JD Edwards EnterpriseOne 9.1 Clustering Best Practices with Oracle WebLogic Server

An Oracle JD Edwards EnterpriseOne Red Paper

December 2012
PURPOSE STATEMENT AND DISCLAIMER

This document provides considerations when reviewing system performance relative to your JD Edwards EnterpriseOne system. Many factors can influence performance and your results may differ depending on many different variables.

This document is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle. Updates and enhancements are provided in accordance with Oracle’s Technical Support Policies at:

www.oracle.com/support/collateral/oracle-technical-support-policies.pdf
JD Edwards EnterpriseOne 9.1 Clustering Best Practices with Oracle WebLogic Server

OVERVIEW

The purpose of this document is to provide all the necessary steps for JDE EnterpriseOne customers to setup Oracle WebLogic Server Clusters. The clusters can then be used for deployment of the JDE E1 HTML application.

WEBLOGIC DEFINITIONS

The first part of this paper discusses each part of a WebLogic Server used in the setup of the JDE E1 clusters. Full WebLogic documentation can be found using this link:

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

What is a cluster?

A WebLogic Server cluster consists of multiple WebLogic Server instances running simultaneously and working together to provide increased scalability and reliability. A cluster appears to clients to be a single WebLogic Server instance. The server instances that constitute a cluster can run on the same machine, or be located on different machines. You can increase a cluster's capacity by adding additional server instances to the cluster on an existing machine, or you can add machines to the cluster to host the incremental server instances. Each server instance in a cluster must run the same version of WebLogic Server.

What is the difference between a Cluster and a Domain?

A cluster is part of a particular WebLogic Server domain.

A domain is an interrelated set of WebLogic Server resources that are managed as a unit. A domain includes one or more WebLogic Server instances, which can be clustered, non-clustered, or a combination of clustered and non-clustered instances. A domain can include multiple clusters. A domain also contains the application components deployed in the domain, and the resources and services required by those application components and the server instances in the domain.

In each domain, one WebLogic Server instance acts as the Administration Server—the server instance which configures, manages, and monitors all other server instances and resources in the domain. Each Administration Server manages one domain only. If a domain contains multiple clusters, each cluster in the domain has the same Administration Server.

All server instances in a cluster must reside in the same domain; you cannot “split” a cluster over multiple domains. Similarly, you cannot share a configured resource or subsystem between domains.

Clustered WebLogic Server instances behave similarly to non-clustered instances, except that they provide failover and load balancing. The process and tools used to configure clustered WebLogic Server instances are the same as those used to configure non-clustered instances. However, to achieve the load balancing and failover benefits that clustering enables, you must adhere to certain guidelines for cluster configuration.

The Cluster Configuration guidelines can be broken out into two methods. One method is to use a hardware load balancer to achieve load balancing and failover. The second method is to use WebLogic Server functionality itself. This paper discusses the steps necessary to achieve load balancing using WebLogic tools only, not load-balancing hardware. Oracle HTTP Server can also be used as a proxy server in front of WebLogic Clusters to load balance. This paper will discuss the steps necessary to setup Oracle HTTP Server.
With the JDE application deployed to the cluster, testing was done to insure that both the WebLogic and Oracle HTTL load balancing solutions worked. No testing was done for failover for the JDE application.

Full details of WebLogic Server clustering, including the steps necessary to implement clustering with load balancing hardware can be found in the Oracle WebLogic Clustering Guide:

http://docs.oracle.com/cd/E21764_01/web.1111/e13709/toc.htm

**Benefits of Clustering**

A WebLogic Server cluster provides these benefits:

- **Scalability** - The capacity of an application deployed on a WebLogic Server cluster can be increased dynamically to meet demand. You can add server instances to a cluster without interruption of service—the application continues to run without impact to clients and end users.

- **High-Availability** - In a WebLogic Server cluster, application processing can continue when a server instance fails. You “cluster” application components by deploying them on multiple server instances in the cluster—so, if a server instance on which a component is running fails, another server instance on which that component is deployed can continue application processing.

**Vertical and Horizontal WebLogic Clusters**

A “vertical cluster” exists when two or more WebLogic Managed Servers on the same physical machine are grouped together. Vertical clusters can help optimize machine resources, as well as provide protection if another managed server on the same machines is unavailable.

