Oracle WebLogic server Administration

HI N Technologies

2014

HI N Technologies
Flat #402/20,16th A Main, 13th Cross,
Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com
www.weblogic4you.blogspot.com

Weblogic Server Admin Guide

By
Harish

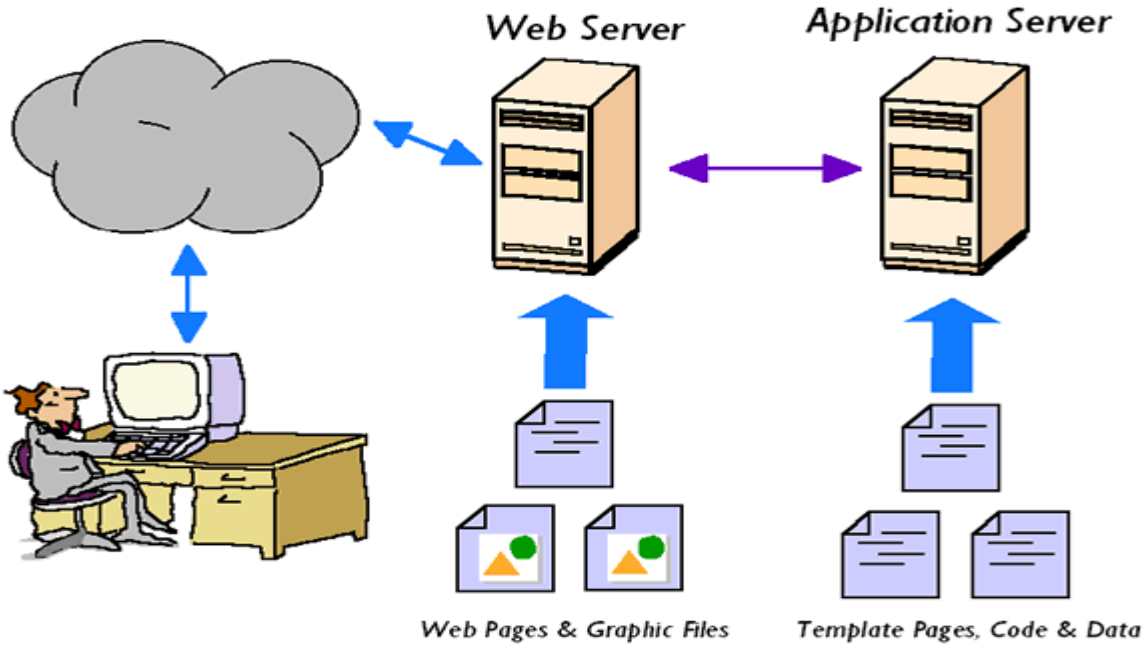
INDEX

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

APPLICATION SERVER:

Application server can expose the dynamic content along with the business logic.

The application server is the middleman between **browser-based front-ends** and **back-end databases** and legacy systems.An application server is a component-based product that resides in the middle-tier of a server centric architecture. It provides middleware services for security and state maintenance, along with data access and persistence. Java application servers are based on the Java™ 2 Platform, Enterprise Edition (J2EE™).

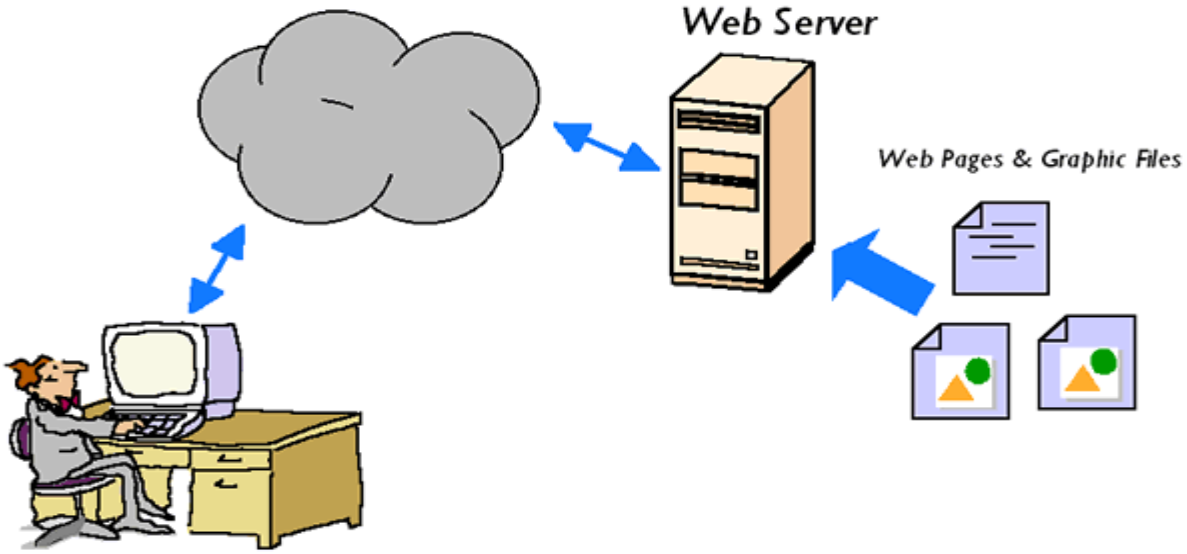


Before 1990's we don't have Application server, we have only webserver was there so that time we use only webserver to do all (ecommerce transactions).
Web server alone provides the online store's functionality. The Web server takes your request, then passes it to a server-side program able to handle the request. The server-side program looks up the pricing information from a database or a flat file. Once retrieved, the server-side program uses the information to formulate the HTML response, and then the Web server sends it back to your Web browser.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

WEBSERVER:

A Web server exclusively handles HTTP/HTTPS requests. It serves content to the web using HTTP/HTTPS protocol.
A webserver you can say it's a computer program or software that serves the webpage (static/dynamic) using http over the world
Web Server is designed to serve HTTP Content.
A Web server handles the HTTP protocol. When the Web server receives an HTTP request, it responds with an HTTP response, such as sending back an HTML page.



Diff b/w Application Server and Webserver:

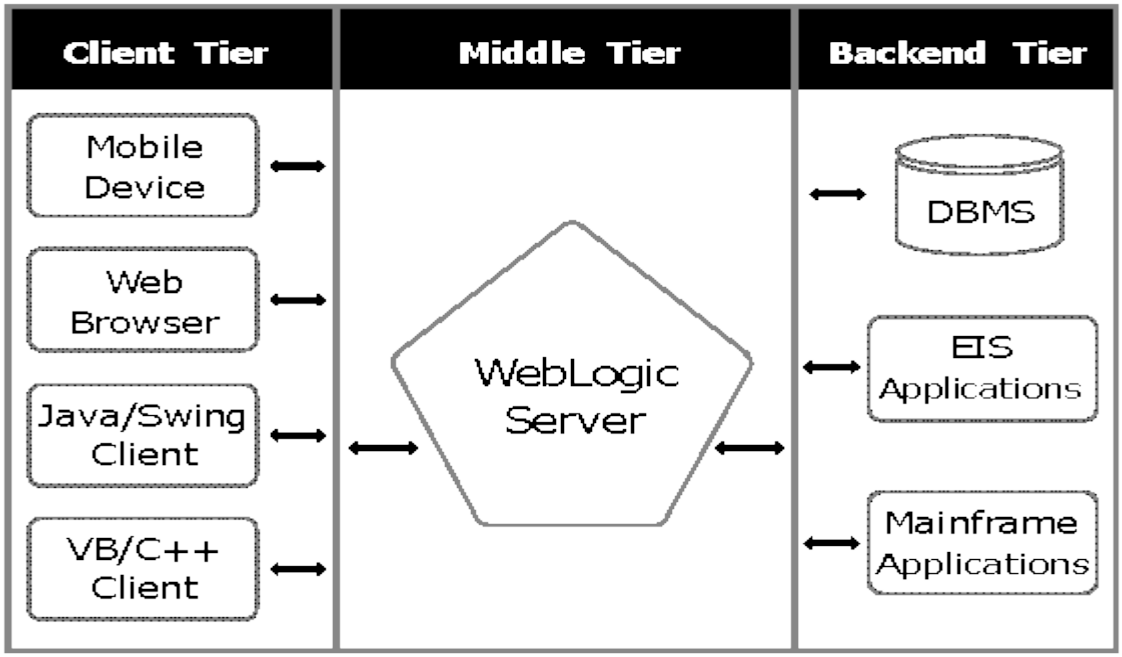
Webserver	Application Server
➢ A webserver you can say it's a computer program or software that serves the webpage (static/dynamic) using http over the world. ➢ Web Server is designed to serve HTTP Content. ➢ A Web server handles the HTTP protocol. ➢ When the Web server receives an HTTP request,it responds with an HTTP response, such as sending back an HTML page. **Ex:** Apache	➢ Application server is much more than serving static/dynamic webpages. ➢ It provides better performance tuning option. ➢ It provides more security options. ➢ It provides scalability high availability which webserver doesn't provide. **Ex**:Weblogic server,wespeare ,jboss

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

INTRODUCTION TO ORACLE WEBLOGIC SERVER:

WebLogic server introduced by WebLogic in 1995.
BEA systems Acquired WebLogic server in 1998.
Oracle has Acquired WebLogic server in 2008.

WebLogic is a server software application that runs on a **middle tier**, between back-end **databases** and related applications and **browser-based thin clients**. WebLogic is a leading **e-commerce** online transaction processing (OLTP) platform, developed to connect users in a distributed computing environment and to facilitate the integration of mainframe applications with distributed corporate data and applications.
WebLogic server is based on Java 2 Platform, Enterprise Edition (J2EE), the standard platform used to create **Java-based multi-tier enterprise applications**. J2EE platform technologies were developed through the efforts of BEA Systems and other vendors in collaboration with the main developer, Sun Microsystems. Because J2EE applications are standardized modules, WebLogic can automate many system-level tasks that would otherwise have demanded programming time.
The main features of WebLogic server include connectors that make it possible for any legacy application on any client to interoperate with server applications, **Enterprise JavaBean (EJB) components,** resource pooling, and connection sharing that make **applications very scalable**. An administration console with a user interface makes management tasks more efficient and features such as Secure Sockets Layer (SSL) support for the encryption of data transmissions, as well as authentication and authorization mechanisms make applications and transactions secure.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Diff b/w WLS Versions (8x,9.x, 10.x and 12c) :

	WLS 8.x	- In wls 8.x, all the jms details will be present in the same config.xml. - In wls 8.x we don't have concept of JMS modules and Subdeployment. - In wls 8.x, we have queues, topics are separately - In WLS 8.x we don't have LOCK & EDIT feature. - In wls 8.x we don't have update application feature, if you want to redeploy you have to delete and deploy application. - In 8.x sever never comes-up, if even one of the deployment fails
	WLS 9.x and 10.x	- In wls 9. x/10.x it will be partly maintained in config.xml and a separate xml files in domain/config/jms folders for each jms module. - In wls 10.x they are clubbed inside a JMS module. - In wls WL 9&10 we have LOCK & EDIT feature. - From 9.x we have update feature available. - In wls 9.x, server gets into ADMIN mode, if deployment fails.
	WLS12c	- JDK version 7 or higher (1.7) is recommended to install WebLogic 12.X - In wls 12c Dynamic Cluster Support (dynamic cluster for a highly scalable systems). - In wls 12c WebLogic Server adds support for Oracle Database 12c. - JMS enhancements(Supports clustered targeted JMS Servers for providing high availability eliminating the need to configure many JMS resources for every single server) -In wls 12c to while applying the weblogic patches we need to change the tool from BSU to OPATCH

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

INSTALLATION GUIDE:

Basically there are three ways of installation modes.

1. GUI Mode Installation
2. Command Mode Installation
3. Silent Mode Installation

Diff ways we are going to install weblogic server like **GUI** it's an general way just clicking next –next process and command mode installation and silent mode installation, for silent mode of installation we have to prepare one

–silent.xml file and we have to execute like below .
./wls.bin –mode=silent.xml /root/harish/silent.xml

-In 64bit operating systems we are going use generic.jar weblogic file ,for installing weblogic server .
For Generic jar file we don't get any inbuilt JDK for this we need to install JDK before installing weblogic server.
Export JAVA_HOME= **/../../bin**

PATH=$JAVA_HOME/bin:$PATH; export PATH

java -jar wls1036_generic.jar

64-Bit Platforms Using a 64-Bit JDK:

JAVA_HOME=path_to_64-bit_JDK; export JAVA_HOME

PATH=$JAVA_HOME/bin:$PATH; export PATH

(UNIX or Linux only) Include the -d64 flag in the installation command when using a 32/64-bit hybrid JDK (such as for the HP-PA, HPIA, and Solaris64 platforms). For example, if installing in graphical mode using the Package installer:

java -d64 -jar wlsversion_generic.jar

Run the java -version command (or java -d64 -version command on UNIX or Linux platforms using a 32/64-bit hybrid JDK) to ensure that your JAVA_HOME refers to a 64-bit JDK.

If you are using the Sun 64-bit JDK, use the following command to install WebLogic Server:
java -Xmx1024m -jar wlsversion_generic.jar
Command Mode Installation (in Linux)
Installing Weblogic server in Linux machine

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

GUI Mode Installation:

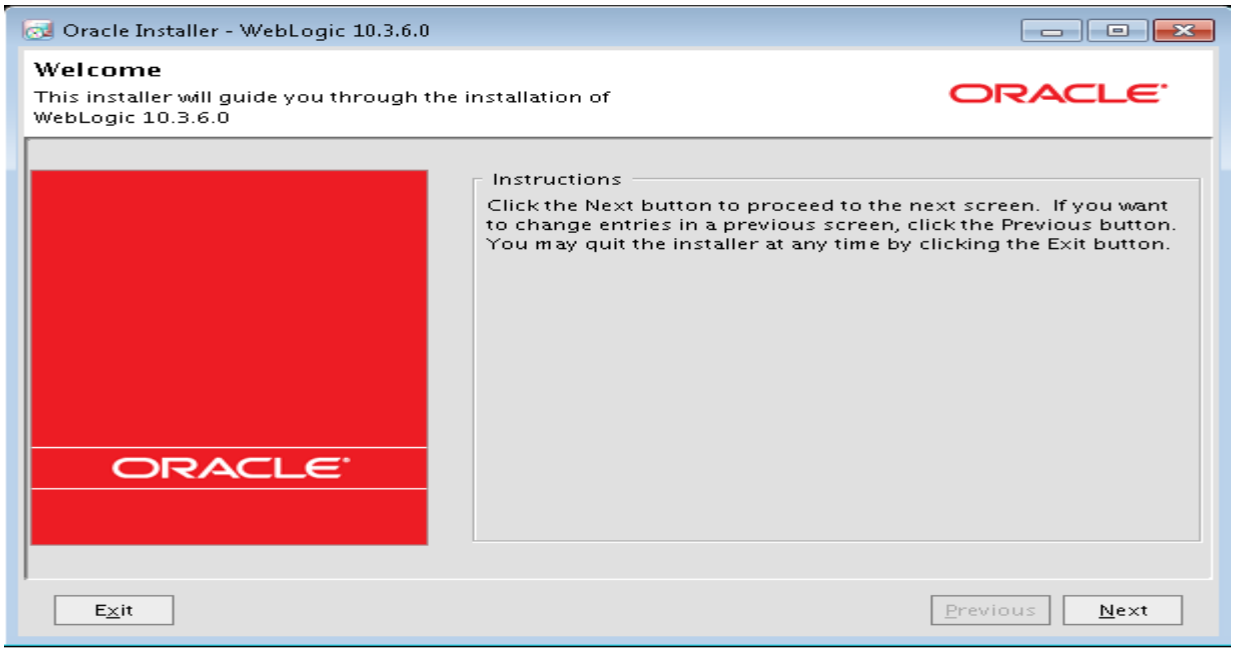
```
-rw-r--r--  1 oracle oinstall 1068506707 Feb 26  2014 wls1036_generic.jar
[oracle@svebs0000028np stage]$ export JAVA_HOME=/oracle/app/stage/jdk1.7.0_71
[oracle@svebs0000028np stage]$ export PATH=$JAVA_HOME/bin:$PATH
[oracle@svebs0000028np stage]$ ll
total 5066116
drwxrwxrwx  8 oracle oinstall       4096 Apr 28 23:55 jdk1.7.0_71
-rwxrwxrwx  1 oracle oinstall  520548890 Apr 24 01:13 ofm_rcu_linux_11.1.1.7.0_64_disk1_1of1.zip
-rwxrwxrwx  1 oracle oinstall 1720182214 Feb 25  2014 ofm_soa_generic_11.1.1.7.0_disk1_1of2.zip
-rwxrwxrwx  1 oracle oinstall 1292438758 Feb 25  2014 ofm_soa_generic_11.1.1.7.0_disk1_2of2.zip
-rwxrwxrwx  1 oracle oinstall  585960613 Feb 11 02:16 p19953598_111170_Generic.zip
drwxr-xr-x 38 oracle oinstall       4096 Mar 13  2013 rcuHome
-rw-r--r--  1 oracle oinstall      22806 Mar 13  2013 readme.html
drwxr-xr-x  9 oracle oinstall       4096 Apr 28 23:26 soa
-rw-r--r--  1 oracle oinstall 1068506707 Feb 26  2014 wls1036_generic.jar
[oracle@svebs0000028np stage]$ $JAVA_HOME/bin/java -d64 -Xmx1024m -jar wls1036_generic.jar
Extracting 0%.....100%
```

oracle@ stage]$ **export JAVA_HOME=/oracle/app/stage/jdk1.7.0_71**

[oracle@ stage]$ **export PATH=$JAVA_HOME/bin:$PATH**

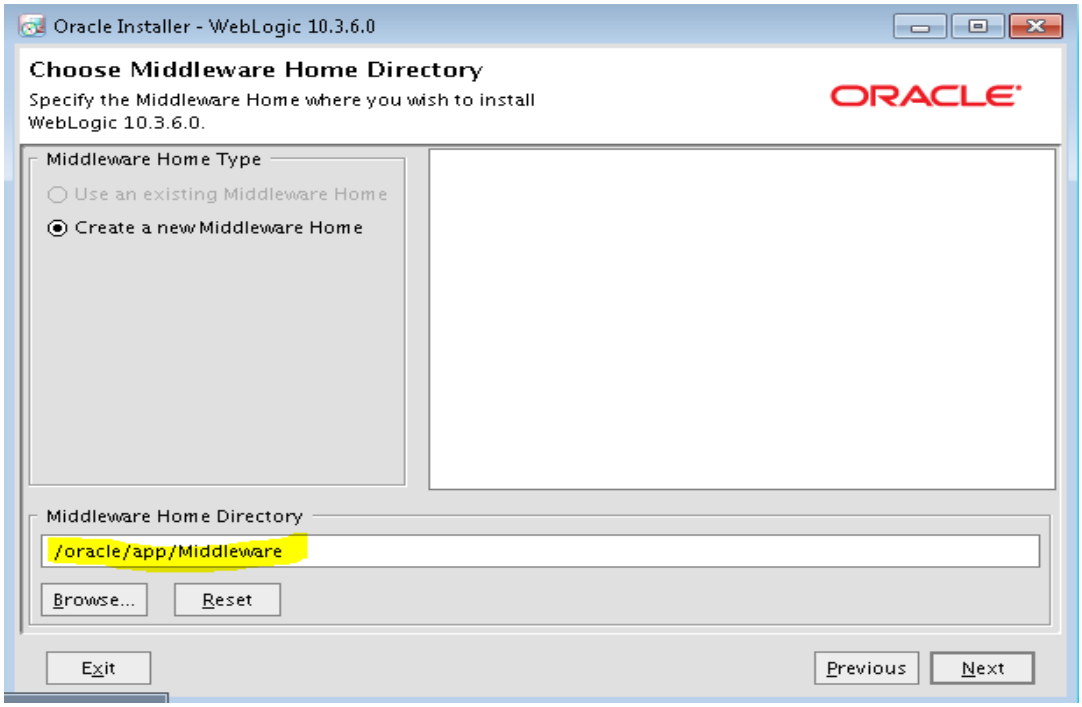
[oracle@ stage]$ **$JAVA_HOME/bin/java -d64 -Xmx1024m -jar wls1036_generic.jar**

1. Execute the generic jar file to the above format .



2. Create Middleware Home

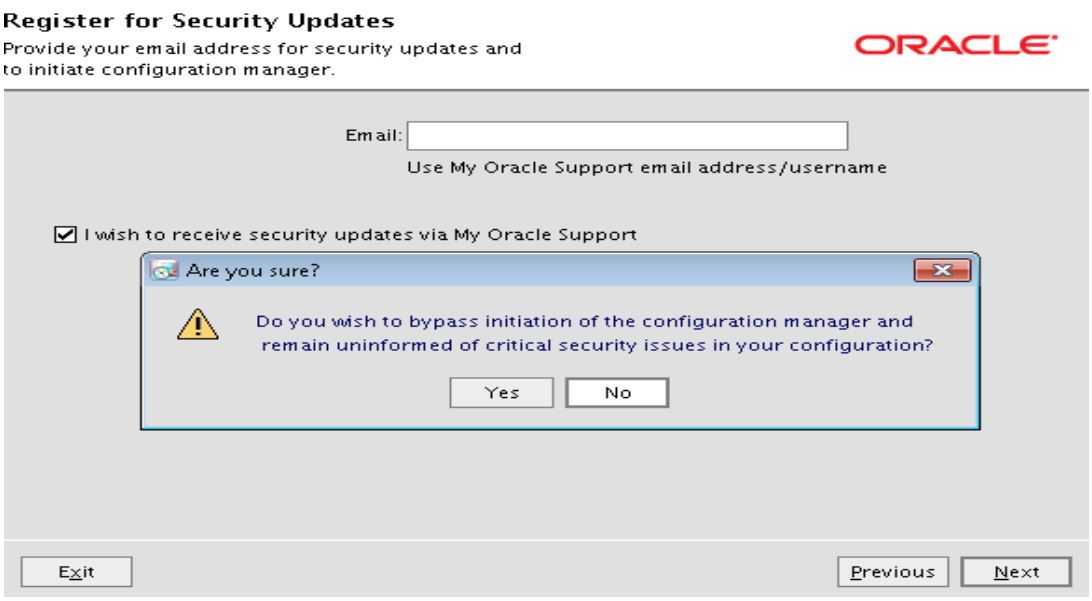
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

