Oracle WebLogic Server Active GridLink for Oracle Real Application Clusters (RAC)

An Oracle White Paper
August 2011

Oracle WebLogic Server Active GridLink for Real Application Clusters (RAC)

Oracle WebLogic Server Active GridLink for Real Application Clusters (RAC)

1.0 INTRODUCTION

High availability and scalability are often the focus in different systems enabling customer applications with cost efficient solutions. There are various known issues in a heterogeneous complex environment involving Java EE middle tier and backend databases. For example, an application request may get blocked for a long period when a database node dies. There is no easy way to tell whether to obtain a fresh new connection after an application received the SQLException. The middle tier applications are unaware of new or restarted database nodes or it executes the work on a slow, hung or dead database node and they often have to rely on waiting for TCP/IP time-outs.

Oracle WebLogic Server 11g provides strong support for the Real Application Clusters (RAC) features in Oracle Database 11g, minimizing database access time while allowing transparent access to rich pooling management functions that maximizes both connection performance and availability.

Oracle WebLogic Server 11g RAC integration solutions have been jointly designed implemented and tested by Oracle Fusion Middle Ware and Database teams. It's not only the best High Availability solution in the market, but also Web Logic Server is the only Application Server who has been fully integrated and certified with Oracle Database RAC 11g without losing any rich functionalities in Java EE implementation with security, transaction, connection pooling , etc management.

There are two data source implementations in Oracle WebLogic Server to support Oracle Real Application Clusters (RAC): the multi data source solution which has been used successfully in customer production deployments since WebLogic Server 8.1 SP5 and the new implementation in Oracle WebLogic 11*g* Release 1 (10.3.4) called Oracle WebLogic Active GridLink for RAC which is the market-leading mid-tier integration solution leveraging additional Oracle RAC advancements. Oracle WebLogic Server Active GridLink for RAC is the strategic solution for supporting Oracle Real Application Clusters recommended by Oracle[1]. It represents the best possible middleware and database integration with features that are not available from other vendors.

The combination of Oracle WebLogic Server Data Source and Connection Pooling solutions and Oracle RAC provides a high-end mission-critical environment offering performance, high scalability and availability features. Load-balancing and Affinity capabilities offer significant performance improvement for online transaction processing scenarios, as well as improving throughput and total response time. Failover solution gives end-to-end rapid failure detection supporting graceful shutdown for planned and unplanned Oracle RAC node outages.

Oracle WebLogic Server 11g provides strong support for Oracle Real Application Clusters (RAC) features in Oracle Database 11g, minimizing database access time while allowing transparent access to rich pooling management functions that maximizes both connection performance and availability.

---

[1] Please refer to the Oracle Fusion Middleware Licensing Information documentation for Active GridLink features
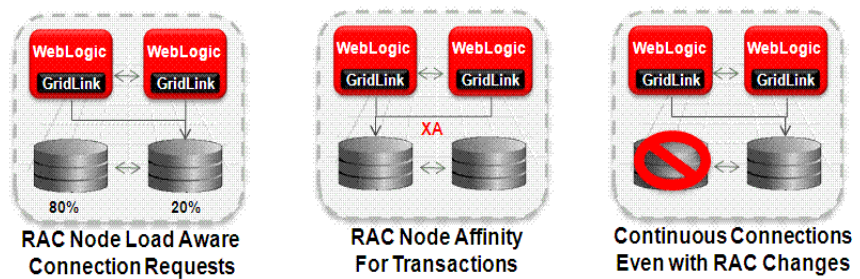
Figure 1

In this article, we start with a brief introduction to Oracle RAC and an overview of the Oracle RAC features supported in Oracle WebLogic Server 11g. We then focus on details of WebLogic Active GridLink for RAC features and the configuration options, with working samples. All the configuration steps and application samples are included in a companion "How-To" that is linked from this article.

In each feature set, the different supports of Multi Data Source and Active GridLink for RAC are covered to clearly explain the functionalities.

2.0 ORACLE REAL APPLICATION CLUSTERS

Oracle RAC enables you to cluster Oracle databases. Single-instance Oracle databases have a one-to-one relationship between the Oracle database and the instance. Oracle RAC environments have a one-to-many relationship between the database and the instance.

