Best Practices for Oracle Fusion Middleware SOA 11*g* Multi Data Center Active-Active Deployment

Oracle Maximum Availability Architecture White Paper
September 2014

Maximum
Availability
Architecture

Oracle Best Practices For High Availability

Introduction

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 miss configuration, 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 a Service Oriented Architecture (SOA) multiple corporate systems may depend on a unique service provider. As the adoption of these architectures grows, 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* SOA system against downtime across multiple locations (referred to as SOA Active-Active Disaster Recovery Solution or SOA 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.

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

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.

In most Multi Data Center Active-Active Deployments or Active-Active Disaster Recovery systems the reality is that the database tier is usually active in only one site. There are alternatives to this approach (Oracle Real Application Clusters on Extended Distance Clusters, Cross-Site Caching (Oracle GoldenGate), and database replication (Streams)) however these solutions are either very demanding in terms of infrastructure required or require specific data types and rules with which not all applications are compliant.

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.

When a failure occurs in the 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 and 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].

[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.

Performance

Besides the common performance paradigms that apply to single-datacenter designs, Oracle Fusion Middleware SOA 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 SOA 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:

- Incoming HTTP invocations from Load Balancers (LBR) or Oracle HTTP Servers (OHS) and HTTP callbacks
- JNDI/RMI and JMS invocations between Oracle WebLogic Servers
- Read/write requests to file systems for JMS and transaction persistent stores as well as for file/FTP adapters

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 HTTP 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).

There are additional types of invocations that may take place between the different SOA 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 and metadata image to all SOA requests, whether served by one site or the other.
- HTTP session replications: some Oracle Fusion Middleware SOA components use stateful web applications (such as Composer, Workspace, etc.) 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 SOA 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. Alternatively both sites can share the same store. The impact of sharing the store will depend on the type of store and the usage pattern by the SOA system.

Administration

Another key aspect of the design and deployment of an Oracle Fusion Middleware SOA 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 SOA 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.

Latency, Jitter, Packet Loss and Bandwidth Across Sites

The overall network throughput of an Oracle Fusion Middleware SOA 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 SOA 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 SOA 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 SOA Multi Data Center Active-Active Deployment because multiple network round trips (mainly from the Oracle Fusion Middleware SOA servers to the SOA database) are required to complete each composite instance. Tests conducted have shown that an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment's performance (where Oracle WebLogic Server SOA servers use a database in a different site) degrades considerably when latency exceeds 5-10 milliseconds. The graphs in Image 1 and Image 2 show the throughput (transactions per second) and average transaction active time for a Fusion Order Demo (FOD) Oracle Fusion Middleware SOA system that uses a database on a different site with different latencies between middle tiers:

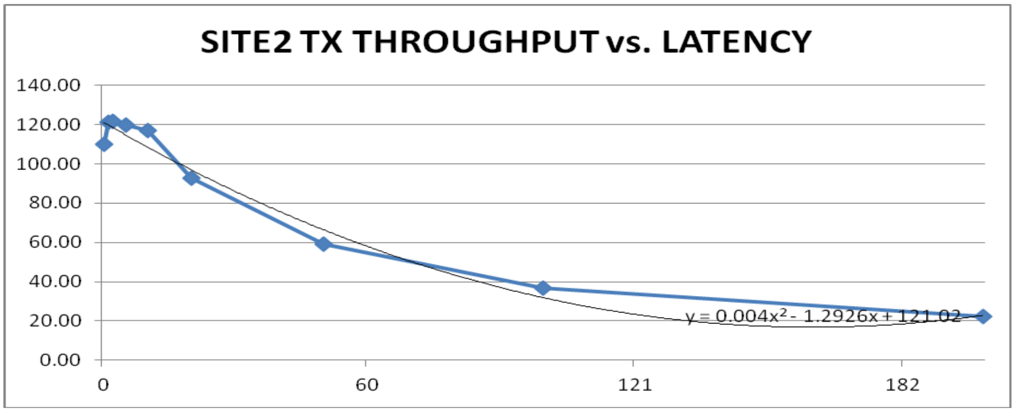


Image 1: Evolution of throughput with different latencies (RTT in msecs.) between middle tiers (FOD)

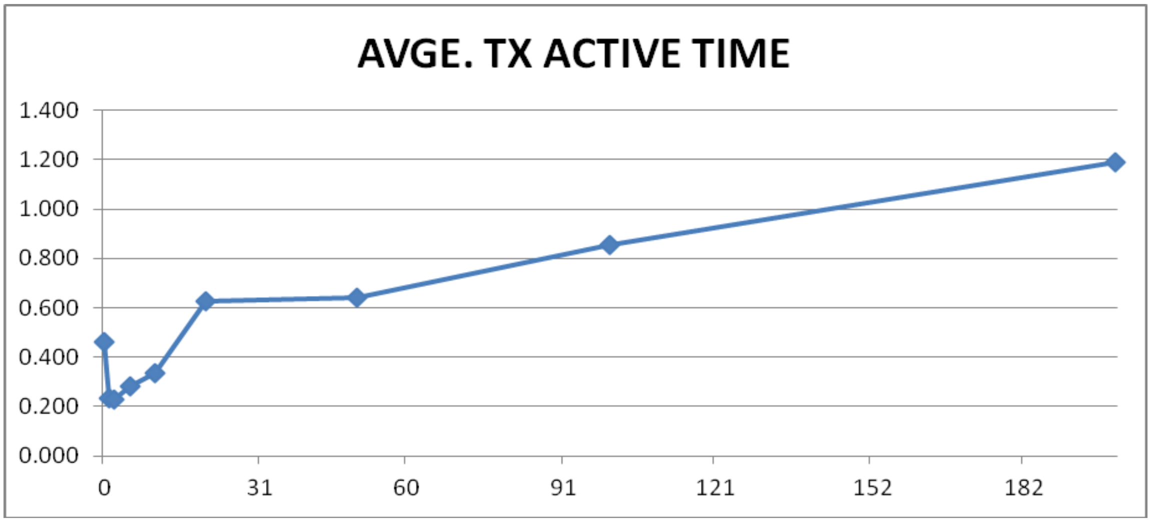


Image 2: Evolution of the time that a transaction remains active as the latency (RTT in msecs.) between the SOA servers and the database is increased

Image 3 shows the degradation observed in the overall system's throughput (both sites working together) for different latencies. Observe that for a latency of around 20 milliseconds RTT the thoughput decreases almost 25%.

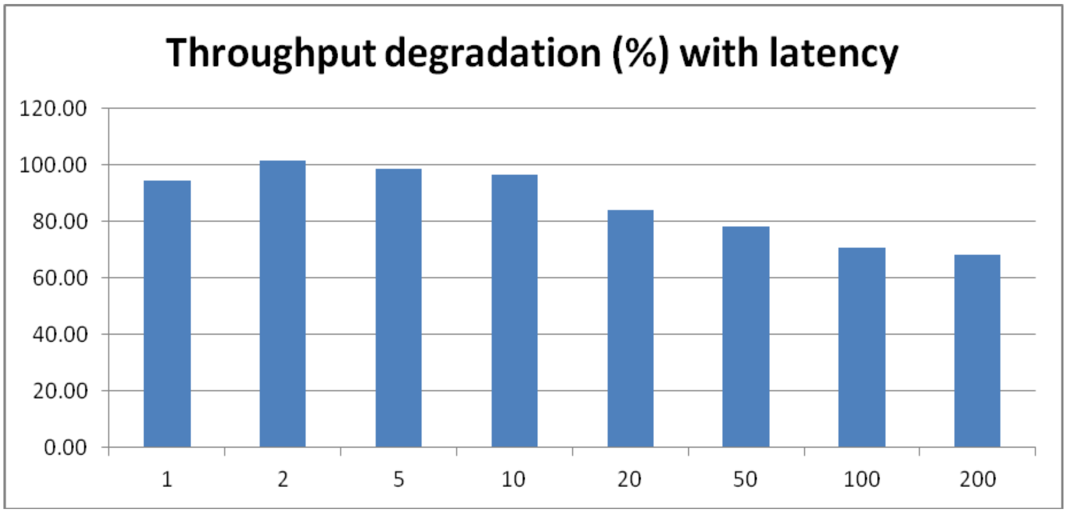


Image 3: Throughput degradation for different latencies (RTT in msecs.)

Image 4 shows the increase in time taken to deploy a composite (as compared to a deployment with all SOA servers and database in the same site).

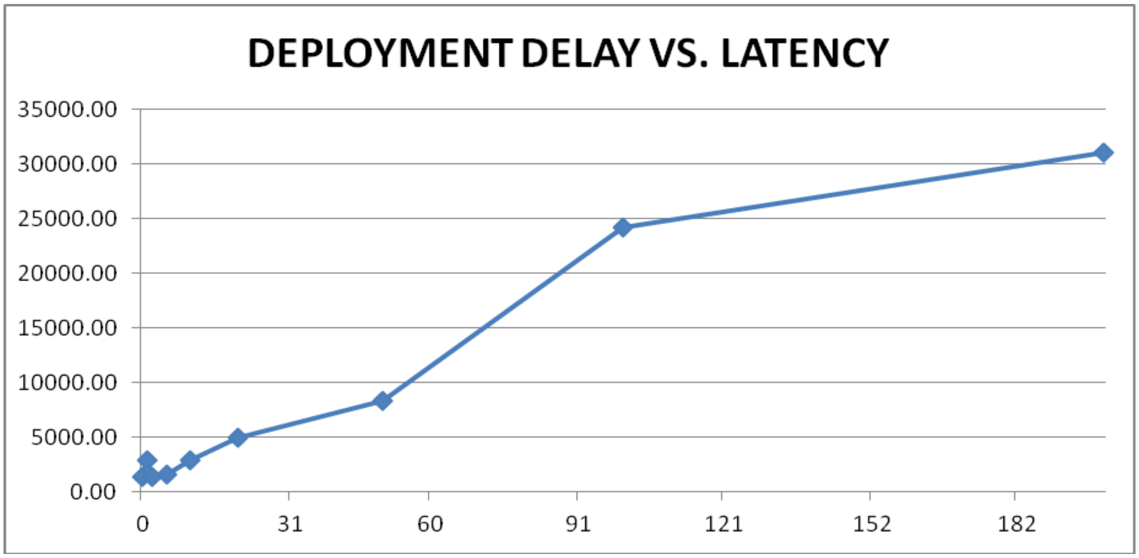


Image 4: Additional time (msecs) consumed for deploying composites with increasing latencies (RTT in msecs.) when the SOA database resides on a different site

When Oracle Data Guard is configured between the two sites, ARCHIVELOG MODE is enabled, and when the latency affects SQL*Net traffic (that is, the same link is used for database access and middle tier communications between sites) the performance degradation is more severe.

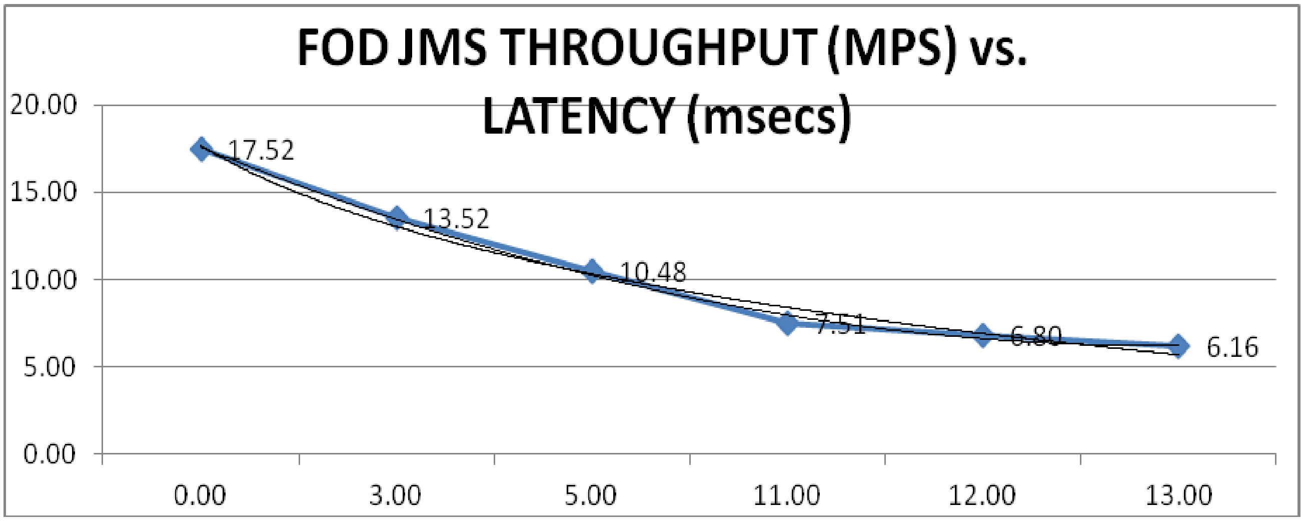


Image 5: Evolution of throughput with increased latencies (RTT in msecs.) for FOD using a SOA database configured with Data Guard and shared link between database and middle tiers

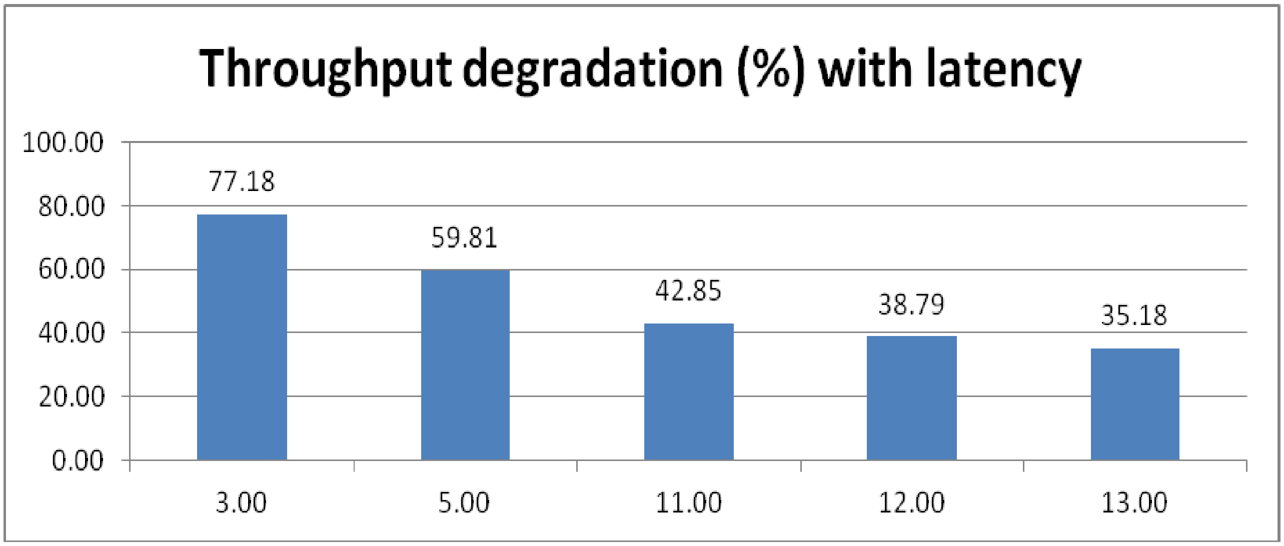


Image 6: Percentage throughput degradation for different latencies (RTT in msecs.) using a SOA database configured with Data Guard and shared link between database and middle tiers

Note that the effect of latency is orthogonal to the bandwidth between two sites (that is, it will affect equally large or small messages and payloads). For example: if a SOA server executes a SQL database query that requests 100 rows of the CUBE_INSTANCE, MEDIATOR_INSTANCE and DLV_MESSAGES tables, one row at a time, over a link with a latency of 60 ms, it takes approximately 6 seconds (60 ms * 100 turns) to complete the transaction *independently of the amount of data in each row*. The same query executed by a user on a LAN connected to the same database server takes less than 2-3 ms to be completed, as the latency due to distance across the LAN is insignificant. This is irrespective of the size of each row. Bigger rows can be retrieved with better bandwidth, but the overall transaction takes the same amount of time.

With all of the above in mind and provided the performance penalties observed in many tests, Oracle recommends not to exceed 10 msecs of latency (RTT) for SOA Multi Data Center Active-Active systems when the latency affects database communications. Systems may operate without issues, but the transaction times will increase considerably. Latencies beyond 10 msecs will also cause problems in the Coherence cluster used for deployment and JTA and web services timeouts for most common composites. This makes the solutions presented in this paper suitable primarily for Metropolitan Area Networks with low latency between sites (for example, the distance from San Francisco to Boston is around 4330 kms and typical latencies are approximately 30-40 msecs).

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 SOA Active-Active Multi Data Center Deployment" section. Each site locally uses an Oracle Fusion Middleware SOA Enterprise Deployment Topology (separation of WSM-PM servers, shared storage and directory structure, etc.). The system requirements are those specified in the Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite.

Additionally, the following requisites must be met:

Network

Latency between the two sites used in the design should not be higher than 10 msecs RTT. The bandwidth requirements will vary based on the type of payloads used by each SOA system.

Shared Storage vs. Database for Transaction Logs and Persistent stores

The topology addressed in this paper was tested using both database-based and file-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 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, deployment plans, and some adapters like File Adapter). Using TLOGs and JMS in the database has a penalty, however, on the system's performance. This penalty is increased when one of the sites needs to cross communicate with the database on the other site. This penalty applies also to JMS Servers that use database persistent stores for AQ destinations. Ideally, from a performance perspective, a shared storage that is local to each site should be used for both types of stores and the appropriate replication and backup strategies at storage level should be provisioned in order to guarantee zero data loss without performance degradation. Whether using database stores will be more suitable than shared storage for a system depends on the criticality of the JMS and transaction data, because the level of protection that shared storage provides is much lower than the database guaranties.

Additionally, 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 SOA Suite* section 2.1.3.1. The global load balancer should allow rules based on the originating server's IPs (an example is provided for F5 Networks).

Oracle Fusion Middleware SOA Components and Versions in Scope

This document is based on Oracle Fusion Middleware SOA 11*g* R1 PS6. Any later release should also work in similar configurations. The Oracle Fusion Middleware SOA components verified are:

- Oracle BPEL
- Oracle Mediator
- Oracle Rules
- Oracle EDN
- Oracle Technology Adapters File, Database, and JMS Adapter
- Oracle MQ Adapter
- Oracle AQ Adapter
- Oracle Apps Adapter

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 Oracle Fusion Middleware SOA Active-Active Multi Data Center Deployment

Image 7 depicts the main pieces (without details on specific routing or Oracle WebLogic Server domain aspects) of the Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment addressed in this paper.

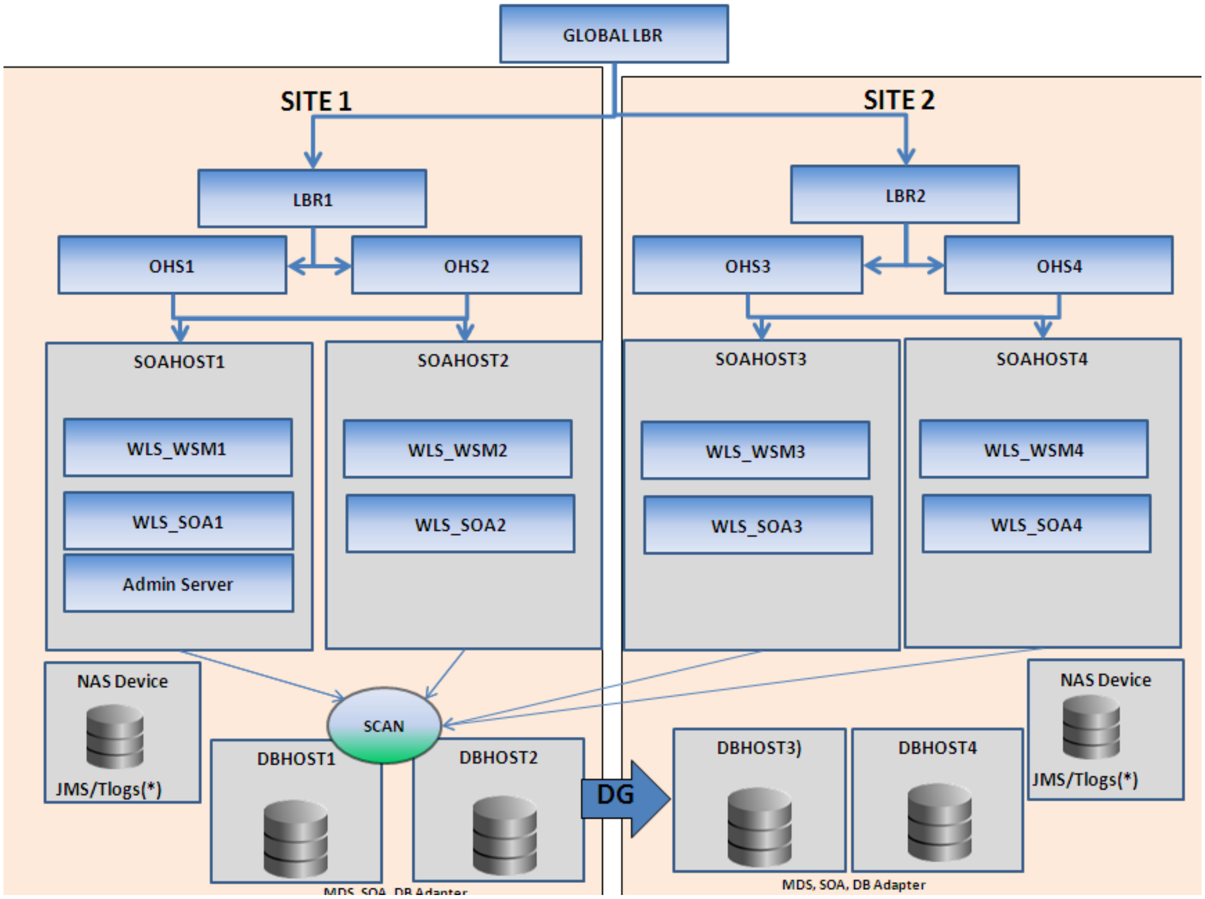


Image 7: Components in an Oracle Fudion Middleware SOA Active-Active Multi Datacenter Deployment

In Image 7 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 SOA components (service engines, adapters, and infrastructure). The two environments share one unique database that is accessed

CONCURRENTLY by servers in both sites. The following sections provide details for each tier.