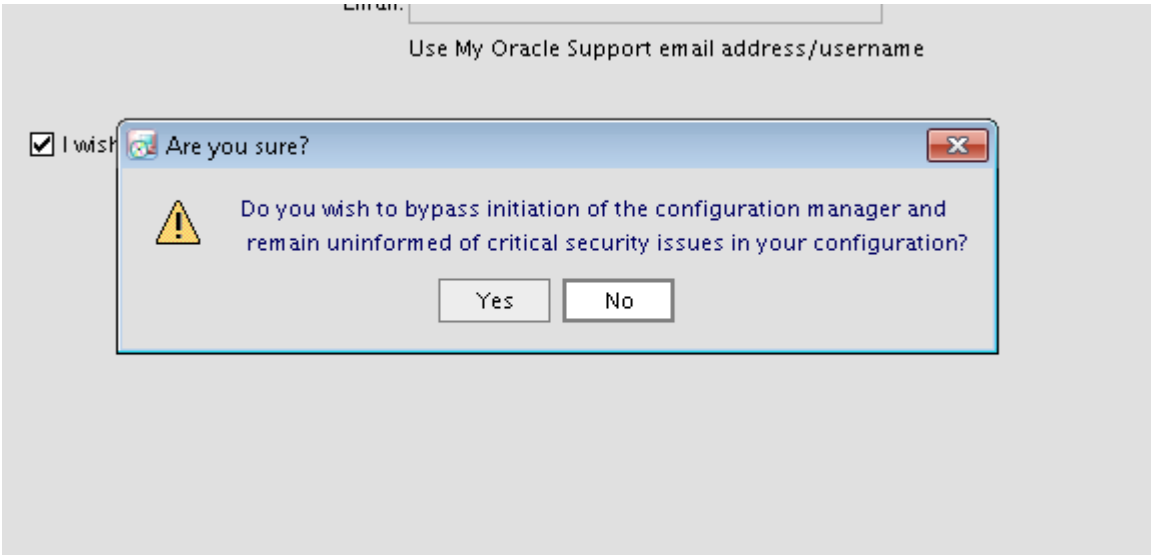


3. Remove the tick mark for security updates.

4. Click on yes (for removing the security updates)

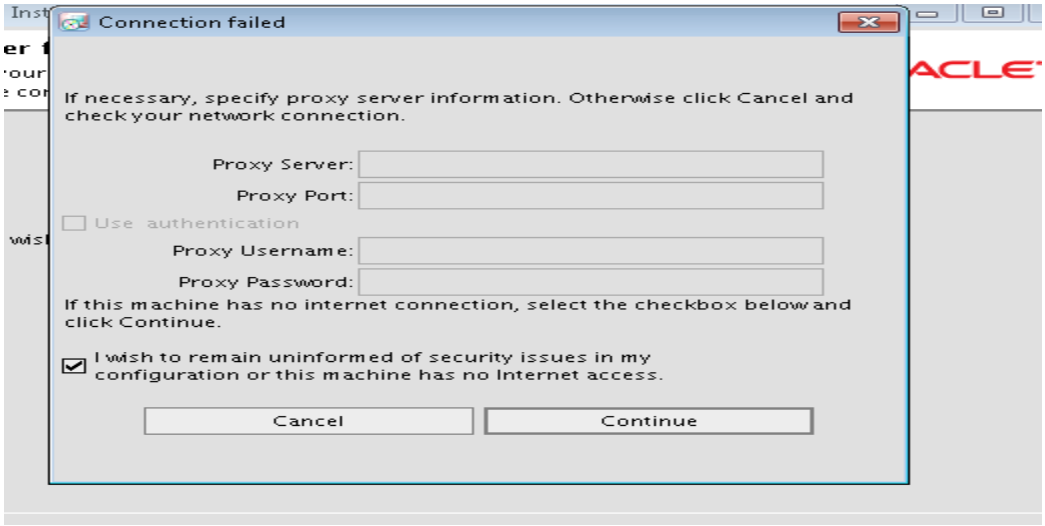
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



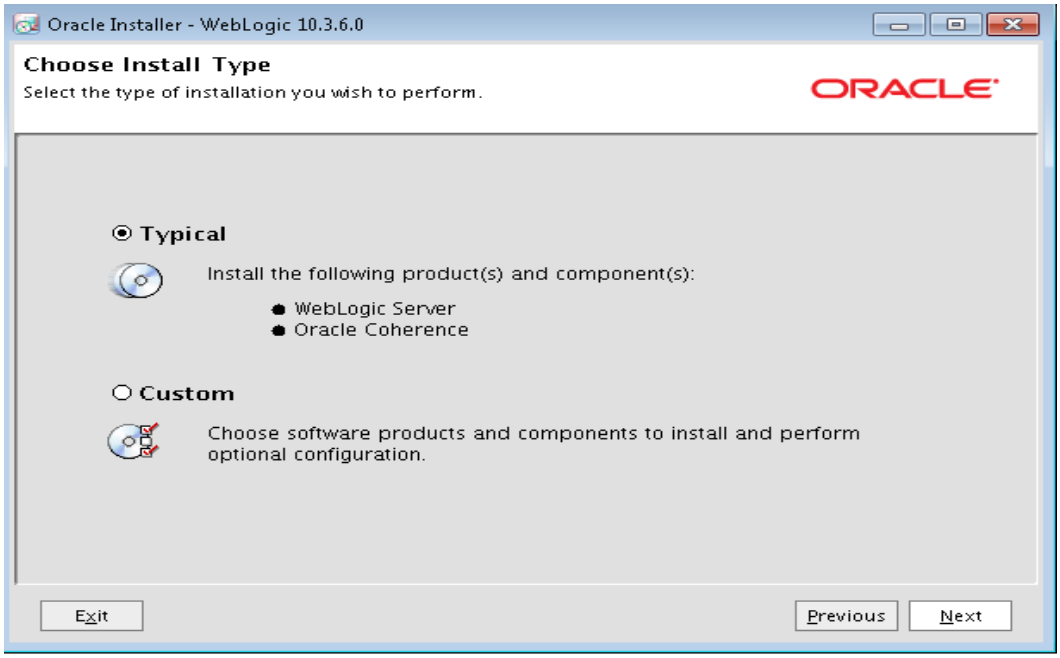


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

5. Click on continue

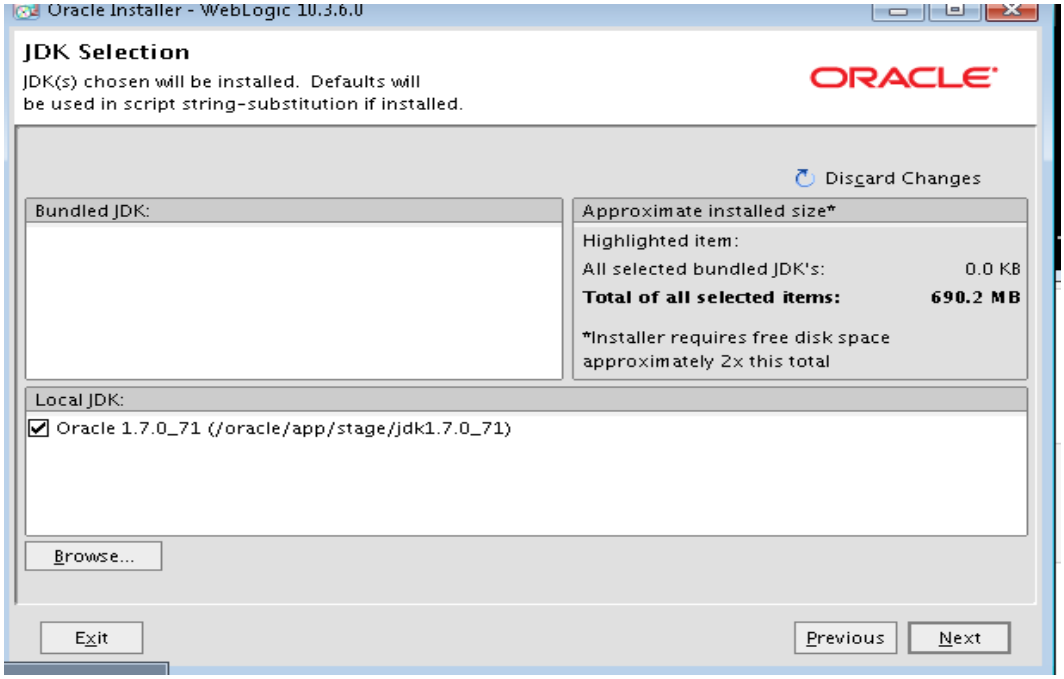


6. Click on the custom installation

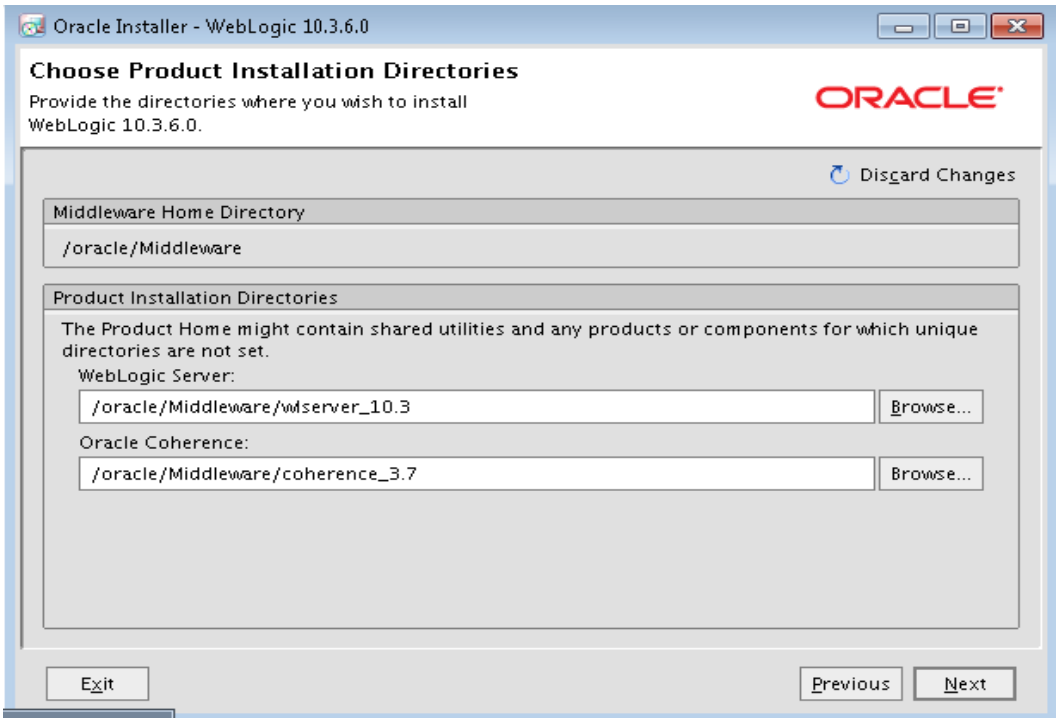


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

7. Select the JDK

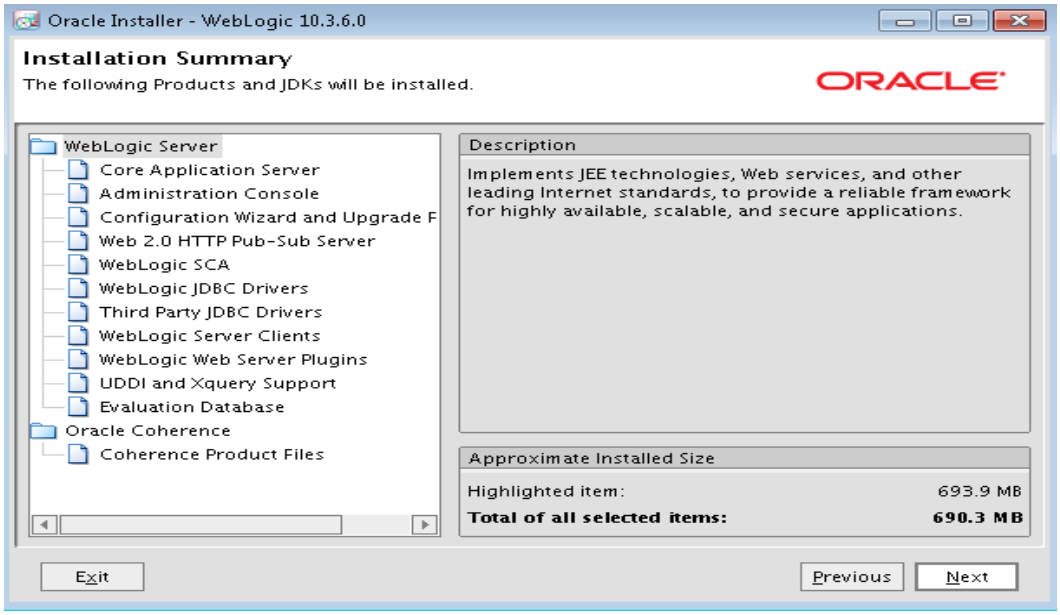


8. Choose the Product installation directories (Java and weblogic).



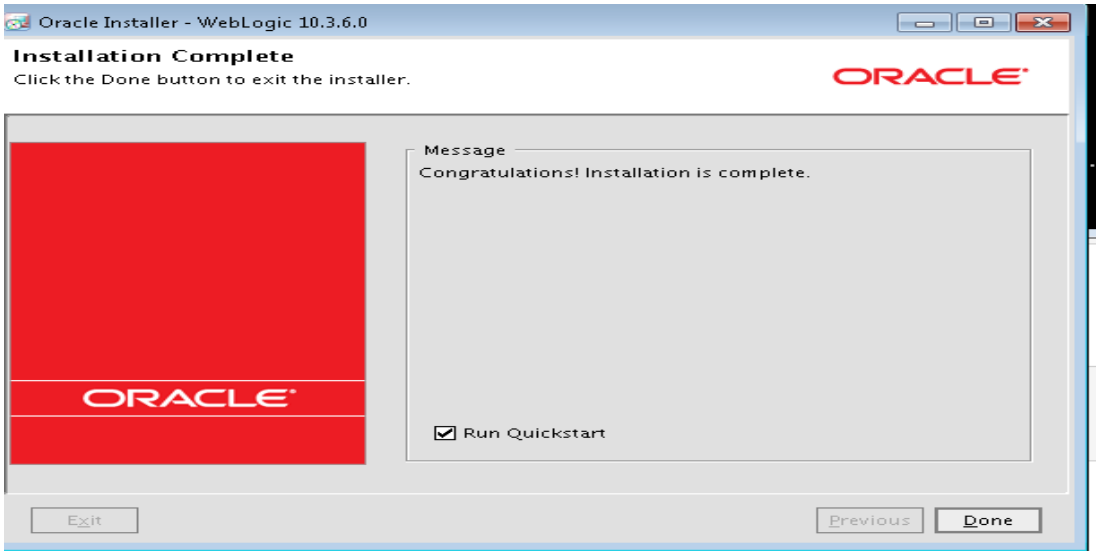
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

9. No the check the summary and processed the installation.



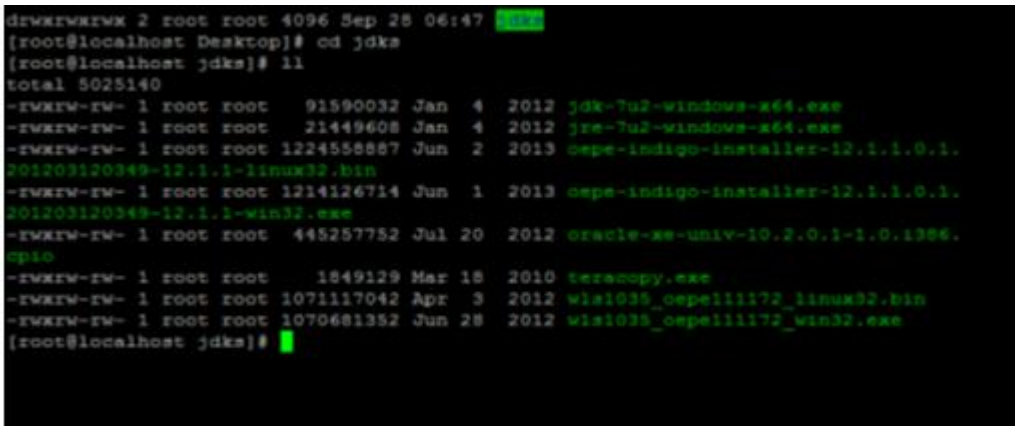
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

10. Now the installation process is done click on done.



Command Mode Installation (in Linux):

Installing WebLogic server in Linux machine from the putty.



Go to the particular directory
To run file you have to use the below format
./ wls1035_oepe111172_linux32.bin

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

```
[root@localhost jdks]# ./wls1035_oepe111172_linux32.bin -mode=console -mode=console
Duplicate command line argument being used: [-mode].
[root@localhost jdks]# ./wls1035_oepe111172_linux32.bin -mode=console
Extracting 0%.....
```

Enter Next

```
<------------------------------------------------------------- Oracle Installer -

Welcome:
--------

This installer will guide you through the installation of WebLogic 10.3.5.0. Type "Ne
previously, type "Previous".  You may quit the installer at any time by typing "Exit"



Enter [Exit][Next]>
```

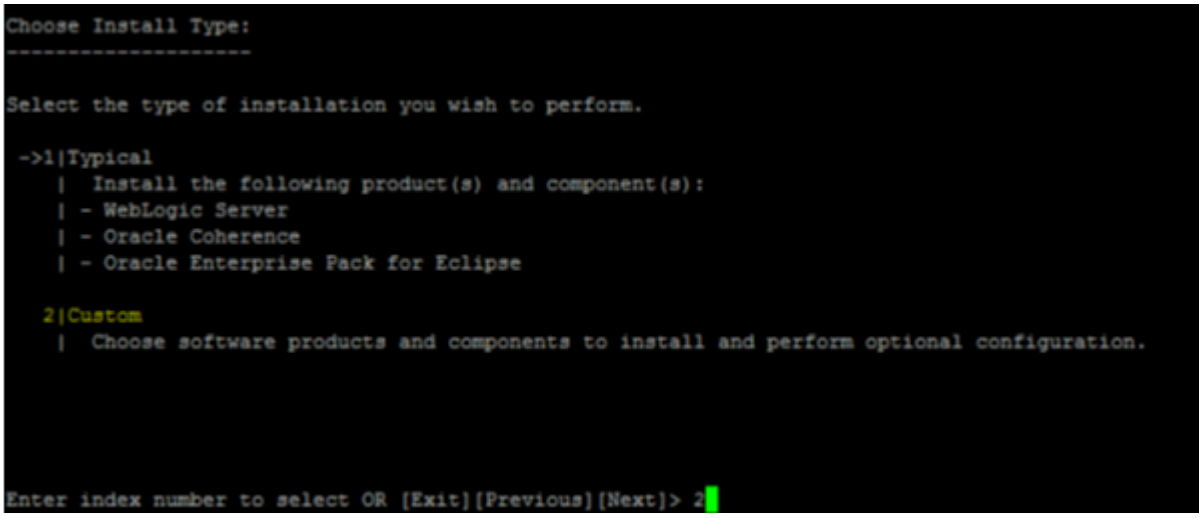
Enter Next

```
<------------------------------------------------------------- Oracle Installer - WebLogic 10.3

Welcome:
--------

This installer will guide you through the installation of WebLogic 10.3.5.0. Type "Next" or enter
previously, type "Previous".  You may quit the installer at any time by typing "Exit".



Enter [Exit][Next]> Next



<------------------------------------------------------------- Oracle Installer - WebLogic 10.3

Choose Middleware Home Directory:
---------------------------------

    "Middleware Home" = [Enter new value or use default "/root/Oracle/Middleware"]
```

Create a new middleware home and click next

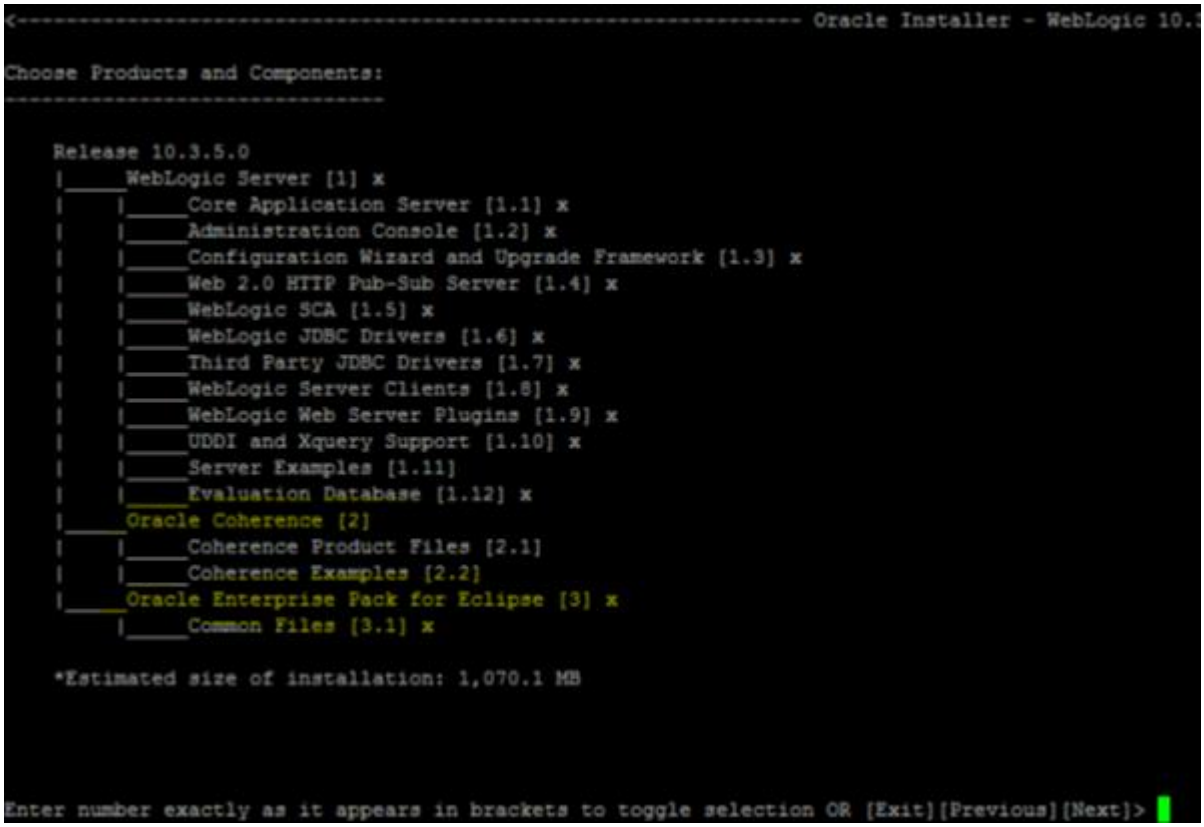
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Remove the software updates.