Figure1 below shows how Oracle RAC is the Oracle Database option that provides a single system image for multiple servers to access one Oracle database. In Oracle RAC, each Oracle instance usually runs on a separate server.
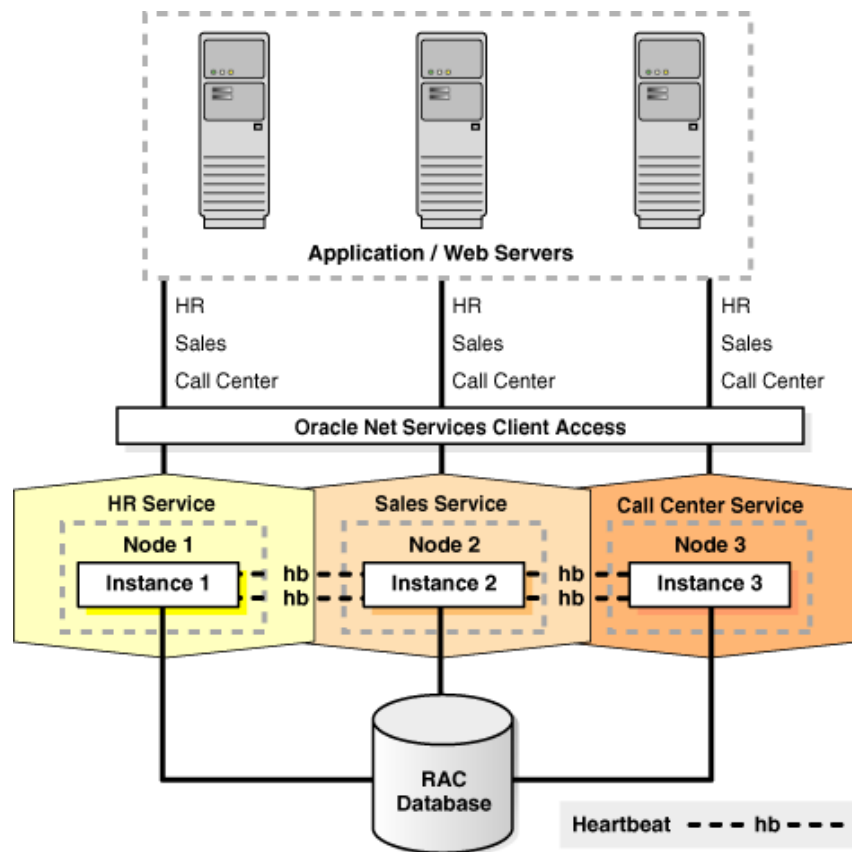
Figure 2

The new features in Oracle RAC 11gR2 include Oracle RAC One Node support which provides enhanced high availability for single-instance databases, protecting them from both planned and unplanned downtime; Single Client Access Name (SCAN) that Oracle Database 11g database clients use SCAN to connect to the database with the capability of resolving to multiple IP addresses, reflecting multiple listeners in the cluster handling public client connections.

2.1 New Support in Oracle RAC 11gR2

The new features in Oracle RAC 11gR2 include:

- Oracle RAC One Node, it provides enhanced high availability for single-instance databases, protecting them from both planned and unplanned downtime.

- Single Client Access Name (SCAN), Oracle Database 11g database clients use SCAN to connect to the database with the capability of resolving to multiple IP addresses, reflecting multiple listeners in the cluster handling public client connections.

- Edition-Based Redefinition, an edition attribute can be specified for a database service using SRVCTL, then all subsequent connections that specify the service use this edition as the initial session edition.

- Enhanced Oracle RAC monitoring and diagnostics in Enterprise Manager.

- Enhanced Oracle RAC Configuration Assistants.

- Enhancements to SRVCTL for Grid Infrastructure Management.

- OCI runtime connection load balancing; which supports for parallel execution processes running on database instances and support for distributed transactions spanning; and the Oracle Database Quality of Service Management Server.

Oracle WebLogic Server 11g and Oracle 11g RAC are fully certified to work together providing high-availability, scalability and performance. The Oracle RAC services, such as failover, runtime connection load-balancing, Affinity etc, are available via Oracle WebLogic Server JDBC data source and connection pool implementations.

3.0 ORACLE WEBLOGIC SERVER AND RAC

In Java EE Application Servers, database interactions are typically handled by data source implementations. You configure and expose a connection to databases as JDBC data sources.

There are two data source implementations in Oracle WebLogic Server to support Oracle Real Application Clusters (RAC): the multi data source solution which has been used successfully in customer production deployments and the new implementation in Oracle WebLogic 11*g* Release 1 (10.3.4) called Oracle WebLogic Active GridLink for RAC which is the market-leading mid-tier integration solution leveraging the latest and greatest Oracle RAC advances.

3.1 Oracle WebLogic Server Multi Data Source

The WebLogic Server JDBC subsystem has supported Oracle RAC since WLS version 8.1 SP5, originally developed for Oracle9*i* RAC. This support is based on a particular type of data source configuration, called a *multi data source*. A multi data source is a data source abstraction over one or more individual data sources. It serves JDBC connections from each of the member data sources according to a specified policy[2]. A RAC multi data source configuration requires that each member data source obtain connections to a particular RAC instance, as illustrated in the following diagram of a three-node RAC cluster configuration.

There are two data source implementations in Oracle WebLogic Server to support Oracle Real Application Clusters (RAC): the multi data source solution and the new implementation in Oracle WebLogic 11 g Release 1 (10.3.4) called Oracle WebLogic Active GridLink for RAC.
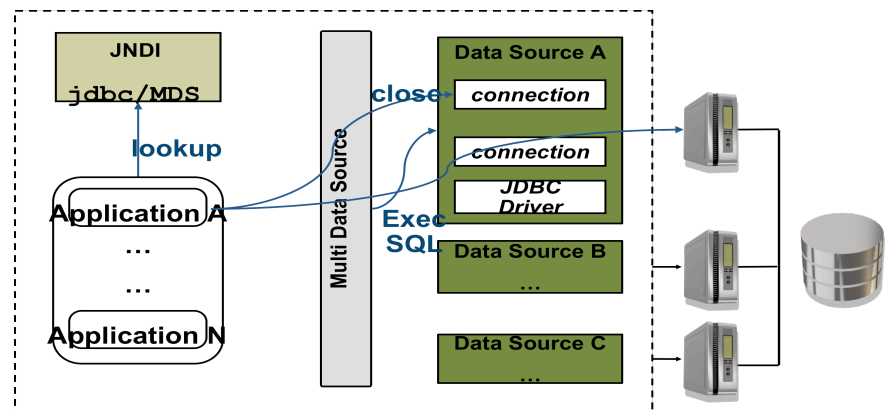

Figure 3

---

[2] Supported multi data source policies include load balancing and failover.