A “horizontal cluster” exists when two more WebLogic Managed Servers on different physical machines are grouped together. Horizontal clusters can help optimize machine and networks resources, and also provides failover protection should other cluster members are unavailable.

**WebLogic Administration Server and Administration Console**

The Administration Server is a special instance of a WebLogic Server in each domain. The Administration Server provides a central point for managing a WebLogic Server domain.

The Administration Console is a Web application hosted by the Administration Server. It can be used to manage and monitor an active domain. The management capabilities include:

- Configuring active domains
- Stopping and starting servers
- Monitoring server health and performance
- Monitoring application performance
- Viewing server logs

**Machines and NodeManager**

A “machine” in WebLogic is a logical representation of the physical machine which hosts a WebLogic server instance. Each Managed Server in a WebLogic Cluster must be assigned to the appropriate machine where the Manager Server is meant to be run. To setup a vertical cluster, the Managed Servers in the cluster need to be assigned to run on the same machine. For horizontal clusters, the Managed Servers need to be assigned to separate machines.

WebLogic Node Manager is used when setting up Managed Servers in a horizontal cluster configuration. “Node Manager” is a WebLogic Server utility that enables you to start, shut down, and restart Administration Server and Managed Server instances from a remote location. A Node Manager process is not associated with
a specific WebLogic domain but with a machine. You can use the same Node Manager process to control server instances in any WebLogic Server domain, as long as the server instances reside on the same machine as the Node Manager process. Node Manager must run on each computer that hosts WebLogic Server instances—whether Administration Server or Managed Server—that you want to control with Node Manager.

**OVERVIEW OF INSTALL**

For this paper, clustering the JDE E1 HTML application with WebLogic Server was accomplished using the following machines.

**dnptlx02**
- Red Hat Enterprise Linux 5.6
- Server Manager Agent 9.1.2
- Oracle WebLogic Server 10.3.5
  - Domain E1_91
    - Administration Server
    - Cluster C1
      - Managed Server C1_VC1
      - Managed Server C1_VC2
  - Node Manager for machine dnptlx02
- Oracle HTTP Server 11.1.1.5 (used for proxy server in front of WebLogic Cluster)

**dnptlx05**
- Red Hat Enterprise Linux 5.6
- Oracle WebLogic Server 10.3.5
  - Domain E1_91
    - Cluster C1
      - Managed Server C1_HC1
  - Node Manager for machine dnptlx05

**coelab2**
- Windows Server 2008 R2 64bit
- Server Manager Console 9.1.2
  - Managed Instance for Oracle WebLogic
  - Managed Instance for E1_C1_JAS (HTML Server deployment)

**JDE Instance details**
- JDE Application Release 9.1 with Tools Release 9.1.2
OVERVIEW OF VERTICAL CLUSTER SETUP

Assumptions

- Oracle WebLogic 10.3.5 64bit is installed on dnptlx02.

Note: Oracle WebLogic Server is offered in several editions, each of which grants license to different levels of functionality. Consult the Oracle Fusion Middleware 11g Licensing Information guide for details about the various editions. The systems described in this document were configured using Oracle WebLogic Server Enterprise Edition.

- Oracle JDE Server Manager Console 9.1.2 is installed and running on coelab2.
- Oracle JDE Server Manager Agent 9.1.2 is installed and running on dnptlx102.
- WebLogic NodeManager on dnptlx102 is up and running.
- JDE EnterpriseOne services are up and running in order to validate the HTML signon with JDE Security Server.
- The person implementing these steps has a thorough understanding of JDE installation, including Server Manager, and WebLogic 11G.

The Vertical Cluster setup will consist of two managed servers running on the same machine. The steps used to set this up are

- Install Oracle WebLogic 10.3.5 64bit on machine dnptlx102
- Create domain E1_91 in WebLogic
- Create machine dnptlx102 in E191 domain
- Create the cluster C1 with the Managed Servers C1_VC1
- Deploy E1 JAS HTML Server to the cluster
- Create Managed Server C1_VC2 and add to cluster C1
- Edit Startup values for the Managed Servers.
- Configure all the JAS settings for E1 JAS HTML Server
- Start the E1 JAS Instance
- Setup HTTP Proxy for load balancing
  - Create Managed Server using WebLogic HTTPClusterServlet, or
  - Install Oracle HTTP Server and configure proxy

For this paper, WebLogic was installed on Linux platform, so all commands and screenshots will be from Linux.