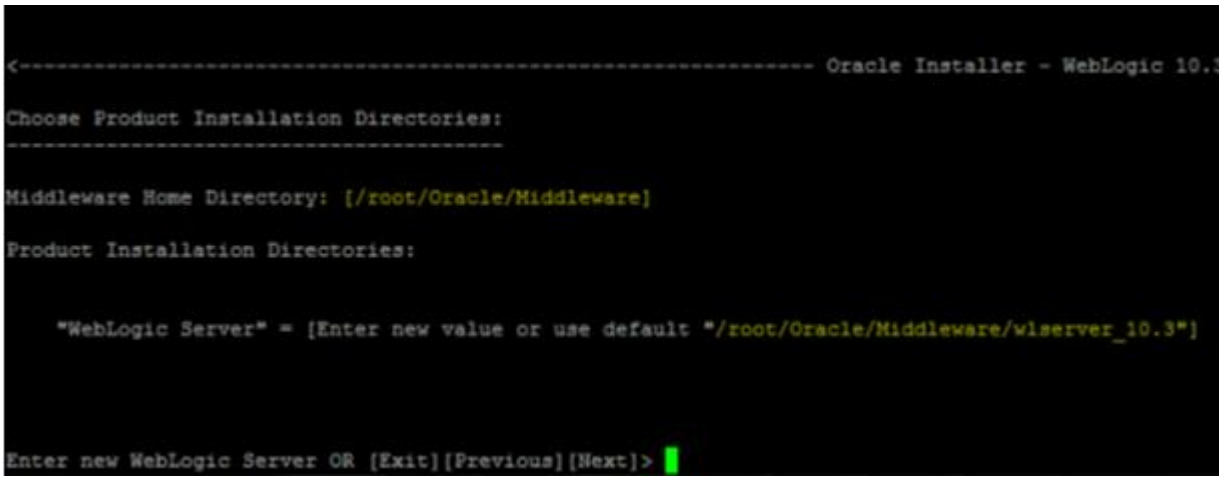


Choose custom installation and click next

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

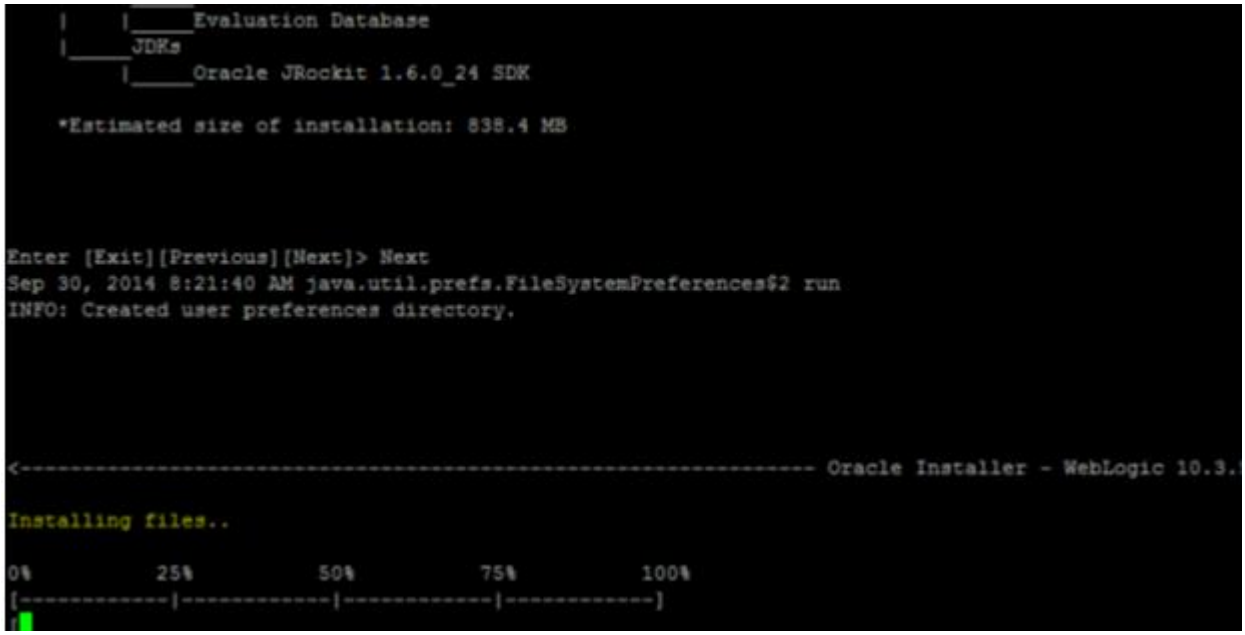


Remove the un-used components and processed .

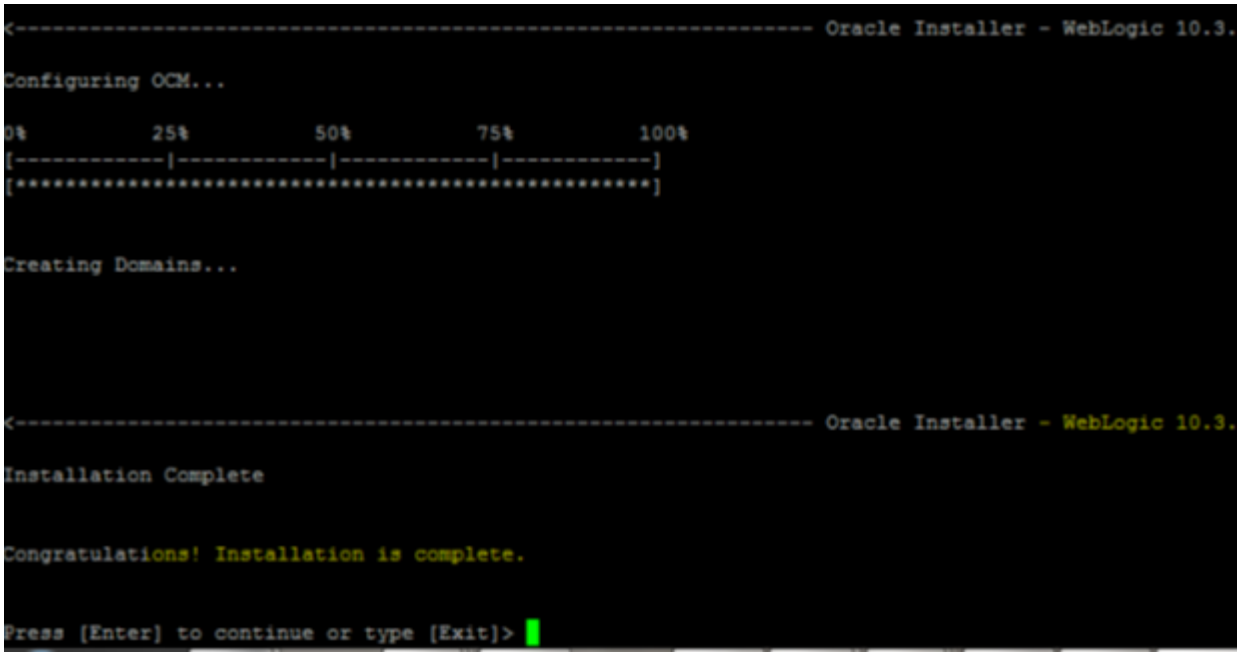


Click Next

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



Installation Process



Installation Complete.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Silent-Mode Installation:

Silent-mode installation is a way of setting installation configurations only once and then using those configurations to duplicate the installation on many machines. During installation in silent mode, the installation program reads the settings for your configuration from an XML file that you create prior to beginning the installation. The installation program does not display any configuration options during the installation process. Silent-mode installation works on both Windows and UNIX systems.

With previous releases of WebLogic Platform, you have the option of running the Configuration Wizard as part of the silent installation process.

Note: The first line of the template file must contain the following XML definition: <?xml version="1.0" encoding="UTF-8"?>. The XML definition must appear at the beginning of the first line of the file; no spaces or line breaks are allowed before it.