3.1.1 Capabilities

WebLogic RAC multi data source configurations provide the following capabilities: load balancing, failover and XA affinity.

Load Balancing
When the Multi Data Source algorithm is set to load balancing, application connection reserve requests are served from each member data source in a round-robin fashion. This allows for improved utilization of the cluster resources than the failover policy, and is supported for XA data sources as well.

Failover
The failover Multi Data Source algorithm causes connection borrow requests to be served from a single member data source until such time as the member is unavailable. Failover can optionally occur when the pool's capacity has been exhausted. The failover policy is typically used with active-passive architectures.

XA Affinity and Failover
When accessed within a global transaction, the member data source from which the JDBC connection was obtained is pinned to the global transaction for the life of the transaction. This ensures that all database operations performed on connections obtained from the Multi Data Source, for a particular transaction, all execute on the same RAC instance. XA affinity results in improved performance and is even a requirement for older versions of RAC, such as prior to 11g.
The XA failover is also supported by the Multi Data Source and transaction manager implementations. If a pinned RAC instance suffers a failure, then a global transaction can complete utilizing a different RAC instance using a connection obtained one of the other member data sources.

3.1.2 Limitations

The WebLogic Multi Data Source is a pure middle tier implementation which does not leverage the Oracle RAC features such as the Oracle Notification Service (ONS). As a result, WebLogic Multi Data Sources do not have immediate knowledge of backend events, and have related functional limitations described below.

Configuration **Complexity**
A WebLogic Multi Data Source requires n+1 JDBC modules, where n is the number of nodes in the RAC cluster. For RAC service configurations, a separate multi data source is required for each defined service. In addition, the configuration itself is static and requires administrative intervention to add or remove data sources when changes are made to the RAC cluster topology.

Connection Polling
Connection polling is the mechanism used in Multi Data Source to determine the viability of individual JDBC connections and to detect changes in the RAC cluster topology. This mechanism is an alternative mechanism to use of backend notification services. Although effective, performing SQL operations on individual connections comes at the expense of additional runtime overhead, and potentially delayed detection of RAC node failures. Also, the potential exists for false positives that could result in the unnecessary disablement of the data source pool and the termination of valid connections that may be in use by applications.

Load Balancing Algorithm
The round-robin load balancing employed by WebLogic Multi Data Source implementation distributes work evenly across all member data sources. Finer grained control is desirable for situations where RAC instances exhibit different performance/response time characteristics.

XA Affinity per MDS
XA affinity is provided by each Multi Data Source. When several Multi Data Sources are enlisted in a global transaction, it is possible that connections could be obtained from different RAC instances. This results in branches of the same global transaction being processed by separate RAC instances. Although supported in later RAC versions, it is less than optimal from a performance perspective.

3.2 Active GridLink for RAC

Oracle WebLogic Server 10.3.4 introduced a single data source implementation to support an Oracle RAC cluster. It responds to FAN events to provide Fast Connection Failover (FCF), Runtime Connection Load-Balancing (RCLB), and RAC instance graceful shutdown. XA affinity is supported at the global transaction Id level. The new feature is called WebLogic Active GridLink for RAC; which is implemented as the GridLink Data Source within WebLogic Server.
The RAC integration capabilities of Universal Connection Pool (UCP) have been utilized by the WebLogic Server GridLink Data Source implementation to provide the FCF, RCLB and Affinity features.

With the key foundation for providing deeper integration with Oracle RAC, this single data source implementation in Oracle WebLogic Server supports the full and unrestricted use of database services as the connection target for a data source.
The active management of the connections in the pool is based on static settings configured on the connection pool itself (min/max capacity, timeouts, etc.) and real time information the connection pool receives from the RAC Oracle Notification Service (ONS) subsystem that advises the "client" of any state changes within the RAC cluster.

The Universal Connection Pool Java library has been integrated with WebLogic Server and been utilized by WebLogic GridLink data source implementation to provide the Fast Connection Failover, Runtime Connection Load Balancing and Affinity features.

Oracle Database services (services) are logical abstractions for managing workloads in Oracle Database. Services divide workloads into logically disjoint groupings. Each service represents a workload with common attributes, service-level thresholds, and priorities. Services are built into the Oracle Database, providing a single system image for workloads, prioritization for workloads, performance measures for real transactions, and alerts and actions when performance goals are violated. Services enable database administrators to configure a workload, administer it, enable/disable it, and measure workload as a single entity.

The GridLink Data Source is associated with a connection pool, which contains a set of heterogeneous connections to the RAC instances that are hidden behind the database service. When an application requests a connection from the data source, a suitable connection is borrowed from the pool and supplied to the application

based on the load balancing information the connection pool has received and the current distributions of connections in use from the pool.