DETAILED SETPS TO SETUP VERTICAL CLUSTER

Create the E1_91 WebLogic Domain

1. As the Linux user who installed WebLogic, run config.sh from WLS_HOME/wls-server_10.3/common/bin. Be sure to set DISPLAY environment variable to a suitable Xwindows target if you are not running the config.sh on the Linux console desktop
2. Select “Create a new WebLogic domain” and click Next
3. Leave the first option “Generate a domain…” selected and click Next.
4. Change the Domain Name to **E1_91** and click **Next**
5. Keep “weblogic” and the User Name and type “admin123” as the user password and confirm the password as well. Click **Next**.
6. Select “Production Mode” for Startup Mode and “Oracle jRockit” for the jdk. Click Next.
7. Include the Administration Server and click **Next**. We will create the cluster, managed servers and machine using the Administration Console even though you can also create using this config.sh program.
8. Change the listen port to an open port and click **Next**. In this example the port is **7501**.
Click **Create** to create the domain.
10. Upon completion you should see the confirmation screen. Take note of the URL for the Administration Console, in this case http://dnptlx102.mlab.jdedwards.com:7501
Start the Administration Server.

On the linux command line in a telnet session on dnptlx102, perform these commands to start the server:

```
Change directories to the domain location
   cd /u01/Oracle/Middleware/user_projects/domains/E1_91
Start the server
   ./startWebLogic.sh
Enter user “weblogic” and password “admin123” when prompted.
```

Create the cluster

1. Log into the Admin Console URL
   http://dnptlx102.mlab.jdedwards.com:7501/console/login/LoginForm.jsp using weblogic/admin123 credentials. You should see something similar to this...
2. Click on Environment, Clusters, Lock & Edit, New. The screen should look like this:

![Cluster creation screen](image)

3. Change the name to C1. The Messaging Mode should be kept as **Unicast**.

   **Note:** Clusters use messaging for sharing session, load balancing and failover, JMS, and other information between cluster members. Clusters can use either Unicast or Multicast messaging. Multicast is a simple broadcast technology that enables multiple applications to subscribe to a given IP address and port number and listen for messages, but requires hardware configuration and support. Unicast does not have these requirements and is recommended for new clusters.

Keep all other options as the default and click **OK**, then click **Activate Changes**. You should see this:
4. Create the machine.
   a. Click on Environment, Machines, Lock & Edit, New
   b. Change machine name to `dnptlx102` and OS to Unix and click Next.
   c. Change the Listen Address to `dnptlx102`. This step becomes important for remotely managing managed servers that exist in WebLogic Domains that span multiple machines.
   d. Click Finish and then Activate Changes.
   e. Screen should look similar to this:

5. Create the Vertical Cluster Managed Server
   a. Click on Environment, Servers, Lock & Edit, New
   b. Change name to `C1_VC1`, change the Server listen port to 7503, select Yes, make this server a member of an existing cluster and select C1, then click Finish. You should see something like this:
c. Click on the server C1_VC1 and change the machine to dnptlx102.
d. Click Save, and then Activate Changes.
e. You should see something similar to this

---

Note: You can access this same screen a number of couple of different ways in the Admin Console. One way is to click on Environments, Clusters, click on C1, then click on the Servers tab, then click on the server name. Another way is to click on Environments, Servers, then click on the server name.

6. Deploy E1 JAS HTML Server to the cluster
   a. Make sure the Server Manager agent is started on dnptlx102.
b. In Server Manager Console on machine coelab2, register Oracle WebLogic Server 11G as a managed instance on dnptlx102. The screen below shows the instance name as **WLS_1035**, the install location as **/u01/Oracle/Middleware**, and uses the same user name and password as when the domain E1_91 was created.

![Create/Register A Managed Instance](image)

```
WLS_1035
Oracle Weblogic Home /u01/Oracle/Middleware
Oracle Weblogic Admin User weblogic
Oracle Weblogic Admin Password ********
```

![Oracle WebLogic Server 11g](image)

c. When clicking on the new managed instance, you should see something similar to the screen below. Note the E1_91 domain with the one managed server C1_VC1 in the cluster C1. The Administration Server is also in the E1_91 domain and it is shown as running because the startWebLogic.sh command was issued after the domain was created.

d. In Server Manager Console, create a new managed instance on dnptlx102 managed home for the E1JAS HTML application:
   - Upload the latest 9.1.2 version of the JDE E1 HTML client and distribute it to dnptlx102.
   - Create a new managed instance called **E1_JAS_C1** under the managed home for dnptlx102, using the j2ee server for C1_VC1, port 7503 and the software component loaded in the previous step. Click **Continue**.
Enter appropriate information for the JDE instance and click **Continue** then confirm the action by clicking **Create Instance**.