Sample Template File for Silent-Mode Installation:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Silent installer option: -mode=silent -silent_xml=/home/me/silent.xml -->
<domain-template-descriptor>
<input-fields>
  <data-value name="BEAHOME"                value="C:\bea" />
  <data-value name="USER_INSTALL_DIR"       value="C:\bea\weblogic81" />
  <data-value name="INSTALL_NODE_MANAGER_SERVICE"   value="no" />
  <data-value name="COMPONENT_PATHS" value="WebLogic Server|WebLogic Workshop|WebLogic
Integration|WebLogic Portal" />
</input-fields>
</domain-template-descriptor>
```


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

DOMAIN CONFIGURATION:

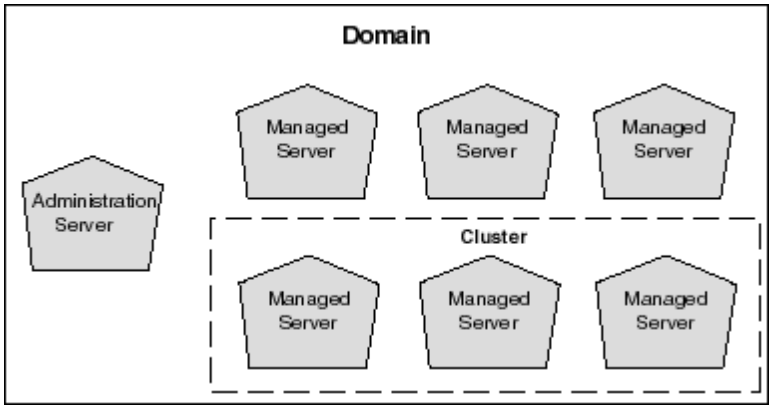
What Is a Domain :

A domain is the basic administration unit for WebLogic Server instances. A domain consists of one or more WebLogic Server instances (and their associated resources) that you manage with a single Administration Server. You can define multiple domains based on different system administrators' responsibilities, application boundaries, or geographical locations of servers. Conversely, you can use a single domain to centralize all WebLogic Server administration activities.

Each domain's configuration is stored in a separate configuration file called config.xml, which is stored on the Administration Server along with other files such as logs and security files. When you use the Administration Server to perform a configuration task, the changes you make apply only to the domain managed by that Administration Server. To manage another domain, use the Administration Server for that domain. For this reason, the servers instances, applications, and resources in one domain should be treated as being independent of servers, applications, and resources in a different domain.You cannot perform configuration or deployment tasks in multiple domains at the same time.

Contents of a Domain:

A domain can include multiple WebLogic Server clusters and non-clustered WebLogic Server instances. A minimal domain can contain only one WebLogic Server instance, which functions both as an administration Server, and as a Managed server—such a domain can be useful while developing applications, but is not recommended for use in a production environment. Although the scope and purpose of a domain can vary significantly, most WebLogic Server domains contain the components described in this section.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

There are several ways we can configure a WebLogic domain.

1. GUI Mode –
2. Command Mode-
3. Silent.xml Mode –
4. WLST –

Creating a Domain Using the Configuration Wizard:

The Configuration Wizard is the easiest and recommended way to create a new WebLogic Server domain. The configuration Wizard interactively queries you about the type of domain you want to create and creates the config.xml file, start scripts, and other files used in a domain. Using the Configuration Wizard you can.

- Create a domain with a single WebLogic Server instance for use in a development environment
- Create a domain with one or more Managed Servers
- Create a domain with one or more clusters
- Configure JDBC connection pools, DataSources, and MultiPools
- Configure JMS
- Add users to the security realm
- Create a domain based on a configuration template. A configuration template defines the full set of resources within a domain, including infrastructure components, applications, services, security options, and general environment and operating system options. You can use one of the provided configuration templates or create your own template.

Creating a Domain Using the weblogic.Server Command:

If you start a WebLogic Server instance from a directory that does not contain a **config.xml** file, the server automatically creates a default configuration, including start scripts and theconfig.xml file, in the directory from which you start the server instance. You can then modify the domain configuration using the Administration Console or other system administration tools.

To create a domain using the **WebLogic.Server** command:

1. Open a command shell.

2. Set the CLASSPATH to include the WebLogic Server classes. The easiest way to set your CLASSPATH is to run the setWLSEnv.cmd (Windows) or the setWLSEnv.sh (UNIX) script. The script is located in the WebLogic Server installation at:

   Oracle/weblogic/server/bin

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

3. Run the following command:

   java weblogic.Server

   When you are prompted for a username and password, enter any values you choose. You will be prompted with:

   Would you like the server to create a default configuration and boot? (y/n)

Note:

If you want to specify a name for your domain add the -Dweblogic.Domain option to the start command. For example, the command

   java -Dweblogic.Domain=HWT weblogic.Server

   creates a domain called "HWT".

4. Answer Y. You will be asked to confirm the password.
5. Enter the same password you entered in step 3. The server starts and creates a default config.xml file in the directory from which you ran the java weblogic.Server command.

The server creates a Boot Identity file (called **boot.properties**) that contains the username and password you entered in step 3. When this file is present in the domain's root directory, the server does not prompt for a username and password during startup.

Domain Types:

There are two basic types of domains:

- **Domain with Managed Servers**: A simple production environment can consist of a domain with several Managed Servers that host applications, and an Administration Server to perform management operations. In this configuration, applications and resources are deployed to individual Managed Servers; similarly, clients that access the application connect to an individual Managed Server.

  Production environments that require increased application performance, throughput, or availability may configure two or more of Managed Servers as a cluster. Clustering allows multiple Managed Servers to operate as a single unit to host applications and resources. For more information about the difference between a standalone and clustered Managed Servers.

- **Standalone Server Domain**: For development or test environments, you may want to deploy a single application and server independently from servers in a production domain. In this case, you can deploy a simple domain consisting of a single server instance that acts as an Administration Server and also hosts the applications you are developing. The examples domain that you can install with WebLogic Server is an example of a standalone server domain.

Note: In production environments, deploy applications only on Managed Servers in the domain. The Administration Server should be reserved for management tasks.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Domain Restrictions:

Many WebLogic Server installations consist of a single domain that includes all the Managed Servers required to host applications. Note the following restrictions for domains:

- Do not create a domain named "weblogic". This name is reserved for internal use by WebLogic Server.
- If you create more than one domain:
    - Each domain requires its own Administration Server for performing management activities. To manage another domain, start the Administration Console hosted by the Administration Server of that domain, or use the URL of the Administration Server in the command line arguments.
    - All Managed Servers in a cluster must reside in the same domain; you cannot "split" a cluster over multiple domains.
    - You cannot share a configured resource or subsystem between domains. For example, if you create a JDBC connection pool in one domain, you cannot use it with a Managed Server or cluster in another domain. (Instead, you must create a similar connection pool in the second domain.)
- Each server instance in your WebLogic environment must have a unique name, regardless of the domain or cluster in which it resides, or whether it is an Administration Server or a Managed Server. Within a domain, each server, machine, cluster, virtual host, and any other resource type must be named uniquely and must not use the same name as the domain.

Diff b/w Production Domain and Development Domain:

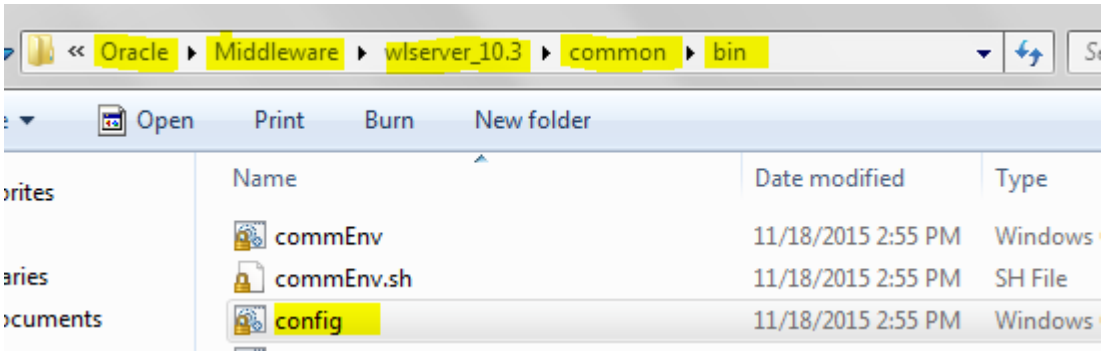
Development domain	Production Domain
▪ The default JDK for development domain is Sun Hotspot ▪ You can use the demo certificates for SSL ▪ Auto deployment is enabled ▪ Server instances rotate their log files on startup ▪ Admin Server uses an automatically created boot.properties during startup ▪ The default maximum capacity for JDBC Datasource is 15 ▪ Lock and Edit feature is disable.	▪ The default JDK for production domain is JRockit ▪ If you use the demo certificates for SSL a warning is displayed ▪ Auto deployment is disabled ▪ Server instances rotate their log files when it reaches 5MB ▪ Admin Server prompts for username and password during startup ▪ The default maximum capacity for JDBC Datasource is 25 ▪ Lock and edit feature is Enable.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Diff b/w Sun JDK and J-Rocket:

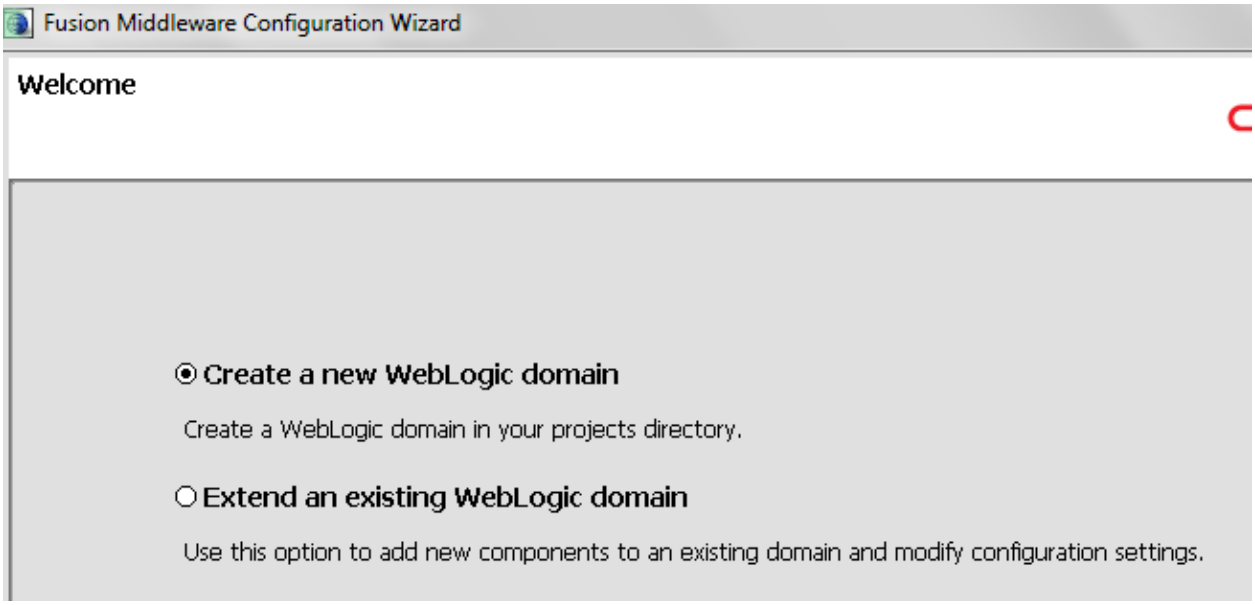
Sun-JDK	J-Rocket
▪ Sun JDK uses interpreter (Interpreter and JIT in previous releases) ▪ In this mechanism, the byte code is read and the translated into machine language, but these results are not saved in the memory. ▪ Every time even if the same method is run again and again, the JVM has to translate the code into machine language. This means machine code will not be reusable as it is not saved anywhere in the memory. ▪ While starting Sun JDk is faster than J-rocket ▪ Sun JDK has the following memory spaces: Eden space, survivior space, tenured generation and permanent generation. The objects move from one space to another according to its age and survival from garbage collection.	▪ Oracle JRockit uses only JIT compiler (Just In Time) ▪ JIT mechanism means, once a method is run, the byte code is translated to machine language and this is saved in the memory. ▪ This means if the method is run again, there is no need for translation and the machine code is reused. ▪ oracle JRockit saves the code, which is why start up takes longer. For the same reason, oracle JRockit uses more memory than sun jdk. ▪ In the long run, JRockit gives a slightly better performance as compared to sun jdk. ▪ JRockit has 2 spaces, young generation and old generation, it uses the same mechanism of garbage collection. There is nothing called as permanent generation in JRockit.

GUI Mode Domain Creation:

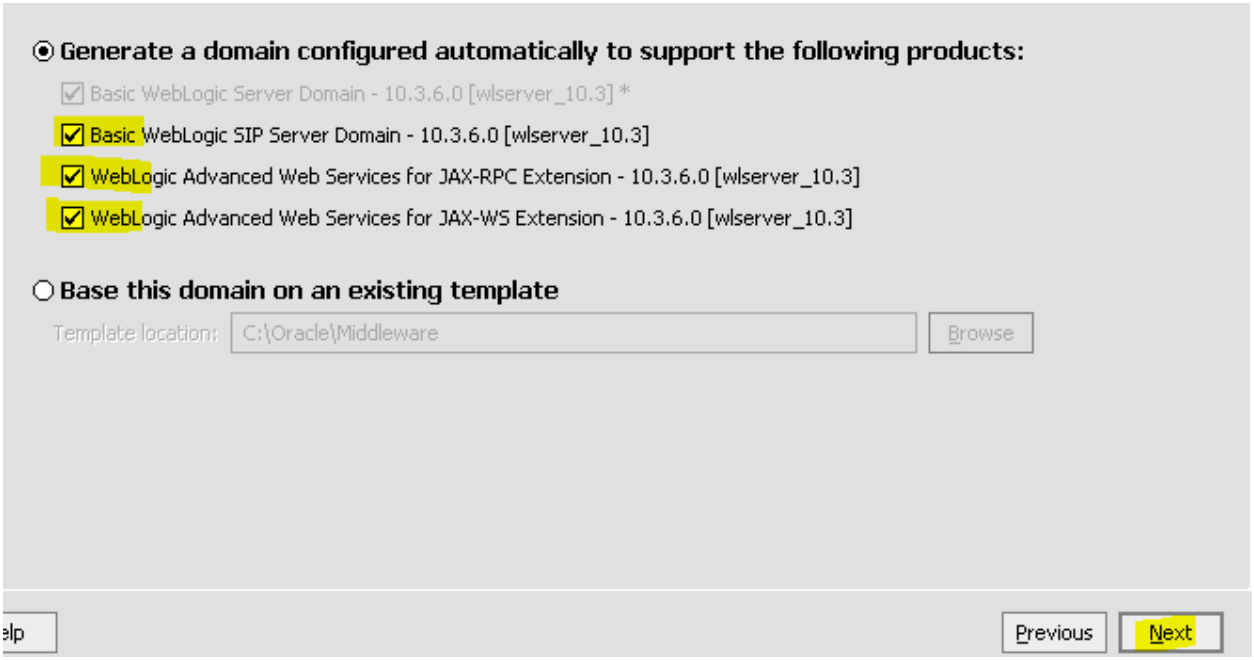


Run the config command.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

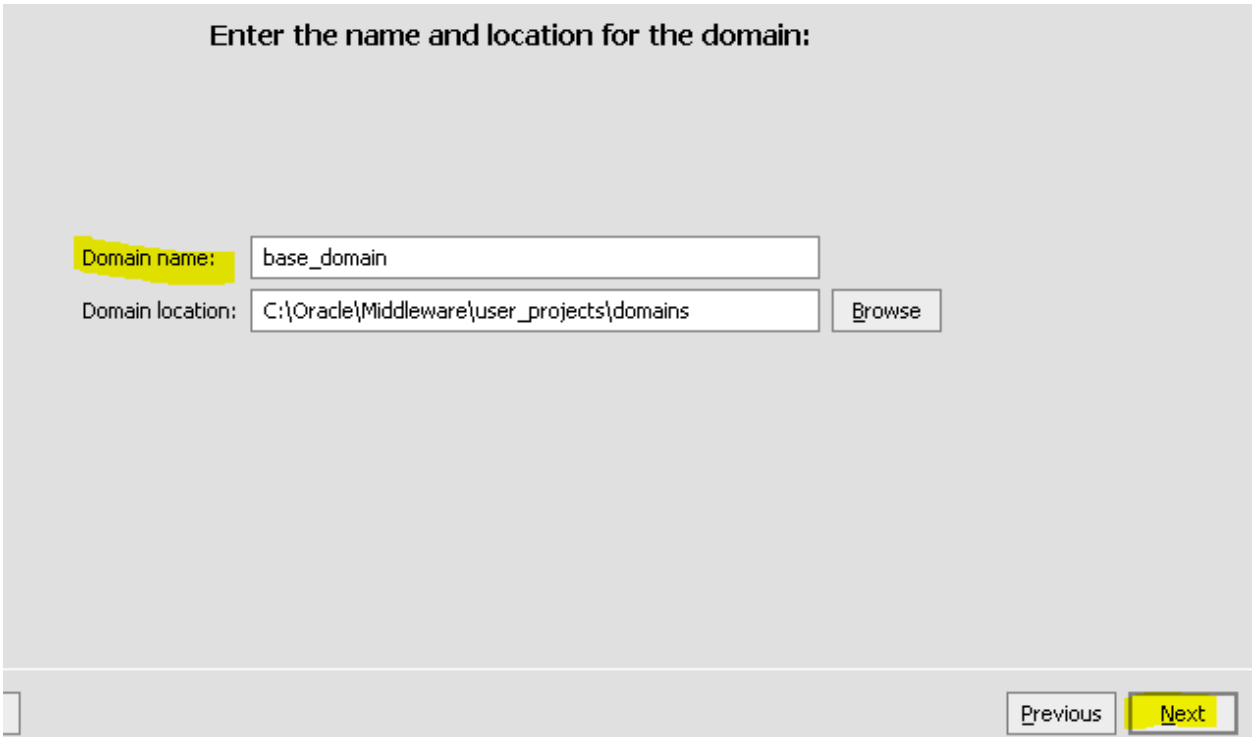


Create the New Domain .

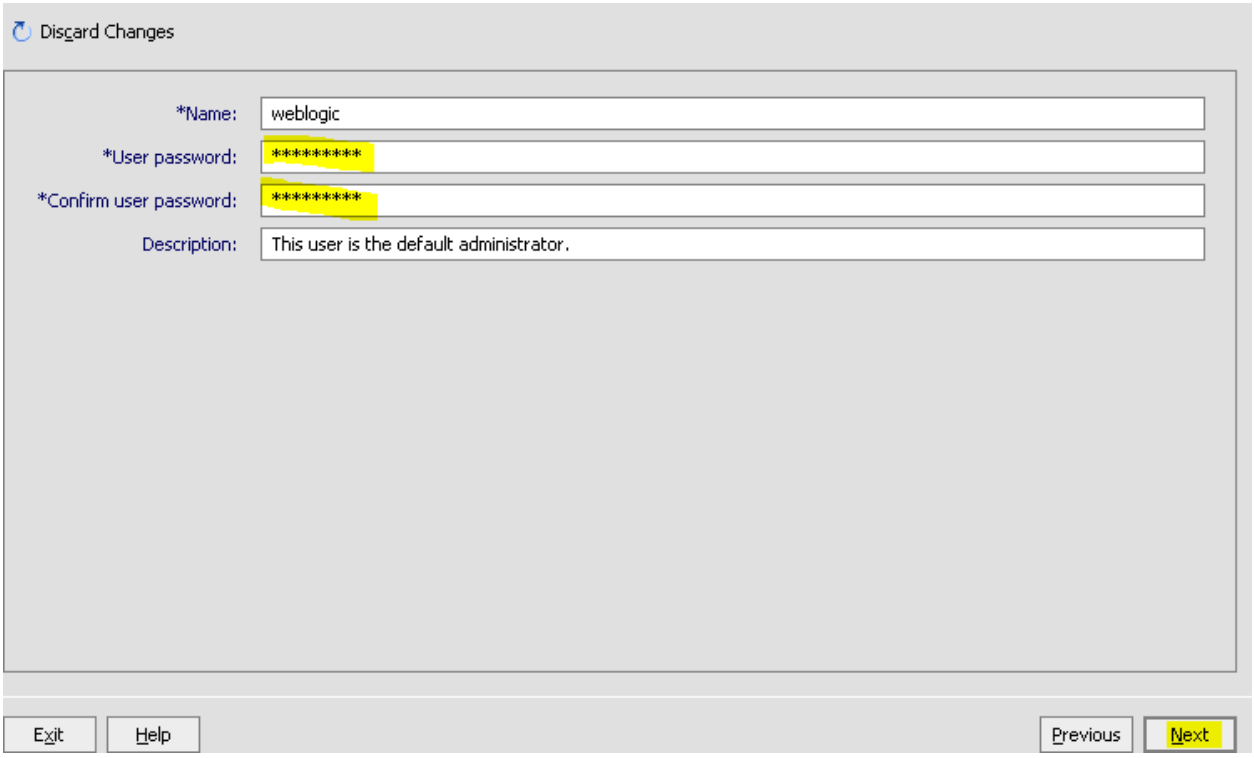


Select the all Components for the new Domain.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

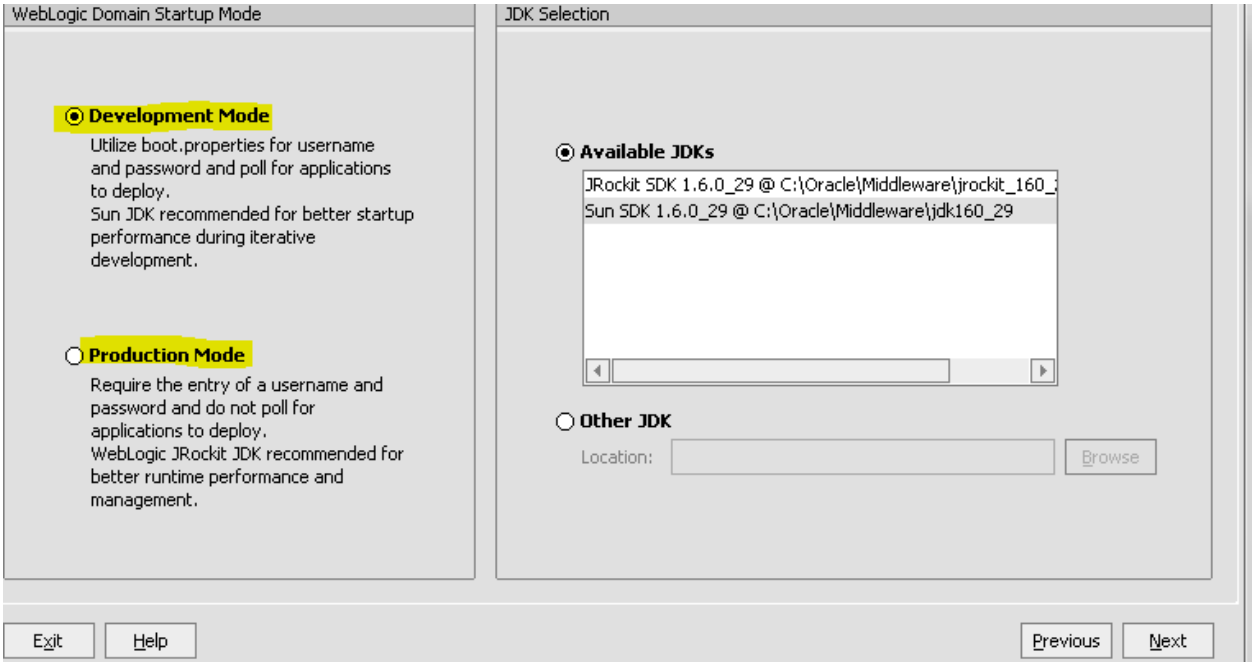


Domain name and Location we can change as per the requirement.

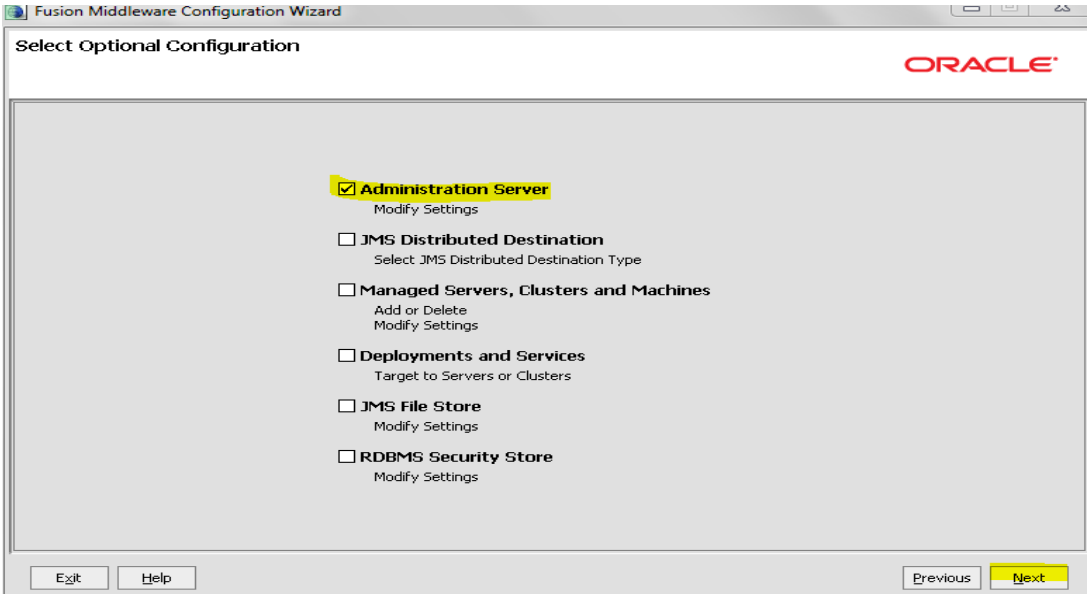


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

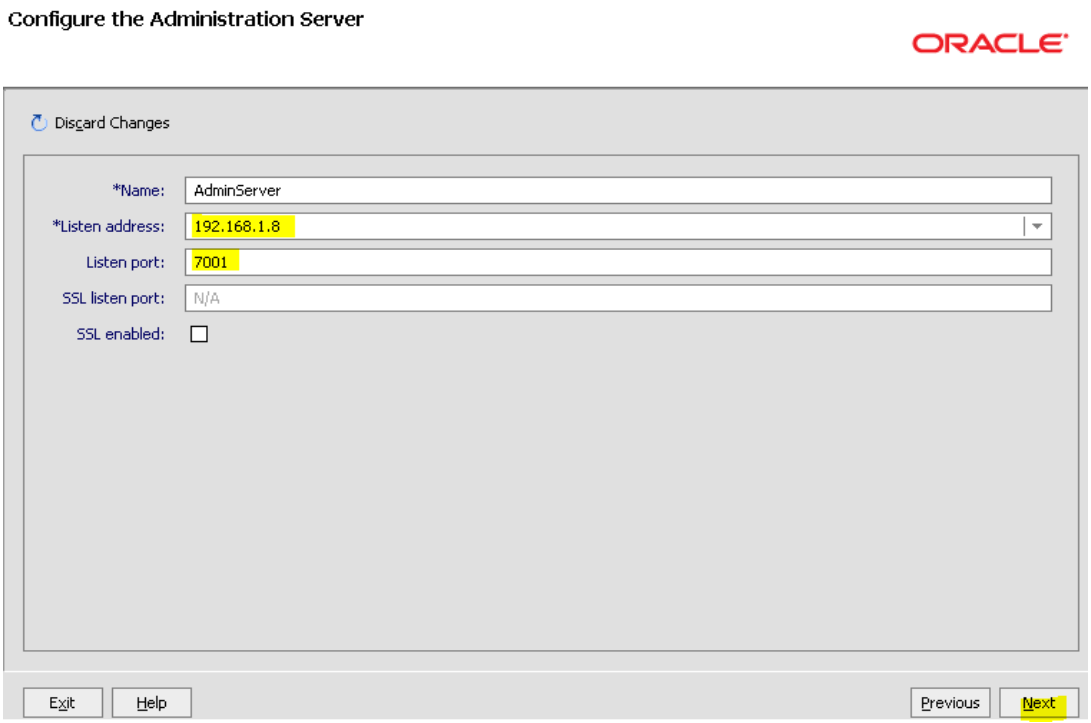
By default Admin user name is weblogic (oracle recommended) and need to create password (alpha numeric 8 Characters)



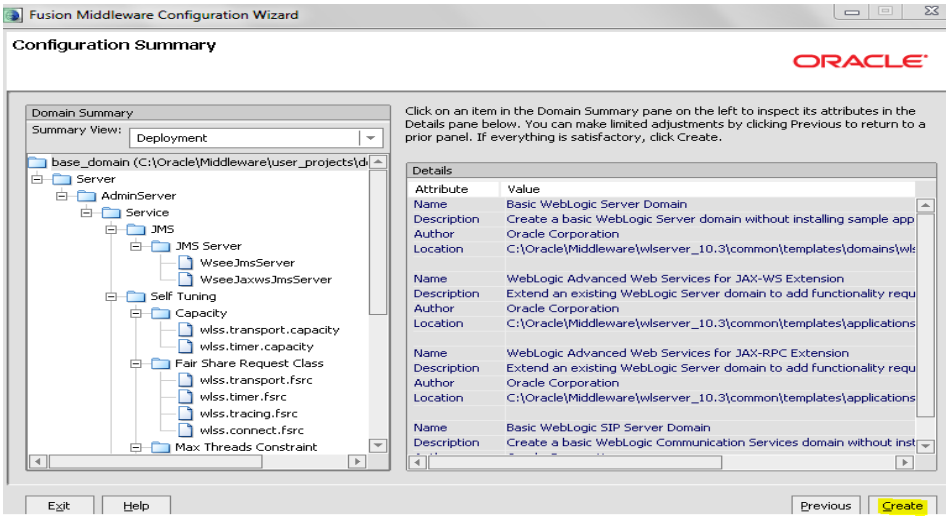
As per the requirement need to select Domain .



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

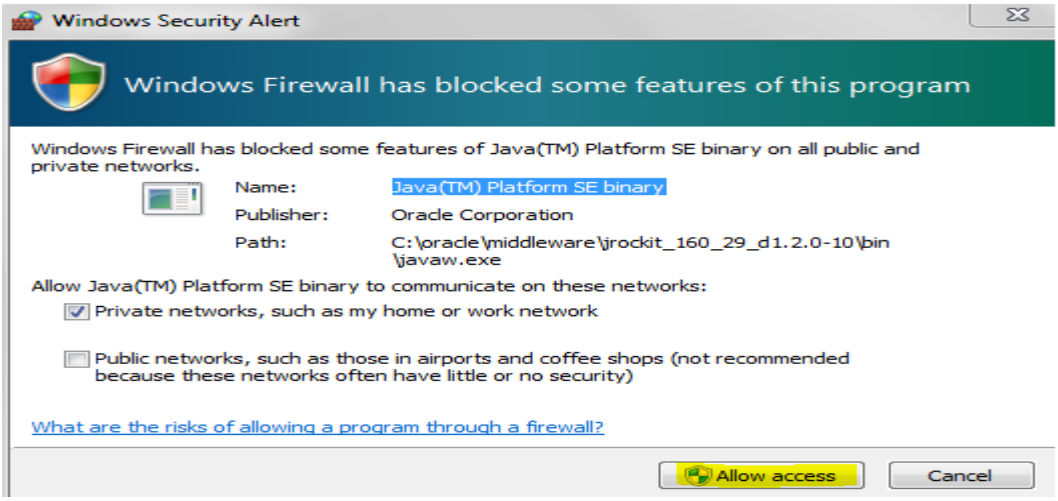


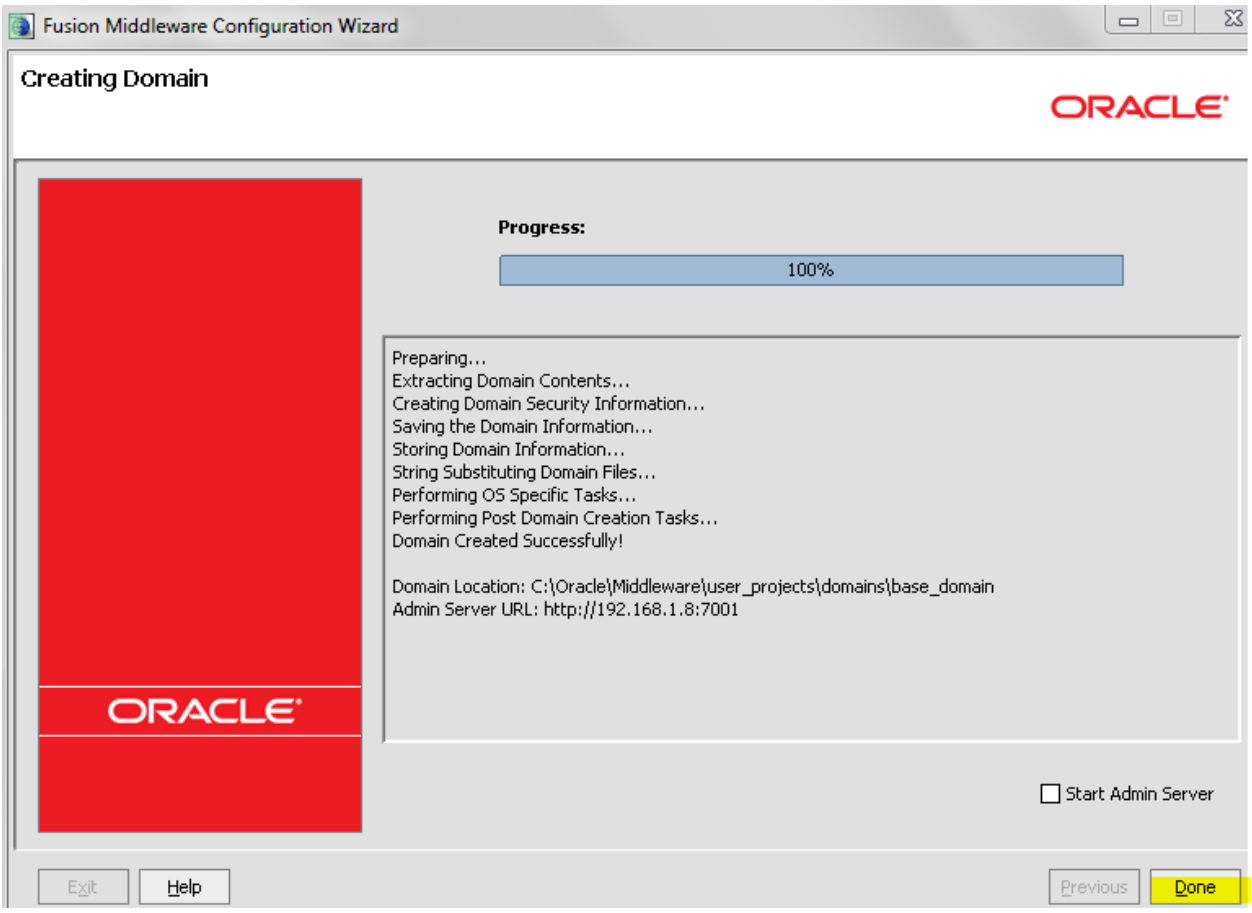
Change the Listen address and port .



Click on Create.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**





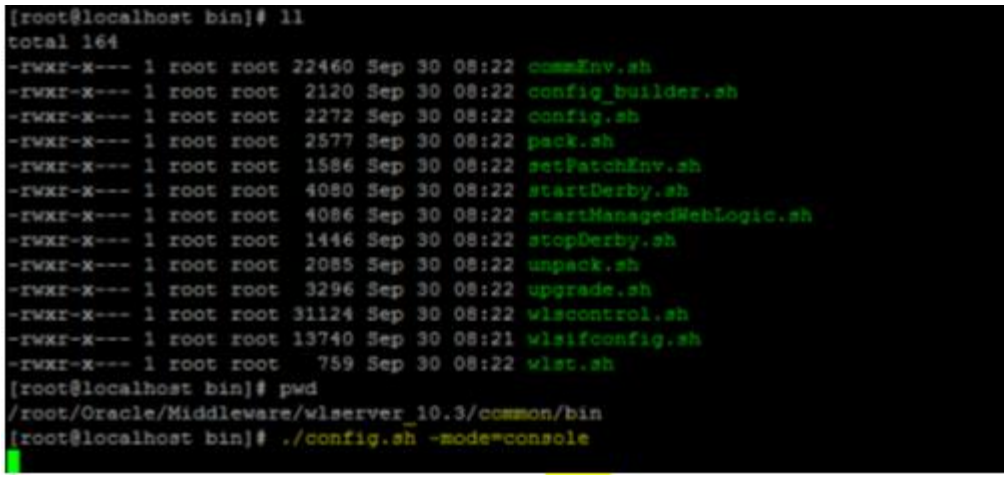
Domain Created successfully.

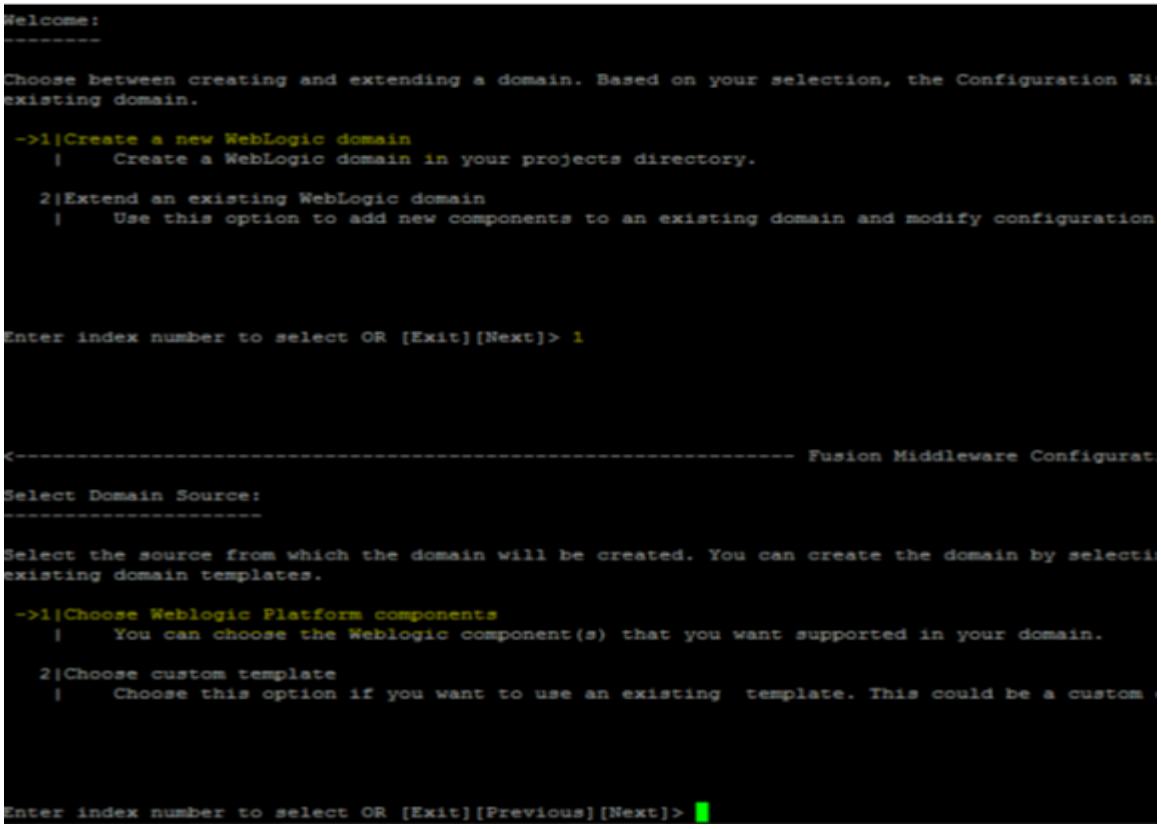
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Command Mode Domain Configuration:

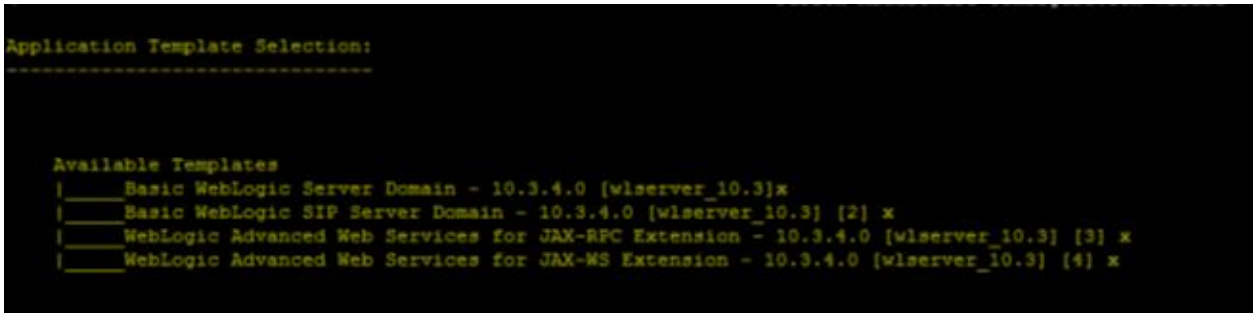
Go to below path
/Oracle/Middleware/wlserver_10.3/common/bin

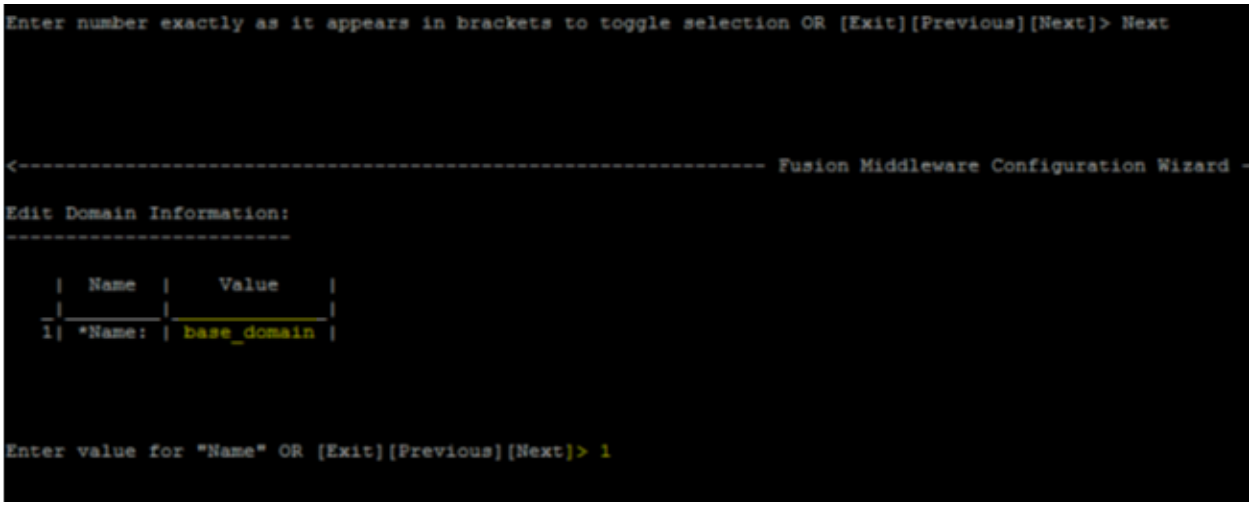
Run ./config.sh



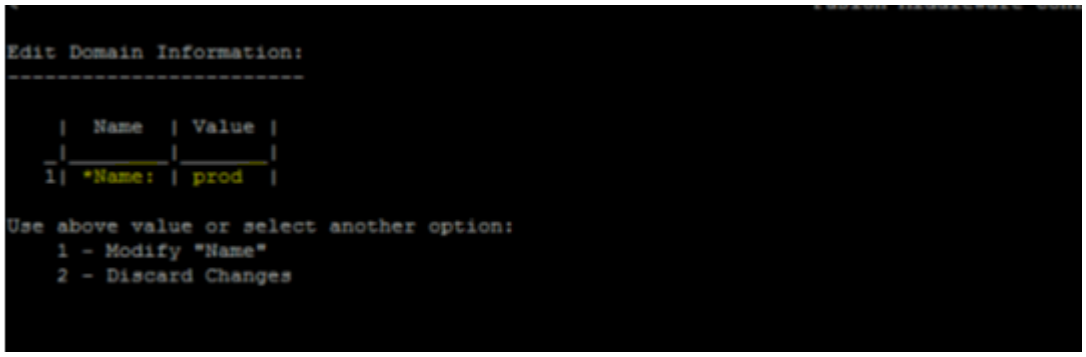


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

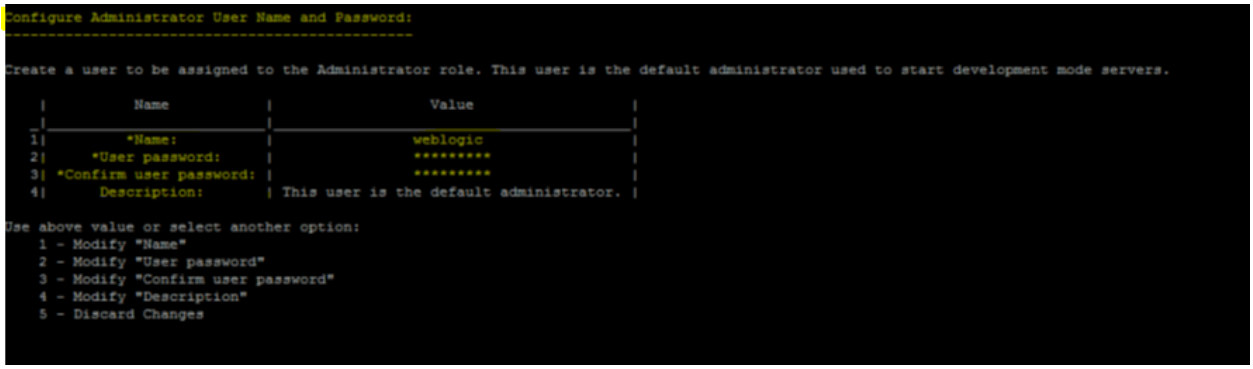




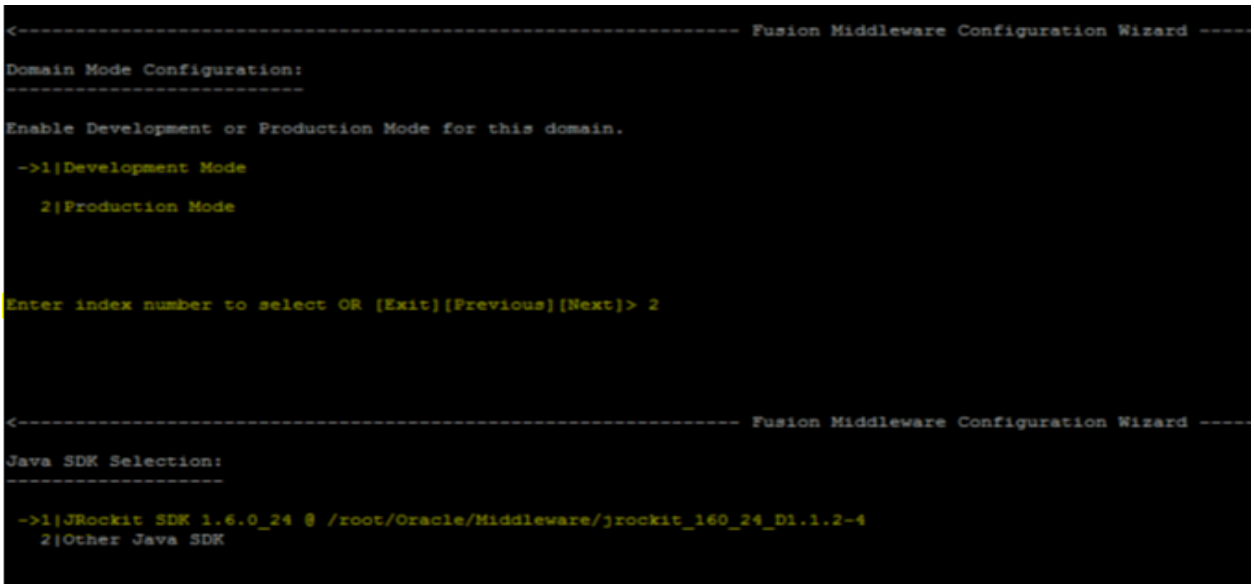
If you want to change the Domain from base domain to any other change it here .



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

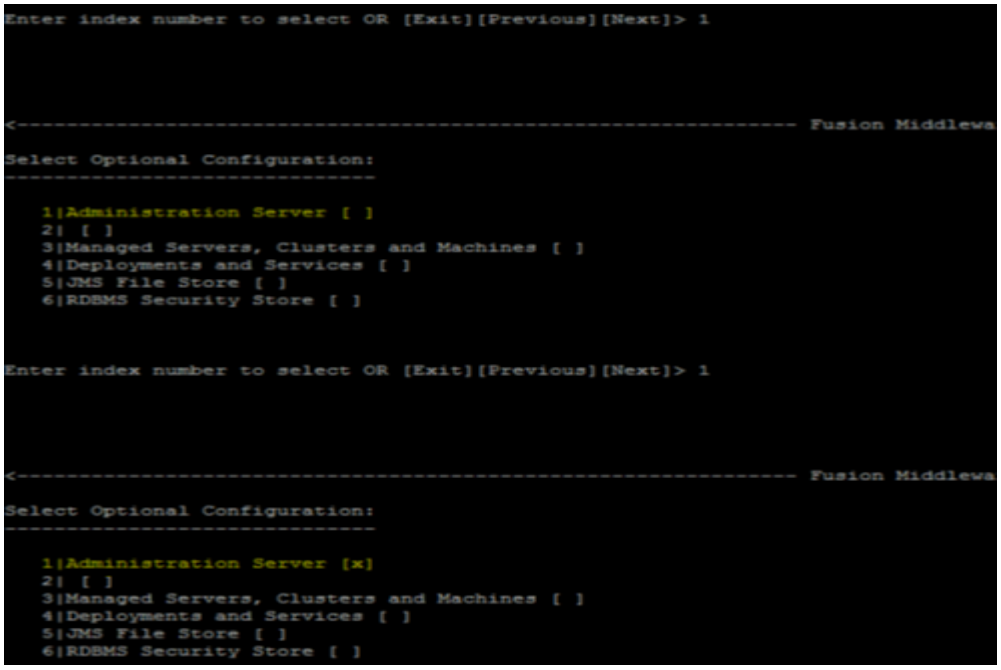


By Default weblogic username is Weblogic and we need to create the password (Password should be alpha numeric with 8 characters)

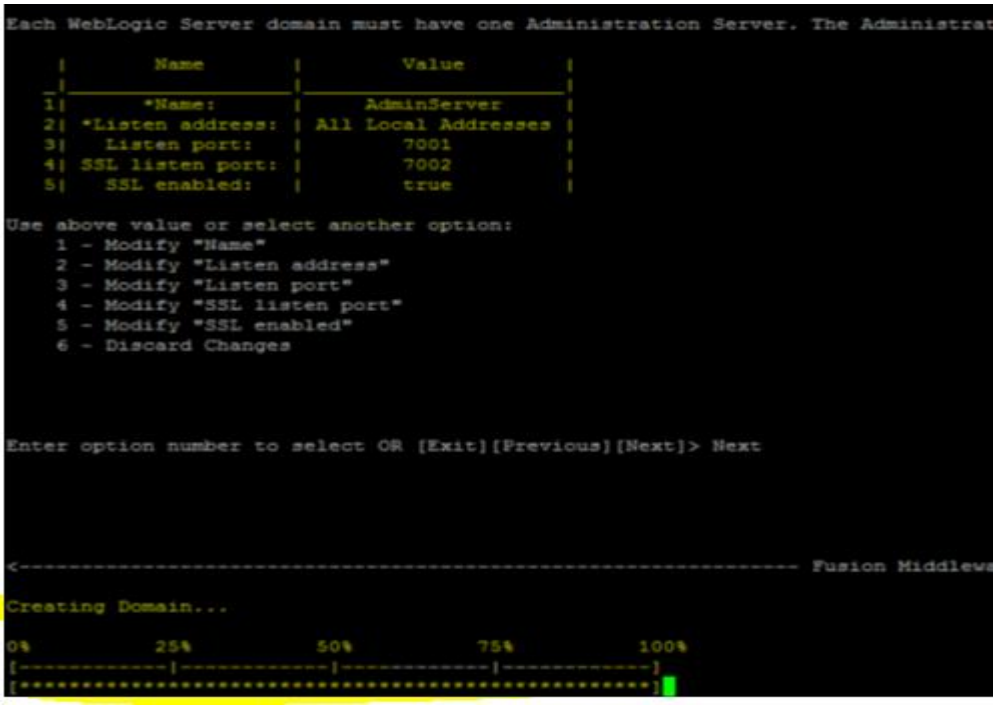


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Based on our requirement need to select the domain either PROD or Dev domain.



Select the components which you want to configuare.



Domain Creation successfully completed.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

START AND STOP SERVICES:

To start and stop Admin Server:

C:\Oracle\Middleware\user_projects\domains\base_domain\bin\

StartWebLogic.cmd or sh (Linux)

To Stop:

stopWebLogic.sh

To start and stop Manage Server:

C:\Oracle\Middleware\user_projects\domains\base_domain\bin\

startManagedWebLogic.sh (Manage server Name)

To Stop:

stopManagedWebLogic.sh (or) cmd (Manage server Name)

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

BOOT.PROPERTIES:

A boot identity file contains the user credentials for starting and stopping an instance of WebLogic Server. An administration server or managed server can refer to this file for user credentials instead of prompting at the command line to provide them. Because the credentials are encrypted, using a boot identity file is more secure than storing plain text credentials in a startup or shutdown script. There can be a different boot identity file for each server in a domain.

If you choose Development Mode when creating a domain by using the Configuration Wizard, a boot identity file is automatically created for the administration server. This tutorial covers creating a boot identity file manually for a Production Mode domain.

If you use Node Manager to start managed servers rather than running start scripts manually, you do not need to create boot identity files for them. Node Manager creates its own boot identity files and stores them under each server's directory in the data/nodemanager subdirectory.

Creating a Boot Identity File:

To create a boot identity file for the administration server, perform the following steps:

Step 1:

Servers should be in shout down state.

$ ./stopWebLogic.sh

Step 2:

In a Terminal window, navigate to the domain directory.

/oracle/Middleware/user_projects/domains/harish_domain/servers/AdminServer/

Note: If your administration server has a different name, cd into that directory under servers.

Create a subdirectory there called security and cd into it:

$ mkdir security
$ cd security