Database Tier

The synchronicity requirements and data types used by the different Oracle Fusion Middleware SOA Suite components limit the possible approaches for the Oracle Fusion Middleware SOA database in a Multi Data Center Active-Active deployment. This document addresses only a solution where the Oracle Fusion Middleware SOA database uses Data Guard to synchronize an active database in Site1 with a passive database in Site2. Although other Active-Active 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 SOA 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 at a 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 Database is configured in each site 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.

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 LBR 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

[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 SOA does not support Oracle Active Data Guard because the SOA components execute and update information regarding SOA composite instances in the database as soon as they are started.

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 SOA services. These GLBR rules can be summarized as follows:

- If requests come from Site1 (callbacks from the SOA 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 SOA server s 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.
- 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 SOA 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 SOA Enterprise Deployment Topology for maximum availability and security. 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 and one single cluster for Oracle Fusion Middleware SOA Suite components. This model is also known as a Stretched Cluster. In this topology, all servers (WSM-PM 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. Each site uses an individual shared storage for JMS and Transactions Logs (TLOGs), or alternatively the database is used as persistent store. For contention and security reasons it is not recommended to use shared storage across sites. Disk mirroring and replication from Site1 to Site2 and vice versa can be used to provide a recoverable copy of these artifacts in each site. 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 WSM-PM ones (WSM_Cluster). 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 SOA and all Oracle WebLogic Servers point to the same SOA and MDS schemas. Image 8 describes the topology.

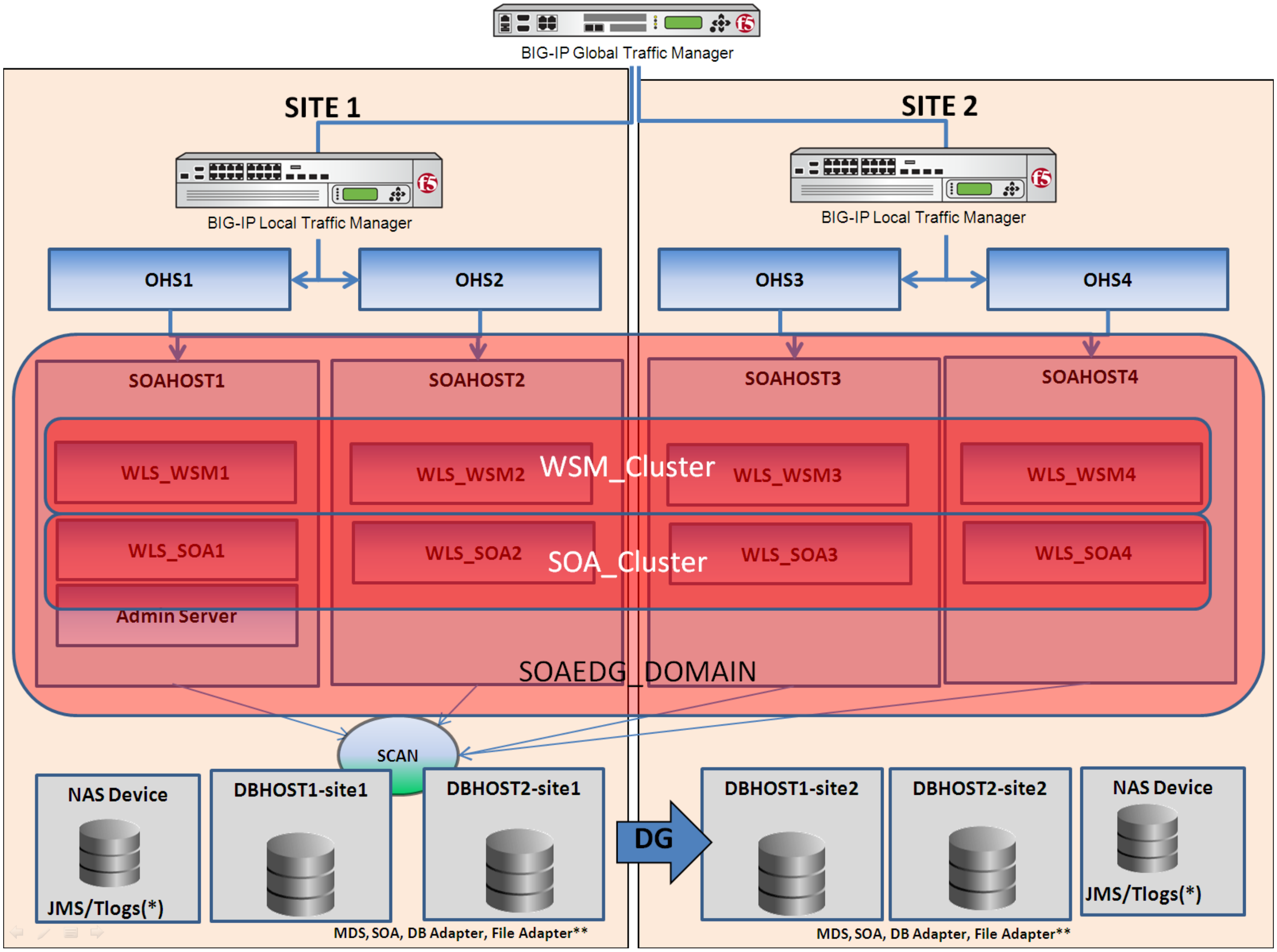


Image 8: Stretched cluster model for Multi Data Center Active-Active Deployment

Characteristics of the Design

Availability: The stretched cluster design uses an OHS configuration 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. 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. Specifically for BPEL, automatic recovery can be achieved from either site irrespective of the node that originated the instance. 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).

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 for SOA 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 (file adapter, TLOGS, etc.), a Multi Data Center Active-Active Deployment is administrated like a standard cluster. This makes its administration overhead very low.

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 SOA Servers and the SOA database. This will be the key to the performance of the Multi Data Center Active-Active Deployment. Image 9 shows the percentage of traffic between the SOA servers in Site2 and the different addresses in Site1 during a stress test. Notice that 98% of the traffic happens between the servers and the database (also located in Site1).

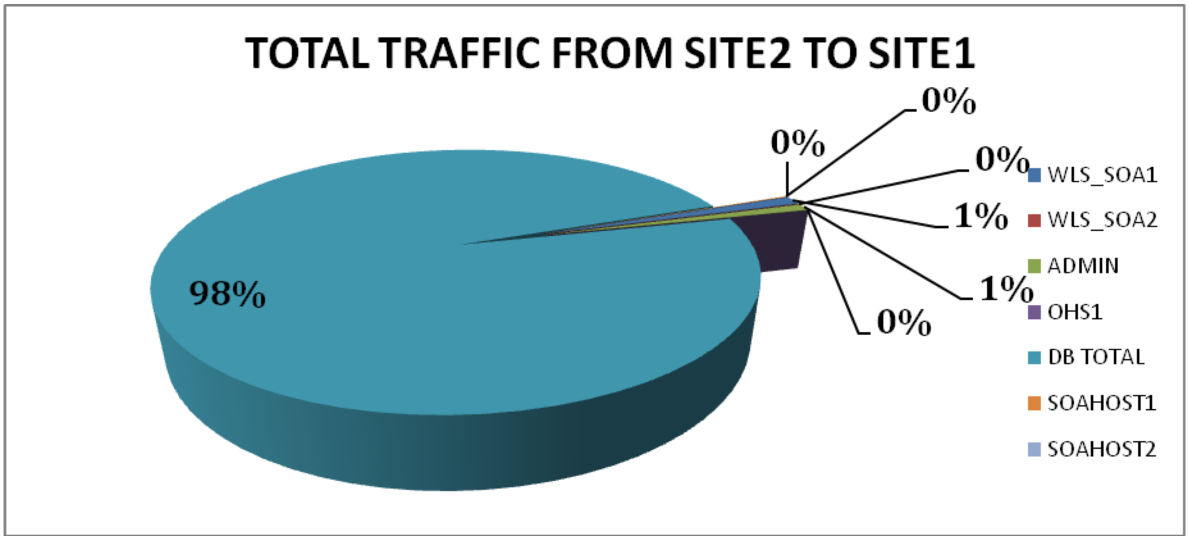


Image 9: Stretched Cluster Model. Traffic percentages to each address in Site1 coming from SOA servers in Site2 during a

[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.

stress test

Other Resources

The two sites may or may not share other external resources. These resources include LDAP, identity stores, policy stores, external JMS destinations, external web services, etc. The configuration details for these external resources are out of the scope of this document. It is required, however, that these resources are consistent in both sites. Notice that asynchronous callbacks may re-hydrate instances that were initiated in a different site. For these to provide a consistent behavior, the same external resources must be available in both sites (this is also required for automatic recovery purposes: any Oracle WebLogic Server can become a cluster master and perform recovery in either site).

Configuring the Oracle Fusion Middleware SOA Active-Active Topology

The following sections provide the steps for configuring an Oracle Fusion Middleware SOA 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 SOA Suite* 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. In summary the steps are:

1. Configure GLBRs and LBRs as per the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* but with the appropriate rules for local routing.
2. Configure OHS as per the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* but with routing restricted to each site.
3. Configure the application tier with special steps for the following:
   - Shared storage/directory configuration
   - Server migration configuration
   - JMS configuration
   - JMS Adapter and File Adapter configuration
   - Data Source configuration
   - 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 for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment

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 SOA 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 SOA Suite*. Because all of the components addressed in this document depend on the availability of the Oracle Fusion Middleware SOA Service Infrastructure application, it is recommended that you set the LBRs to monitor the `/soa-infra/` URL[4] to determine the availability of a SOA server. This will eliminate undesired routings when the SOA Oracle WebLogic Server is RUNNING but the SOA subsystem is not really available (these routings can occur when the monitor is set on the root context (`/`) URL).

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[5].

1. It is assumed that the appropriate listener already exists in the GTM
2. In the Global Traffic 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 and links that share the same subnet on the network). The defaults are appropriate.
3. In the Global Traffic 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 system on the network) as follows:
    a. Use the address of the first site's Local LBR for this server.
    b. For Product, select BIG-IP System (single).
    c. 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).
    d. Add as virtual server the address on which the local LBR is listening for SOA requests. For this virtual server, 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).
4. In the Global Traffic Menu of the F5 Administration Console create a new pool (for future reference we will call it the MDCpool). 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

[4] Include the backslash at the end of the URL, or the monitor will fail due to the Oracle WebLogic Server redirecting to the front end address.

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

number that points to a specific resource on the network.
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.

5. In the Global Traffic Menu 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 as follows:
    a. Use the FQDN that will be used to access the SOA Multi Data Center Deployment system.
    b. Add the pool previously created to the Wide IP.
    c. Enable persistence for the Pool.
    d. Use Round Robin as the load balancing method.

With these settings the F5 GTM should round robin request to both sites or datacenters. To do the smart routing required for callbacks, internal web service invocations, etc., define two more pools:

- A pool containing ONLY the LBR in Site1 (Site1pool)
- A pool containing ONLY the LBR in site 2 (Site2pool)

Add the following iRule for Global Traffic and assign it to the Wide IP for the system.

```
when DNS_REQUEST {
  if { [IP::addr [IP::client_addr] equals 10.10.10.10/24 ] } {
    pool Site1pool
  } elseif { [IP::addr [IP::client_addr] equals 20.20.20.10/24] } {
    pool Site2pool
  } else {
    pool   MDCPool
  }
```

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.

Configuring Oracle HTTP Server for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment

In a stretched cluster model, and to reduce traffic across sites, the Oracle HTTP Servers (OHS) do not use dynamic cluster notifications, instead they get configured with a static list of servers. This has the caveat of slower failure detection (when a SOA server crashes the HTTP server takes more time to detect the failure than with a dynamic list) and requires updates in the configuration if new servers are added. However, it improves the system's performance. The following excerpts from the mod_wl_ohs.conf files in the OHS provide an example of the required configuration for routing to the soa-infra web application.

Site1:

```
# SOA soa-infra app
<Location /soa-infra>
    SetHandler weblogic-handler
    WebLogicCluster Site1_server1.mycompany.com:8001,Site1_server2.mycompany.com:8001
   DynamicServerList OFF
</Location>
```

Site2:

```
# SOA soa-infra app
<Location /soa-infra>
    SetHandler weblogic-handler
    WebLogicCluster Site2_server1.mycompany.com:8001,Site2_server2.mycompany.com:8001
DynamicServerList OFF
</Location>
```

Configuring the Application Tier of an Oracle Fusion Middleware SOA AA DR System for a Stretched Cluster

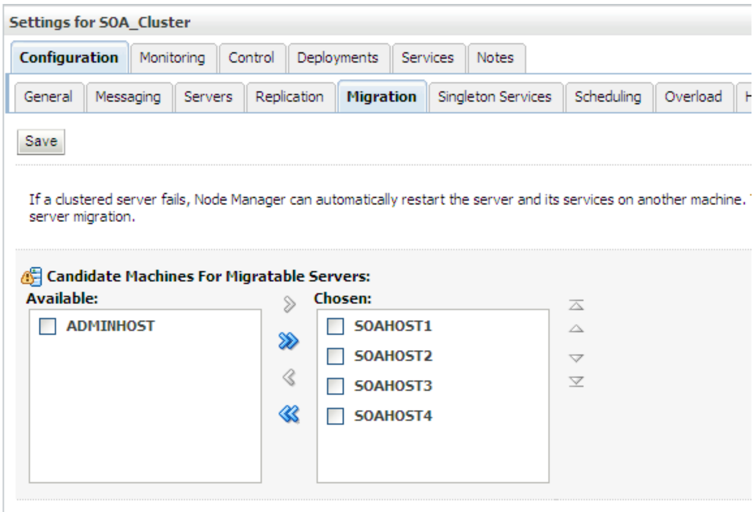
The stretched cluster design is an Enterprise Deployment Topology scaled out to two additional servers in Site2. There are some aspects to consider that can make the system more scalable and that will minimize the possible performance degradation caused by the latency across sites.

The first site is installed as described in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*. Variations in terms of installing the Oracle WSM-PM Cluster collocated with the SOA Cluster and other high availability topologies are allowed. The second site (Site2) is configured using the steps provided in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* section 16.6. The following aspects need to be considered:

- **Binaries/Installations:** The second site uses its own redundant binaries (that is, at least 2 binary installations should be used per site for high availability).
- **Paths:** The binary installations, `aserver` domain directory, deployment plans, transaction logs, and file adapter directories in Site2 should use the same path as Site1.
- **Shared Storage:** The binary installations, `aserver` domain directory, deployment plans, transaction logs, and file adapter directories reside on shared storage as indicated in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*, but each site uses a shared storage local to the site. That is, designs where servers in Site2 need to access shared storage on Site1 and vice versa **should be avoided**. This means that in the Stretched Cluster a "manual" separation of artifacts (file adapter, deployment plans, JMS, and TLOGs) is done. With this configuration, automatic server migration across sites is not possible because transaction and JMS message recovery is precluded (as opposed to a typical scale out scenario in the context of one single site, where server migration is configured using all machines as candidate).
- **Oracle HTTP Server Configuration:** Observe the details in Configuring Oracle HTTP Server for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment when configuring OHS routing.
- **SOA Cluster Configuration:** Specify the GLBRs virtual server's address created in the Configuring Load Balancers and Global Load Balancers for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment sections as callback URL (that is, as front end address for the SOA cluster) that will front end the system. For the SOA cluster's Cluster Address include **all the servers** in the Stretched Cluster:

```
Site1_server1.mycompany.com:8001,Site1_server2.mycompany.com:8001,
Site2_server1.mycompany.com:8001,  Site2_server2.mycompany.com:8001
```

- **Server Migration configuration:** In the stretched domain design servers use only those machines in their same site as candidates for migration. Use the steps in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* for configuring server migration with these additional considerations:
  - For the Cluster do not specify any machines as candidates as shown here:



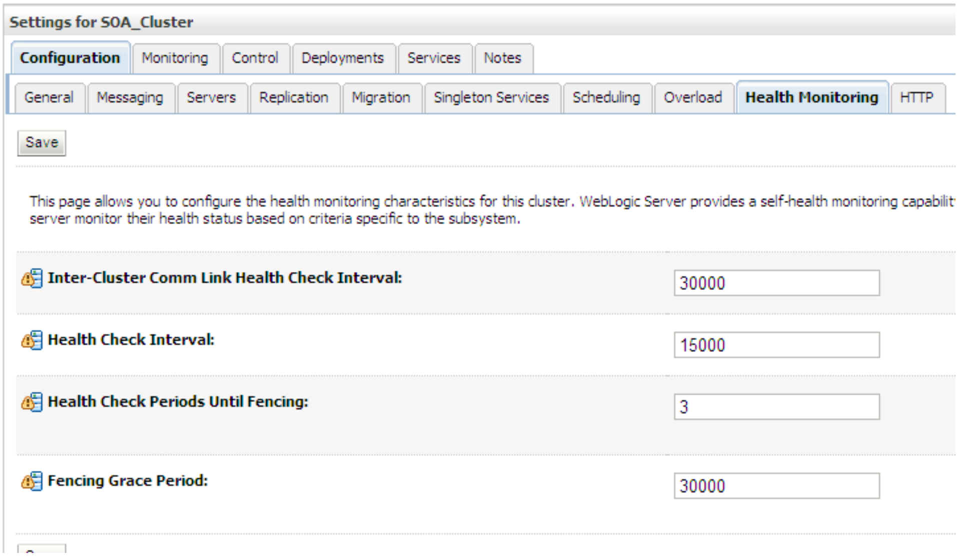
If a clustered server fails, Node Manager can automatically restart the server and its services on another machine.
server migration.

Candidate Machines For Migratable Servers:

Available:
☐ ADMINHOST

Chosen:
☐ SOAHOST1
☐ SOAHOST2
☐ SOAHOST3
☐ SOAHOST4

  - 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. To increase the Health Check Interval use the Health Monitoring tab for the cluster as shown here:



Settings for SOA_Cluster

This page allows you to configure the health monitoring characteristics for this cluster. WebLogic Server provides a self-health monitoring capabilit
server monitor their health status based on criteria specific to the subsystem.

Inter-Cluster Comm Link Health Check Interval:	30000
Health Check Interval:	15000
Health Check Periods Until Fencing:	3
Fencing Grace Period:	30000

- **Transaction and Persistence logs configuration:**
  - For file-based persistence stores use two shared storages, each one local to each site (to enable server migration inside each site). For consistency, and to simplify backup procedures and other administration operations, the same paths can be used in both sites since persistence stores are qualified with the server's name. However, using a different path identifying each site is advisable to facilitate recovery (scenarios where Site2 needs to recover TLOGs from Site1 or vice versa). For example, in Site1 use the following path for the persistence store (See section 9.8 in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*):

```
ORACLE_BASE/admin/domain_name/soa_cluster_name/tlogs_Site1
```

And in Site2 use:

```
ORACLE_BASE/admin/domain_name/soa_cluster_name/tlogs_Site2
```

When a server fails and other servers in the same site remain available, server migration takes care of resuming transactions. When all of the servers in one site are unavailable and another site needs to resume transactions it is necessary to start MANUALLY the appropriate servers in the available site, as described in the following procedure:

1. Make the appropriate transaction logs and persistent stores available in the new site (either through the appropriate disk replication or backups).
2. Enable the server's virtual host name in the appropriate node

NOTE: Because different sites or locations may use different gateways, netmasks, etc. it may not be possible to use the same VIP for the server in Site2. If a virtual host name was used for the server in Site1, the server may use a different virtual IP on Site2 while preserving the host name as a listen address.