- The E1 HTML code will be deployed to the C1_VC1 server.

- A warning message will appear concerning the web server configuration files are out of date on the remote server and they need to be synchronized. The configuration files (jas.ini, jdbj.ini, jdelog.properties, etc) are copied from the server manager agent folders to the classes folder where the HTML application code resides when the Synchronize Configuration button is clicked. These actual locations are discussed in step 9. The Synchronization will occur at a later step, so no need to click the button at this time.

- Once the HTML server is deployed, the server will look like this
7. Create Managed Server C1_VC2 and add to cluster C1
   a. Back in the WebLogic Admin Console, click on Environment, Servers, Lock & Edit, New
   b. Change name to C1_VC2, change the Server listen port to 7505, select Yes, make this server a member of an existing cluster and select C1, then click Finish.
   c. Click on the server C1_VC2 and change the machine to dpntlx102.
   d. Click Save, then Activate Changes.
   e. You should see something similar to this:

8. Edit Startup values for the Managed Servers
The startup values for the servers need to be changed in order to uniquely identify each cluster member for Server Manager. The unique identification is used to separate the cluster members for both log separation and runtime metrics displays within Server Manager Console.

a. In WebLogic Admin Console, click on Environment, Servers, C1_VC1, Lock & Edit.
b. Click on the Server Start tab, and place the following in Arguments box:
   -Xms32m -Xmx768m -DcloneId=C1_VC1

c. Click Save and Activate Changes.
d. Repeat the same steps for C1_VC2, using
   -Xms32m -Xmx768m -DcloneId=C1_VC2

Note: The –DcloneId is case-sensitive. If there are any errors or special characters in the Arguments, the server will not start. If the server is not starting then it is recommended to delete the values for startup and retype them in the Arguments box.

9. Configure the JAS settings for E1 JAS HTML Server

   a. Using the Server Manager console, edit your Configuration items for the EnterpriseOne HTML Server E1_JAS_C1 managed instance. Make sure all the correct information is entered for Network Settings, JDBJ Database Configuration, etc. and save the changes.

Since the current configuration is connecting to an Oracle Database, appropriate database connect strings must be entered in the tnsnames.ora section of the JDBJ Database Configuration. Also, the fully qualified path to the tnsnames.ora file must be removed. This step is necessary in order for the jdbj.ini file to find the proper location to the tnsnames.ora file after the synchronization of the configuration files occurs.

When the configurations are saved within Server Manager, the files are stored in

<SM Agent Home>/targets/E1_JAS_C1/config

b. Click on the Managed Instance Name E1_JAS_C1 in Server Manager, and then click on the Synchronize Configuration button. The synchronization process will take a few minutes to complete. The configuration files are copied to a location within the Server Manager Agent folders, as well as a location within the WebLogic folders.

For Server Manager instance E1_JAS_C1, the configuration files are found in the classes folder in

<SMAgentHome>/targets/E1_JAS_C1/owl_deployment/webclient.ear/app/webclient.war/WEB-INF

For each managed server, there is another copy of the deployed code because the WebLogic deployment method being used is the staged method, meaning the code is copied to a staged area for each server. After synchronization, the configuration files for the WebLogic deployment area are found in the classes folder in

<WLHOME>/user_projects/domains/E1_91/servers/C1_VC1/stage/E1_JAS_C1/app/webclient.war/WEB-INF and
10. Start the E1 JAS Instance

   a. The first time the E1_JAS_C1 managed instance is started within Server Manager, the WebLogic servers need to be started instead of the actual HTML Instance. Subsequent Stop/Start requests can be handled directly by the E1_JAS_C1 instance.

   b. In Server Manager, navigate to your WebLogic managed instance WLS_1035, locate your two managed server names, check the boxes to the left of the names, and then click Start.