```
[oracle@host01 ~]$ cd /u01/app/oracle/Middleware/user_projects/
[oracle@host01 user_projects]$ cd domains/dizzyworld/
[oracle@host01 dizzyworld]$ cd servers/AdminServer/
[oracle@host01 AdminServer]$ mkdir security
[oracle@host01 AdminServer]$ cd security
[oracle@host01 security]$
```


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Note: If there is already a security subdirectory under the administration server's directory, you do not need to create it.

In the security directory, create a text file called boot.properties and edit it. In this tutorial, the gedit editor is used.

$ vi boot.properties

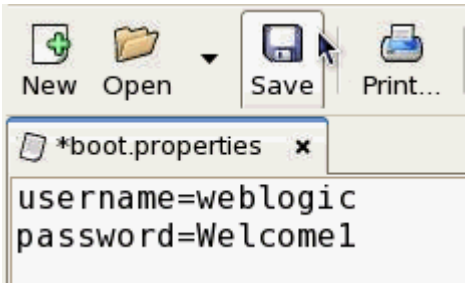
```
[oracle@host01 security]$ gedit boot.properties
```

Step 3 :

In the file, put two lines:

username=weblogic
password=Welcome1

Save the file and exit the editor.



Note: The administrator username and password for your domain may be different. Make sure the values following the = are correct for your domain.

Step 4:

Now restart the administration server. In a Terminal window, navigate to the domain directory, and enter the command:

$ ./startWebLogic.sh

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