This new implementation simplifies the use of Oracle RAC database with WLS through the GridLink Data Source approach, which in turn reduces the configuration and management complexity required to use Oracle RAC. Note that utilization of Multi Data Source configurations for RAC environments will continue to be supported. Upgrades from RAC Multi Data Sources to Grid Link Data Sources are straight-forward and involve creating a single Grid Link Data Source with the same JNDI name as the Multi Data Source, which reduces the number of configuration artifacts to maintain.

3.2.1 Fast Connection Failover

The Fast Connection Failover (FCF) feature is a Fast Application Notification (FAN) client implemented through the Universal Connection Pool. The feature requires the use of an Oracle JDBC driver and an Oracle RAC database or use of Oracle Restart on a single instance database.

WebLogic GridLink Data Source has been integrated with FCF from Universal Connection Pool implementation and uses FCF to:
- Provide rapid failure detection
- Abort and remove invalid connections from the connection pool quickly
- Perform graceful shutdown for planned and unplanned Oracle RAC node outages
- Adapt to changes in topology, such as adding or removing a node
- Distribute runtime work requests to all active Oracle RAC instances, including those rejoining a cluster

ONS is used by the Oracle RAC database to broadcast events that describe a change of state. GridLink Data Sources can register to receive notifications from ONS and therefore quickly become aware of any state changes in a RAC database. Using these state change notification events, GridLink Data Sources can intelligently adapt its connection pools so that it provides continuous, reliable and efficient access to the RAC database as changes happen.

An adaptive response to state changes in the RAC cluster allows WebLogic Server to handle outages by immediately retracting, closing and discarding connections to RAC instances that have been stopped or taken out by an unplanned outage, without needing to periodically poll the connections to ensure they are valid, or affecting uninvolved connections to surviving nodes. This eliminates the need to to test connections to ensure applications are not given dead connections and quickly removes dead connections from RAC node failures, which in some failure modes, might otherwise hang for minutes.

Further, it allows WebLogic Server to proactively reapportion its set of connections to support scenarios where new RAC instances are added or are restarted after an outage. This results in WLS being able to make full use of the resources within the RAC database. Furthermore, using the database service model, this allows database administrators to make changes to the RAC service/instance allocations, which are then seamlessly applied through the affected WLS connection pools without needing to make configuration changes to the connection pool configuration. It

also removes the need to create complex arrangements of multiple data sources to represent a dedicated instance of the RAC database.

The WebLogic GridLink Data Source provides Fast Connection Failover capabilities and responds to RAC database service and node events {UP, DOWN} to ensure that the reserve o physical connections in the pool are always pointing to a valid database node; and it ensures th the reserve of physical connections are well distributed across the available database nodes. Th Fast Connection Failover behavior is enabled as a configuration setting on the GridLink Data Source.

With the Fast Connection Failover capability enabled, the following scenarios are supported:

- Planned down Event - Planned outages are defined as database maintenance or other activities that are needed to perform at a known point in time. Support for these events is available where an Oracle RAC service can be gracefully shutdown. In such scenarios, any borrowed or in-use connections are not interrupted and closed until work is completed and control of the connection is returned to the pool. This provides an extremely efficient way in large heterogeneous customer environments to manage planned outages.

- Unplanned down Event - Support for unplanned outages is provided by detecting and removing stale connections to an Oracle RAC cluster. Stale connections include connections that do not have a service available on any instance in an Oracle RAC cluster due to service-down and node-down events. Borrowed connections and available connections that are stale are detected, and their network connection is severed before removing them from the pool. These removed connections are not replaced by the pool. Instead, the application must retry connections before performing any work with a connection.

  The primary difference between unplanned and planned shutdown scenarios is how borrowed connections are handled. Stale connections that are idle in the pool (not borrowed) are removed in the same manner as the unplanned shutdown scenario.

- Up Event - Oracle RAC Instance Rejoin and New Instance Scenarios - Scenarios where an Oracle RAC cluster adds instances that provide a service of interest are supported. The instance may be new to the cluster or may have been restarted after a down event. In both cases, WebLogic Connection Pool for JDBC recognizes the new instance and creates connections to the node as required.

3.2.2 Runtime Connection Load Balancing

WebLogic GridLink Data Sources and JDBC connection pools leverage the load balancing functionality provided by an Oracle RAC database to provide better throughput and more efficient use of resources. Runtime connection load balancing requires the use of an Oracle JDBC driver and an Oracle RAC database.

Oracle performance analysis has revealed significant performance benefits from the use of runtime connection load balancing vs. a static round-robin algorithm. These benefits are observed even when nodes in the RAC cluster are balanced from a hardware perspective, and when the average load on the nodes on the cluster are expected to be reasonably uniform on average. Transient differences in load characteristics are often sufficient to make runtime connection load balancing the optimal load balancing mechanism for RAC clusters.