Once the servers have started, then the E1_JAS_C1 will be started and the JDE HTML Application server can be accessed on these two URLs:

   http://dnptlx102:7503/jde/owhtml and
   http://dnptlx102:7505/jde/owhtml

   It is good idea to test these URLs to make sure the E1 HTML application is accessible. However, the URLs should not be used by actual users. The HTTP Proxy (discussed later) will be used to setup load balancing to have all users access the HTML application.

   The URL on the Server Manager managed instance page only points to the port where the HTML application was initially deployed.

   The servers can be stopped and started successfully using both Server Manager and the WebLogic Console.

   In Server Manager, successfully started E1 HTML Instance will look like this
11. Setup HTTP Proxy for load balancing

There are a couple of ways to load balance the connections to the WebLogic clustered servers.

One way involves using the WebLogic-supplied proxy plug-in HTTPClusterServlet and creating an additional WebLogic managed server to proxy the connections. The details for this method can be found in the WebLogic Cluster Guide found here:

http://docs.oracle.com/cd/E21764_01/web.1111/e13709/setup.htm#CLUST466

Another way to load balance the connections is using Oracle HTTP server in front the WebLogic server. The Oracle HTTP server can be installed on the same server as WebLogic or a separate server altogether, allowing more flexibility in your overall architecture.

There is a specific module in Oracle HTTP Server called mod_wl_ohs which allows requests to be proxied from Oracle HTTP Server to Oracle WebLogic Server. The module is documented in the Oracle Guide accessible here:

http://docs.oracle.com/cd/E21764_01/web.1111/e10144/getstart.htm#BEHGIDCB

Use the following steps to setup mod_wl_ohs to load balance using the Oracle HTTP server:

a. The first thing to check is if the WebLogic domain containing the cluster includes JRF (Java Runtime Files). The domain must include these files in order for this method to work properly.

   Try to extend the domain using config.sh (the same command used to create the domain). Choose to extend an existing domain and click **Next**.
Select your domain and click **Next**.
If the option to add Oracle JRF is available and unchecked then check the box and run through the steps accepting the defaults to extend the domain and then skip to the below Install Oracle 11G WebTier section. If you have a domain that spans machines (as in Horizontal Cluster) then both installs on each machine need the JRF files.

If the JRF files do not show up as an option, they need to be installed to the WebLogic Home folder. To do this, download the file from Oracle Fusion Middleware downloads:


The file is found under Application Development Runtime download.

- Extract the jar file and run the RunInstaller program in the Disk1 folder and install the files to your WebLogic Home folder.
- Restart Admin Console in the domain, then follow Step 1 above to extend the domain.

b. Install Oracle 11G WebTier which includes the Apache-based Oracle HTTP Server.

- Obtain the software from Oracle Technology Network (http://technet.oracle.com) or the Oracle Software Delivery Cloud (http://edelivery.oracle.com).
- Expand the downloaded file.
- Run the runInstaller program in the Disk1 folder and click Next.
Select Skip Software Updates and click Next.

Select Install and Configure and click Next.
- Click **Next** after the Prerequisite Check.

- Keep that defaults for locations and click **Next**.
- Uncheck the box regarding security updates and click **Next**. Click **Yes** on the pop up confirming your choice if prompted.

- Uncheck the box for **Oracle Web Cache** and click **Next**.
- Enter your WebLogic domain information where your cluster is located and click Next. If the domain is reported not to have the JRF files, restart the Administration Server for the domain and try again.

- Keep the defaults for Specify Component Details and click Next.

- Keep the defaults for Configure Ports and click Next.
Click **Install** on Installation Summary.

Click **Next** and then **Finish** on the Installation Complete screen. The OHS instance should be started and you should now be able to access http://hostname:7777 Fusion Middleware page. The port may not be 7777 depending on the other installations on the machine. In the example for this paper the port is 7778.

Edit the file `mod_wl_ohs.conf` found in `BEA_HOME/instances/<Instance Name>/config/OSH/<ohs name>`
Add this parameter substituting your server names and ports in your cluster.

- `<Location /jde>`
  - SetHandler weblogic-handler
  - WebLogicCluster dnptlx102:7503,dnptlx102:7505
- `</Location>`

Restart Oracle HTTP Server by command line or shortcuts from Program Files (if you are on Windows). Use these commands to stop and start from the command line:

- `$ORACLE_INSTANCE/bin/opmnctl stopproc ias-component=ohs1`
- `$ORACLE_INSTANCE/bin/opmnctl startproc ias-component=ohs1`

You should now be able to access http://hostname:7778/jde and Oracle HTTP Server will load balance the connections to the servers in your cluster.