```
.
.
JAVA Memory arguments: -Xms256m -Xmx512m  -XX:MaxPermSize=256m
.
WLS Start Mode=Production
.
CLASSPATH=/u01/app/oracle/Middleware/patch_wls1211/profiles/default/sys_m
anifest_classpath/weblogic_patch.jar:/u01/app/oracle/Middleware/patch_oep
e100/profiles/default/sys_manifest_classpath/weblogic_patch.jar:/u01/app/
oracle/Middleware/patch_ocp371/profiles/default/sys_manifest_classpath/we
blogic_patch.jar:/u01/app/oracle/Middleware/jdk160_29/lib/tools.jar:/u01/
<Mar 20, 2012 7:05:05 PM UTC> <Notice> <WebLogicServer> <BEA-000329> <Sta
rted the WebLogic Server Administration Server "AdminServer" for domain "
dizzyworld" running in production mode.>
<Mar 20, 2012 7:05:05 PM UTC> <Notice> <WebLogicServer> <BEA-000365> <Ser
ver state changed to RUNNING.>
<Mar 20, 2012 7:05:05 PM UTC> <Notice> <WebLogicServer> <BEA-000360> <The
 server started in RUNNING mode.>
```

Notice that you are not prompted for the username and password.

Step 5:

To see that the username and password are safe in the file, in a Terminal window, return to the security directory and display the contents of the boot.properties file.

$ cat boot.properties