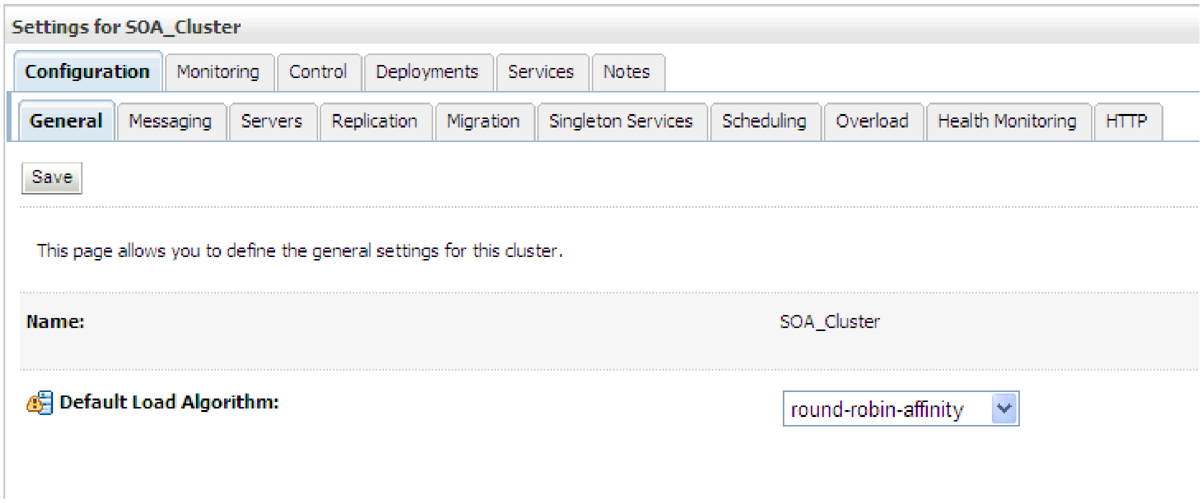
3. Change the machine assigned to the failed server so that it uses a machine in the failover site.
4. Start the server normally using the Oracle WebLogic Server Administration Console (pending transactions should be resumed appropriately).

   NOTE: It is expected that the exact same resources are available in both sites (for example, files if the file adapter is used). This is a requirement irrespective of transaction log configurations since otherwise BPEL could not perform auto recovery indistinctly in either site.

- For database JMS and TLOG stores, servers in the two sites can point to the same database and the same schema.

  NOTE: As described in the Shared Storage section, using TLOGs in the database has the advantage of incorporating the propagation of transaction logs to Site2 using Data Guard; however, this can cause an average performance degradation of around 5-10% for the FOD example (the impact will vary depending on the application or composite type). This effect is aggravated when servers need to access the database in another site. Using Oracle WebLogic Server JMS with database persistence in Multi Data Center Active-Active Deployment will cause an even larger performance impact, especially with large payloads. The benefits of automatic preservation of messages vs. the performance degradation caused should be considered.

- **Replicated services:** To eliminate cross-site traffic, it is recommended that you use local affinity for JNDI context factory resolution. To do this, set the Default Load Algorithm to "round-robin affinity" (the default is round_robin) or to any "affinity-based" algorithm. This can be done in the General tab for the SOA Cluster Configuration as shown:



This page allows you to define the general settings for this cluster.

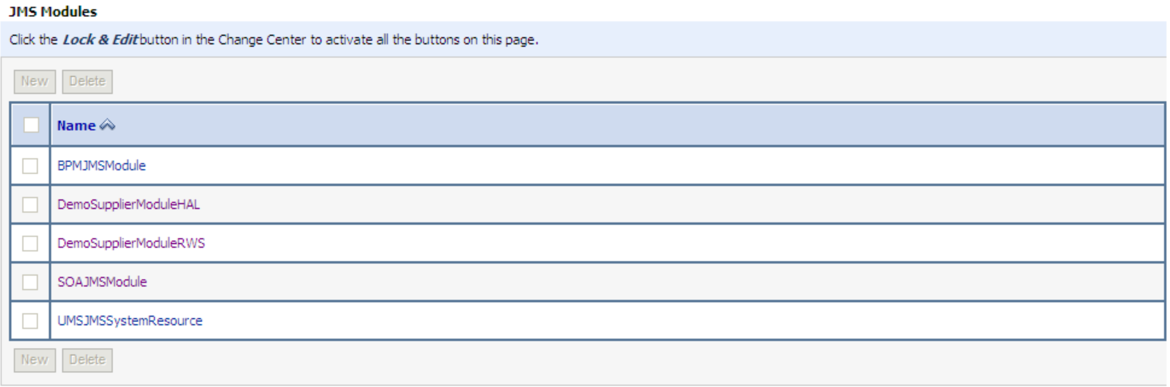
Name: SOA_Cluster

Default Load Algorithm: round-robin-affinity

- **JMS Destinations:** Oracle WebLogic Server provides initial context and server affinity for client connections. JMS connection factories can be used further to restrict connections and provide site-affinity for a stretched cluster. Connection factories are replicated objects in the cluster. Clients get JNDI contexts, and from these JNDI contexts obtain a reference to the appropriate connection factory. By setting the default load balancing algorithm for the cluster to any affinity-type protocol (round-robin-affinity, weight-affinity, random-affinity) connection factories' stubs will be generated from a local server as first option. The connection factory client stubs then use the list of all servers that the connection factory is targeted to. By using local affinity at the connection factory level (which is set by default), the connection factory will connect to JMS servers that are local as first preference. It is therefore recommended to use affinity at cluster level (configured in the previous section) and also in the connection factories (which is the default). This will reduce undesired cross-site communication for Uniformed Distributed Destinations (UDDs) and Uniform Distributed Topics (UDTs). This mechanism, however, is not deterministic. For systems that use JMS destinations intensively and with large payloads it is recommended to completely avoid cross-site traffic for JMS invocations. This can be achieved using a selective targeting of subdeployment modules in each site. Consider for example the case where a specific UDD (for example, DemoSupplierQueueUDD) needs to be available in both sites, but the system is using large JMS payloads intensively. Traffic can be deterministically restricted to one site by using separate JMS modules in each site. Follow the following steps for this type of approach:

  NOTE: This configuration forces destinations to be isolated (they are just available locally in each server); hence it may not be applicable for some systems where subsequent processing may depend on specific messages properties or when clients are remote to the destinations

1. Using the Oracle WebLogic Server Administration Console select Services > Messaging > JMS Modules and create a separate JMS module for each site. For each module select the servers in each site respectively as targets.

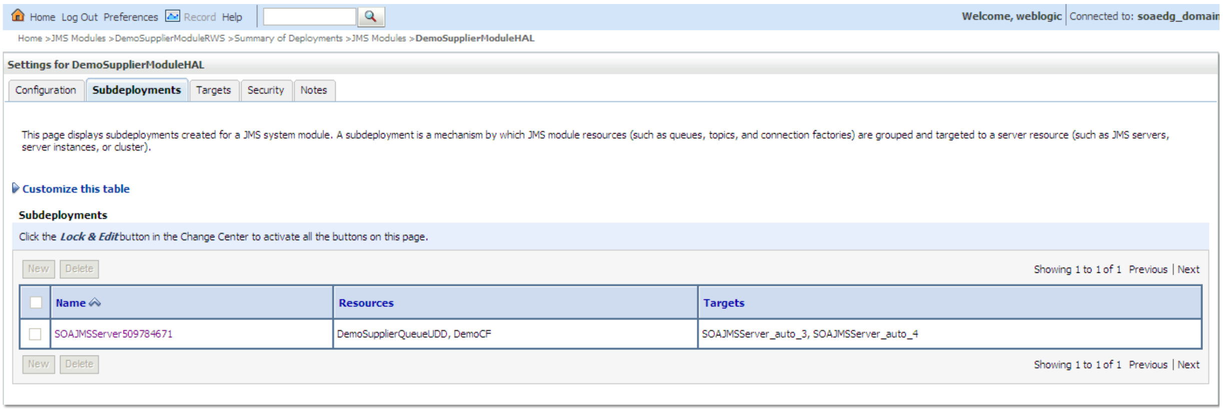


JMS Modules

Click the ***Lock & Edit*** button in the Change Center to activate all the buttons on this page.

□	Name
□	BPMJMSModule
□	DemoSupplierModuleHAL
□	DemoSupplierModuleRWS
□	SOAJMSModule
□	UMSJMSSystemResource

2. Create a separate subdeployment module for those JMS servers that will host the destinations.



Home Log Out Preferences Record Help
Welcome, weblogic
Connected to: soaedg_domain

Home >JMS Modules >DemoSupplierModuleRWS >Summary of Deployments >JMS Modules >**DemoSupplierModuleHAL**

Settings for DemoSupplierModuleHAL

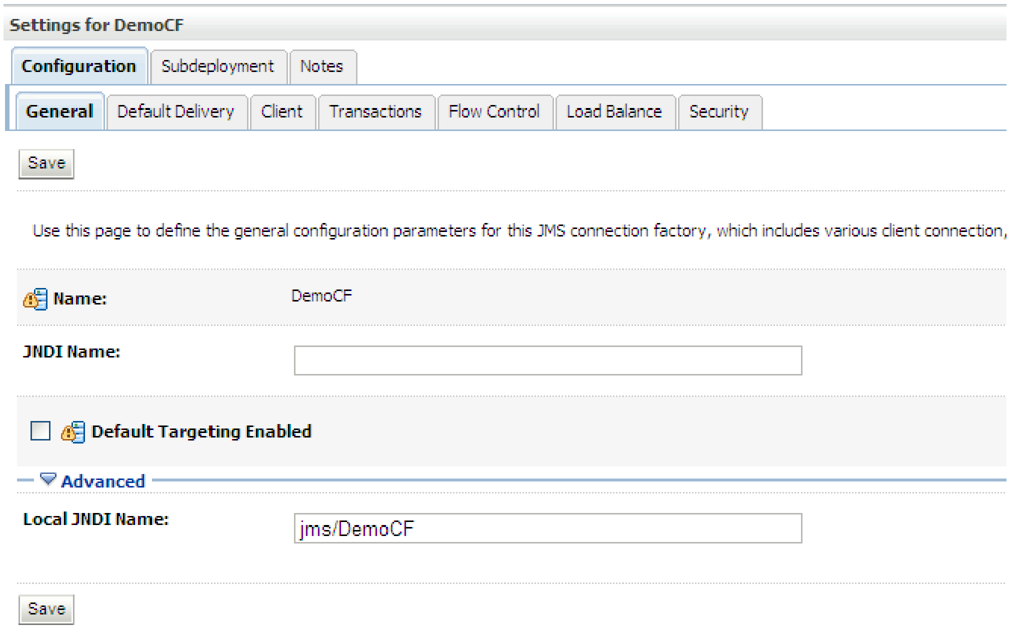
This page displays subdeployments created for a JMS system module. A subdeployment is a mechanism by which JMS module resources (such as queues, topics, and connection factories) are grouped and targeted to a server resource (such as JMS servers, server instances, or cluster).

Subdeployments

Click the ***Lock & Edit*** button in the Change Center to activate all the buttons on this page.

□	Name	Resources	Targets
□	SOAJMSServer509784671	DemoSupplierQueueUDD, DemoCF	SOAJMSServer_auto_3, SOAJMSServer_auto_4

3. Create the required UDDs and the pertaining connection factories. For UDDs use the same JNDI name in both sites. For connection factories, specify only local JNDI name. This name will be bound only on the local server instance and will not be propagated to the rest of the cluster. Thus the connection factory stubs will hold a reference only to the local JMS server when establishing a connection.



Settings for DemoCF

Use this page to define the general configuration parameters for this JMS connection factory, which includes various client connection,

Name: DemoCF

JNDI Name:

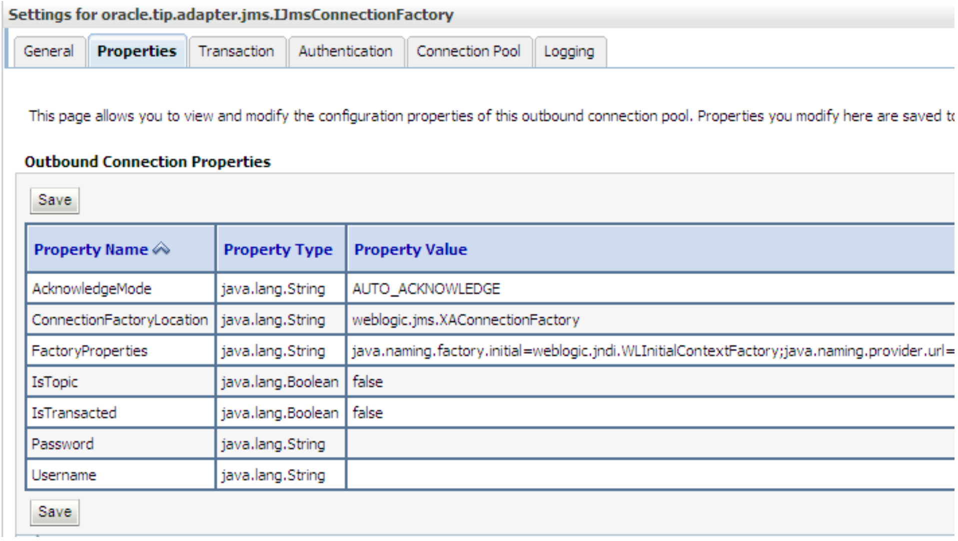
□ Default Targeting Enabled

Local JNDI Name: jms/DemoCF

4. Assign the subdeployment created in the previous step to the connection factories and UDDs and activate all changes.

This will guarantee that each server uses only local destinations avoiding cross-site JMS invocations.

- **JMS Adapter configuration:** The JMS adapter requires configuring specific connection factory properties that include the list of servers available for JNDI context retrieval.



Settings for oracle.tip.adapter.jms.IJmsConnectionFactory

This page allows you to view and modify the configuration properties of this outbound connection pool. Properties you modify here are saved t

Outbound Connection Properties

Property Name	Property Type	Property Value
AcknowledgeMode	java.lang.String	AUTO_ACKNOWLEDGE
ConnectionFactoryLocation	java.lang.String	weblogic.jms.XAConnectionFactory
FactoryProperties	java.lang.String	java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.provider.url=
IsTopic	java.lang.Boolean	false
IsTransacted	java.lang.Boolean	false
Password	java.lang.String	
Username	java.lang.String	

Besides the considerations explained in the "JMS Destinations" bullet above, it is recommended that the JNDI URL used by the adapter contains a list of "local" servers in each site (that is, it uses only servers in Site1 for the configuration in Site1 and only servers in Site2 for the configuration on Site2). This will guarantee site context affinity[6]. To achieve this configuration follow these steps:

1. Update the Outbound Connection Pool properties for the instance that the adapter will use (as described in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*, Section 9.9.2) specifying as java.naming.provider.url the list of server in Site1). For example:
   ```
   java.naming.provider.url=t3://Site1_server1:8001,Site1_server2:8001
   ```

2. Save the update in the Administration Console.

3. Copy the generated deployment plan to the mirror location on Site2. For example, from server1_Site1:
   ```
   scp /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/JMSPlan.xml
   Site2_server1: /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/
   ```

4. Edit the deployment plan in Site2 and replace the server list with the list of servers in Site2.

   Site1 Deployment Plan excerpt:

   ```
   <name>ConfigProperty_FactoryProperties_Value_13243793917130</name>
           <value>java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
   rovider.url=
   t3://Site1_server1:8001,Site1_server2:8001;java.naming.security.principal=weblogic;java.naming.
   security.credentials=welcome1
           </value>
   ```

[6] For consumer scenarios, the JMS Adapter implicitly sets up at least one consumer on each member in the Distributed Destinations so the "locality" effect is mitigated.

Site2 Deployment plan excerpt:

```
<name>ConfigProperty_FactoryProperties_Value_13243793917130</name>
        <value>java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
rovider.url=
t3://Site2_server1:8001,Site2_server2:8001;java.naming.security.principal=weblogic;java.naming.
security.credentials=welcome1
        </value>
```

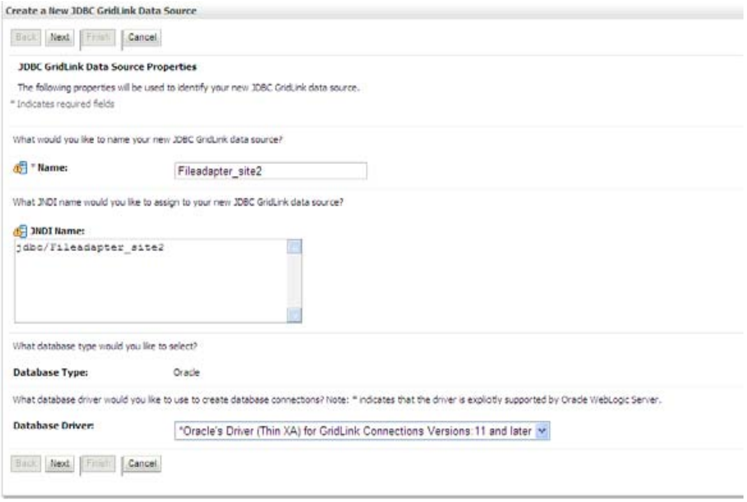
5. Update the JMS Adapter deployment using the modified deployment plan (the same location in both sites, but effectively a different file). The update will use the deployment plan file in Site1 for the servers in Site1 and the deployment plan file in Site2 for the servers in Site2.
   NOTE: If manual migration of a server across sites is necessary, the list of servers must be updated to contain the new member in the site.

- **File Adapter considerations:** Although the two sites act separately processing files (using separate share storage for each site), by default the same Data Source is used with the same schema and tables for file locking and file mutex. This schema is used for guaranteeing that the same file is processed only by one server at a time and that two adapter instances will not write to the same file concurrently. It is appropriate to use the same database for outbound operations since a unique sequence is used for the mutex that is used to prevent overwriting files. For inbound operations, however, and since the two sites use (by default) the same schema in the same database, "under processing" scenarios could take place. This is because file adapter instances in either site can mark a file name as "blocked". Because the same file name can be used in both sites, this can block its processing in both locations but the file will be consumed in only one of them. To avoid these situations, there are multiple alternatives:
  - Guaranteeing unique file names for input operations across sites (notice that the path is the same since the corresponding jca file is unique in a stretched cluster).
  - Using different schemas in the same database to guarantee proper processing.
  - Using a separate database for mutex and locks to provide better performance than the previous alternatives and allow total parallel processing without races for locks if the same file is present in both site's input directory.

    NOTE: If due to middle tier failure at site level a server from Site1 needs to be started in Site2 (or the other way around), the Data Source must be updated to point to a local database/schema for improved performance.

In order to use different databases or separate schemas in the SOA database, the appropriate schema owner and tables need to be created and a new Data Source needs to be used for the File Adapter. The following steps provide the details for reconfiguring the adapter.

1. Site1 will use the default schema.
2. For Site2, use a different schema from the existing one (SOINFRA) (if you are using different databases you can actually use the same one). You can reuse other existing schemas (such as the one used for server migration leasing) or create a new one using the steps in section 14.2 in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*.
3. Connect to the database with the new schema.
4. Use the FILEADAPTER script in Appendix A: File Adapter Locks and Muxers to create the appropriate mutex and lock database objects.
5. Create a new GridLink Data Source for the schema. This must be of type “Oracle's Driver (Thin XA) for GridLink Connections Versions: 11 or later".



Create a New JDBC GridLink Data Source

JDBC GridLink Data Source Properties

The following properties will be used to identify your new JDBC GridLink data source.

* Indicates required fields

What would you like to name your new JDBC GridLink data source?

* Name: Fileadapter_site2

What JNDI name would you like to assign to your new JDBC GridLink data source?

JNDI Name: jdbc/Fileadapter_site2

What database type would you like to select?

Database Type: Oracle

What database driver would you like to use to create database connections? Note: * indicates that the driver is explicitly supported by Oracle WebLogic Server.

Database Driver: *Oracle's Driver (Thin XA) for GridLink Connections Versions: 11 and later

6. Enable FAN for the Data Source. You can copy the JDBC URL and ONS properties from the existing SOA schemas.
7. Target the Data Source to the SOA_Cluster.
8. Make a backup of the deployment plan for the File Adapter.

```
cp /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/FileAdapter.xml
/u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/FileAdapter.xml.orig
```

9. Update the inbound Data Source with the JNDI name used in the new one.

   ```
   jdbc/Fileadapter_Site2
   ```

10. Save the update in the Administration Console.
11. Copy the generated deployment plan to the mirror location on Site2. For example:

    ```
    scp /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/FilePlan.xml
    Site2_server1: /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/
    ```

12. Revert to the original file in Site1

    ```
    cp
    /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/FileAdapter.xml.orig
    /u01/app/oracle/admin/soaedg_domain/soa_cluster/dp/FileAdapter.xml
    ```

13. Access the Deployments screen in the Oracle WebLogic Server Administration Console and update the File Adapter deployment with the modified Deployment plan. Site1 will use the original Data Source while Site2 will use the new one.

- **Configuring OWSM's Java Object Cache for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment:** Oracle Web Services Manager (OWSM) uses the Java Object Cache to improve performance and for guaranteed consistency when creating or updating policies. WSM-PM servers read from this in-memory cache instead of reading from the database, in most cases improving performance. In a typical configuration, the Java Object Cache (JOC) is configured as a distributed cache among all WSM-PM servers in a cluster. In an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment these synchronized caches allow for the sharing of objects and write to the same database. To configure JOC across both sites, follow the steps in section 8.5.5 of the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*. This will imply additional network traffic between the two sites, depending on the frequency of policy updates this may cause additional overhead in the system.