After logging into http://dnptlx102:7778/jde two times in separate browsers, you will see in the Server Manager Console one user signed into each cluster member. Note also the log separation of the two cluster members.
When drilling down into the Runtime Metrics, information is also separated by cluster member.
OVERVIEW OF HORIZONTAL CLUSTER SETUP

The vertical cluster is now setup with two managed servers running on the same physical machine. The next section will discuss what is necessary to add another managed server to the same cluster, with the new managed server running on a separate physical machine. When this is complete we will have a horizontal cluster in place.

Assumptions

- Oracle WebLogic 10.3.5 64bit is installed on dnptlx105. WebLogic must be installed in the same directory structure on each machine. This is mandatory, and if there is any deviation in the install location between the two machines then the clustered managed server will not work on dnptlx105.

  **Note:** Oracle WebLogic Server is offered in several editions, each of which grants license to different levels of functionality. Consult the Oracle Fusion Middleware 11g Licensing Information guide for details about the various editions. The systems described in this document were configured using Oracle WebLogic Server Enterprise Edition.

- Oracle WebLogic 10.3.5 is installed and running on dnptlx102. The Admin Console is running on port 7501. The two vertical managed servers discussed in the previous section are installed.
- Oracle JDE Server Manager Console 9.1.2 is installed on coelab2
- Oracle JDE Server Manager Agent 9.12 is installed on dnptlx102
- WebLogic NodeManager on dnptlx102 is up and running
- WebLogic NodeManager on dnptlx105 is up and running
- JDE EnterpriseOne services are up and running in order to validate the HTML signon with JDE Security Server
- E1_JAS_C1 managed instance is stopped in Server Manager
- The person implementing these steps has a thorough understanding of JDE installation, including Server Manager, and WebLogic 11G.

The steps used to set this up are

- Install Oracle WebLogic 10.3.5 64bit on machine dnptlx105. WebLogic must be installed in the same directory structure as installed on dnptlx102. This is mandatory, and if there is any deviation in the install location between the two machines then the clustered managed server will not work on dnptlx105.

  **Note - The JRF Files must be installed in the WebLogic Home as well (see step 11a above regarding the “Setup HTTP Proxy for load balancing”).**

  **Note – The NodeManager must be started on the horizontal cluster machine.**

- Create machine dnptlx105 in E1900 domain using Admin Console
- Create Managed Server C1_HC1 and add to the cluster C1
- Edit Startup values for the Managed Server C1_HC1
- Create the domain E1_91 on dnptlx105
- Start the E1 JAS Instance
- Setup HTTP Proxy for load balancing

DETAILED STEPS TO SETUP HORIZONTAL CLUSTER

When using these steps to setup the horizontal cluster, it is mandatory that the same release or Oracle WebLogic is installed on both physical machines involved in the cluster. Also, the machines where WebLogic is installed
should share the same hardware and operating system characteristics. While it is possible to spread the cluster across heterogeneous machines, it is recommended that all machines are similar. In the setup described in this paper, both machines are similar RedHat Linux machines.

1. Create machine dnptlx105 in E1900 domain using Admin Console

The following step is important because the machines in WebLogic are used to direct the Admin Console to the NodeManager process on the appropriate machine in order for administration of the cluster and the cluster members to happen properly.

a. Log into WebLogic Admin Console running on dnptlx102.

b. Click on Environment, Machines, Lock & Edit, New

c. Change machine name to dnptlx105 and OS to Unix and click OK.

d. Click on dnptlx105, click on the Node Manager tab and change the Listen Address to dnptlx105. This step becomes important for remotely managing managed servers that exist in WebLogic Domains that span multiple machines.

e. Click Save and Activate Changes.

f. Screen should look similar to this

![Settings for dnptlx105](image)

2. Create Managed Server C1_HC1 and add to the cluster C1

a. In WebLogic Admin Console, Click on Environment, Servers, Lock & Edit, New

b. Change name to C1_HC1, change the Server listen port to 7507, select Yes, make this server a member of an existing cluster and select C1, then click Finish.
c. Click on the server C1_HC1 and select the machine dnptlx105.
d. Click Save, then Activate Changes.
e. You should see something similar to this

3. Edit Startup values for the Managed Server C1_HC1.
a. In WebLogic Admin Console, click on Environment, Servers, C1_HC1, Lock & Edit.
b. Click on the Server Start tab, and place the following in Arguments box:
   -Xms32m -Xmx768m -DcloneId=C1_HC1

   The –DcloneId is case-sensitive. If there are any errors or special characters in the Arguments, the server will not start. If the server is not starting then it is recommended to delete the values for startup and retype them in the Arguments box.

c. Click Save and Activate Changes.

