{"key":"authorization","value":"BASI
C encodedPassword"}}}'
```

Where

AgreementName is an arbitrary name you assign to the agreement.

ClusterID1 is the Cluster ID of your master site.

ClusterID2 is the Cluster ID of your clone site.

encodedPassword is the encoded value of your oamadmin account and password (see step above)

For example:

```
curl -u oamadmin -H 'Content-Type: application/json' -X POST
'http://iadadmin1.example.com/oam/services/rest/_replication/setup' -d
'{"name":"DC1toDC2",
"source":"EDGMDC1","target":"EDGMDC2","documentType":"ENTITY","config":{"entry":{"key":"au
thorization","value":"BASIC b2FtYWRtaW46UGFzc3dvcmQxIA0K"}}}'
```

After executing the command you should see something like:

```
{"enabled":"true","identifier":"201607221101462625","ok":"true","pollInterval":"900","star
tingSequenceNumber":"860","state":"READY"}
```

Make a note of the Identifier above.

Validate that the agreement is Setup and Ready

Now that the agreement is created query its status from a Master and Consumer perspective.

Master:

```
curl -u oamadmin -H 'Content-Type: application/json' -X GET
'http://iadadmin1.example.com/oam/services/rest/_replication/201607221101462625'
```

You should see a response similar to:

{"enabled":"true","identifier":"201607221101462625","ok":"true","pollInterval":"3600","startingSequenceNumber":"860","state":"ACTIVE"}

Consumer:

```
curl -u oamadmin -H 'Content-Type: application/json' -X GET
'http://iadadmin1.example.com/oam/services/rest/_replication/201607221101462625?type=consumer'
```

You should see a response similar to:

{"enabled":"true","identifier":"201607221101462625","ok":"true","pollInterval":"900","startingSequenceNumber":"860","state":"READY"}

Disable Clone Writes

In an OAM Multi Datacenter deployment, all policy creation operations must occur at the nominated master site. These will automatically be propagated to the MDC clones. By default there is nothing to prevent you performing policy maintenance operations on the clones, this can result in replication issues. Therefore writing to clone sites must be prevented.

This is achieved by running the following wlst command on each clone.