Configuring Data Sources for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment

The Data Sources used by Oracle Fusion Middleware SOA Suite should be configured to automate failover of connections in case there is failover or switchover of the active database. A Data Guard database role transition (where a standby database becomes the new primary database) can be performed without restarting Oracle WebLogic Servers[7]. The following Data Sources need to be configured properly to automate this failover.

[7] Refer to Database Failures: Data Guard Switchover and for details on server migration implications when a database role change takes place.

- EDNDataSource
- EDNLocalTxDataSource
- mds-owsm
- mds-soa
- OraSDPMDataSource
- SOADataSource
- SOALocalTxDataSource

Additionally any persistence store using database and the leasing Data Source used for server migration (if both sites use the same database for leasing) should also be configured to automate failover. The Oracle WebLogic Server and Highly Available Oracle Databases: Oracle Integrated Maximum Availability Solutions white paper contains all of the details required for configuring Data Sources for transparent fail over in disaster recovery scenarios for the SOA database. Beyond the required database configuration (services, Oracle Net, and Data Guard), and since the default recommendation is to use GridLink Data Sources, the modifications on the middle tier should typically be limited to:

- Updating the ONS configuration to include both production and standby site ONS
- Updating the JDBC URL to include the appropriate services in both sites

For database connection pool sizing, consider that the total number of processes in the database must sustain the addition of the pools of both sites. The following is a sample JDBC URL for the SOA Data Source in an Oracle Fusion Middleware SOA Multi Data Center Active-Active configuration where the database uses Data Guard:

```
jdbc:oracle:thin:@
(DESCRIPTION_LIST =
        (LOAD_BALANCE = off)
        (FAILOVER = on)
        (DESCRIPTION =
                (CONNECT_TIMEOUT = 10)
                (TRANSPORT_CONNECT_TIMEOUT = 3)
                (RETRY_COUNT = 3)
                (ADDRESS_LIST =
                        (LOAD_BALANCE = on)
                        (ADDRESS =
                                (PROTOCOL = TCP)
                                (HOST = scanSite1.mycompany.com)
                                (PORT = 1521)
                        )
                )
                (CONNECT_DATA =
                        (SERVICE_NAME =soaedg.mycompany.com)
                )
        )
        (DESCRIPTION =
                (CONNECT_TIMEOUT =10)
                (TRANSPORT_CONNECT_TIMEOUT = 3)
                (RETRY_COUNT = 3)
                (ADDRESS_LIST =
                        (LOAD_BALANCE = on)
                        (ADDRESS =
                                (PROTOCOL =TCP)
                                (HOST = scanSite2.mycompany.com)
                                (PORT = 1521)
                        )
                )
                (CONNECT_DATA =
                        (SERVICE_NAME = soaedg.mycompany.com))
                )
        )
```

Composite and MDS Deployments and Updates: Oracle Coherence Configuration

Oracle Fusion Middleware SOA Suite uses Oracle Coherence for propagating both composite deployments and metadata (MDS) updates across a cluster. Oracle recommends using unicast for Coherence communications in context of the Oracle FMW SOA Suite EDG. Unicast is most often used when multicast networking is undesirable or unavailable in an environment or when an environment is not properly configured to support multicast. This is typically the case for systems that span a MAN. The Well Known Addresses (WKA) feature is a Coherence mechanism that allows cluster members to discover and join a cluster using unicast instead of multicast. All multicast communications are disabled for the coherence cluster if WKA is enabled.

The Oracle Fusion Middleware SOA Multi Data Center Active-Active Stretched Cluster design use a single Coherence cluster for SOA that spans the two sites involved. Coherence is sensitive to delays in cluster formation and responsiveness to heartbeats from members that are part of the cluster. With latencies ranging from 5/10 msecs RTT no issues were detected for different operations involving the Coherence infrastructure used by SOA (cluster creation, new nodes joining cluster across sites, MDS updates, and composite deployments). For latencies above 10 msecs RTT, Coherence may report errors both in cluster formation and also in keeping members updated. Expect messages like the following on the servers that come up first in such situations:

```
<TIME> <Warning> <Coherence> <BEA-000000> <TIME Oracle Coherence GE 3.7.1.6
<Warning> (thread=Cluster, member=n/a): This Member(Id=0, Timestamp=TIME,
Address=X.X.X.X:8088, MachineId=43916,
Location=site:,machine:HOSTNAME,process:29028, Role=WebLogicServer) has been
attempting to join the cluster using WKA list [Site1_server1:8088,
Site1_server2:8088, Site2_server1:8088, Site2_server2:8088] for 30 seconds without
success; this could indicate a mis-configured WKA, or it may simply be the result
of a busy cluster or active failover.>
```

If there are multiple failed tries for joining the cluster, the nodes trying to join will reports messages like the following:

```
<TIME> <Warning> <Coherence> <BEA-000000> <TIME Oracle Coherence GE 3.7.1.6
<Warning> (thread=Cluster, member=n/a): Received a discovery message that indicates
the presence of an existing cluster that does not respond to join requests; this is
usually caused by a network layer failure:
Message "SeniorMemberHeartbeat"
  {
  FromMember=Member(Id=2, Timestamp=TIME, Address=X.X.X.X:8088, MachineId=7604,
Location=site:,machine:HOSTNAME,process:18544, Role=WebLogicServer)
  FromMessageId=0
  Internal=false
  MessagePartCount=0
  PendingCount=0
  MessageType=17
  ToPollId=0
  Poll=null
  Packets
    {
    }
  Service=ClusterService{Name=Cluster, State=(SERVICE_STARTED, STATE_ANNOUNCE),
Id=0, Version=3.7.1}
  ToMemberSet=null
  NotifySent=false

  LastRecvTimestamp=TIME
  MemberSet=MemberSet(Size=2, ids=[2, 3])
  }>
```

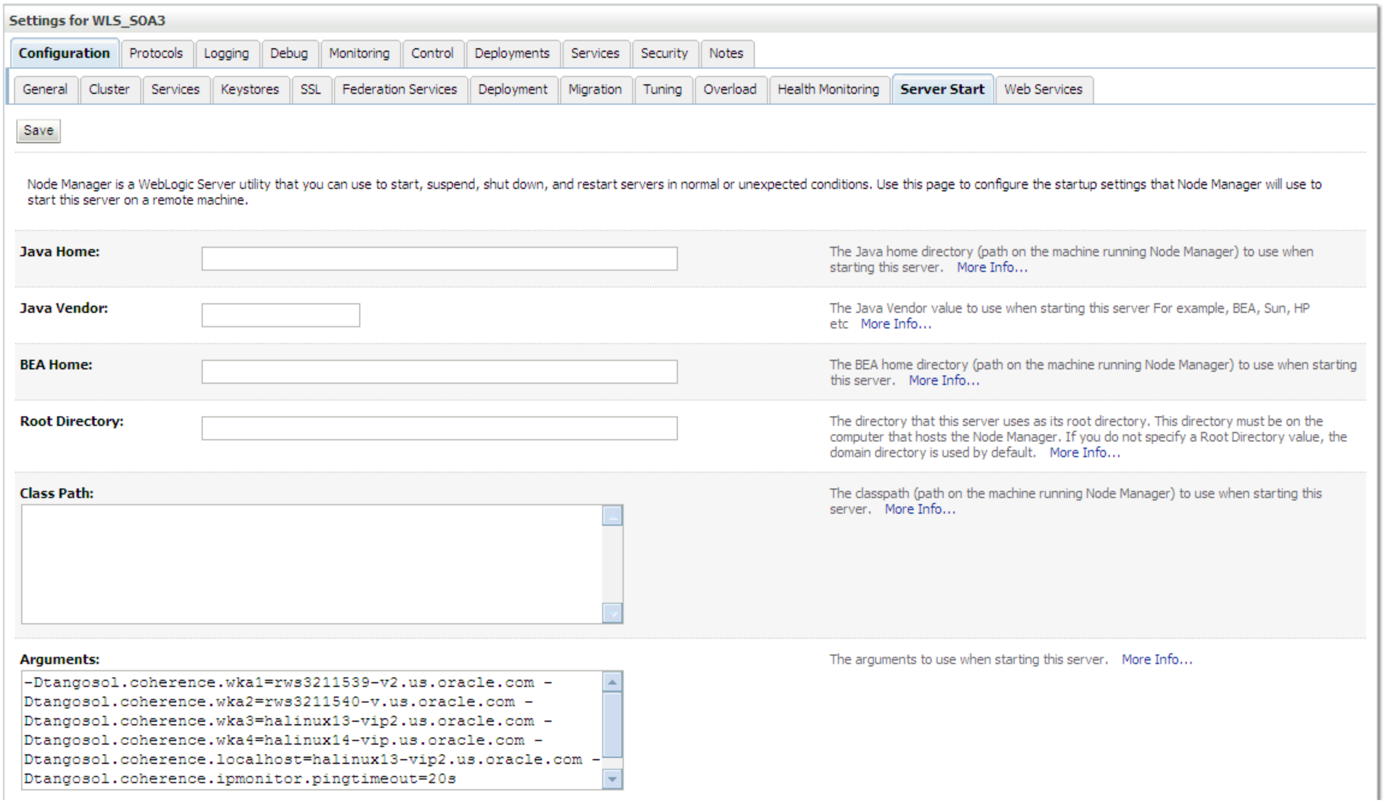
Finally, if the error persists, the new member trying to join will report a failure and the Coherence service will be stopped, and after some time reports the following:

```
<TIME > <Error> <Coherence> <BEA-000000> <TIME Oracle Coherence GE 3.7.1.6 <Error>
(thread=Cluster, member=n/a): Failure to join a cluster for 1000 seconds; stopping
cluster service.>
<TIME PDT> <Error> <Coherence> <BEA-000000> <TIME Oracle Coherence GE 3.7.1.6
<Error> (thread=Cluster, member=n/a): Failure to join a cluster for 1000 seconds;
stopping cluster service.>
<TIME> <Error> <Coherence> <BEA-000000> <TIME Oracle Coherence GE 3.7.1.6 <Error>
(thread=[ACTIVE] ExecuteThread: '1' for queue: 'weblogic.kernel.Default (self-
tuning)', member=n/a): Error while starting cluster:
com.tangosol.net.RequestTimeoutException: Timeout during service start:
ServiceInfo(Id=0, Name=Cluster, Type=Cluster
  MemberSet=MasterMemberSet(
    ThisMember=null
    OldestMember=null
    ActualMemberSet=MemberSet(Size=0
      )
    MemberId|ServiceVersion|ServiceJoined|MemberState
    RecycleMillis=2600000
    RecycleSet=MemberSet(Size=0
      )
    )
)
```

In these situations it is recommended that you increase the join-timeout and IP-timeout periods for the Coherence cluster. Despite being originally designed for multicast configurations, join-timeout can also be set in unicast cases. It is recommended that you upgrade the Coherence infrastructure used by SOA to version 3.7.1.6 (patch number 14468425) for optimized behavior of the IP-timeout parameter (the version included with Oracle Fusion Middleware SOA PS5 is 3.7.1.1.0). To upgrade the Coherence libraries used by Oracle Fusion Middleware SOA, download the provided patch and make a copy of your existing coherence.jar files in the following locations:

MW_HOME/coherence_3.7.orig/lib/coherence.jar
MW_HOME /oracle_common/modules/oracle.coherence/coherence.jar
MW_HOME /coherence_3.7/lib/coherence.jar
MW_HOME /wlserver_10.3/server/lib/console-ext/autodeploy/coherence.jar

Replace the libraries with the new coherence.jar. Also, make a copy of the MW_HOME/coherence_3.7 directory and replace it with the new one that is included in the patch. To increase IP-timeout (default 5 seconds) both a tangosol-coherence-override.xml file and a command line Java option can be used. The tangosol.coherence.ipmonitor.pingtimeout system property is used to specify a timeout instead of using the operational override file. For example, add -Dtangosol.coherence.ipmonitor.pingtimeout=20s in the SOA server start options to increase to 20s the tcpring listener timeout.



Node Manager is a WebLogic Server utility that you can use to start, suspend, shut down, and restart servers in normal or unexpected conditions. Use this page to configure the startup settings that Node Manager will use to start this server on a remote machine.

Java Home:		The Java home directory (path on the machine running Node Manager) to use when starting this server. More Info...
Java Vendor:		The Java Vendor value to use when starting this server For example, BEA, Sun, HP etc More Info...
BEA Home:		The BEA home directory (path on the machine running Node Manager) to use when starting this server. More Info...
Root Directory:		The directory that this server uses as its root directory. This directory must be on the computer that hosts the Node Manager. If you do not specify a Root Directory value, the domain directory is used by default. More Info...
Class Path:		The classpath (path on the machine running Node Manager) to use when starting this server. More Info...

Arguments:	-Dtangosol.coherence.wka1=rws3211539-v2.us.oracle.com -
	Dtangosol.coherence.wka2=rws3211540-v.us.oracle.com -
	Dtangosol.coherence.wka3=halinux13-vip2.us.oracle.com -
	Dtangosol.coherence.wka4=halinux14-vip.us.oracle.com -
	Dtangosol.coherence.localhost=halinux13-vip2.us.oracle.com -
	Dtangosol.coherence.ipmonitor.pingtimeout=20s
		The arguments to use when starting this server. More Info...

Using a Coherence override also allows you to change the number of attempts before determining that a computer that is hosting cluster members has become unreachable. The values of the ip-timeout and ip-attempts elements should be permissive enough for the network latency between the two sites. The same applies to the join-timeout. To use a Coherence override, it is necessary to update the tangosol-coherence-override.xml file included in the fabric-runtime.jar file in

ORACLE_HOME/soa/modules/oracle.soa.fabric_11.1.1/fabric-runtime.jar. Make a copy of the original JAR, extract its contents, and update the tangosol-coherence-override.xml file as shown here:

```
<!DOCTYPE coherence PUBLIC "-//Tangosol, Inc.//DTD Tangosol Coherence 3.0//EN"
"http://www.tangosol.com/dtd/coherence_3_0.dtd">
<coherence>
  <configurable-cache-factory-config>
    <class-name>com.tangosol.net.DefaultConfigurableCacheFactory</class-name>
    <init-params>
      <init-param>
        <param-type>java.lang.String</param-type>
        <param-value>soa-coherence-cache-config.xml</param-value>
      </init-param>
    </init-params>
  </configurable-cache-factory-config>
  <cluster-config>
  <multicast-listener>
         <join-timeout-milliseconds>600000</join-timeout-milliseconds>
  </multicast-listener>
  <tcp-ring-listener>
         <ip-timeout system-
property="tangosol.coherence.ipmonitor.pingtimeout">60s</ip-timeout>
         <ip-attempts>3</ip-attempts>
         <listen-backlog>10</listen-backlog>
   </tcp-ring-listener>
      <packet-delivery>
            <packet-publisher>
                  <timeout-milliseconds>650000</timeout-milliseconds>
            </packet-publisher>
      </packet-delivery>
    <unicast-listener>
      <well-known-addresses>
…
[rest of the contents remain the same
```

Repack the file in the fabric-runtime.jar archive and replace the older version located at ORACLE_HOME/soa/modules/oracle.soa.fabric_11.1.1/fabric-runtime.jar. Restart the SOA servers for the change to be effective. If needed, tcp-ring death detection can be disabled. Disabling death detection can alleviate network traffic but also makes the detection of failed members take longer. If disabled, a cluster member uses the packet publisher's resend timeout interval to determine that another member has stopped responding to UDP packets. By default, the timeout interval is set to 5 minutes. Refer to the coherence documentation for changing the packet resend timeout. To disable death detection, tangosol-coherence-override.xml file and add an `enabled` element that is set to false. For example:

```
<?xml version='1.0'?>

<coherence xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns="http://xmlns.mycompany.com/coherence/coherence-operational-config"
   xsi:schemaLocation="http://xmlns.mycompany.com/coherence/
   coherence-operational-config coherence-operational-config.xsd">
   <cluster-config>
      <tcp-ring-listener>
         <enabled>false</enabled>
      </tcp-ring-listener>
   </cluster-config>
</coherence>
```

Composite deployments (first version or updates to newer versions) are not activated in the SOA servers until they are available in all members of the cluster (servers must be in the RUNNING state and with the soa-infra application in the Activestate). The composite version will be listed by the soa-infra application and reported as Loaded in the SOA server's output file, but invoking this version while it is being deployed to other servers generates a SOA fault (that is, activation is precluded until all servers complete the deployment). In a Multi Data Center Deployment this is especially relevant because remote access to the MDS schemas may cause long delays in activating composites in those servers with higher latency to the database. Planned dowtimes or activation windows should be planned based on these “slowest” members.

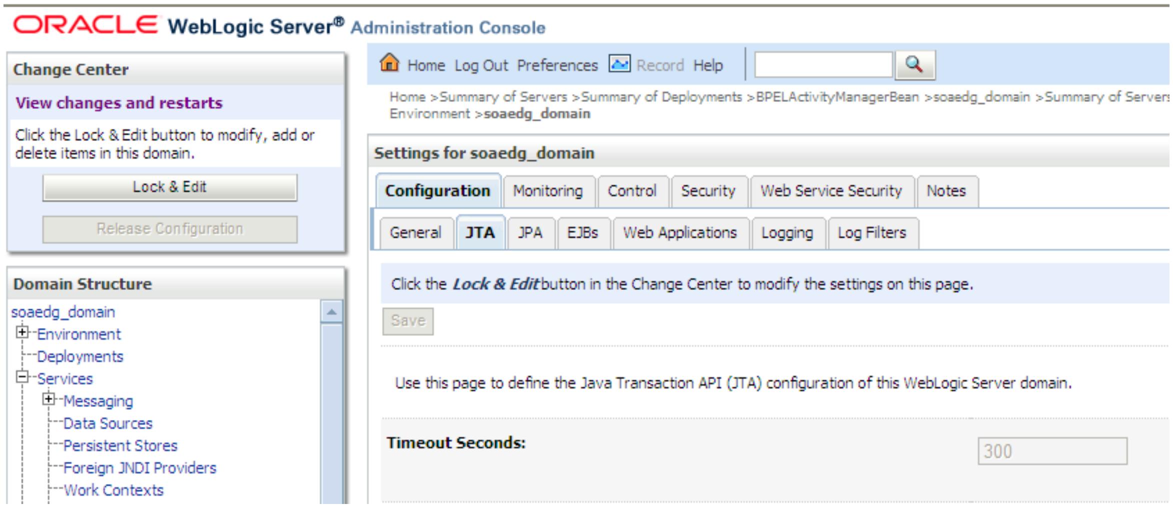
Setting Appropriate Timeouts for Synchronous and Asynchronous Operations

Timeouts may occur in different layers of the Oracle Fusion Middleware stack in a running Oracle Fusion Middleware SOA system. There are timeout periods specified for transactions in the database, for transactions branches, EJB method invocations, web services, etc. Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployments are especially sensitive to timeout settings because multiple operations performed by some servers need to access the database in a different location. Timeouts may need to be increased in these types of systems due to the latencies involved. In the Stretched Cluster model, domain settings are shared for both sites; hence it is required to use timeouts that account for the worst case scenario. Additionally it is required that timeouts are configured in the different layers of the SOA system so that the appropriate embracing tiers behave properly (for example, if the database timeout is set to a value lower than the Global WLS timeout, it may occur that a transaction ID is "removed" from the database before the work on other branches completes). In summary timeouts need to be configured such that:

- They account for the latency in the system
- They expire properly in the chain of invocations across different layers

There are different parameters that may be configured for expiring requests in different layers. These are the main ones:

- Timeout in the application server invocations (also known as global transaction timeout). This is the Oracle WebLogic Server's global transaction timeout in seconds for active transactions. It is configured through the Oracle WebLogic Server Administration Console. In the Oracle WebLogic Administration Console select Domain in the Navigation tree on the left, then Services > JTA > Timeout Seconds.



Home >Summary of Servers >Summary of Deployments >BPELActivityManagerBean >soaedg_domain >Summary of Server
Environment >**soaedg_domain**

Settings for soaedg_domain

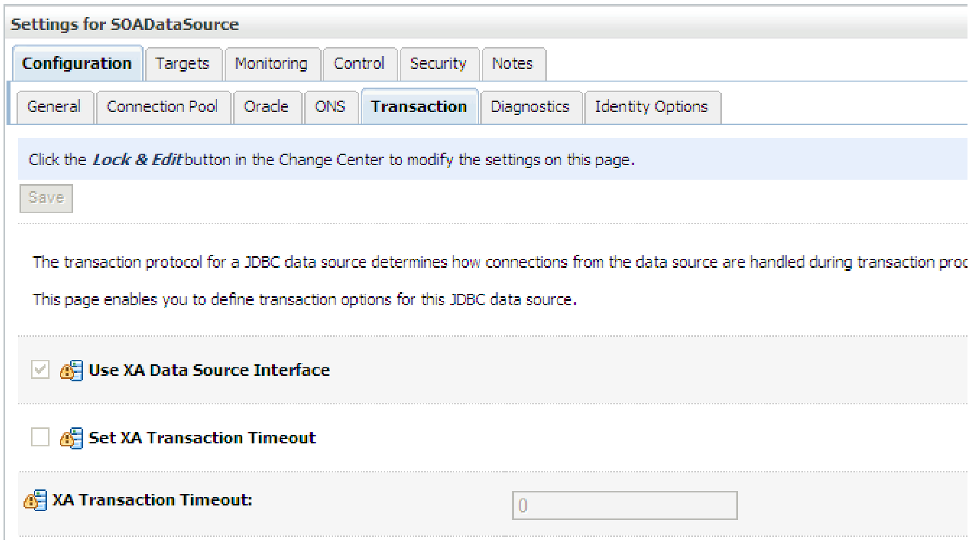
Click the Lock & Edit button to modify, add or delete items in this domain.