4. Create the domain E1_91 on dnptlx105

Now that the steps to create the cluster are complete on dnptlx102, the next thing that has to happen is to distribute the domain to dnptlx105. We will use the pack and unpack commands to accomplish the distribution.

a. In a telnet session on dnptlx102, change directories to the WL_HOME/common/bin folder

   cd /u01/Oracle/Middleware/wlserver_10.3/common/bin

b. Pack the domain into a Jar file. There are no line breaks in the following command. Create the user_templates folder if it does not exist on dnptlx102.

   ./pack.sh -managed=true -domain=/u01/Oracle/Middleware/user_projects/domains/E1_91 -template=/u01/Oracle/Middleware/user_templates/E1_91.jar -template_name="E1_91"

   [oracle@dnptlx102 bin]$ ./pack.sh -managed=true -domain=/u01/Oracle/Middleware/user_projects/domains/E1_91 -template=/u01/Oracle/Middleware/user_templates/E1_91.jar -template_name="E1_91"
   ✓ read domain from "/u01/Oracle/Middleware/user_projects/domains/E1_91"
   ✓ succeed: read domain from "/u01/Oracle/Middleware/user_projects/domains/E1_91"
   ← set config option Managed to "true"
   ✓ succeed: set config option Managed to "true"
   ← write template to "/u01/Oracle/Middleware/user_templates/E1_91.jar"
   ........................................................................................................................................
   ✓ succeed: write template to "/u01/Oracle/Middleware/user_templates/E1_91.jar"
   ✓ succeed: close template
   [oracle@dnptlx102 bin]$ 

c. ftp the /u01/Oracle/Middleware/user_templates/E1_91.jar file to dnptlx105 machine to the same location /u01/Oracle/Middleware/user_templates. Create the user_templates folder if it does not exist on dnptlx105.

d. Unpack the jar file on dnptlx105 using a telnet session on dnptlx105.

   Change directory to common/bin in WL_HOME

   cd /u01/Oracle/Middleware/wlserver_10.3/common/bin

   Unpack the JAS.jar file. There are no line breaks in the following command

   ./unpack.sh -domain=/u01/Oracle/Middleware/user_projects/domains/E1_91 -template=/u01/Oracle/Middleware/user_templates/E1_91.jar
5. Start the E1 JAS managed Instance using Server Manager Console.

a. Since we added another server to the C1 Cluster, the WebLogic servers again need to be started instead of the actual HTML Instance. Subsequent Stop/Start requests can be handled directly by the E1_JAS_C1 instance.

b. In Server Manager, navigate to your WebLogic managed instance WLS_1035, locate the three managed server names, check the boxes to the left of the names, and then click Start.

Upon starting the E1_JAS_C1 managed instance, all the servers in the C1 will be started and become available for service. The vertical cluster members will be started and accessible as before using the direct URL. The new cluster member running on dnptlx105 will be accessible via this URL.

http://dnptlx105:7507/jde

The configuration files (jas.ini, jdbj.ini, jdlogs.properties, etc) for the server C1_HC1 will be located on the dnptlx105 machine in the following location

<WLHOME>/user_projects/domains/E1_91/servers/C1_HC1/stage/E1_JAS_C1/app/webclient.war/WEB-INF/classes

If a change is made to the configuration files using Server Manager Console, the Synchronize Configuration warning will again appear, and when this process is completed all the configuration files will be copied over to the same classes folder on dnptlx105.