The load balancing advisory service issues FAN events that advise clients on the current state of the cluster including advice on where to direct connections to. WebLogic Server connection pool receives load balancing advisory events issued by the database, and distributes connections to the RAC nodes accordingly as shown in the diagram below.
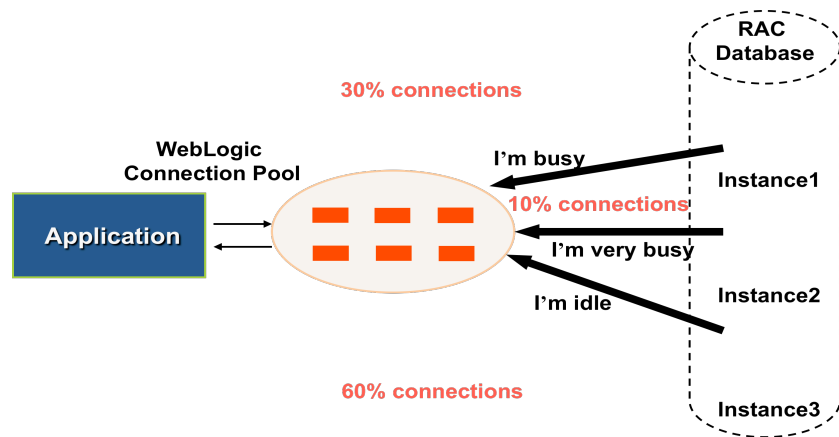
Figure 4

Runtime connection load balancing provides the following benefits:

- Manages pooled connections for high performance and scalability
- Receives continuous recommendations on the percentage of work to route to database instances
- Adjusts distribution of work based on different back-end node capacities such as CPU capacity or response time
- Reacts quickly to changes in cluster reconfiguration, application workload, overworked nodes, or hangs
- Receives metrics from the Oracle RAC Load Balance Advisory. Connections to well performing instances are used most often. New and unused connections to under-performing instances will gravitate away over time. When distribution metrics are not received, connection is selected using a random choice.

3.2.3 Connection Affinity

WebLogic GridLink Data Sources leverage affinity functionality provided by an Oracle RAC database. Connection affinity requires the use of an Oracle JDBC driver and an Oracle RAC database version 11.1.0.6 or higher.

Connection affinity allows a connection pool to select connections that are directed at a specific Oracle RAC instance to provide the best performance for the customer applications. The pool uses run-time connection load balancing to select an Oracle RAC instance to create the first connection and then subsequent connections are created with an affinity to the same instance.

WebLogic GridLink Data Sources supports transaction-based affinity.

Transaction-Based Affinity

Transaction-based affinity is an affinity to an Oracle RAC instance that can be released by either the client application or a failure event. Applications typically use this type of affinity when long-lived affinity to an Oracle RAC instance is desired or when the cost (in terms of performance) of being redirected to a new Oracle RAC instance is high. WebLogic XA connections that are enlisted in a distributed transaction keep an affinity to the Oracle RAC instance for the duration of the transaction. In this case, an application would incur a significant performance cost if a connection is redirect to a different Oracle RAC instance during the distributed transaction.

The affinity will be established based on the global transaction id, instead of by individual data source, to ensure that connections obtained from different data sources that are configured for the same RAC cluster are all associated with the same RAC instance. The LLR two-phase commit optimization will be supported by the RAC data source and will also participate in XA affinity.
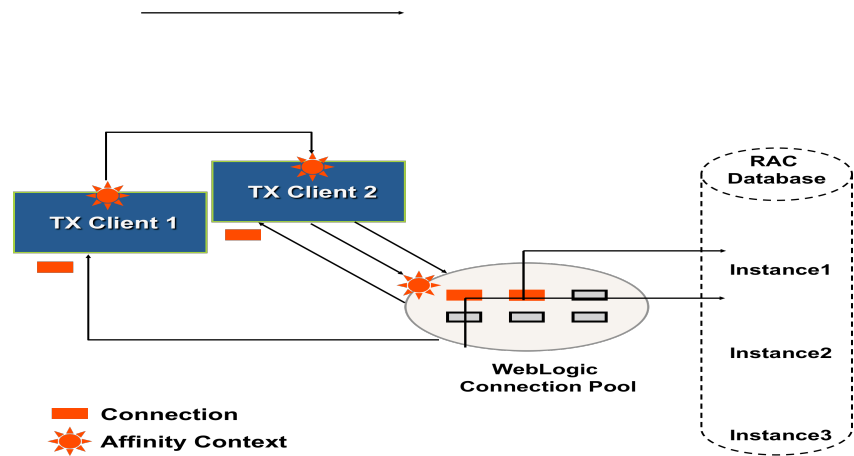

Figure 5

4.0 ORACLE WEBLOGIC SERVER ACTIVE GRIDLINK CONFIGURATION

4.1. Single Client Access Name (SCAN)

Single Client Access Name (SCAN) is a new Oracle Real Application Clusters (RAC) 11g Release 2 feature that provides a single name for clients to access Oracle Databases running in a cluster. The benefit is that the client's connect information does not need to change if you add or remove nodes in the cluster. Having a single name to access the cluster allows clients to use the EZConnect client and the simple JDBC thin URL to access any database running in the cluster, independently of which server(s) in the cluster the database is active. SCAN provides load balancing and failover for client connections to the database. The SCAN works as a cluster alias for databases in the cluster.

EZconnet **sqlplus system/manager@sales1-scan:1521/oltp**
JDBC connect **jdbc:oracle:thin:@sales1-scan:1521/oltp**

Figure 6

WebLogic Server GridLink Data Source supports using SCAN for Data Source configurations.