Click the ***Lock & Edit*** button in the Change Center to modify the settings on this page.

Use this page to define the Java Transaction API (JTA) configuration of this WebLogic Server domain.

Timeout Seconds: 300

- Timeouts in XA Data Sources: This is used in Oracle WebLogic Server to set a transaction branch timeout for Data Sources (You may want to set it if you have long-running transactions that exceed the default timeout value on the XA resource). It is configured in the Transaction tab for each XA Data Source.



Settings for SOADataSource

Click the *Lock & Edit* button in the Change Center to modify the settings on this page.

The transaction protocol for a JDBC data source determines how connections from the data source are handled during transaction proc

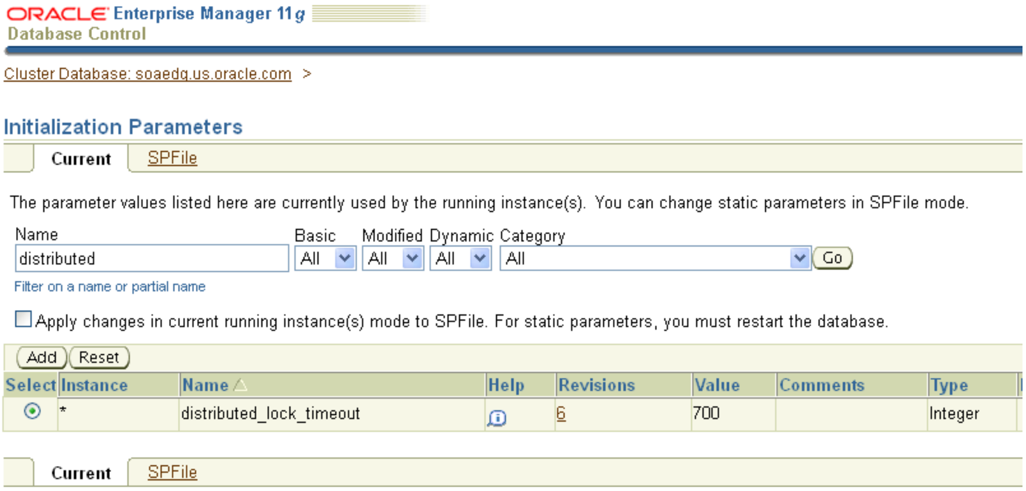
This page enables you to define transaction options for this JDBC data source.

☑ Use XA Data Source Interface

□ Set XA Transaction Timeout

XA Transaction Timeout: 0

- Timeouts for distributed locking in the database (distributed_lock_timeout): this specifies the amount of time (in seconds) for distributed transactions to wait for locked resources. It can be modified with the appropriate ALTER statements or using Oracle Enterprise Manager Database Control as shown:


ORACLE Enterprise Manager 11g
Database Control
Cluster Database: soaedg.us.oracle.com >
Current
SPFile
Go
Add
Reset

Initialization Parameters

The parameter values listed here are currently used by the running instance(s). You can change static parameters in SPFile mode.

Name: distributed Basic: All Modified: All Dynamic: All Category: All

Filter on a name or partial name

□ Apply changes in current running instance(s) mode to SPFile. For static parameters, you must restart the database.

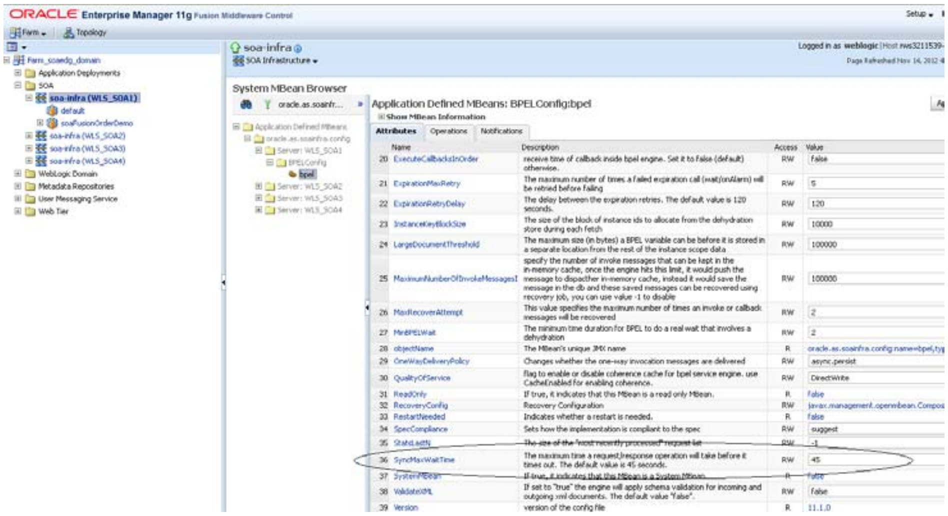
Select	Instance	Name △	Help	Revisions	Value	Comments	Type
○	*	distributed_lock_timeout	ⓘ	6	700		Integer

In an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment, you should set the distributed locking timeout for the database so that it accounts for the longest running database transactional operation possible (accounting for the delays introduced by the latency across sites.) Once this value is set, configure the XA Data Sources and Global Transaction Timeouts to lower values:

distributed_lock_timeout >= XA DS Timeout >= Global Transaction Timeout

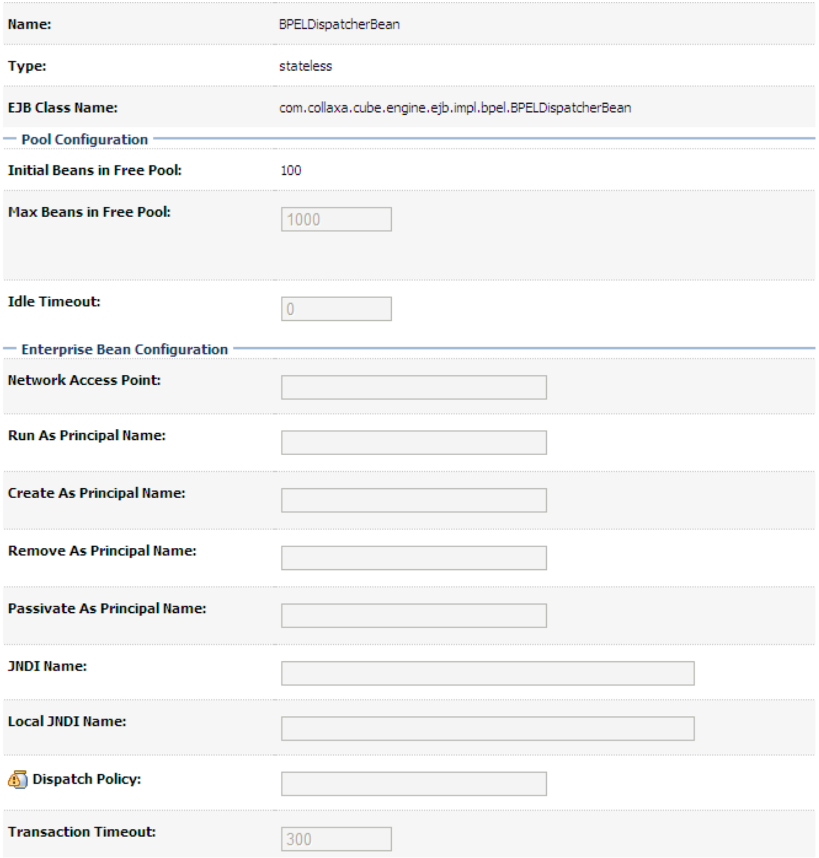
Additionally, and for BPEL synchronous processes, there are other parameters that may control timeouts:

- Time that a synchronous client should wait for an answer (syncMaxWaitTime). This property defines the maximum time a request and response operation takes before timing out. If the BPEL process service component does not receive a reply within the specified time, then the activity fails. To modify it using Oracle Enterprise Manager Fusion Middleware Control right click soa-infra then select SOA Administration > BPEL Properties > More BPEL Configuration Properties > syncMaxWaitTime.



	Name	Description	Access	Value
20	ExecuteCallbacksInOrder	receive time of callback inside bpel engine. Set it to false (default) otherwise.	RW	false
21	ExpirationMaxRetry	The maximum number of times a failed expiration call (wait/onAlarm) will be retried before failing	RW	5
22	ExpirationRetryDelay	The delay between the expiration retries. The default value is 120 seconds.	RW	120
23	InstanceKeyBlockSize	The size of the block of instance ids to allocate from the dehydration store during each fetch	RW	10000
24	LargeDocumentThreshold	The maximum size (in bytes) a BPEL variable can be before it is stored in a separate location from the rest of the instance scope data	RW	100000
25	MaximumNumberOfInvokeMessagesI	specify the number of invoke messages that can be kept in the in-memory cache, once the engine hits this limit, it would push the message to dispacther in-memory cache, instead it would save the message in the db and these saved messages can be recovered using recovery job, you can use value -1 to disable	RW	100000
26	MaxRecoverAttempt	This value specifies the maximum number of times an invoke or callback messages will be recovered	RW	2
27	MinBPELWait	The minimum time duration for BPEL to do a real wait that involves a dehydration	RW	2
28	objectName	The MBean's unique JMX name	R	oracle.as.soainfra.config:name=bpel,typ
29	OneWayDeliveryPolicy	Changes whether the one-way invocation messages are delivered	RW	async.persist
30	QualityOfService	flag to enable or disable coherence cache for bpel service engine. use CacheEnabled for enabling coherence.	RW	DirectWrite
31	ReadOnly	If true, it indicates that this MBean is a read only MBean.	R	false
32	RecoveryConfig	Recovery Configuration	RW	javax.management.openmbean.Compos
33	RestartNeeded	Indicates whether a restart is needed.	R	false
34	SpecCompliance	Sets how the implementation is compliant to the spec	RW	suggest
35	StatsLastN	The size of the "most recently processed" request list	RW	-1
36	SyncMaxWaitTime	The maximum time a request/response operation will take before it times out. The default value is 45 seconds.	RW	45
37	SystemMBean	If true, it indicates that this MBean is a System MBean.	R	false
38	ValidateXML	If set to "true" the engine will apply schema validation for incoming and outgoing xml documents. The default value "false".	RW	false
39	Version	version of the config file	R	11.1.0

- Timeouts for EJBs (applicable especially to BPEL): when the BPEL EJBs methods are involved, this specifies the transaction timeout period in seconds (default is 300 for all BPEL EJBs). It can be modified using the Oracle WebLogic Server Administration Console by selecting **Deployment** then expand **soa_infra** application, and then click in the specific BPEL EJB.



Name:	BPELDispatcherBean
Type:	stateless
EJB Class Name:	com.collaxa.cube.engine.ejb.impl.bpel.BPELDispatcherBean
Pool Configuration	
Initial Beans in Free Pool:	100
Max Beans in Free Pool:	1000
Idle Timeout:	0
Enterprise Bean Configuration	
Network Access Point:	
Run As Principal Name:	
Create As Principal Name:	
Remove As Principal Name:	
Passivate As Principal Name:	
JNDI Name:	
Local JNDI Name:	
Dispatch Policy:	
Transaction Timeout:	300

To elude exceptions and SOA faults in an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment where transactions are dropped before the processes are completed (typically reported as "thread is NOT associated with a transaction") the recommendation is to set

syncMaxWaitTime < BPEL EJB's transaction timeout < Global Transaction Timeout

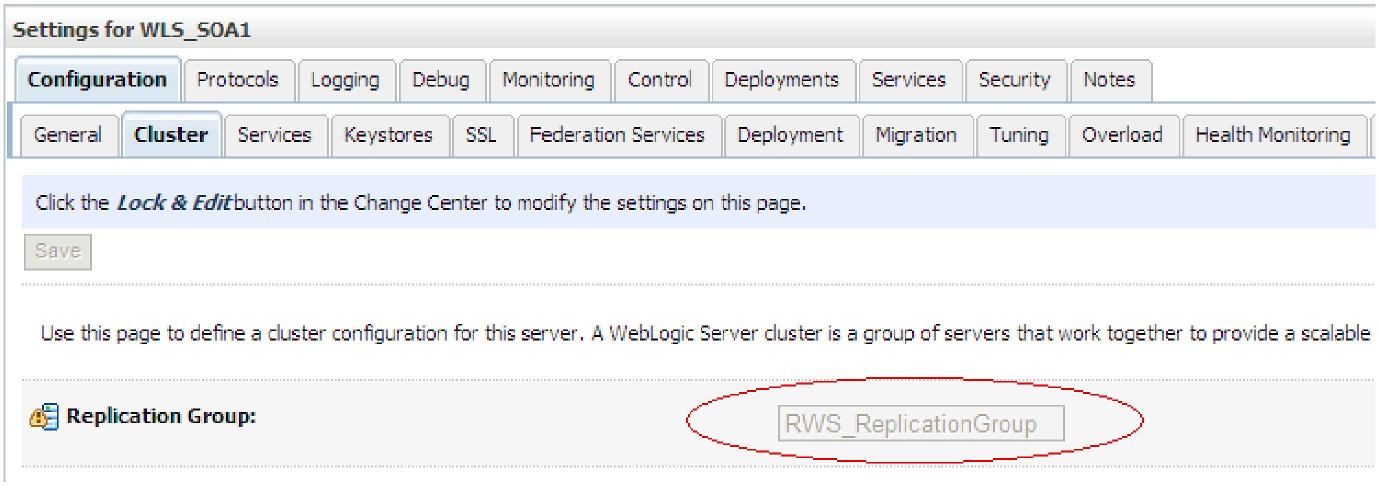
Also, any web services clients that the system exposes must be configured with a timeout that is permissive enough with worst latency (that is, if the invocation takes 3 seconds though Site1, but 10 seconds when it goes though Site2, the latter will set the general limit for invocations).

Finally, if the BPEL system is using specific Request-reply (synchronous) and In-only receive (asynchronous) timeouts, they must be defined based on the worst case scenarios for a SOA invocation in context of the Multi Data Center Deployment topology. This would be the case where the invocation is routed to the site with the highest latency in accessing the SOA database. These settings are part of the BPEL process definition, and since there is a unique MDS/composite repository, they cannot be customized for each site. Refer to the *Oracle*

Fusion Middleware Developer's Guide for Oracle SOA Suite for more details on using events and timeouts in BPEL processes.

Session Replication Implications

There are some applications (especially administration consoles like SOA's composer, Oracle Business Process Management BPM composer, BPM workspace, and the Oracle B2B console) that make intensive use of HTTP session objects. One of the advantages of an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment is the ability to seamlessly failover users from one site to another when they are accessing these consoles. However, session replication across data centers may cause serious performance degradation in the system. Oracle recommends defining two different replication groups (one for each site) to minimize the possibility of replication happening across the two sites. To configure session replication groups, use the Oracle WebLogic Administration Console to access Environment > Servers > Server Name > Cluster SubTab as shown:


Settings for WLS_SOA1

Click the ***Lock & Edit*** button in the Change Center to modify the settings on this page.

Use this page to define a cluster configuration for this server. A WebLogic Server cluster is a group of servers that work together to provide a scalable

Replication Group: RWS_ReplicationGroup

For each server in Site1 enter the same replication group name (for example Site1RG). Repeat the operation for servers in Site2 using a common name for all of them but different from the one used in Site1 (for example Site2RG).

NOTE: Using replication groups is a best effort to replicate state only to servers in the same site, but is not a deterministic method. If one single server is available in one site, and others are available in the other, replication will occur across the MAN and will continue for that session even if servers come back online in the same site.

Optimizing Oracle Net Services Performance

Operating System and Oracle Net tuning play a critical role in data transmission across Metropolitan Area Networks (MAN). However, the effect of these adjustments is more relevant for higher latencies between JDBC clients and servers. In an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment the servers with the highest latency in accessing the database become the drivers for network buffers and Oracle Net adjustments. Some operating system default configurations are not optimized for current Ethernet speeds and it becomes very important to adjust some parameters to make data transfer more efficient. One of the key metrics in determining the optimum configuration is the Bandwidth-delay product. This is the product of network bandwidth and the round trip time/delay of data going over the network. A simple way to determine the round trip time (RTT) is to use a command such as *ping* from one host to another and use the response times returned. Ideally this parameter should be obtained as an average over a few minutes to eliminate momentary deviations. In Linux, this average can be obtained using the command ping directly.

```
[orcl@host1_Site1~]$ ping host1_Site2

64 bytes from host1_Site2 icmp_seq=0 ttl=61 time=7.64 ms
64 bytes from host1_Site2 icmp_seq=1 ttl=61 time=8.43 ms
64 bytes from host1_Site2 icmp_seq=2 ttl=61 time=7.62 ms
64 bytes from host1_Site2 icmp_seq=3 ttl=61 time=7.76 ms
64 bytes from host1_Site2: icmp_seq=4 ttl=61 time=6.71 ms

--- host1_Site2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4014ms
rtt min/avg/max/mdev = 6.715/7.634/8.430/0.552 ms, pipe 2
```

TCP socket buffer settings control how many packets are sent at one time over the network. In some operating systems default settings need to be increased in order to improve utilization of available bandwidth. When network latency is high, larger socket buffer sizes are better to fully utilize network bandwidth. The optimal socket buffer size is two times the size of the Bandwidth Delay Product (BDP) (which would typically be RTT*BW since RTT is 2*delay) and should be set both for the database server and the Oracle Fusion Middleware nodes. Larger buffers than 2xBDP are beneficial in most cases (overall with higher latencies that can present larger deviations in RTT) but also consume more memory. Resource utilization should be monitored, and if memory is available increasing the buffer improves the data transmission behavior. The size of the socket buffers should be adjusted both in Oracle Fusion Middleware nodes and database servers.