The tnsnames.ora file does not get copied to the above folder where the other configuration files are located. The tnsnames.ora file must be manually copied to dnptlx105, in the same folder structure where it is located on dnptlx102. This will be fixed in a later release of Server Manager.
In the meantime, create the server manager agent target config folder on dnptlx105 in <Server Manager Home>/targets/E1_JAS_C1/config and copy the trsnames.ora file from dbptlx102 to dnptlx105 in the newly created folder. The target folder on dnptlx105 is /u01/Oracle/Oracle_SMA/SCFHA/targets/E1_JAS_C1/config

```bash
[oracle@dnptlx105 config]$ pwd
/u01/Oracle/Oracle_SMA/SCFHA/targets/E1_JAS_C1/config
[oracle@dnptlx105 config]$ ls -altg
total 12
drwxr-xr-x 2 oinstall 4096 Nov 5 07:33 .
-rw-r--r-- 1 oinstall 130 Nov 5 07:33 trsnames.ora
drwxr-x--- 4 oinstall 4096 Nov 5 07:32 ...
```

When the release level of the HTML server is changed via Server Manager Console, and the HTML managed instance is deployed to a cluster, upon deployment of the new HTML release all of the managed servers in the cluster will be updated.

6. Notes about Server Manager and Horizontal Cluster Members

Runtime Metrics

Server Manager Console receives instance information and runtime metrics from the Server Manager Agents running on each of the managed home machines. For the vertical cluster members, all metrics are transferred successfully due to the fact that we have an agent running on dnptlx102.

However, no Server Manager agent has been installed on dnptlx105 (the machine containing the horizontal cluster member) which means no data will be reported to the Server Manager Console. We can correct this situation by simply copying 1 file from dnptlx102 to dnptlx105.

The file is “agent.properties”, and is located in <Server Manager Home>/config on dnptlx102. The file contains the name of the Server Manager Console Machine, and Management Server JMX port, which is the port the Manager Server listens to for incoming connections.

By copying this agent.properties file to the machine hosting the horizontal cluster member, we are sending data to the server manager on the port listed and therefore the information will be displayed in Server Manager console.

The file must be in the same physical location on each of the two machines, and it must be owned by the same user account that runs WebLogic.

Action ➔ Create the file structure on your horizontal machine and copy the properties file to that location.

After the horizontal server install, Server Manager will display this:
Logs

One drawback for horizontal cluster members is that the logs from Managed Server running on the horizontal cluster machines will not be brought into the Server Manager console for viewing. The functionality of displaying logs is performed by the Server Manager Agent, and since there is no additional agent installed on the horizontal cluster machine, no logs will be available in Server Manager Console. To view these logs, simply log into the machine as the user running WebLogic, change directories to the log location (as specified by jdelog.properties file), and open in an editor. Functionality to view horizontal cluster member logs will be included in a future release of Server Manager.

The screen below shows that only the Vertical Member logs are displayed.

JDBC Drivers

Proper JDBC drivers must be installed on all physical machines that host managed servers in the cluster. The recommended method of installing JDBC drivers is via Server Manager Console. Currently, this method requires a Server Manager Agent running on the target machine for successful JDBC file copy.
Similar to the situation with logs, the JDBC drivers will not be copied to machines which host horizontal cluster members because there is no Server Manager Agent running on that machine.

The workaround is to manually copy the required JDBC file to the horizontal cluster machine. The file must be identical to the file on the vertical cluster machine, owned by the user running WebLogic, and placed in the same folder structure as exists on the vertical cluster machine.

The proper folder to install the JDBC file is

<Server Manager Home>/targets/<WebLogic Managed Instance>/config/jdbc/<WLS Domain Name>/<WLS Managed Server Name>/

Action ➔ Create the folder above on the horizontal cluster machine, then copy the JDBC file to that folder.

7. Setup HTTP Proxy for load balancing

The Oracle HTTP Server was setup to load balance the two vertical servers in step 11 of Setting up Vertical clusters. The configuration needs to be modified so that the new horizontal server is included in the rotation. The following steps are used to accomplish adding the horizontal server.

   a. Stop the Oracle HTTP Server on dnptlx102 using the command line or shortcuts from Program Files (if you are on Windows).

   b. Edit the file mod_wl_ohs.conf found in BEA_HOME/instances/<Instance Name>/config/OSH/<ohs name>

   c. Add the horizontal server name and port in your cluster

       <Location /jde>
           SetHandler weblogic-handler
           WebLogicCluster dnptlx102:7503,dnptlx102:7505,dnptlx105:7507
           DynamicServerList OFF
       </Location>

   d. Start the Oracle HTTP Server by command line or shortcuts from Program Files (if you are on Windows).

   e. You should now be able to access http://dnptlx102:7778/jde and Oracle HTTP Server will load balance the connections to the servers in your cluster.