```
[oracle@host01 security]$ cat boot.properties
#Tue Mar 20 19:04:55 UTC 2012
password={AES}TVOpZ4CVTEkJwXUVU9guh9HtLKZOE4+Mmwu2FjI6/zM\=
username={AES}MhcB2OWcAqaKUpc26w439VmIYqwvbaX01IFVGHi5/jI\=
```

You can see that the username and password have been encrypted by the server. A comment has also been added with the timestamp when the encryption occurred.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com www.weblogic4you.blogspot.com

OVERVIEW OF NODE MANAGER:

Nodemanager is an a utility to start and stop manage servers from the console. Nodemanager having one additional feature call crash recovery enable(Any unexpected failure nodemanager will restart automatically)

Node Manager is a Java utility that runs as separate process from WebLogic Server and allows you to perform common operations tasks for a Managed Server, regardless of its location with respect to its Administration Server. While use of Node Manager is optional, it provides valuable benefits if your WebLogic Server environment hosts applications with high availability requirements.

If you run Node Manager on a machine that hosts Managed Servers, you can start and stop the Managed Servers remotely using the Administration Console or from the command line. Node Manager can also automatically restart a Managed Server after an unexpected failure.

NODE MANAGER IS DOMAIN-INDEPENDENT:

Nodemanager is an Machine dependent so one nodemanager is enough for one machine.

A Node Manager process is not associated with a specific WebLogic domain. Node Manager resides outside the scope of a domain, and you can use a single Node Manager process to start Managed Servers in any WebLogic Server domain that it can access, Managed Server 2 and Managed Server 3 could be in separate domains, and controlled by a single Node Manager process.

SHUT DOWN FAILED MANAGED SERVERS:

Node Manager periodically checks the self-reported heath status of Managed Servers that it has started. By default, Node Manager issues a health query to a Managed Server every 180 seconds.

Node Manager automatically kills a Managed Server that reports its health state as "failed", if the Managed Server's Auto Kill If Failed attribute is true. By default, the Auto Kill If Failed attribute is false. If you want Node Manager to restart a Managed Server that is hung, set Auto Kill If Failed to true, on the Server—>Configuration—>Health Monitoring tab for the Managed Server.

If a Managed Server does not respond to three consecutive health queries in a row, Node Manager considers the Managed Server to be "failed", and shuts it down, if Auto Kill If Failed is set.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

CONFIGURE NODEMANAGER:

Step 1:

Configure a Machine in the console.

Login to console>Click Machines in the domain dir>Take lock @ edit> Create New Machine>

Machine Name : MachineA
Hostname: 192.168.1.*
Type : SSL/Plain
Port : 556

Step 2:

Start Nodemanager under the location .
/Oracle/app/Middleware/wlserver-10.3/server/bin > $startNodemanager.sh
Once you will run the Nodemanager then the nodemanager properties will get create under the nodemanager folder.
/Oracle/app/Middleware/wlserver-10.3/common/Nodemanager/nodemanager.properties

Edit the below Nodemanager properties file as per your machine configuration in the console.
Listen Address: *******
Port: ********
SSL : true/false (Based on your machine configuration in the console)
Crash Recovery Enable: false/true (By default this property is false change it to true)
Start script Enable: true/false ()

Step 3:

After modify the properties need to start Nodemanager.

Step 4:

Now verify the nodemanager is reached or not .
Go to > Console> Machine>Monitoring>
Here you will able to see Reachable if it reached or you will able to see the java error.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

WEBLOGIC SERVER CLUSTER:

A cluster is a group of Managed Servers running simultaneously and working together to provide increased scalability and reliability.

Benefits of Clustering:

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.

Communication in a Cluster:

When servers are in a cluster, these member servers communicate with each other by sending heartbeats and indicating that they are alive. For this communication between the servers, either unicast or multicast messaging is used. This is chosen from the admin console in Cluster -> Configuration -> Messaging -> Messaging Mode.

There are two types of communication:

-Unicast Communication
-Multicast Communication

Unicast Communication:

For the member servers in the cluster, group leaders are chosen and only those group leaders communicate with the servers among the group and these leaders notify each other about the availability of all the other servers.