- Multi Data Source – support SCAN configuration, no connect time listener failover and load balancing. The runtime functionalities are supported by Multi Data Source middle tier implementation only.
- Active GridLink- fully supports SCAN. Below shows a configuration example.

```
ORCLservice =
      (DESCRIPTION =
        (ADDRESS = (PROTOCOL = TCP)(HOST = sales1
                    scan.example.com)(PORT = 1521))
          (CONNECT_DATA =
            SERVER = DEDICATED)
            (SERVICE_NAME = MyORCLservice)
        ))
```
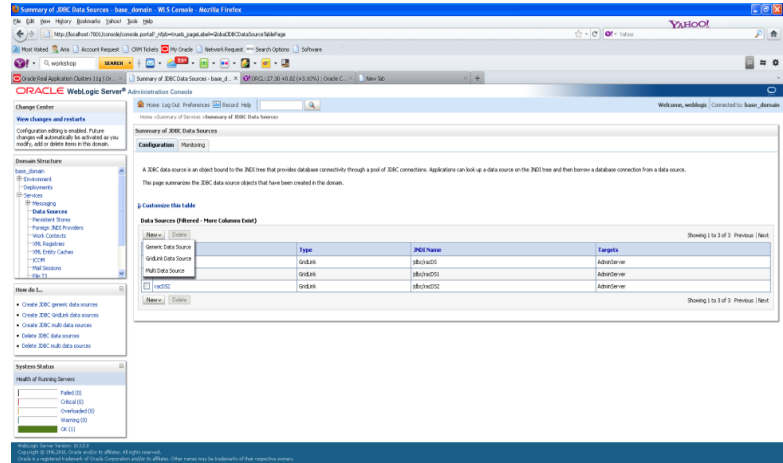
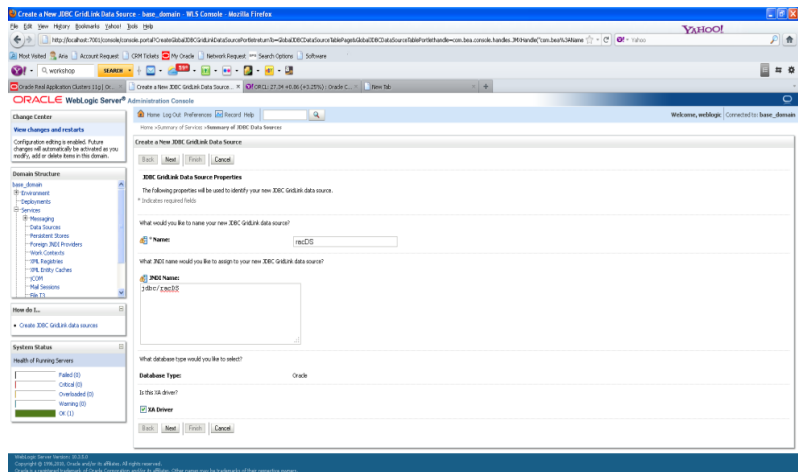Figure 7

4.2. Configuring GridLink Data Source

The Administration Console or the WebLogic Scripting Tool (WLST) can be used for creating a GridLink Data Source.

There are three types of data sources available for the configuration. The generic data source is the implementation for single database access. The multi data source

is the native WebLogic middle tier implementation for Oracle RAC integration which doesn't leverage Oracle Notification Service. The GridLink Data Source is the new Active GridLink implementation which takes advantages of Oracle RAC supporting FCF, RCLB, and Affinities. It supports using SCAN for the configuration.


The GridLink Data Source can be configured simply using a Data Source name and JNDI location as it's shown below.

To be able to leverage Oracle Notification Service, both listener and ONS will need to be configured. The SCAN address could be used for both.
The figure below shows an Example.
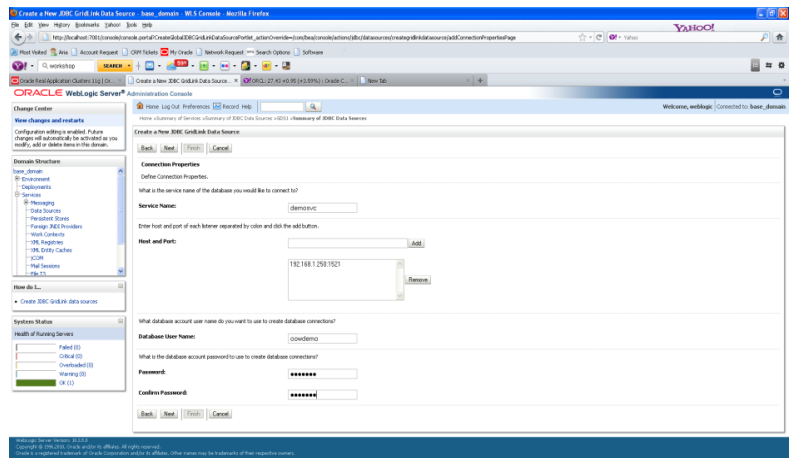
The Service Name is the service created in your RAC database. The Host:Port would be Your_SCAN:1521. The ONS configuration will use the same SCAN address with ONS port, such as Your_SCAN:6200.
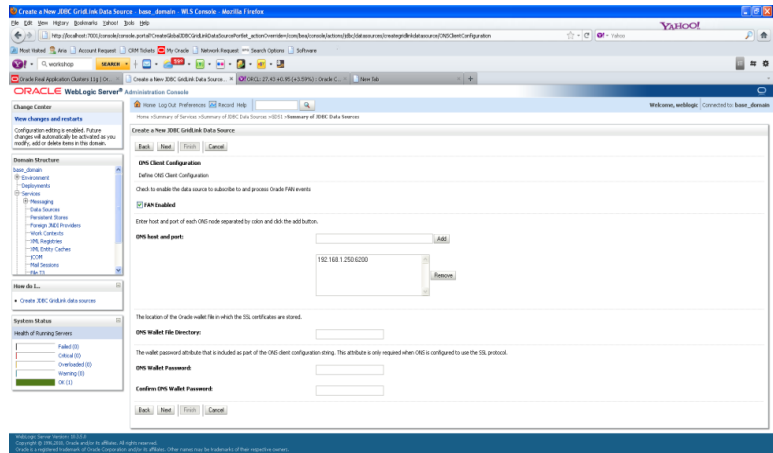