```
setMultiDataCenterWrite(WriteEnabledFlag="false")
```

Restart Domains

Restart the Admin and Managed servers on each domain. Create a webgate agent on the Master site and check that it is propagated to the cloned site.

Configuring Oracle Identity Manager

Creating the OIM Stretched Cluster

OIM uses a stretched cluster design. The stretched cluster design is an Enterprise Deployment Topology scaled out to additional servers in Site2. There are some aspects to consider that can make the system more scalable and that will minimize the possible performance degradation caused by the latency across sites.

You create the OIM domain and Configure that domain using the steps listed in the Enterprise Deployment Guide for Identity and Access Management, specifically:

Chapter 15 Creating Domains for an Enterprise Deployment

Chapter 16 Setting Up Node Manager for an Enterprise Deployment

Chapter 19 Configuring Oracle Identity Manager

Chapter 20 Configuring BI Publisher

Follow these instructions with the following changes:

Deviations from the Standard Enterprise Deployment

- **OIM Cluster Configuration:** In the configuration wizard when creating the IAMGovernanceDomain include ALL servers (Site 1 and Site 2) when specifying the OIM Cluster. For example:

  oimhost1vhn1.example.com:14000, oimhost2vhn1.example.com:14000, oimhost3vhn1.example.com:14000, oimhost4vhn1.example.com:14000

- **SOA Cluster Configuration:** In the configuration wizard when creating the IAMGovernanceDomain include ALL servers (Site 1 and Site 2) when specifying the SOA Cluster. For example:

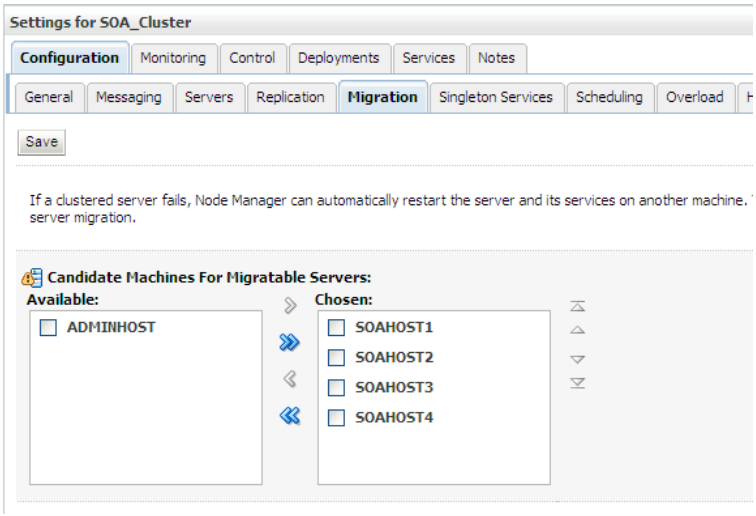
  oimhost1vhn2.example.com:8001, oimhost2vhn2.example.com:8001, oimhost3vhn2.example.com:8001, oimhost4vhn2.example.com:8001

- **BI Cluster Configuration:** In the configuration wizard when creating the IAMGovernanceDomain include ALL servers (Site 1 and Site 2) when specifying the BI Cluster. For example:
- oimhost1vhn3.example.com:9704, oimhost2vhn3.example.com:9704, oimhost3vhn3.example.com:9704, oimhost4vhn3.example.com:9704
- JMS File Locations do not need to be specified in the config wizard as they will be residing in the database.
- Configure JMS to use the Database, creating JMS stores for All the OIM/BI/SOA managed servers.
- When creating jms_custom.ddl ensure that the file exists at the same location on both sites.
- When creating the JMS Data Sources be sure to specify the OIM service name not the database service name.
- When entering the SOA Server coherence parameters be sure to include ALL hosts in the list of coherence servers.
- When running the OIM Configuration Wizard in 19.2 Configuring Oracle Identity Manager, use the OIM service name in the database connection screen.
- When Configuring OIM to Reconcile from the ID Store include ALL hosts in the OIMProviderURL.
- There is no need to perform the actions in 19.8 Configuring the default persistence store as you will already have configured it to use the Database persistence store.
- When Integrating OIM with OAM
  - Copy the keystores from the local OAM deployment.
  - It is not necessary to Import the Certificates into Mobile Security Suite as Mobile Security Suite is not supported in an MDC configuration.
  - When specifying ACCESS_SERVER_HOST in oimitg.props specify the load balancer name oam.example.com
  - When specifying the MDS_DB_URL use the OIM service name.
  - OIM_MSM_REST_SERVER_URL is not required.
- There is no need to create OMSS Helpdesk User and Roles as OMSS is not supported by MDC.
- When Configuring BI Publisher in Chapter 20 Configuring BI Publisher:
  - In Chapter 20.2.2 Configuring JMS It is not necessary to perform steps 8 and 9. In step 11 choose the Database JMS you will already have created in the Create Domain Chapter.

Server Migration

In the stretched domain design servers use only those machines in their same site as candidates for migration. Use the steps in the *Chapter 21 Configuring Server Migration for an Enterprise Deployment* for configuring server migration with these additional considerations:

- For the Cluster do not specify any machines as candidates as shown here:



- For servers on Site1, chose only machines in Site1 as candidates.
- For servers on Site2, chose only machines in Site2 as candidates.
- Depending on the latency across sites, you may need to increase the Health Check Interval for server migration. The default is 10000msecs which should be adequate in most cases; however, busy periods and overloads may require using a higher value depending on each case. Notice that this setting affects the health checks for all of the servers in the Stretched Cluster, hence it will increase the time it takes to detect crashes of all of the servers.

Update OIM Data Sources

When the OIM domain was created a number of datasources will have been created which use the oim.example.com database service. These services will all be pointing at the primary OIM database. In order to facilitate a seamless failover the standby database needs to be added in to the data source configurations. A typical database connection which includes both a primary and standby database looks like:

```
jdbc:oracle:thin:@
(DESCRIPTION_LIST=(LOAD_BALANCE=OFF)(FAILOVER=ON)(DESCRIPTION=(CONNECT_TIMEOUT=3)(RETRY_CO
UNT=3)(ADDRESS_LIST=(LOAD_BALANCE=ON)(ADDRESS=(PROTOCOL=TCP)(HOST=iagdb-
scan.example.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=oim.example.com)))(DESCRIPTION=(
CONNECT_TIMEOUT=3)(RETRY_COUNT=3)(ADDRESS_LIST=(LOAD_BALANCE=ON)(ADDRESS=(PROTOCOL=TCP)(HO
ST=iagdbdg-scan.example.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=oim.us.oracle.com))))
```

To change the OIM data sources perform the following steps:

1. Login to the WebLogic Console.
2. Click **Lock and Edit.**
3. In the Domain Tree, expand **Services** – **Data Sources.**
4. On the Summary of JDBC Data sources page, click on the name of a Data source for example ApplicationDB
5. Click on the **Connection Pool** tab.
6. Update the URL to reflect the above database connection example.
7. Click **Save.**
8. Validate the data source by clicking on the **Monitoring** tab, and the **Testing** sub tab.
9. Select one of the servers and click on **Test Data Source**, make sure the test is successful before continuing.
10. Click on the **ONS** Tab.

11. Add the standby database to the ONS Nodes field separating each host/port with a comma. For Example primaryDB-scan:6200,standbyDB-scan:6200
12. Click **Save**.
13. Repeat for each data source
14. Click **Activate Changes**.

Update JPS Security Files

In addition to updating the weblogic data sources you must also update the data source information in the files jps-config.xml and jps-config-jse.xml which are located in the directory IGD_ASERVER_HOME/config/fmwconfig

Before editing these files take a backup of them first!

Edit the files and update the jdbc.url property with the value above.

After editing the file will look something like:

```
        <propertySet name "props.db.1">
            <property name "jdbc.url"
value "jdbc:oracle:thin(DESCRIPTION_LIST=(LOAD_BALANCE=OFF)(FAILOVER=ON)(DESCRIPTION=(CONN
ECT_TIMEOUT=3)(RETRY_COUNT=3)(ADDRESS_LIST=(LOAD_BALANCE=ON)(ADDRESS=(PROTOCOL=TCP)(HOST=i
agdb-
scan.example.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=oim.example.com)))(DESCRIPTION=(
CONNECT_TIMEOUT=3)(RETRY_COUNT=3)(ADDRESS_LIST=(LOAD_BALANCE=ON)(ADDRESS=(PROTOCOL=TCP)(HO
ST=iagdbdg-
scan.example.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=oim.us.oracle.com))))/>
            <property name "oracle.security.jps.farm.name" value "cn=IAM"/>
```

Save the file.

Pack/Unpack the domain

For the above changes to be propogated to the managed servers you will need to pack and unpack the IAMGovernanceDomain.

Restart the Administration Server and All Managed servers in the IAMGovernanceDomain.

Moving the BI Publisher Shared Configuration Location

Rather than following the instructions in 20.1 Moving reports to a Shared Directory follow the steps below.

Ensure that the RT_HOME directory is replicated between site 1 and site 2. The best way of achieving this is to use disk replication technology.

1. Copy the contents *of IGD_ASERVER_HOME*/config/bipublisher to the directory *RT_HOME*.
2. On oimhost1 login to BI publisher using the url http://oimhost1vhn3.example.com:9704/xmlpserver using the user xelsysadm.
3. Click the **Administration** Tab.
4. Under System Maintenance, click on **Server Configuration**.
5. In the Path Field under the heading Configuration Folder enter *RT_HOME*/bipublisher/repository.
6. In the Path Field under the heading Catalog enter *RT_HOME*/bipublisher/repository.

7. Click **Apply**.
8. This will have updated the configuration file which is located in the directory *IGD_MSERVER_HOME*/config/bipublisher. When BI Managed servers start they obtain their configuration from the file located in the *IGD_ASERVER_HOME*/config/bipublisher directory. So following the changes made above the configuration file which is called xmlp-server-config.xml must be manually copied from the MSERVER_HOME directory to the ASERVER_HOME directory. For example:

```
cp IGD_MSERVER_HOME/config/bipublisher/xmlp-server-config.xml
IGD_ASERVER_HOME/config/bipublisher
```

9. Restart each of the BI managed servers and check that the entry that appears in the server configuration screen is *RT_HOME*/bipublisher/repository.

Disable the OIM Job Scheduler on Site2

The OIM job scheduler uses the database intensively. In order to keep inter site traffic to a minimum, the job scheduler should be disabled on the site where the database is not primary. This step is not necessary if you plan to leave the OIM Managed servers shutdown on Site 2.

The jobscheduler is disabled by adding the following parameter to the server startup arguments.

```
-Dscheduler.disabled=true
```

To do this you need to perform the following steps:

- Log in to the WebLogic Administrative Console.
- In left pane, click **Environment**, **Servers**.
- Click the name of the managed server in which you want to disable the scheduler for.
- Click **Lock and Edit** in the left tab.
- Click **Configuration**, **Server start** tab in the right pane.
- In the Argument Text box, add -Dscheduler.disabled=true, and save.
- Click **Activate Change** in the left pane.

After switching over the database, this parameter should be removed from these servers and added into the server definitions of the now standby site.

Conclusion

Having followed the above configuration steps, you will have built an identity management application suitable for use in a typical enterprise. If the network latency is very low between the two sites then it may be possible to run OIM in an active/active configuration, however Active/Passive is the preferred deployment.

This solution is made simpler by having :

- Two different entry points for OAM and OIM
- Using a global load balancer with geographic affinity.
- Having an extra load balancer for OAM OAP calls.

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

Hardware and Software, Engineered to Work Together

Michael Rhys