Configuring I/O Buffer Size in the Database Server

Since the database server primarily writes data to clients, setting the SEND_BUF_SIZE parameter on the server-side is typically enough. If the database server is receiving large requests, then also set the RECV_BUF_SIZE parameter. It is recommended to set them at the Oracle Net level in the database so that normal TCP sessions such as telnet, ssh, etc. do not use additional memory unless these other protocols are in similar need for optimization. To configure the database server, set the buffer space size in the listener.ora and sqlnet.ora files. In the listener.ora file, specify the buffer space parameters for a particular protocol address or for a description. The following is an example of the settings for a typical Oracle RAC-scan listener configuration:

```
LISTENER_SCAN3 =

  (DESCRIPTION =

    (ADDRESS = (PROTOCOL = IPC)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(KEY = LISTENER_SCAN3))
)

LISTENER_SCAN2 =

  (DESCRIPTION =

    (ADDRESS = (PROTOCOL = IPC)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(KEY = LISTENER_SCAN2))

  )

LISTENER_SCAN1 =

  (DESCRIPTION =

    (ADDRESS = (PROTOCOL = IPC)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(KEY =
LISTENER_SCAN1))
```

Configuring I/O Buffer Size on the Oracle Fusion Middleware Nodes

Unfortunately, as of Oracle 11*g*R2 Oracle JDBC thin clients cannot specify socket buffer sizes, hence buffers need to be adjusted in the operating system. Recent versions of Linux (version 2.6.17 and later) use auto tuning with a 4 MB maximum buffer sizes. Enabling or disabling of auto tuning is determined by the /proc/sys/net/ipv4/tcp_moderate_rcvbuf parameter. If the parameter tcp_moderate_rcvbuf is present and has a value of 1 then auto tuning is enabled. With auto tuning, the receiver buffer size (and TCP window size) is dynamically updated (auto-tuned) for each connection. Sender-side auto tuning has been present and unconditionally enabled for many years in Linux kernels. The per connection memory space defaults are set with two 3 element files:

/proc/sys/net/ipv4/tcp_rmem - memory reserved for TCP receive buffers

/proc/sys/net/ipv4/tcp_wmem - memory reserved for TCP send buffers

These files contain three values: minimum, initial, and maximum buffer size. They are used to set the bounds on auto tuning and balance memory usage while under memory stress. The maximum values must be larger than the maximum value of 2xBDP. With auto tuning, an excessively large initial buffer wastes memory and can hurt performance. Additionally, the maximum buffer size that applications can request can also be limited with /proc variables:

/proc/sys/net/core/rmem_max - maximum receive window

/proc/sys/net/core/wmem_max - maximum send window

The following are sample recommendations for TCP optimization in Linux:

- Use TCP auto-tuning in kernel (2.4.27, 2.6.7)
  ```
  /proc/sys/net/ipv4/tcp_moderate_rcvbuf (1=on)
  ```
- Tune TCP Max Memory:
  ```
  /proc/sys/net/ipv4/tcp_rmem and tcp_wmem
  – 4096 87380 174760
  ```
  Set the maximum (last value in the example=174760) to a value larger than 2xBDP.
- Tune the socket window sizes.
  ```
  /proc/sys/net/core/rmem_max and wmem_max
  ```
  Set this to larger than 2xBDP.
- Ensure that TCP Performance features are enabled (set all of the following to 1).

```
/proc/sys/net/ipv4/tcp_sac
/proc/sys/net/ipv4/tcp_window_scaling
/proc/sys/net/ipv4/tcp_timestamps
```

Configuring Session Data Unit

Oracle Net sends data in packages with a specific size. Oracle Net waits for these units to be “filled” before sending them across the network. Each of these buffers is called a session data unit (SDU). Adjusting the size of the SDU to the amount of data provided to Oracle Net can improve performance, network utilization, and memory consumption in an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment with higher latencies than 5 msecs RTT. The SDU size can be set from 512 bytes to 65535 bytes (in Oracle Database 11*g* R2). The default SDU for the client and a dedicated server is 8192 bytes. The default SDU for a shared server is 65535 bytes. The actual SDU size used is negotiated between the client and the server at connect time and is the smaller of the client and server values. Configuring an SDU size different from the default requires configuring the SDU on both the client and server computers, unless you are using shared servers. For shared servers, only the client value must be changed because the shared server defaults to the maximum value. Oracle recommends setting the SDU to the maximum value possible (64k) as this has been verified to provide the best results in systems with different latencies. To set the SDU on the Oracle Fusion Middleware servers, update the JDBC URL connection string in each of the connection pools used by SOA Data Sources. For example:

```
jdbc:oracle:thin:@
(DESCRIPTION_LIST =
        (LOAD_BALANCE = off)
        (FAILOVER = on)
        (DESCRIPTION =
                (CONNECT_TIMEOUT = 10)
                (TRANSPORT_CONNECT_TIMEOUT = 3)
                (RETRY_COUNT = 3)
                (ADDRESS_LIST =
                        (LOAD_BALANCE = on)
                        (ADDRESS =
                                (PROTOCOL = TCP)
                                (HOST = scanSite1.mycompany.com)
                                (PORT = 1521)
                                (SDU=65535)
                        )
                )
                (CONNECT_DATA =
                        (SERVICE_NAME =soaedg.mycompany.com)
                )
        )
        (DESCRIPTION =
                (CONNECT_TIMEOUT =10)
                (TRANSPORT_CONNECT_TIMEOUT = 3)
                (RETRY_COUNT = 3)
                (ADDRESS_LIST =
                        (LOAD_BALANCE = on)
                        (ADDRESS =
                                (PROTOCOL =TCP)
                                (HOST = scanSite2.mycompany.com)
                                (PORT = 1521)
                                (SDU=65535)
                        )
                )
                (CONNECT_DATA =
                        (SERVICE_NAME = soaedg.mycompany.com))
                )
```

To set the SDU on the database servers, you can modify the listener configuration file (`LISTENER.ORA`) (this will set the SDU on a "per connection basis", or you can set the SDU for all Oracle Net connections with the profile parameter DEFAULT_SDU_SIZE in the sqlnet.ora file. The following example sets the SDU to its recommended value for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment in each of the SCAN listeners for the Oracle RAC Database:

```
LISTENER_SCAN3 =

  (DESCRIPTION =

    (ADDRESS = (PROTOCOL = IPC)(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(KEY =
LISTENER_SCAN3))  )

LISTENER_SCAN2 =

  (DESCRIPTION =

    (ADDRESS = (PROTOCOL = IPC)(SDU=65535)(SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(KEY =
LISTENER_SCAN2))

  )

LISTENER_SCAN1 =

  (DESCRIPTION =

    (ADDRESS = (PROTOCOL = IPC)(SDU=65535) (SEND_BUF_SIZE=10485760)(RECV_BUF_SIZE=10485760)(KEY =

LISTENER_SCAN1))
```

The benefit of larger SDU values should be noticeable in scenarios where large payloads are transferred to the database. In all cases, the SDU size should be less than or equal than the socket buffer to avoid fragmentation.

Failures in Different Tiers and Switchover/Failover Behavior

The Oracle Fusion Middleware SOA Multi Data Center Active-Active topology is resilient to failures in any component. Since each site uses the Oracle Fusion Middleware SOA EDG Topology, failures at component level (LBR, OHS instance, Oracle WebLogic Server, database instance) should not cause any disruption because each site provides local redundancy for each component. When an entire tier fails in a site, different aspects need to be considered as discussed in the following sections.

Failure in All OHS Instances in One Site

OHS instances typically reside in isolated hardware (for security and administration purposes). It may occur that all instances of OHS in one site become unavailable due to a failure in the hardware hosting them. If a site loses all instances, the Oracle WebLogic Server SOA servers may continue processing SOA instances as long as no HTTP callbacks occur. If internally-generated callbacks occur, because the LBR is configured with rules to route to the site originating the callback, they will fail. However, the server may continue processing JMS and locally optimized invocations. If restoration of the lost OHS instances is not possible in the short term, the instances in the other site may be used to route requests to the SOA servers on the "OHS-orphan" site. In the stretched cluster model, this would require setting the DyanmicServerList to ON. This change can be applied in a rolling manner to eliminate downtime. Additionally, and because rules have been defined to perform smart routing to each site based on the request's origin, make sure that the appropriate monitors are set in the GLBR to stop routing to OHS when this type of failure occurs.

Failure in All Oracle WebLogic Server SOA Servers in One Site

When all the Oracle WebLogic Server SOA servers are down in one site, the other site continues processing requests. The local LBRs should stop routing to the corresponding OHS servers (GLBR should stop routing to that site). From the BPEL perspective if the automatic recovery cluster master was hosted in the failed servers, a new cluster master will arise in the available site. This server can perform automated recovery of instances initiated on the other site. Because server migration across sites should be avoided, pending transactions will not be recovered or resumed until the appropriate servers are restarted. Alternatively, and if resource consistency is guaranteed across sites, transactions can be restarted in the available site by making TLOGs and persistent stores available to the appropriate server (refer to the transaction logs paragraphs in the configuration sections above).

Administration Server Failure

The standard consideration for Administration Server failure scenarios that apply in a single data center topology apply to Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment. Node failures should be addressed with the standard failover procedures described in *Oracle Fusion Middleware High Availability Guide* and *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* (restarting the Administration Server in another node that resides in the same data center pointing to the shared storage that hosted the Administration Server domain directory). Additionally, the appropriate backup and restore procedures should be deployed to make regular copies of the Administration Server's domain directory. In case of a failure that affects the site hosting the Administration Server (involving all nodes), it may be required to restart the server in a different site. To do this, follow these steps:

1. Make the backup (or disk mirror/copy) of your Administration Server's domain directory available in the failover site.

```
scp /u01/orcl/backups/ Multi Data Center
Deployment_soaedg_domain_STRETCHED.gz
server1_Site2:/u01/app/oracle/admin/soaedg_domain/aserver/
```

2. Restore the aserver/ directory (including both the soaedg_domain and applications directory) in the failover site so that the exact same domain directory structure is created for the Administration Server's domain directory as in the original site.

```
cd /u01/app/oracle/admin/soaedg_domain/
tar -xzvf /u11/app/oracle/admin/soaedg_domain/ Multi Data Center
Deployment_soaedg_domain_STRETCHED.gz
```

3. Restart Node Manager in the node where the Administration Server will be restored.

```
cd WL_HOME/server/bin
export JAVA_OPTIONS="-DDomainRegistrationEnabled=true"
./startNodeManager.sh
```

   NOTE: -DDomainRegistrationEnabled=true needs to be used whenever Node Manager is used to manage the Administration Server in a domain structure like the EDG's.

4. Likely, the Administration Server will be failed over to a different subnet, requiring the use of a different virtual IP (VIP) that is reachable by other nodes. Make the appropriate changes in the hostname resolution system in this subnet so that this VIP will map to the original Virtual Hostname that the Administration Server used as listen address in Site1 (for example, in Site1, ADMINHOSTVHN1 may map to 10.10.10.1 while in Site2 either local /etc/hosts or DNS server will have to be updated so that ADMINHOSTVHN1 maps to 20.20.20.1). All servers will use ADMINHOSTVHN1 as the address to reach the Administration Server). If the Administration Server is front ended with an OHS and LBR as prescribed in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*, clients will be agnostic to this change. If clients directly access the Administration Server's listen hostname, they must be updated in their DNS resolution also.

   Sample /etc/hosts in Site1:

```
10.10.10.1 ADMINHOSTVHN1.mycompany.com ADMINHOSTVHN1
```

Sample /etc/hosts in Site2:

```
20.20.20.1 ADMINHOSTVHN1.mycompany.com ADMINHOSTVHN1
```

Additionally, if host name verification is enabled for the Administration Server, the appropriate trust stores and key stores must be updated with new certificates. Use the instructions in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*.

5. Start WLST and connect to Node Manager with nmconnect and the credentials set for the Administration Server using nmstart. Enter the Node Manager user name and password used in the original site.

```
cd ORACLE_COMMON_HOME/common/bin
./wlst.sh
```

Once you are in the WLST shell:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
'SOAHOST1','5556','domain_name','/u01/oracle/admin/soaedg_domain/aserver/
soaedg_domain')

wls:/nm/domain_name nmStart('AdminServer')
```

Verify that the Administration Server is working properly by accessing both the Oracle WebLogic Server Administration Console and the Oracle Enterprise Manager Fusion Middleware Control.

Database Failures: Data Guard Switchover and Failover

The JDBC URL string and ONS configuration provided for the SOA Data Sources should guarantee that reconnection happens automatically when a failover or a switchover takes place at the database level. However, all SOA servers configured with database leasing for server migration will shut themselves down if the database takes more than the default fencing period for server migration (30 seconds). It is not recommended to increase the fencing period to adapt to the database because that would imply slower detection of regular failures (like process deaths, Oracle WebLogic Server nodes crashes, etc.) for intra-site server migration. Using larger values for the server migration fencing penalizes more common failure detections in benefit of better reactions to a database switchover (which should not be a frequent scenario). A database switchover for a SOA system will typically take more than a couple of minutes, using a fencing period beyond such a value would imply that server migration would take at least that time to be triggered which could cause overloading of the remaining servers and cascade fall of the system depending on the situation. During Data Guard switchover or failover of the SOA database, servers will be shutdown automatically by the Oracle WebLogic Server infrastructure and should be started when the database operation completes. Alternatively, the servers can be gracefully shut down before the switchover operation (does not apply to failover scenarios). Either of the two approaches can be used depending on the load on the system or the business needs.

Performance and Scalability Implications for an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment

Capacity Usage and Planning

A Multi Data Center Deployment design needs to account for the throughput penalty introduced by the latency between sites. The worst case scenario will be such that the middle tiers residing in the same site as the active database (Site1 for this example) become unavailable while this database keeps running. In this case, Site2 will have to sustain the load that the two sites were processing together, and it is expected that the average response times will worsen due to the latency because all requests must access a remote database. For such cases it may be recommended to switch over the database until the middle tiers in Site1 come back online. Site2 will have to have enough spare capacity during normal operation to sustain the added load when a failover from Site1 middle tiers occur. Alternatively, requests will have to be throttled in such a failover mode. The best approach is to throttle requests in the entry points to the system (GLBR or resource adapter access points, such like file system for file/FTP adapters or JMS clients). Requests can be throttled in the GLBR using the appropriate rules for routing. Similarly, clients producing JMS messages (for JMS adapter), database entries (for database adapter) or Files (to file or FTP adapters) can use the appropriate IP filtering to decrease the load in the system. Defining these rules is out of the scope of this document.

Start Latencies

A considerable amount of time is spent by servers in creating the connection pools' initial capacity for the different Data Sources used by SOA components. By default most SOA Data Sources use a zero initial capacity for its connection pool. However, to reduce the response time of the system during runtime, it may be advised to increase the initial capacity. For servers that reside in a "remote" site (remote to the SOA database) higher initial pool capacity will cause increased delays in starting the servers. A balanced decision needs to be made between the improved response time during normal operation and the time that it may take a system to recover in order to come up with the ideal initial capacity settings. In stretched clusters, because the initial capacity is set at the Data Source level (connection pool), the settings for initial capacity will affect the start period for all servers in a cluster. Image 10 shows the increased time that it takes to restart a server (compared to a server local to the database) as the latency between the server and the database becomes worse.

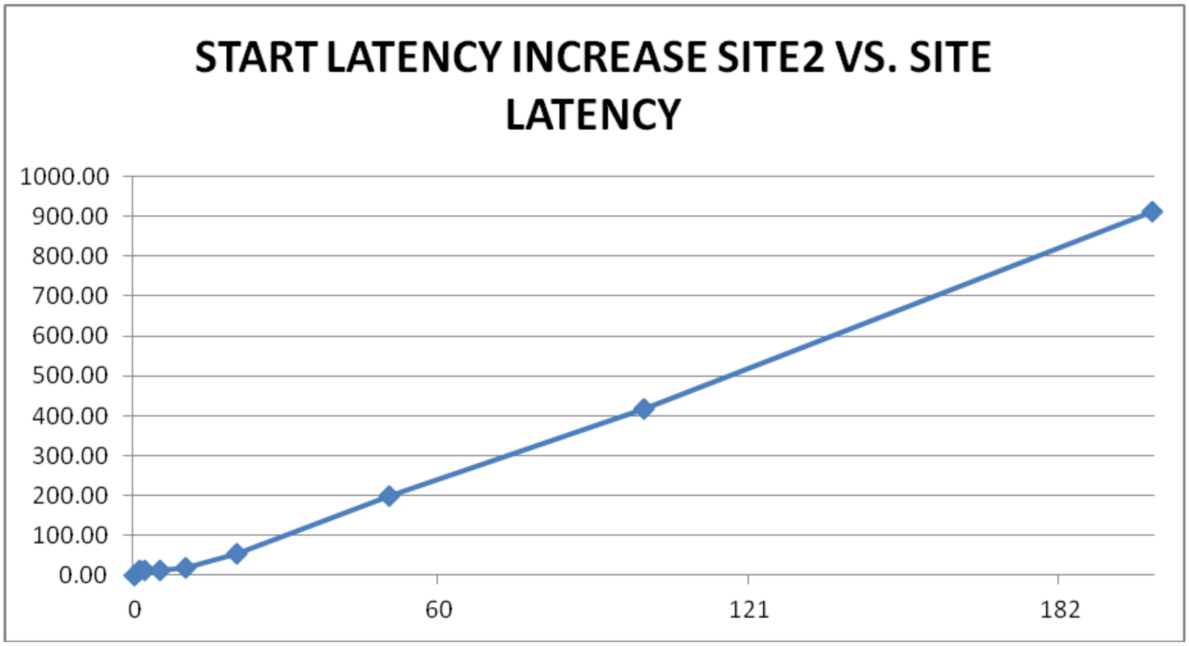


Image 10: Increase in restart time (seconds) as latency (RTT in msecs) grows for a SOA server using initial capacity=50 for Data Sources)

The process to restart a SOA server involves bringing up the appropriate Oracle WebLogic Server and also loading and activating the appropriate composites. Once the Oracle WebLogic Server hosting SOA reaches the RUNNING state, nothing prevents OHS from routing to the SOA applications whether composites are available or not. SOA, however, has been enabled to reject the requests for composites that are not loaded yet with a “503 Service unavailble” HTTP code. OHS should automatically retry the request on other available servers and process it successfully (if any other servers have completed the loading of the composite). The following excerpt from an OHS server routing to a SOA server that has not completed loading composites shows the expected behavior.

OHS routes to a SOA Server that is RUNNING but has not loaded the composites yet:

```
2012-09-30T10:12:00.5698-07:00 <253641351617120207> got a pooled connection to
server '‘server1_in_cluster’/8001/0' from general list for '/soa-
infra/services/soaFusionOrderDemo/OrderBookingComposite/orderprocessor_client_ep',
Local port:31906
…
2012-09-30T10:12:00.5709-07:00 <253641351617120207> Reader::fill(): sysRecv
returned 199
```

SOA returns a 503 error because the composite is not loaded yet:

```
2012-09-30T10:12:00.5709-07:00 <253641351617120207> URL::parseHeaders:
CompleteStatusLine set to [HTTP/1.1 503 Service Unavailable]
…
2012-09-30T10:12:00.5709-07:00 <253641351617120207> Exiting method
BaseProxy::sendRequest
2012-09-30T10:12:00.5709-07:00 <253641351617120207> sendResponse() : r->status =
'503'
```

OHS marks the server as unavailable until next proxy plugin update and retries on a different server:

```
2012-09-30T10:12:00.5709-07:00 <253641351617120207> Marking
‘server1_in_cluster’:8001 as unavailable for new requests
2012-09-30T10:12:00.5709-07:00 <253641351617120207> *******Exception type
[FAILOVER_REQUIRED] (Service Unavailable) raised at line 175 of
../common/BaseProxy.cpp
2012-09-30T10:12:00.5710-07:00 <253641351617120207> got exception in sendResponse
phase: FAILOVER_REQUIRED [line 175 of ../common/BaseProxy.cpp]: Service Unavailable
at line 554
2012-09-30T10:12:00.5710-07:00 <253641351617120207> Failing over after
FAILOVER_REQUIRED exception in sendResponse()
2012-09-30T10:12:00.5711-07:00 <253641351617120207> attempt #1 out of a max of 5
2012-09-30T10:12:00.5712-07:00 <253641351617120207> keepAlive = 1, canRecycle = 0
2012-09-30T10:12:00.5712-07:00 <253641351617120207> general list: trying connect to
‘server2_in_cluster’/8001/0 at line 2372 for '/soa-
infra/services/soaFusionOrderDemo/OrderBookingComposite/orderprocessor_client_ep'
2012-09-30T10:12:00.5720-07:00 <253641351617120207> URL::Connect: Connected
successfully
…
```

OHS fails over the request transparently to the client and marks the servers that are loading composites as “bad” (for 10 seconds, the default MaxSkipTime period in the OHS configuration) before retrying that bad server. Because composites are loaded from a remote database, the total amount of time that it takes a SOA server to be effectively available after a restart will be affected mainly by the Data Source pool’s initial capacity (for the server to be available to OHS) and the composite's sizes (for the SOA system to stop rejecting request with 503 codes).

Average Active Time for Transactions and Transaction Recovery

In an Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment, transactions remain active for longer periods of time in the site with higher latency to the database. Transactions that remain active for longer periods are more likely to be affected by failures. Image 11 shows the evolution of the average time that transactions remain active for the FOD example as the latency between Site2 and the database is increased.

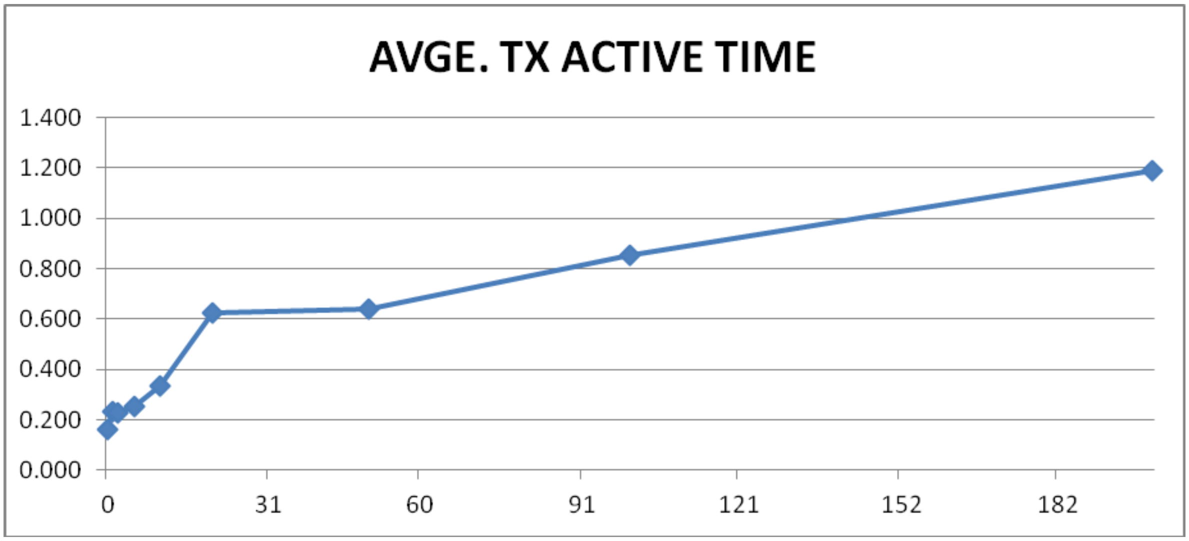


Image 11: Average time that transactions remain active in a Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment with FOD as the latency (RTT in msecs) across sites increases

It is important that the appropriate transaction logs protection mechanisms are provided when zero data loss is a business requirement. In these cases, using a JDBC persistent store for JMS and transactions logs is recommended for protection. For single server failures, server migration should take place and recovery will happen automatically. When an entire middle tier in one site is lost, and because server migration is prevented from one site to the other, the failed servers must be started manually on the site that remains available. In a Stretched Cluster system the read and resume of logs should be transparent.

Summary

Oracle Fusion Middleware SOA Suite can be used in multi-datacenter deployments where multiple Oracle Fusion Middleware SOA servers are actively processing requests using a central database located in one of those datacenters. This configuration uses a single Oracle WebLogic Server domain model where servers in all sites participate in the same cluster (also known as Stretched Cluster) and rely on Data Guard to provide protection for the SOA database. The network latency between sites needs to be sufficiently low to overcome the performance penalty introduced by the delay in invocations and to eliminate possible inconsistencies in deployment and runtime scenarios. Oracle recommends using these topologies in Metropolitan Area Networks with latencies between SOA servers and the database below 5 msecs (RTT). It is recommended that for each SOA multi data center the deployment, payload, composite types, and throughput requirements are analyzed, and parameters such as timeouts are fine tuned to ensure optimum fail over and performance of the system.

Appendix A: File Adapter Locks and Muxers

The following script can be used for creating file adapter mutex and lock tables:

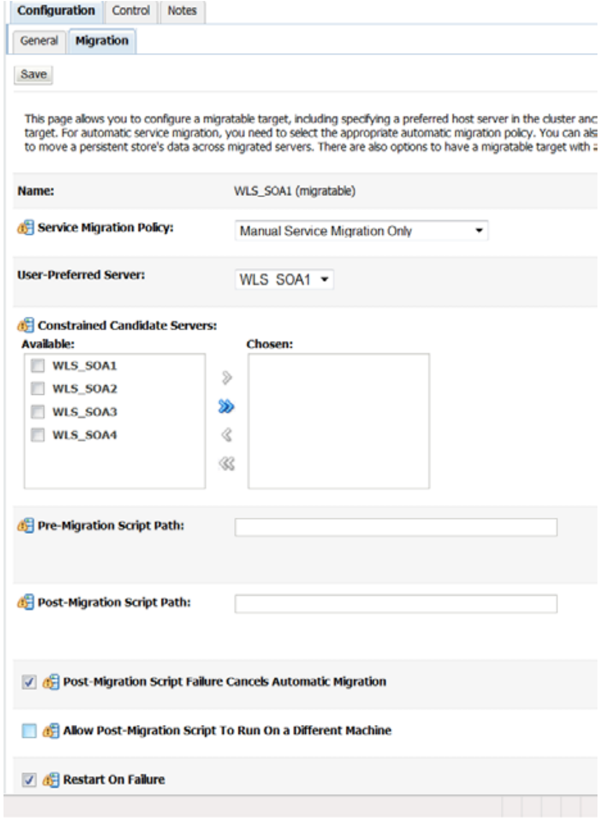
```
CREATE TABLE FILEADAPTER_IN
(
   FULL_PATH VARCHAR2(4000) NOT NULL,
   ROOT_DIRECTORY VARCHAR2(3000) NOT NULL,
   FILE_DIRECTORY VARCHAR2(3000) NOT NULL,
   FILE_NAME VARCHAR2(1000) NOT NULL,
   FILE_ENDPOINT_GUID VARCHAR2(2000) NOT NULL,
   FILE_LAST_MODIFIED NUMBER,
   FILE_READONLY CHAR(1),
   FILE_PROCESSED CHAR(1) DEFAULT '0',
   CREATED NUMBER NOT NULL,
   UPDATED NUMBER
);
ALTER TABLE FILEADAPTER_IN ADD CONSTRAINT FILEADAPTER_IN_PK PRIMARY KEY (FULL_PATH);
CREATE INDEX IDX_ROOT_DIRECTORY ON FILEADAPTER_IN  (ROOT_DIRECTORY );
CREATE INDEX IDX_FILE_DIRECTORY ON FILEADAPTER_IN (FILE_DIRECTORY );
CREATE INDEX IDX_FILE_PROCESSED ON FILEADAPTER_IN  (FILE_PROCESSED );
CREATE INDEX IDX_FILE_READONLY ON FILEADAPTER_IN (FILE_READONLY );
-- ---------------------------------------------------------------------
-- FILEADAPTER_MUTEX
-- ---------------------------------------------------------------------
CREATE TABLE FILEADAPTER_MUTEX
(
   MUTEX_ID VARCHAR2(4000) NOT NULL,
   MUTEX_CREATED TIMESTAMP,
   MUTEX_LAST_UPDATED TIMESTAMP,
   MUTEX_SEQUENCE NUMBER
)
;
ALTER TABLE FILEADAPTER_MUTEX ADD CONSTRAINT FILEADAPTER_MUTEX_PK PRIMARY KEY
(MUTEX_ID);
```

Appendix B: Configuring in-place restart for JMS JDBC persistent stores

To configure in-place restart for JMS JDBC persistent stores, the pertaining JMS server and persistent store must be targeted to a migratable target and this migratable target must use “Restart on Failure”[8]. To configure this, follow these steps:

Enable restart on failure for migratable targets:

1. Log in to the WLS Administration Console
2. Click Lock and Edit
3. On the navigation tress on the left, expand Environment and click on Migratable Targets
4. Click on WLS_SOA1 (migratable)
5. Click on the Migration Tab
6. Check the “Restart On Failure” box at the bottom



This page allows you to configure a migratable target, including specifying a preferred host server in the cluster anc target. For automatic service migration, you need to select the appropriate automatic migration policy. You can als to move a persistent store's data across migrated servers. There are also options to have a migratable target with a

Name: WLS_SOA1 (migratable)

Service Migration Policy: Manual Service Migration Only

User-Preferred Server: WLS SOA1

Constrained Candidate Servers:

Available:

□ WLS_SOA1
□ WLS_SOA2
□ WLS_SOA3
□ WLS_SOA4

Chosen:

Pre-Migration Script Path:

Post-Migration Script Path:

☑ Post-Migration Script Failure Cancels Automatic Migration

□ Allow Post-Migration Script To Run On a Different Machine

☑ Restart On Failure

Image 12: Migration configuration

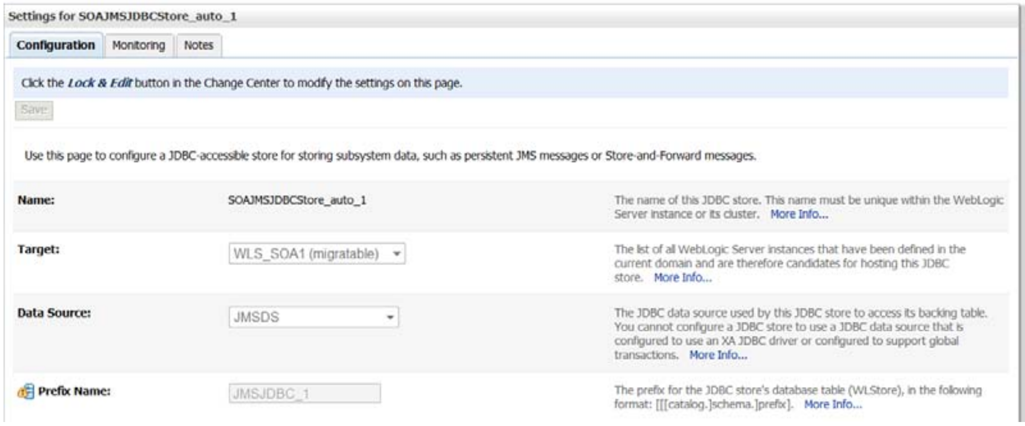
7. Click Save

[8] Restart on Failure with manual service migration requires a patch. Refer to bug number 17702917 for the pertaining patch for the different WLS versions

8. Repeat steps 4-7 for all the SOA JMS servers using the appropriate migratable targets (WLS_SOA1(migratable), WLS_SOA2(migratable) etc depending on the WLS server that the JMS servers is related to)
9. Activate the changes

Re-target the JMS Servers and Persistent Stores:

1. Log in to the WLS Administration Console
2. Click Lock and Edit
3. On the navigation tree on the left, expand Services->Messaging and click on JMS Servers
4. On the JMS Servers table, Click on SOAJMSServer_auto_1
5. Select the Targets Tab
6. Select "WLS_SOA1 (migratable)" as target



Settings for SOAJMSJDBCStore_auto_1

Click the *Lock & Edit* button in the Change Center to modify the settings on this page.

Use this page to configure a JDBC-accessible store for storing subsystem data, such as persistent JMS messages or Store-and-Forward messages.

Name:	SOAJMSJDBCStore_auto_1	The name of this JDBC store. This name must be unique within the WebLogic Server instance or its cluster. More Info...
Target:	WLS_SOA1 (migratable)	The list of all WebLogic Server instances that have been defined in the current domain and are therefore candidates for hosting this JDBC store. More Info...
Data Source:	JMSDS	The JDBC data source used by this JDBC store to access its backing table. You cannot configure a JDBC store to use a JDBC data source that is configured to use an XA JDBC driver or configured to support global transactions. More Info...
Prefix Name:	JMSJDBC_1	The prefix for the JDBC store's database table (WLStore), in the following format: [[[catalog.]schema.]prefix]. More Info...

Image 13: Migratable target configuration

7. Click Save (you may ignore the error about the JMS server or SAF agent SOAJMSServer_auto_1 not being targeted to the same target as its persistent store)
8. Repeat steps 4-7 for all JMS servers using JDBC persistent stores
9. On the navigation tree on the left, expand Services and click on Persistent Stores
10. Click on the persistent store associated to the SOAJMSServer_auto_1 JMS server (the persistent store can be determined form the Services->Messaging ->JMS Servers table)
11. Select "WLS_SOA1(migratable)" as target
12. Click Save
13. Repeat steps 10-12 for all the JMS JDBC persisten stores using the appropriate migratable targets

(WLS_SOA1(migratable), WLS_SOA2(migratable) etc)

14. Activate the Changes
15. Restart the SOA servers

Once in place restart is configured, the behavior of the JMS JDBC persistent store can be verified setting the “-*Dweblogic.debug.DebugSingletonServices=tru*e” and *“-Dweblogic.StdoutDebugEnabled=true*” start properties for the servers. Upon a database failure, the following events should be logged in the server’s out file:

Add store to migratable target:

```
<Debug> <SingletonServices> <BEA-000000> <MigratableGroup: adding migratable SOAJMSJDBCStore_auto_1 to group WLS_SOA1 (migratable) Migratable class - weblogic.management.utils.GenericManagedService$ManagedDeployment>
```

Reach out to DB:

```
<Info> <Store> <BEA-280071> <JDBC store "SOAJMSJDBCStore_auto_1" opened table "JMSJDBC_1WLStore" and loaded 0 records. For additional JDBC store information, use the diagnostics framework while the JDBC store is open.>
```

Crash f DB:

```
<Emergency> <Store> <BEA-280060> <The persistent store "SOAJMSJDBCStore_auto_1" encountered a fatal error, and it must be shut down: weblogic.store.PersistentStoreFatalException: [Store:280065]java.sql.SQLException: Connection has been administratively destroyed. Reconnect. (server="WLS_SOA1" store="SOAJMSJDBCStore_auto_1" table="JMSJDBC_1WLStore"):(Linked Cause, "java.sql.SQLException: Connection has been administratively destroyed. Reconnect.")
```

Failed Store:

```
<Error> <Store> <BEA-280074> <The persistent store "SOAJMSJDBCStore_auto_1" encountered an unresolvable failure while processing transaction "BEA1-148701EFAE50DC04072E". Shutdown and restart to resolve this transaction. weblogic.store.gxa.GXAException: weblogic.store.PersistentStoreException: weblogic.store.PersistentStoreFatalException: [Store:280032]The persistent store suffered a fatal error and it must be re-opened
```

Deactivate Store:

```
<Debug> <SingletonServices> <BEA-000000> <MigratableGroup: Going to call migratableDeactivate on SOAJMSJDBCStore_auto_1 for WLS_SOA1 (migratable)>
```

Reactivate Store:

```
<Debug> <SingletonServices> <BEA-000000> <MigratableGroup: activating migratable 'SOAJMSJDBCStore_auto_1' for WLS_SOA1 (migratable)>
```

Ownership lock on table and failure to activate (first try):

```
<Warning> <Store> <BEA-280076> <Database table "JMSJDBC_1WLStore" for store "SOAJMSJDBCStore_auto_1" is currently owned by "[name={server=WLS_SOA1!host=192.168.48.58!domain=soaexa_domain!store=SOAJMSJDBCStore_auto_1!table=JMSJDBC_1WLStore}:random=5074588073335957339:timestamp=1401991210748]". Trying to wait for ownership.>
```

New try to activate Persistent/ Store:

```
<Debug> <SingletonServices> <BEA-000000> <MigratableGroup: activating migratable 'SOAJMSJDBCStore_auto_1' for WLS_SOA1 (migratable)>
```

Successful activation of store:

```
<Info> <Store> <BEA-280071> <JDBC store "SOAJMSJDBCStore_auto_1" opened table "JMSJDBC_1WLStore" and loaded SOAJMSJDBCStore_auto_1 records. For additional JDBC store information, use the diagnostics framework while the JDBC store is open.>
```

To reduce logging overhead, remove the “*-Dweblogic.debug.DebugSingletonServices=true*” and “*-Dweblogic.StdoutDebugEnabled=true*” flags from the server’s start properties once the correct behavior is verified.

Appendix C: Considerations for Oracle FMW SOA Multi Data Center deployments on Exalogic

When deploying Multi Data Center systems in Active-Active configurations on Exalogic the base domain is created following the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2. The stretched cluster design is basically an Enterprise Deployment Topology on Site1/ Exalogic RACK1 scaled out to two additional compute nodes in Site2/Exalogic RACK 2. The following diagram describes the topology:

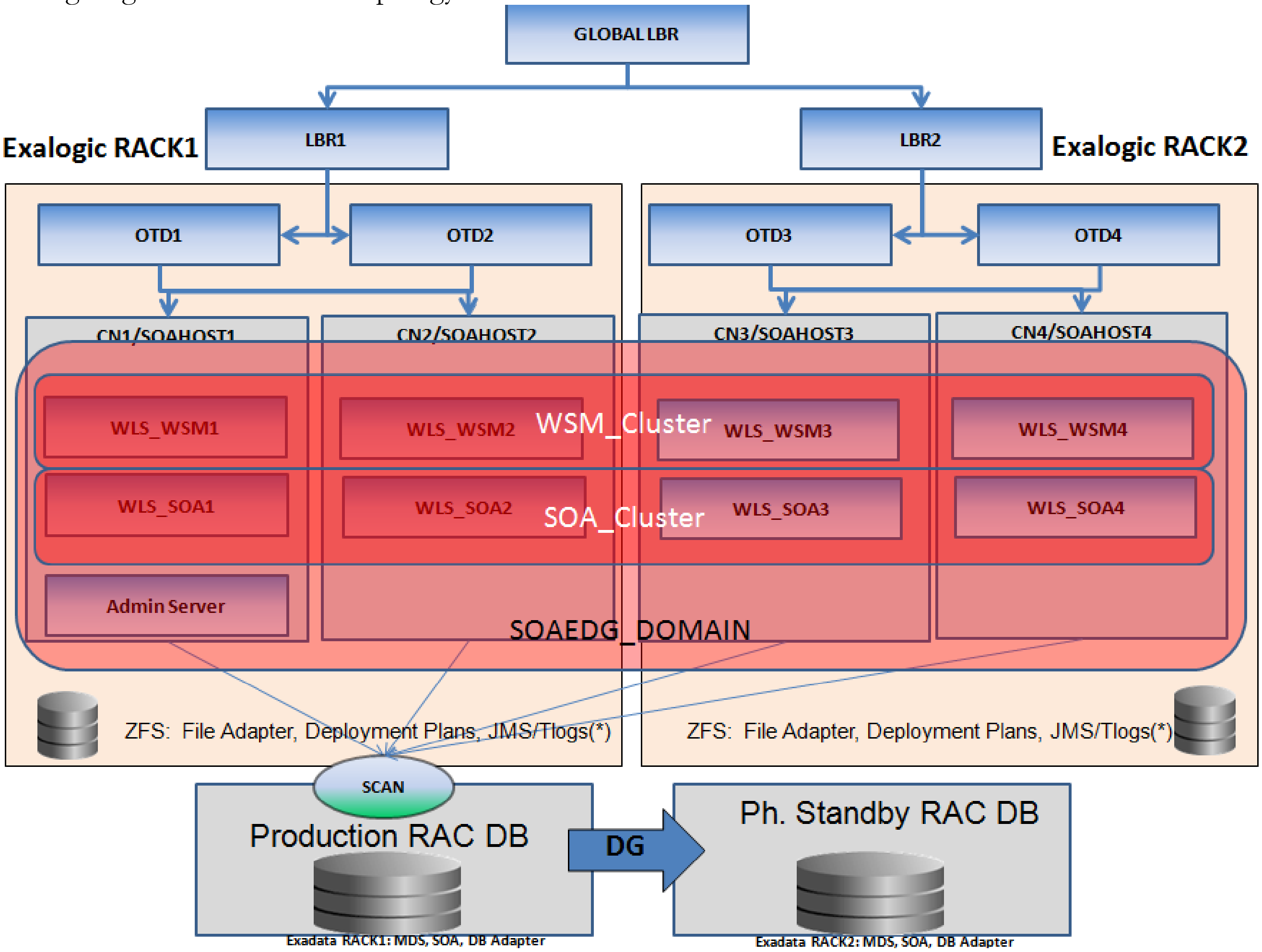
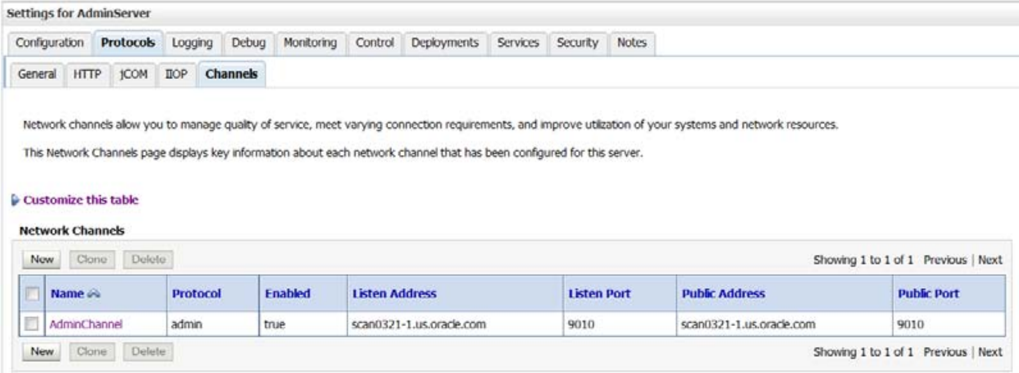


Image 14: Multidatacenter deployment on Exalogic

The Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2 setup, needs to be slightly modified, however, because IPoIB communications are precluded across sites. The following additional configuration steps are needed and differ from the domains described for non-Exalogic Multi Data Center Topologies and the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite:

1. An Administrative Channel has to be created in the Administration Server using its virtual hostname (the same as described in the Exalogic EDG for SOA) using an SSL port. This channel is required so that deployment and configuration management operations succeed across sites (the default listen address used for Managed Servers in the SOA EDG on Exalogic, use IPOIB addresses that are non reachable from both sites). Refer to the Oracle® Fusion Middleware Configuring Server Environments for Oracle WebLogic Server 11g Release 1 (10.3.6) documentation for steps and details about administrative channels. Keep in mind that using an administrative channel requires that you establish an SSL connection to the Administration Server in order to start any Managed Server in the domain. This applies whether you start Managed Servers manually, at the command line, or using Node Manager. Thus, it is required to create the appropriate certificate and trust stores for the servers in both Exalogic racks.



Network channels allow you to manage quality of service, meet varying connection requirements, and improve utilization of your systems and network resources.

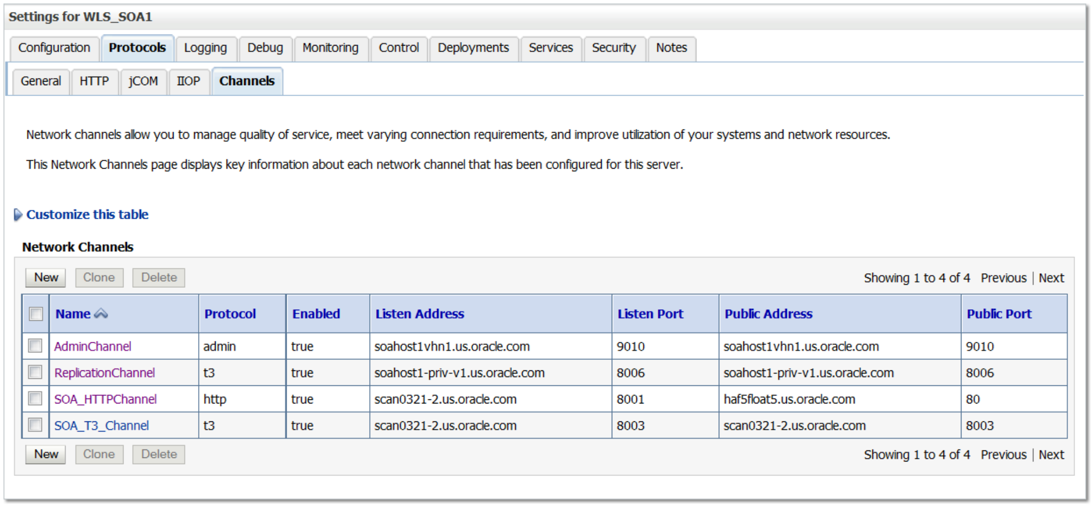
This Network Channels page displays key information about each network channel that has been configured for this server.

Network Channels

□	Name	Protocol	Enabled	Listen Address	Listen Port	Public Address	Public Port
□	AdminChannel	admin	true	scan0321-1.us.oracle.com	9010	scan0321-1.us.oracle.com	9010

Image 15: Administrative Channel created on the Administration Server

2. SOA Servers keep their default listen address on IPoIB and use the following channels:
    a. An additional administrative channel to be managed by the Administration Server.
    b. The channels on EoIB and IpoIB described in the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2 for session replication and to be accessible from external jms/rmi clients



Network channels allow you to manage quality of service, meet varying connection requirements, and improve utilization of your systems and network resources.

This Network Channels page displays key information about each network channel that has been configured for this server.

Network Channels

□	Name	Protocol	Enabled	Listen Address	Listen Port	Public Address	Public Port
□	AdminChannel	admin	true	soahost1vhn1.us.oracle.com	9010	soahost1vhn1.us.oracle.com	9010
□	ReplicationChannel	t3	true	soahost1-priv-v1.us.oracle.com	8006	soahost1-priv-v1.us.oracle.com	8006
□	SOA_HTTPChannel	http	true	scan0321-2.us.oracle.com	8001	haf5float5.us.oracle.com	80
□	SOA_T3_Channel	t3	true	scan0321-2.us.oracle.com	8003	scan0321-2.us.oracle.com	8003

Image 16: Channels used for the SOA Servers

3. Node Managers in both sites use an EoIB addresses as listen address so that they can be contacted by the Administration Server (in the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2 they use IPoIB addresses). They listen on the compute nodes' EoIB hostname.
4. The WSMPM servers also change their listen addresses from IPoIB to EoIB for the appropriate JOC cluster to work across sites. They listen also on the compute nodes' EoIB hostname. JOC cluster configuration needs to be re-run with this configuration. Refer to section Configuring the Java Object Cache for Oracle WSM in the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2.
5. The coherence cluster used by SOA servers for deployments and MDS updates, needs to use EoIB instead of IPoIB. This is needed so that deployments and updates are properly propagated across sites/racks. For example the coherence start arguments used for WLS_SOA1 changes from:

```
-Dtangosol.coherence.wka1=SOAHOST1-PRIV-V1
-Dtangosol.coherence.wka2=SOAHOST2-PRIV-V1
-Dtangosol.coherence.localhost=SOAHOST1-PRIV-V1
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

to:

```
-Dtangosol.coherence.wka1=soahost1vhn1.example.com
-Dtangosol.coherence.wka2=soahost2vhn1.example.com
-Dtangosol.coherence.wka3=soahost3vhn1.example.com
-Dtangosol.coherence.wka4=soahost4vhn1.example.com
-Dtangosol.coherence.localhost=soahost1vhn1.example.com
```

```
    -Dtangosol.coherence.localport=8089
    -Dtangosol.coherence.wka1.port=8089
    -Dtangosol.coherence.wka2.port=8089
    -Dtangosol.coherence.wka3.port=8089
    -Dtangosol.coherence.wka4.port=8089
```

Other aspects as explained in section “Configuring the Oracle Fusion Middleware SOA Active-Active Topology” (pages 20-36) apply also to Exalogic Systems with the following additional considerations:

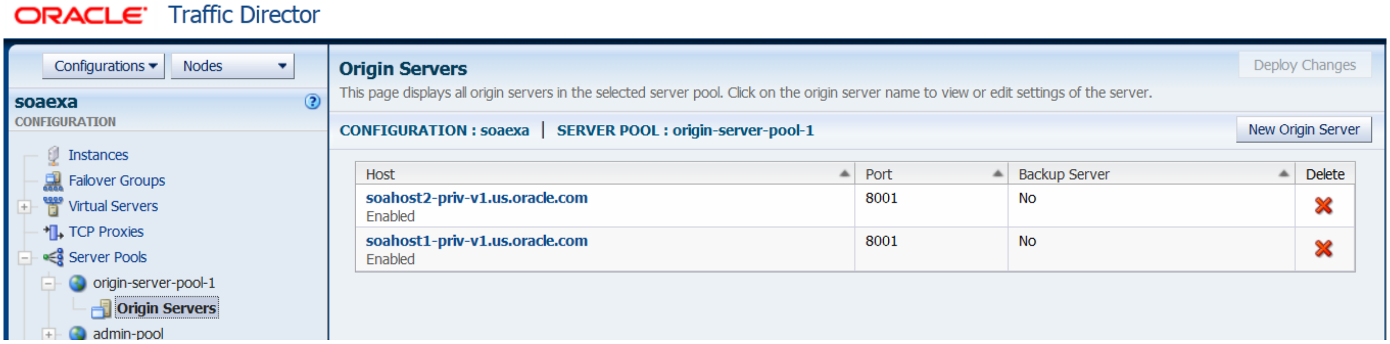
Load Balancer considerations

Configure the GLBR and the LBRs as described in section “Configuring Load Balancers and Global Load Balancers for Oracle Fusion Middleware SOA Multi Data Center Active-Active Deployment”. Notice that the LBR pools will use the appropriate OTD addresses (instead of OHS addresses) for routing requests. Local routing should be achieved similarly as in the commodity hardware example by defining two pools, one for OTDs in Site1/RACK1 and another for OTDs in Site2/RACK2 and using a similar rule as the one explained for OHS (page 22 in this document)

WebTier/OTD Considerations

The OTDs in either site use only local SOA servers as members in the pools (i.e. OTD in Site1/RACK1 points to SOA servers in Site1/RACK1 and OTD in Site2/RACK2 uses a pool with only SOA servers in Site2/RACK2). This pool use the IPoIB listen address of the SOA servers. Here are the example pools:

Site1/RACK1



Origin Servers

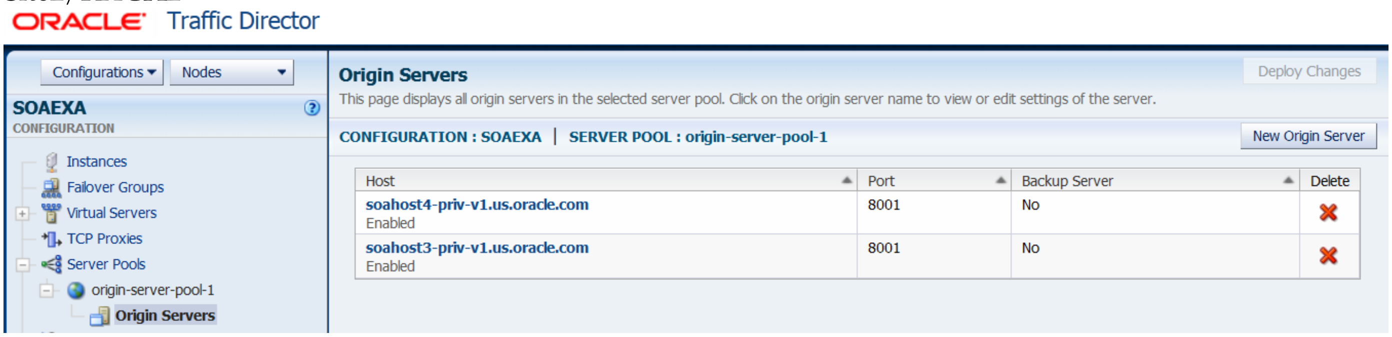
This page displays all origin servers in the selected server pool. Click on the origin server name to view or edit settings of the server.

CONFIGURATION : soaexa | SERVER POOL : origin-server-pool-1

Host	Port	Backup Server	Delete
soahost2-priv-v1.us.oracle.com Enabled	8001	No	✖
soahost1-priv-v1.us.oracle.com Enabled	8001	No	✖

Image 17: Server pool for OTDs in Site1

Site2/RACK2



Origin Servers

This page displays all origin servers in the selected server pool. Click on the origin server name to view or edit settings of the server.

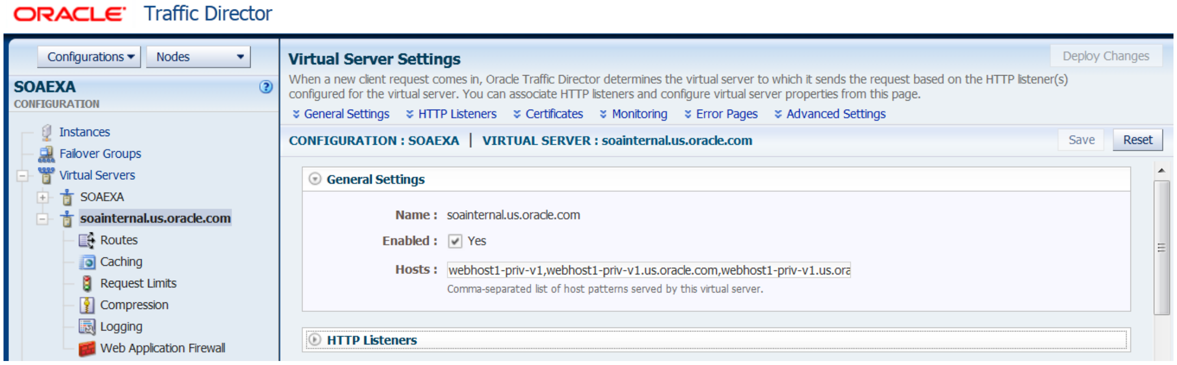
CONFIGURATION : SOAEXA | SERVER POOL : origin-server-pool-1

Host	Port	Backup Server	Delete
soahost4-priv-v1.us.oracle.com Enabled	8001	No	✖
soahost3-priv-v1.us.oracle.com Enabled	8001	No	✖

Image 18: Server Pool for OTDs in Site2

Monitors, virtual servers etc are configured in a similar fashion in both sites. The soainternal virtual server, needs to use the exact same hostname alias in both sites so that SOA can loop back to its local OTD (only one address can be specified as server URL for a SOA Cluster). Refer to section Section 9.7, "Configuring Oracle Traffic Director with the Extended Domain" in the the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2 for details. Here is an example of sointernal.mycompany.com virtual server's definition:

Site1/RACK1



Virtual Server Settings

When a new client request comes in, Oracle Traffic Director determines the virtual server to which it sends the request based on the HTTP listener(s) configured for the virtual server. You can associate HTTP listeners and configure virtual server properties from this page.

CONFIGURATION : SOAEXA | VIRTUAL SERVER : soainternal.us.oracle.com

General Settings

Name : soainternal.us.oracle.com

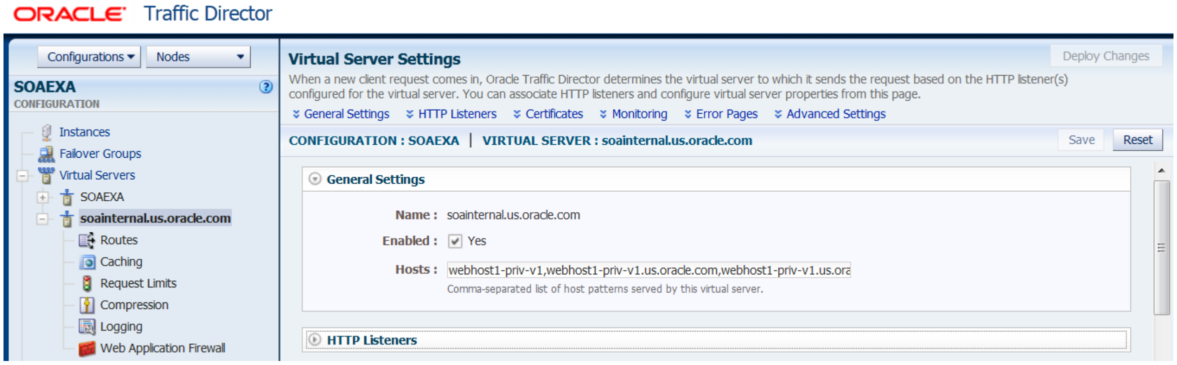
Enabled : ☑ Yes

Hosts : webhost1-priv-v1,webhost1-priv-v1.us.oracle.com,webhost1-priv-v1.us.ora

Comma-separated list of host patterns served by this virtual server.

HTTP Listeners

Site2/RACK2



Virtual Server Settings

When a new client request comes in, Oracle Traffic Director determines the virtual server to which it sends the request based on the HTTP listener(s) configured for the virtual server. You can associate HTTP listeners and configure virtual server properties from this page.

CONFIGURATION : SOAEXA | VIRTUAL SERVER : soainternal.us.oracle.com

General Settings

Name : soainternal.us.oracle.com

Enabled : ☑ Yes

Hosts : webhost1-priv-v1,webhost1-priv-v1.us.oracle.com,webhost1-priv-v1.us.ora

Comma-separated list of host patterns served by this virtual server.

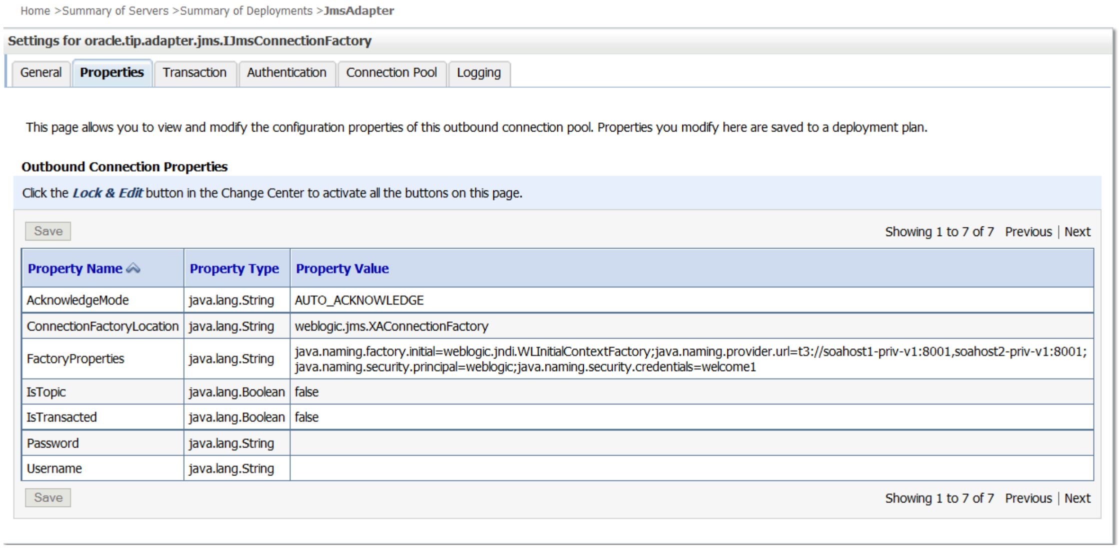
HTTP Listeners

Application Tier considerations.

Shared Storage/Directory Configuration: The same binary, path and shared storage considerations explained for Multi Data Center Active-Active deployments on commodity hardware apply to Exalogic configuration: each Site/Exalogic RACK should use its own redundant binaries and paths should be the same for domain locations and ORACLE HOMES. For all the examples and applications tested by Oracle, the throughput differences between File vs. Db stores (ZFS vs. Exadata in context of Exalogic Systems) is even less significant than on commodity hardware. In any case, the applicability of each type of store may depend on the application/business requirements. Refer to section "Shared Storage vs. Database for Transaction Logs and Persistent stores" for details

Server migration configuration: Just as for commodity hardware, in the stretched domain design servers use only those machines in their same site as primary candidates for migration. Use the steps in Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2 for configuring server migration with the same considerations explained in the "Configuring the Application Tier of an Oracle Fusion Middleware SOA AA DR System for a Stretched Cluster" section. Notice that servers in the Exalogic Elastic Cloud Deployment Guide for Oracle SOA use additional channels/listen address to the one used by default in the standard Enterprise Deployment Guide for Oracle SOA Suite (listen address on both EoiB and IpoIB are used) hence it is needed to add multiple interfaces to the Node Manager configuration.

JMS configuration, JMS Adapter and File Adapter configuration: The same JMS and File Adapter considerations explained in the "Configuring the Application Tier of an Oracle Fusion Middleware SOA AA DR System for a Stretched Cluster" apply on Exalogic systems. The only difference for Exalogic configurations is that the JMS adapter is configured with the IPoIB addresses used by servers for the initial JNDI context retrieval.


Home >Summary of Servers >Summary of Deployments >JmsAdapter
General
Properties
Transaction
Authentication
Connection Pool
Logging
Save
Showing 1 to 7 of 7 Previous | Next

Settings for oracle.tip.adapter.jms.IJmsConnectionFactory

This page allows you to view and modify the configuration properties of this outbound connection pool. Properties you modify here are saved to a deployment plan.

Outbound Connection Properties

Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

Property Name	Property Type	Property Value
AcknowledgeMode	java.lang.String	AUTO_ACKNOWLEDGE
ConnectionFactoryLocation	java.lang.String	weblogic.jms.XAConnectionFactory

FactoryProperties	java.lang.String	java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.provider.url=t3://soahost1-priv-v1:8001,soahost2-priv-v1:8001;
		java.naming.security.principal=weblogic;java.naming.security.credentials=welcome1

IsTopic	java.lang.Boolean	false
IsTransacted	java.lang.Boolean	false
Password	java.lang.String	
Username	java.lang.String	

Data Source configuration: As explained for non Exalogic configuration (page 34) the Data Sources used by Oracle Fusion Middleware SOA Suite should be configured to automate failover of connections in case there is failover or switchover of the active database. The same type of failover jdbc connect strings should be used for Exalogic Systems. It is not recommended to use compound connect strings that allow connecting to the same database using SDP or TCP depending on the IPoIB connectivity between the Exalogic RACKs and Exadata. The reason is that OracleNet will traverse and test the list of listeners with every connection request (until the first one that succeeds) and this may affect performance. Given this, the recommendation is using a standard failover JDBC url using EoIB addresses as described in page 36.

Oracle FMW SOA Suite ServerURL: As described in the Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2 SOA Servers are configured with an IPoIB address available in OTD as their Server and HTTPServer URL address. This allows restraining invocations from SOA-server-to-SOA-server inside the Infiniband fabric (instead of looping back to the front end GLBR). The Server and HTTPServer URL address are unique for each SOA Cluster. In order to make Site2/RACK2 generate requests also to its local OTD, it is necessary to alias the hostname used for the Server URL to the local OTD listen address. This can be done using /etc/hosts hostname resolution. For example, if webhost1-priv-v1 is used as Server URL and HTTPServer URL for SOA (see http://docs.oracle.com/cd/E18476_01/doc.220/e47690/extend_soa.htm#CHDHGAHC) then this hostname should map to different addresses in each site (the address that corresponds to the OTD's soainternal.us.oracle.com virtual server):

```
root@cn1site1 ~]# cat /etc/hosts | grep webhost1-priv-v1
```

```
192.168.48.56 webhost1-priv-v1.example.com webhost1-priv-v1

[root@cn1site2]# cat /etc/hosts | grep webhost1-priv-v1
192.168.41.72  webhost1-priv-v1.example.com webhost1-priv-v1
```

References

1. *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*

http://docs.oracle.com/cd/E23943_01/core.1111/e12036/toc.htm

2. *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*

http://docs.oracle.com/cd/E23943_01/admin.1111/e10226/toc.htm

3. *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite 11g Release 1 (11.1.1.6)*

http://docs.oracle.com/cd/E23943_01/dev.1111/e10224/fod_hi_level_fod.htm#CIHGDIII

4. *Oracle Fusion Middleware User's Guide for Technology Adapters*

http://docs.oracle.com/cd/E23943_01/integration.1111/e10231/toc.htm

5. *F5's Big IP Global Traffic Manager Documentation*

http://support.f5.com/kb/en-us/products/big-ip_gtm.html

6. *Oracle WebLogic Server and Highly Available Oracle Databases: Oracle Integrated Maximum Availability Solutions*

http://www.oracle.com/technetwork/database/features/availability/wlsdatasourcefordataguard-1534212.pdf

7. *Best Practices for Active-Active Fusion Middleware: Oracle WebCenter Portal*

http://www.oracle.com/technetwork/database/availability/webcenteractiveactive-1621358.pdf

8. *Oracle Coherence Developer's Guide*

http://docs.oracle.com/cd/E24290_01/coh.371/e22837/toc.htm

9. *Oracle® Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite Release EL X2-2 and EL X3-2*

http://docs.oracle.com/cd/E18476_01/doc.220/e47690/toc.htm

Best Practices for Oracle FMW SOA 11g Multi Data Center Active-Active Deployment
September 2014

Author: Fermin Castro
Contributing Authors: Susan Kornberg, Richard del Val, Michael Rhys, Pradeep Bhat, Virginia Beecher, Vikas Anand

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

Worldwide Inquiries:
Phone: +1.650.506.7000

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

0109