All the Active GridLink performance features, such as Fast Connection Failover, Runtime Connection Load Balancing, Affinities, are enabled by one flag, i.e., 'FAN Enabled'. Different performance or HA features will kick in automatically depending upon the application scenarios. For example, FCF will kick in when a failure case happens. If the application has Global Transaction opened, the XA Transaction Affinity will kick in based on the RCLB event for the first connection. For the Web Session scenarios, Session Affinity will apply first, then RCLB. The goal is to provide the best high availability, scalability and performance solution for the heterogeneous systems and applications.

When the listener and ONS are configured, connections can be tested easily with the WebLogic Server Administration Console.

The figure below shows the listener connection testing.
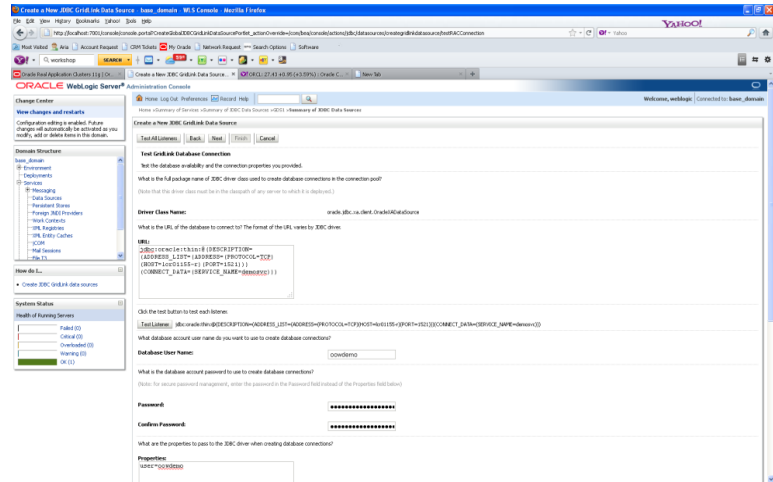
The ONS configuration could be tested as well.
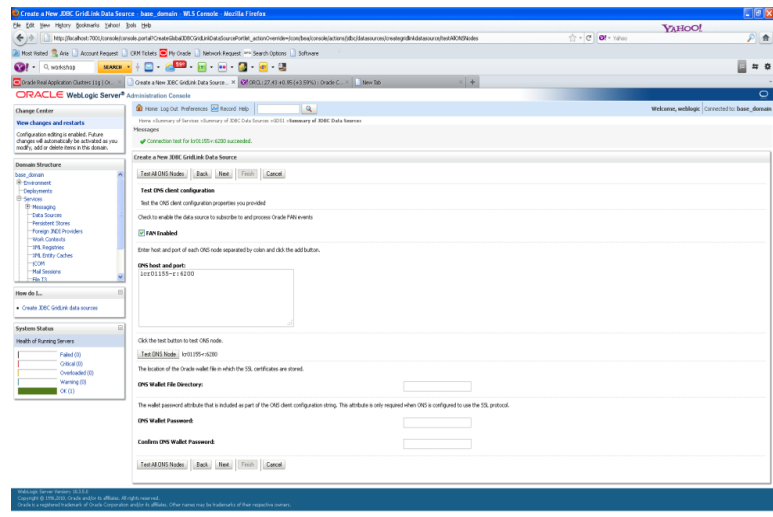
5.0 HIGH AVAILABILITY WITH ORACLE DATA GUARD

Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data. Data Guard provides a comprehensive set of services that create, maintain, manage, and monitor one or more standby databases to enable production Oracle databases to survive disasters and data corruptions. Data Guard maintains these standby databases as copies of the production database. Then, if the production database becomes unavailable because of a planned or an unplanned outage, Data Guard can switch any standby database to the production role, minimizing the downtime associated with the outage. Data Guard can be used with traditional backup, restoration, and cluster techniques to provide a high level of data protection and data availability.

A Data Guard configuration consists of one production database and one or more standby databases. The databases in a Data Guard configuration are connected by Oracle Net and may be dispersed geographically. There are no restrictions on where the databases are located, provided they can communicate with each other. For example, a standby database could be located on the same system as the production database, along with two standby databases on other systems at remote locations.

A high availability architecture requires a fast failover capability for databases and database clients. Client failover encompasses failure notification, stale connection cleanup, and transparent reconnection to the new primary database. Oracle Database provides the capability to integrate database failover with failover procedures that automatically redirect clients to a new primary database within seconds of a database failover.

5.1. WebLogic Active GridLink for RAC and Data Guard

WebLogic Active GridLink fully supports Data Guard with the integration solution implemented with Oracle Universal Connection Pool.
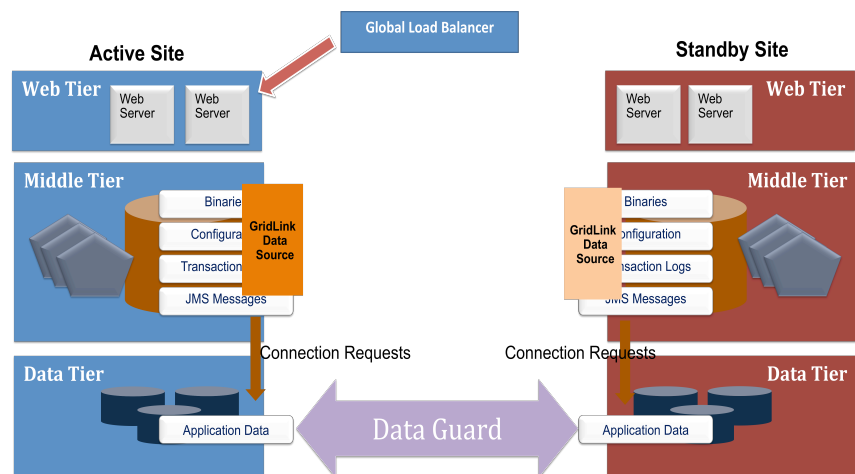

Figure 8

The configuration is done by specifying two SCAN address in the connection url that one represents the primary data center; the other one represents the standby data center. The backend has the knowledge of which address is for the primary and which is for the standby. The failure events are notified to the middle tier via the Oracle Notification Service.

The figure below shows a configuration example:

```
  SALES= (DESCRIPTION_LIST= 12
     (LOAD_BALANCE=off)
     (FAILOVER=on)
     (DESCRIPTION=
(CONNECT_TIMEOUT=5)(TRANSPORT_CONNECT_TIMEOUT=3)
     (RETRY_COUNT=3)
     (ADDRESS_LIST= (LOAD_BALANCE=on)
        (ADDRESS=(PROTOCOL=TCP)
        (HOST=Austin-scan)(PORT=1521)))
     (CONNECT_DATA=(SERVICE_NAME=oltpworkload))) (
        DESCRIPTION=
(CONNECT_TIMEOUT=5)(TRANSPORT_CONNECT_TIMEOUT=3)
        (RETRY_COUNT=3) (ADDRESS_LIST= (LOAD_BALANCE=on)
          ADDRESS=(PROTOCOL=TCP)(HOST= Houston-scan)(PORT=1521)))
     (CONNECT_DATA=(SERVICE_NAME=oltpworkload))))
```

Figure 10

5.2. WebLogic Multi Data Source and Data Guard

WebLogic Multi Data Source supports Oracle Data Guard with the following
limitations.

- Only failover policy is supported.
- Only one RAC instance is allowed in the primary data center.

6.0 HIGH AVAILABILITY WITH ORACLE RAC ONE NODE

Oracle Real Application Clusters One Node is a new option to Oracle Database
11g Release 2 Enterprise Edition. It improves upon many of the benefits of server
Virtualization and extends them to databases running in physical server
environments.
Oracle RAC One Node enables:

- Better server consolidation
- Enhanced protection from failures
- Greater flexibility and workload management
- Better online maintenance

In addition it allows customers to virtualize database storage, standardize their
database environment, and, should the need arise, upgrade to a full multi-node
Oracle RAC database without downtime or disruption. Further, it is fully
compatible with and complementary to Oracle Virtual Machine (VM), allowing
customers to build environments that leverage the strengths of both Oracle RAC
One Node and server virtualization.

WebLogic Server Active GridLink for RAC integrates and fully supports Oracle
RAC One Node.
WebLogic Multi Data Source does not support RAC One Node.

**Oracle WebLogic Server 11g
provides strong support for
Oracle Real Application Clusters
(RAC) features in Oracle
Database 11g, minimizing
database access time while
allowing transparent access to
rich pooling management
functions that maximizes both
connection performance and
availability.**

7.0 SUMMARY

Oracle WebLogic Server and Oracle RAC are designed to work together to provide an environment for highly available and scalable applications. Oracle WebLogic Server Active GridLink for RAC provides the best available support for the Real Application Clusters (RAC) features in Oracle Database 11g, minimizing database access time while allowing transparent access to rich pooling management functions that maximizes both connection performance and availability.

8.0 REFERENCES

How-to Configure Oracle WebLogic Server with GridLink Data Source:

http://www.oracle.com/technetwork/middleware/weblogic/wls-jdbc-gridlink-howto-333331.html

Oracle Database 11gR2 documentation:

http://www.oracle.com/pls/db112/homepage

Oracle WebLogic Server 10.3 documentation:

http://download.oracle.com/docs/cd/E21764_01/wls.htm

Single Client Access Name White Paper:

http://www.oracle.com/technetwork/database/clustering/overview/scan-129069.pdf

Oracle RAC One Node White Paper:

http://www.oracle.com/technetwork/database/clustering/overview/twp-rac1nodev1-1-130698.pdf

ORACLE FUSION MIDDLEWARE