Multicast Communication

In this mechanism there will be no leader and no member .All manage servers act as a leader .

Each server communicates with every member server in the cluster. Which means heartbeats are sent to every server.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Cluster Types:

There are two types of cluster in WebLogic.

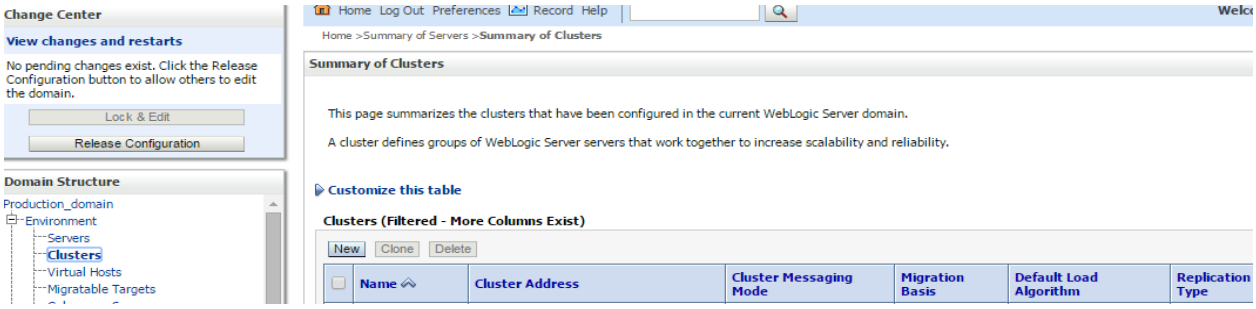
- VERTICAL CLUSTER
- HORIZONTAL CLUSTER

VERTICAL CLUSTER :

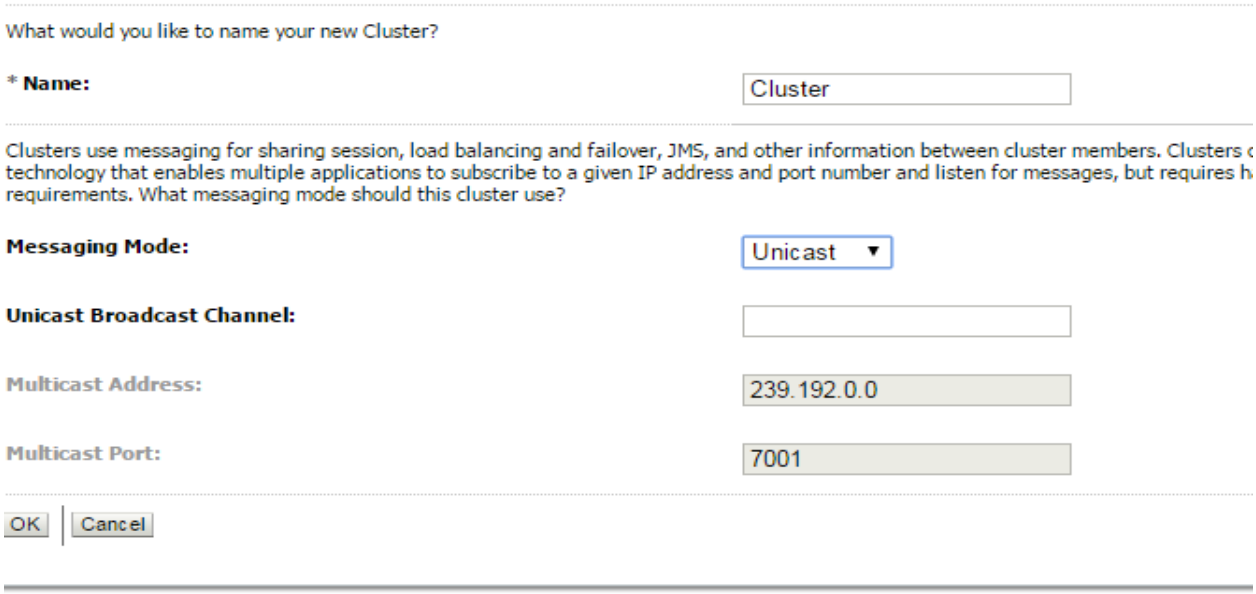
In Vertical clustering, multiple application server instances are hosted on the same physical machine. This type of clustering provides increased efficiency, load balancing and process failover. However, if hardware fails then there may not be ready alternative.

Vertical Cluster Configuration:

Step 1: Login to the weblogic console.

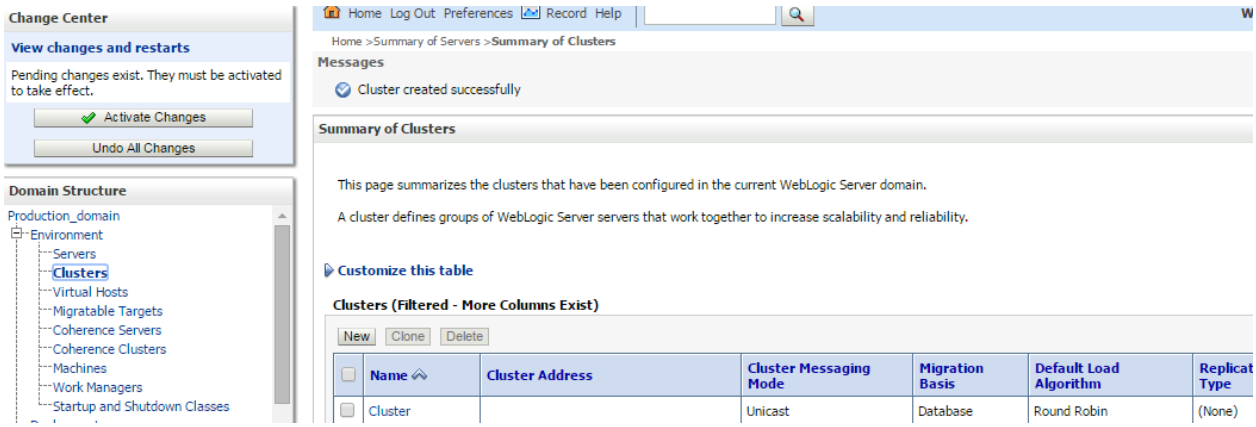


Take Lock & Edit session and create new cluster.

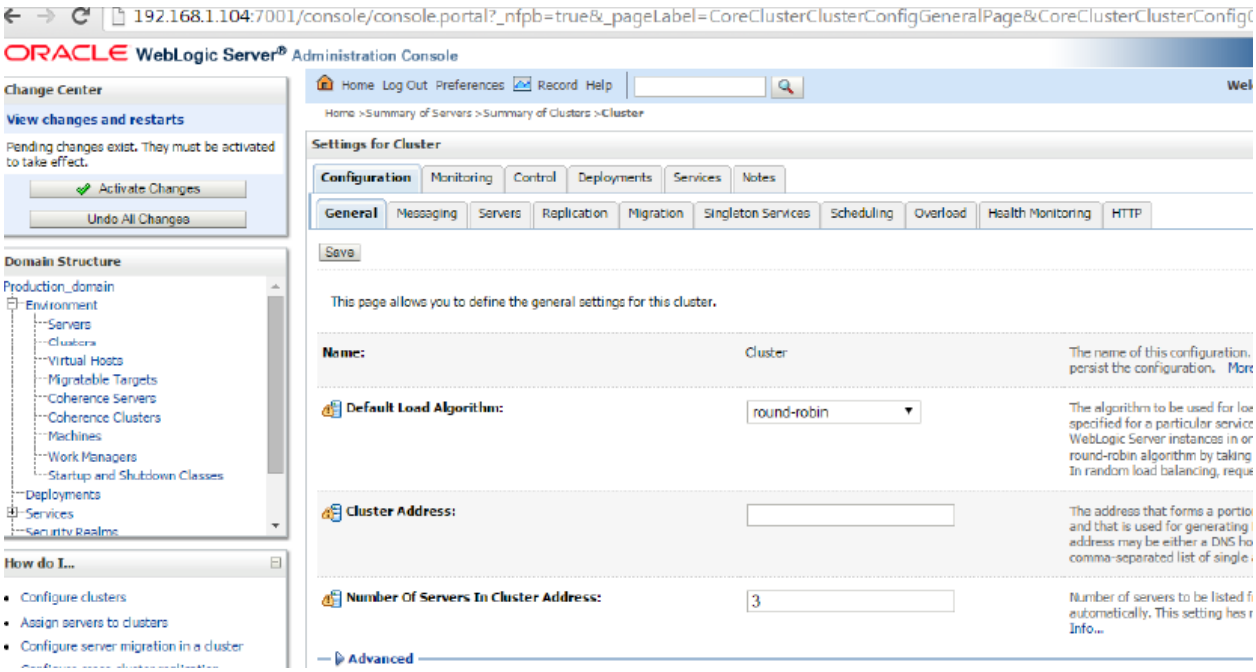


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

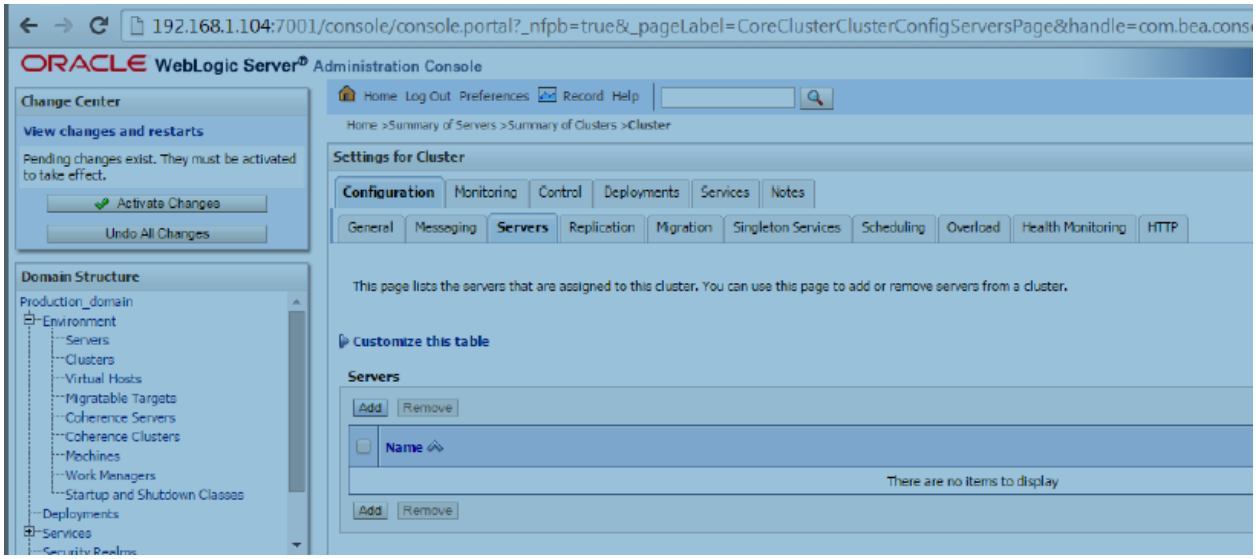
Name=Cluster
Messaging Mode=Unicast or Multicast (as per your requirment)
Click OK



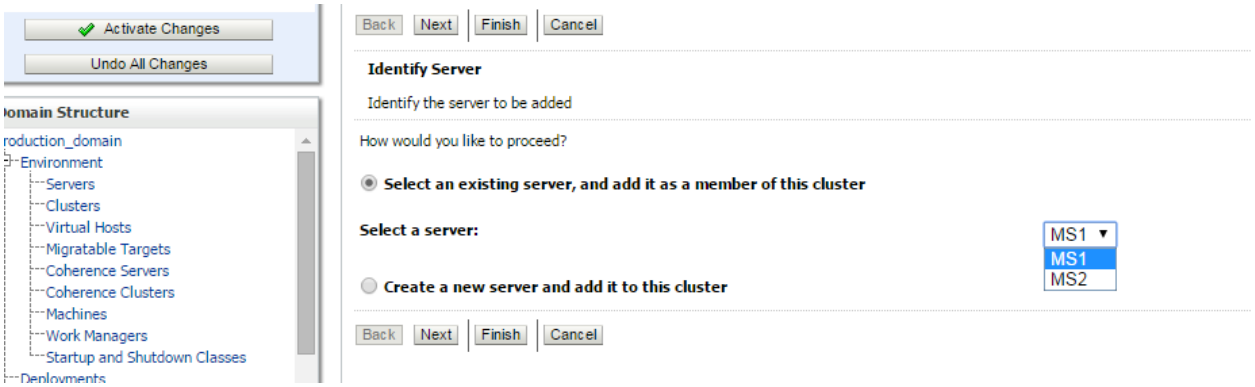
Click on the cluster (which u created)



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



Click on add



Select the servers and finish.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

HORIZONTAL CLUSTER:

In horizontally clustered environment, cluster-enabled application is deployed on multiple physical machines. Each machine is available for requests. Horizontal clusters offers protection over hardware failure, increases efficiency, provides load balancing and process failover. However, since there are many number of physical machines involved the installation and maintenance cost increases proportionally.

Horizontal cluster Configuration:

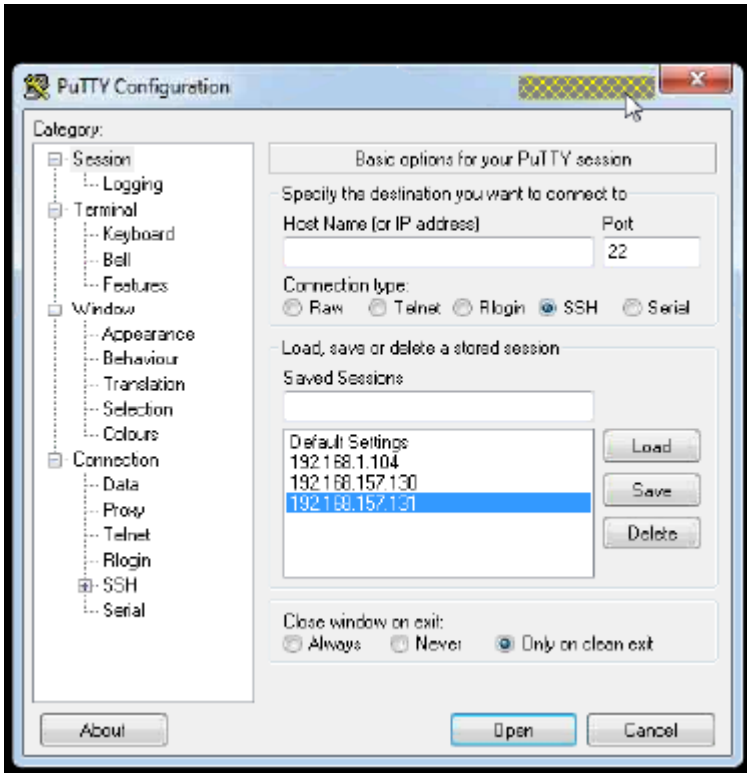
Step-1:

Here I have two vm's –

192.168.157.130

192.168.157.131

Now I am creating new Domain in Machine-A(192.168.1.130)



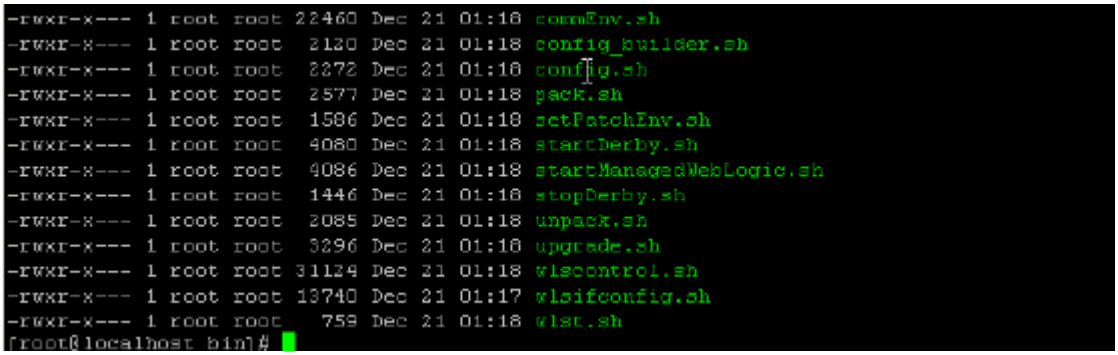
HI N Technologies

Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068

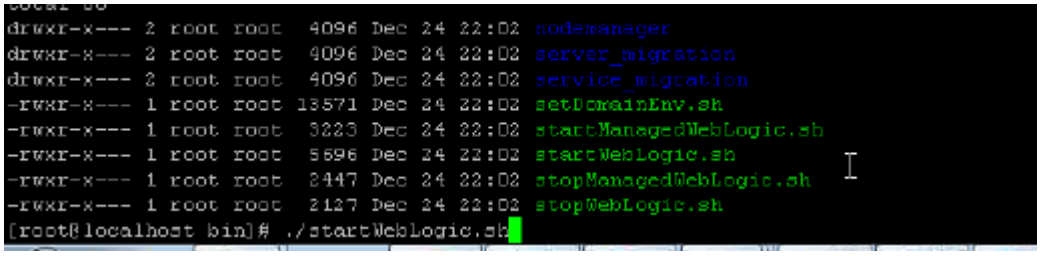
PH: 080-42288565 Mob: +91-7676333847

www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

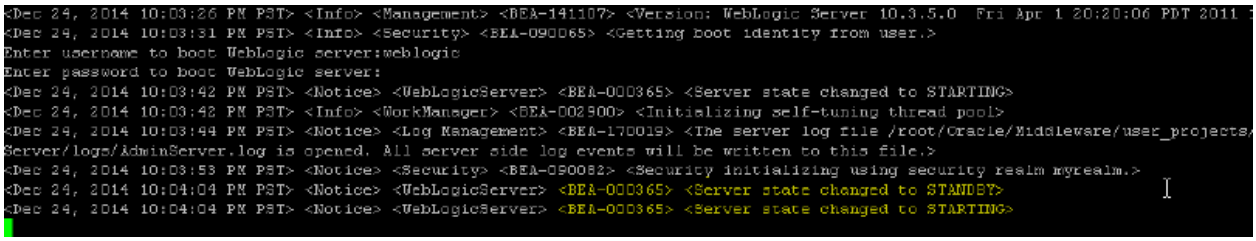
Login in to Machine-A via putty.



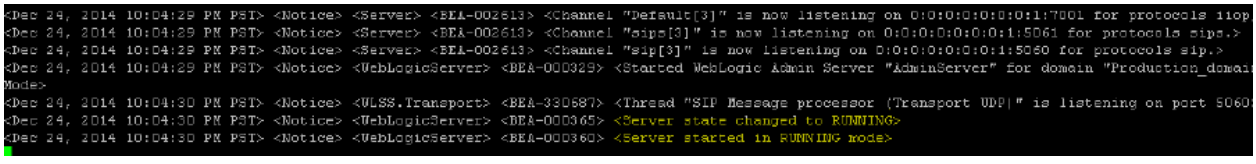
Step 2:
Create new Domain
Domain name =Production_domain



After creating the domain start the Admin server.
/root/Oracle/Middleware/user_projects/domain/Production_domain/bin
./startWeblogic.sh



Starting

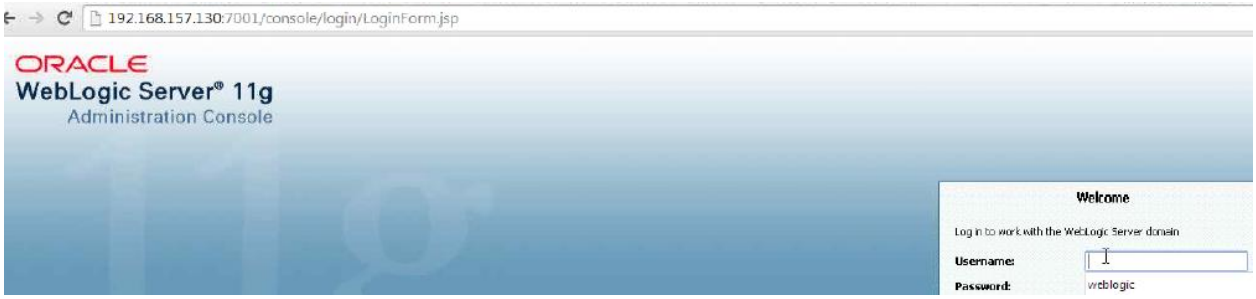


Running Mode.

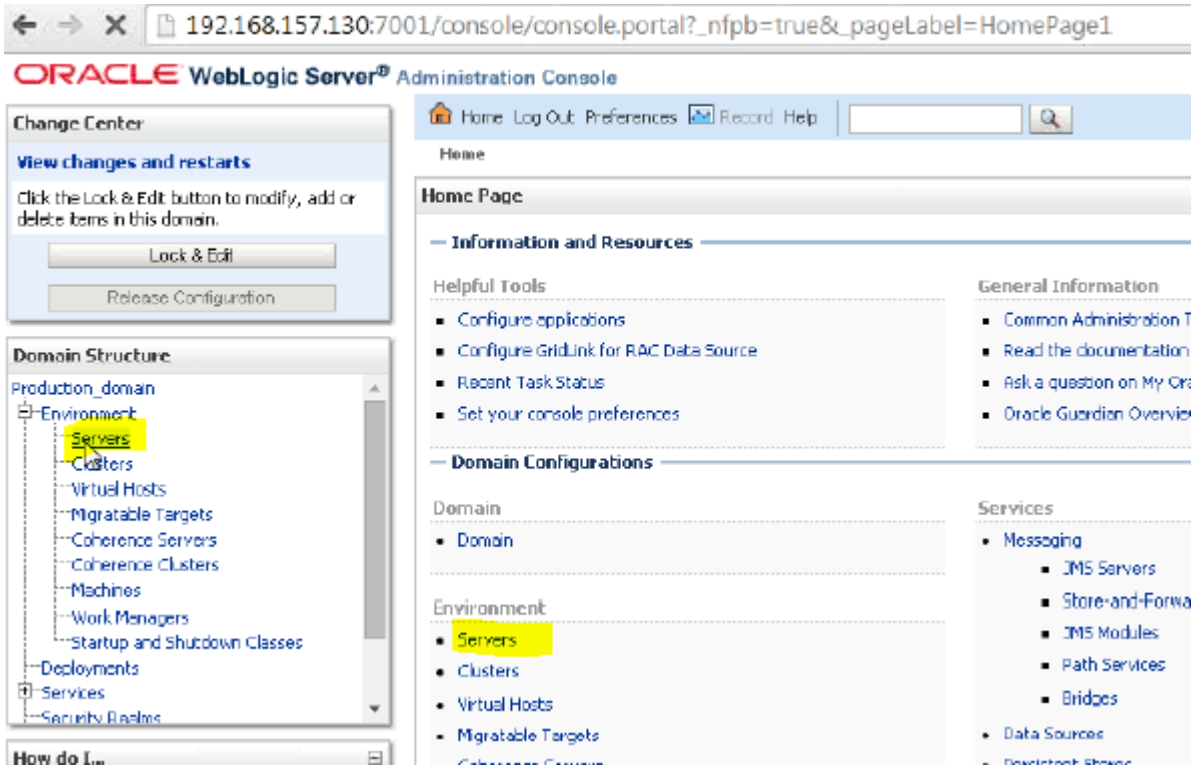
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Step 3:

Login to WebLogic console.



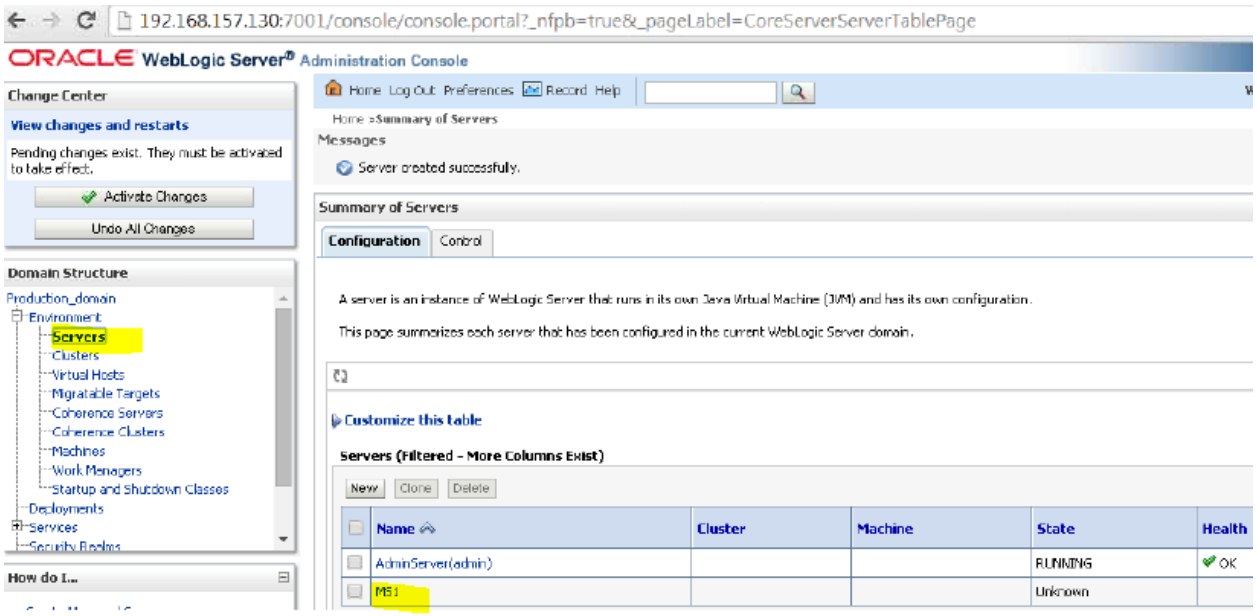
Login with the weblogic and password.



Step 4:

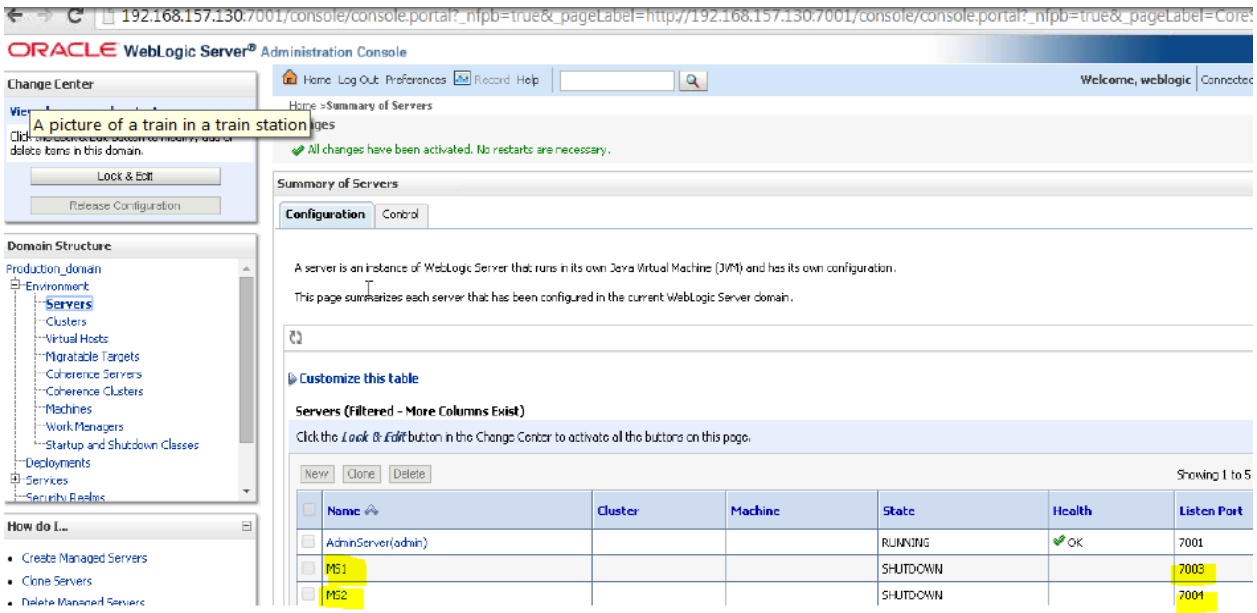
Create servers

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



Create MS1, MS2, like below.
MS1 give Machine-A ip address
MS2 give Machine –B ip address

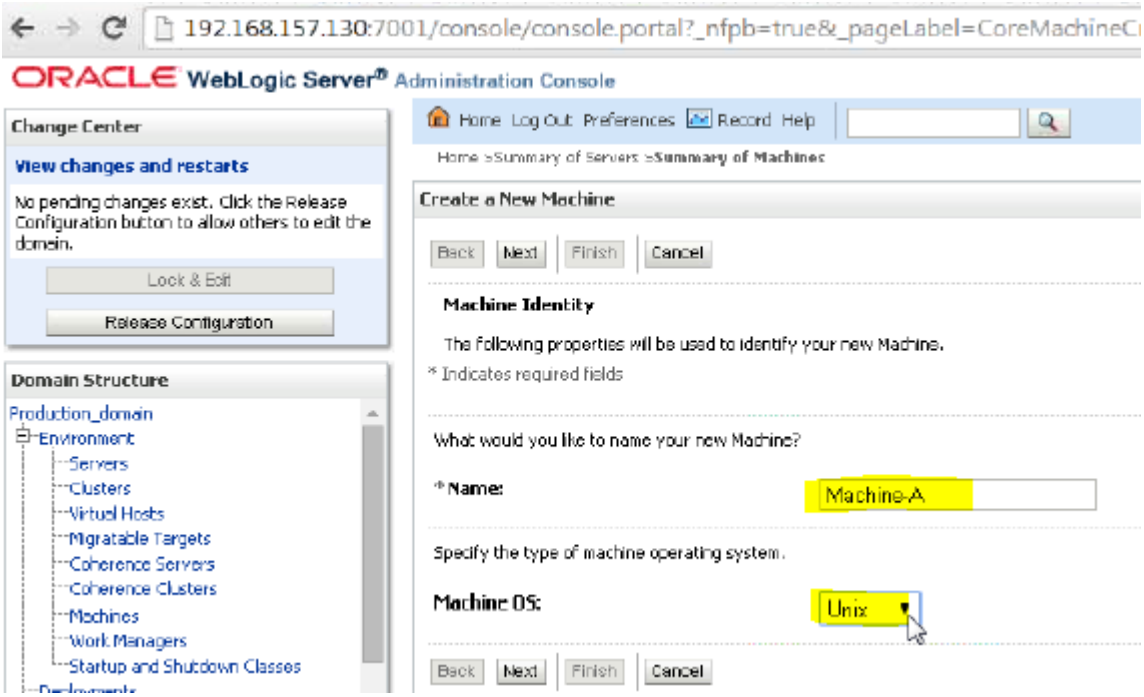
Manage server Name	IP Address	PORT
MS1	192.168.157.130	7004
MS2	192.168.157.131	7004

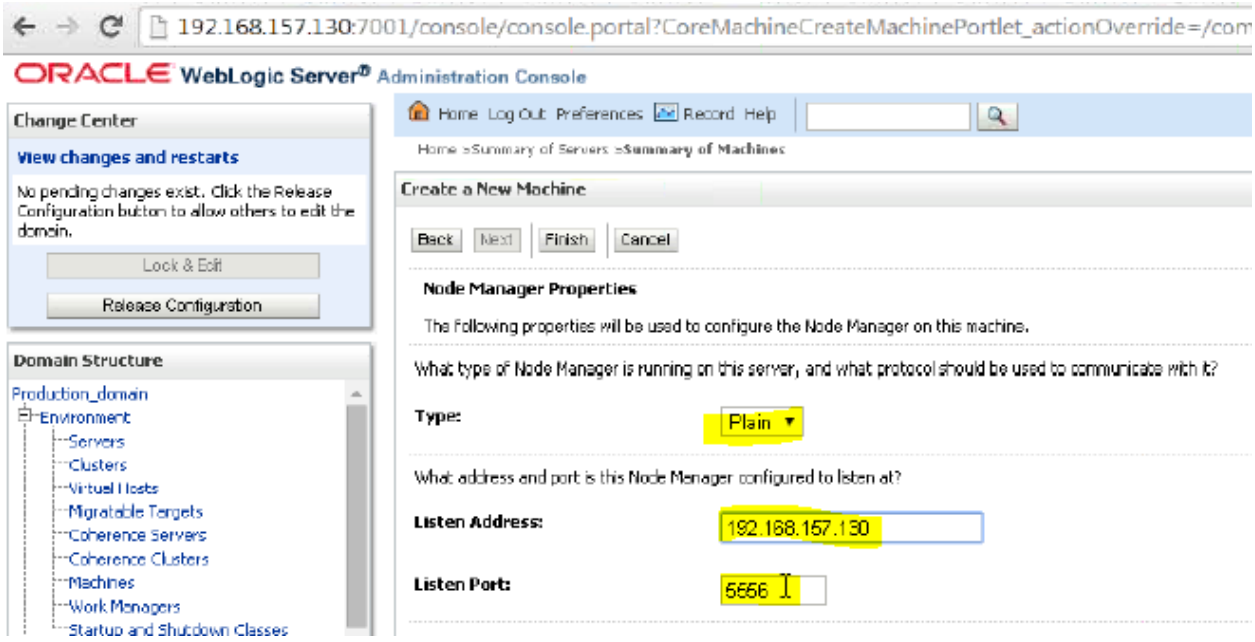


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

After creating manager you can able to see the above screen.

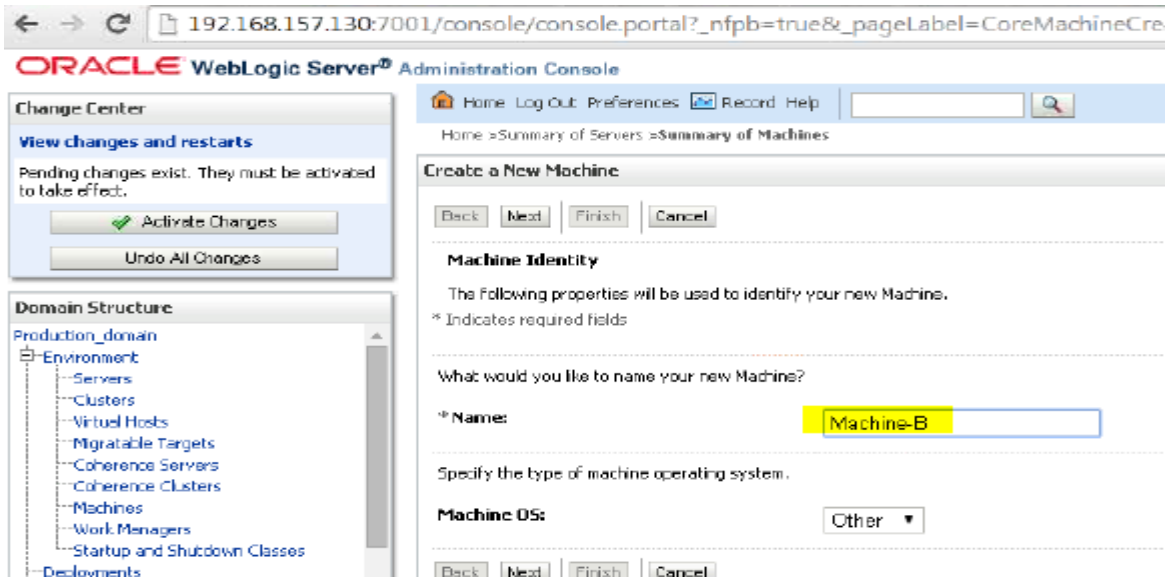
Step-5:
Create Machine in the Weblogic console
Click on the machines in the and create 2 machines

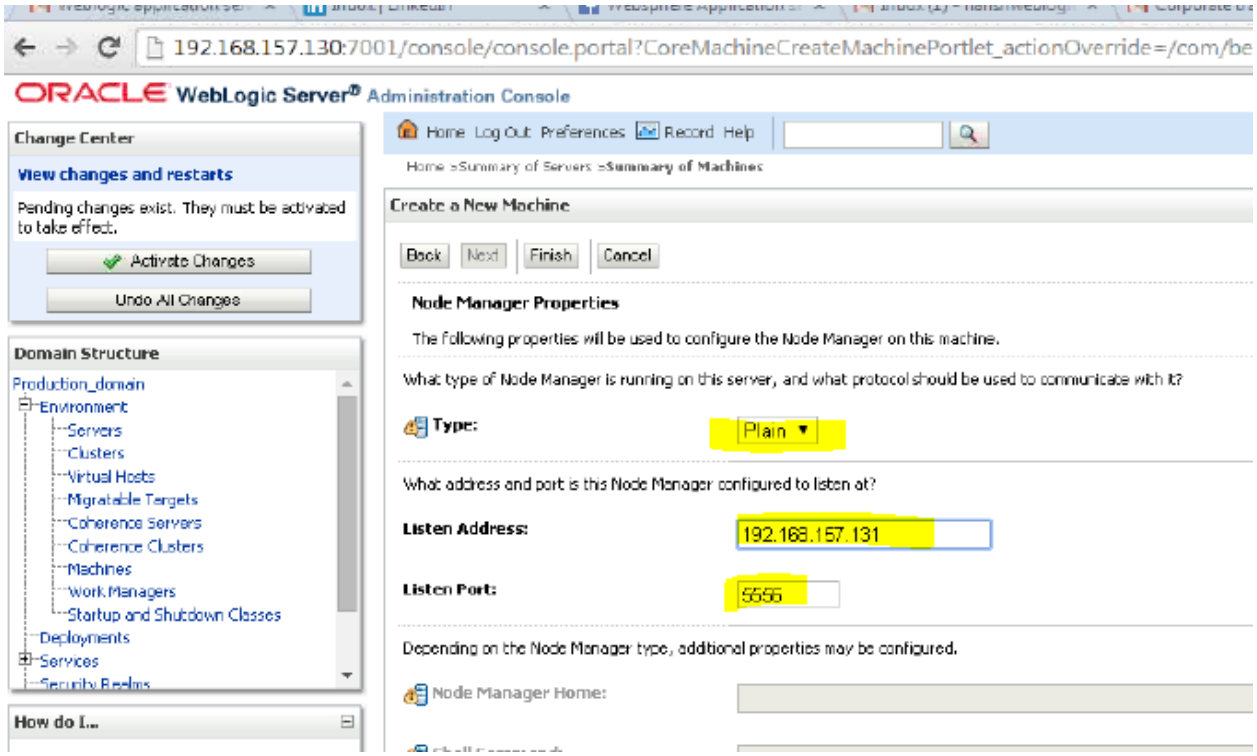




HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

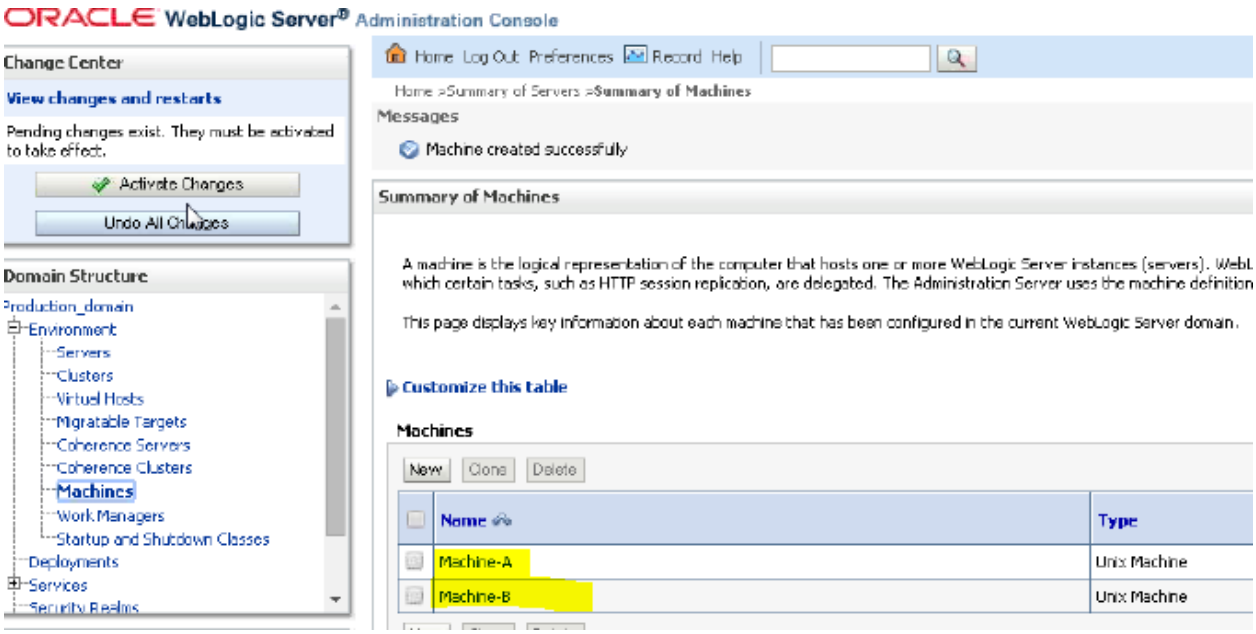
Click next
Type=Plain (because we are not using ssl certificates)
Listen Address=192.168.157.130
Now create machine B.





HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Click next
Type=Plain (because we are not using ssl certificates)
Listen Address=192.168.157.131



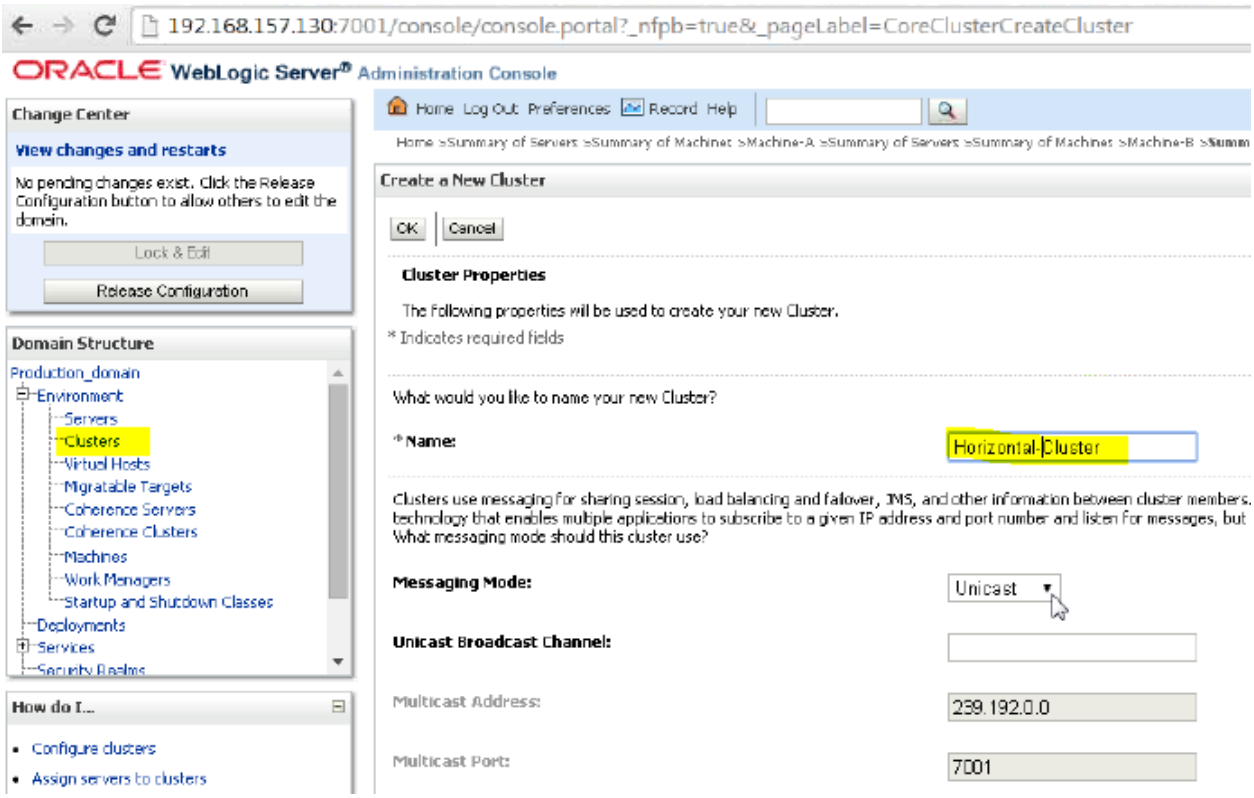
Assign manage server to the machine – MS1 belongs to Machine-A and MS2 belongs to Machine-B.

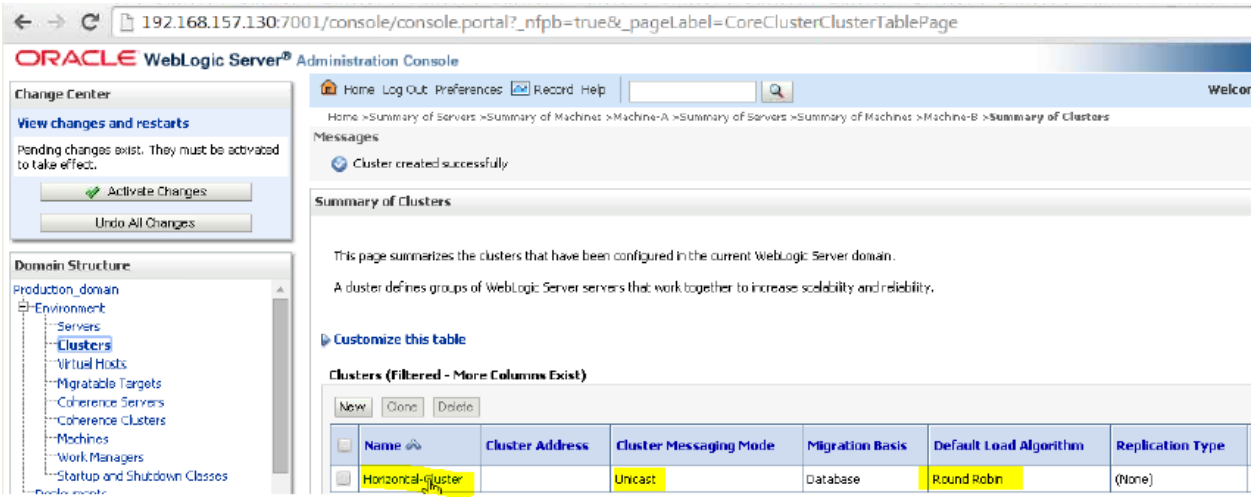
Manage-server	**Machine**	IP-address	port
MS1	Machine-A	192.168.157.130	7003
MS2	Machine-A	192.168.157.131	7004

Step 6:

Now create cluster.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

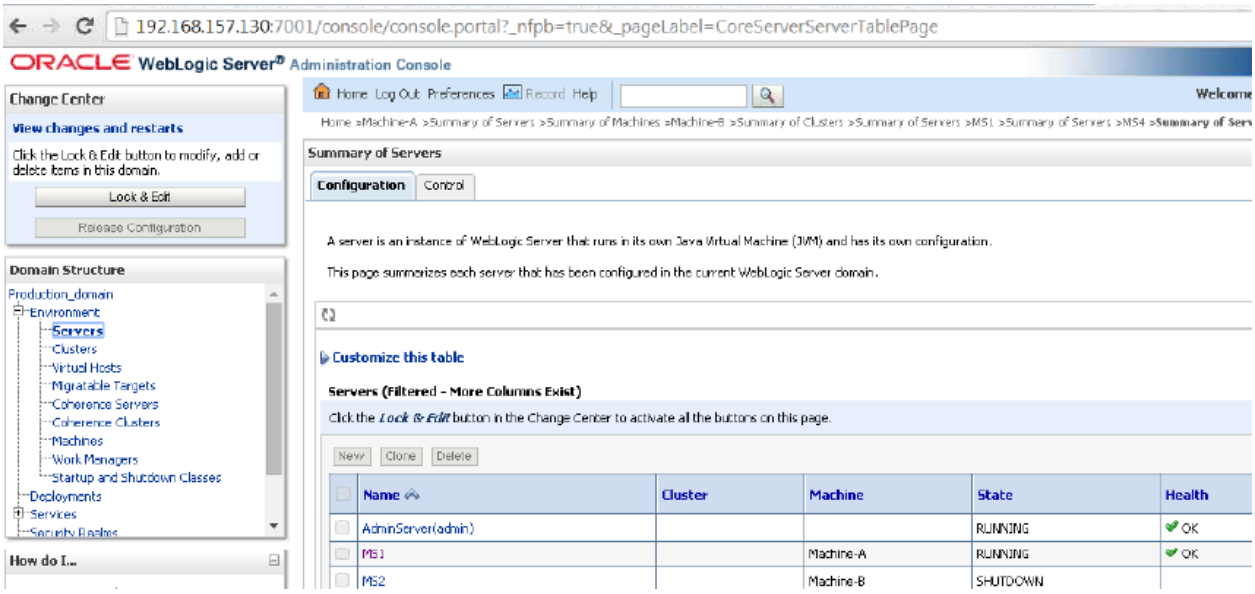




HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

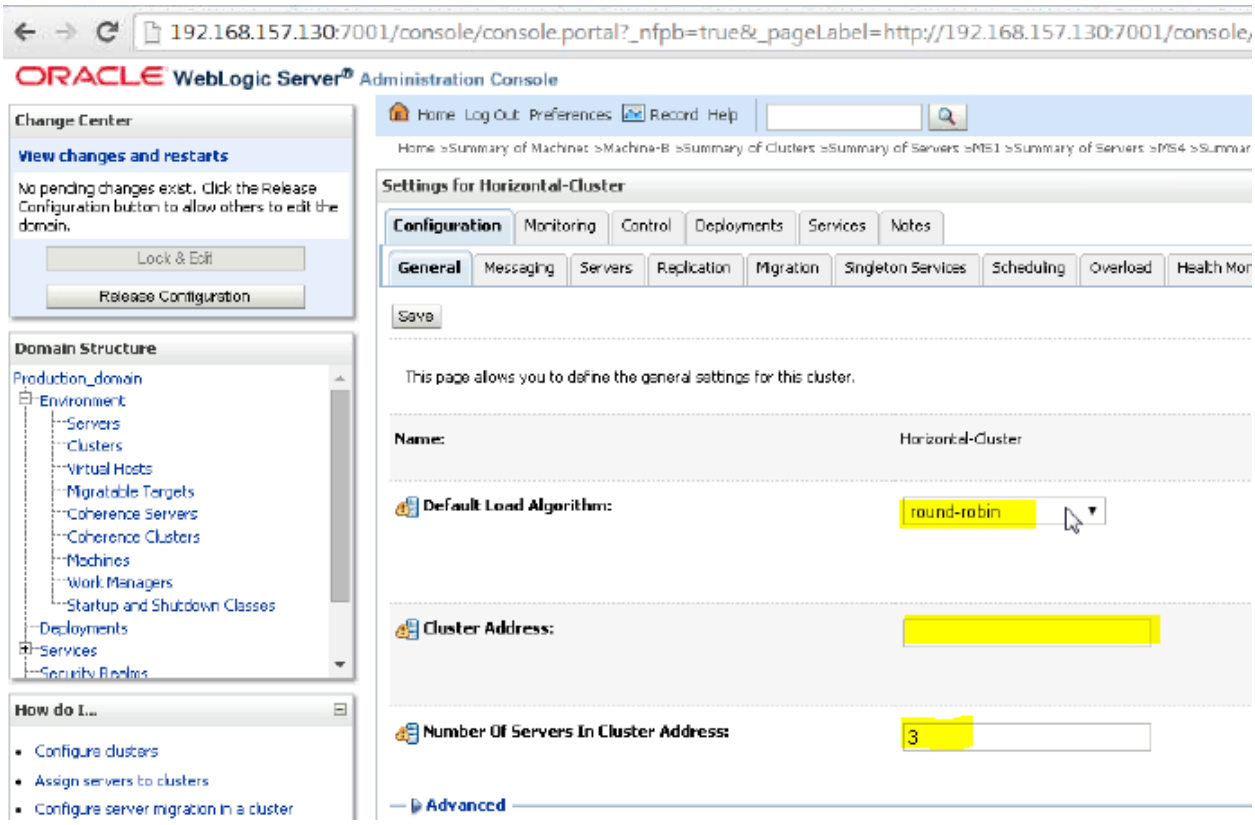
Horizontal cluster is created.
Now add all manage server in to Cluster

Manage server	Cluster	Machine	Ip-address	Port
MS1	Horizontal -Cluster	Machine-A	192.168.157.130	7003
MS2	Horizontal -Cluster	Machine-A	192.168.157.131	7004



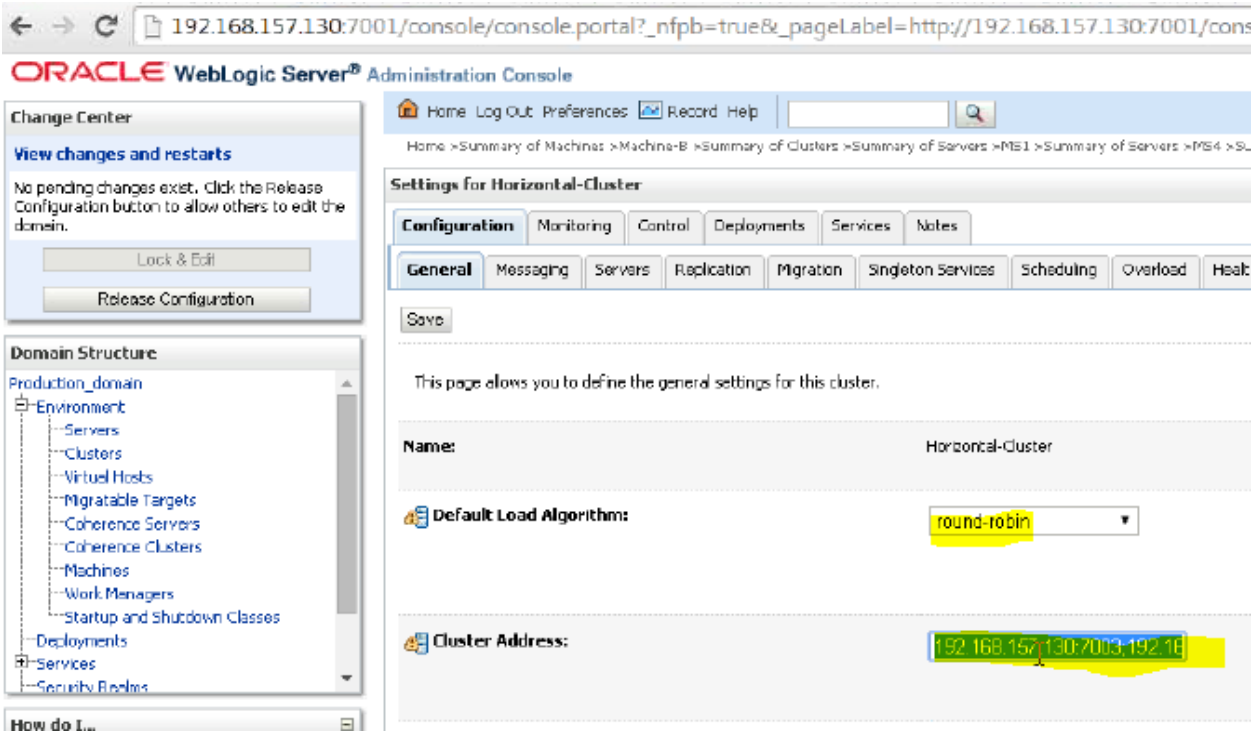
Step-7:
Now add the cluster address in the Horizontal Cluster>General

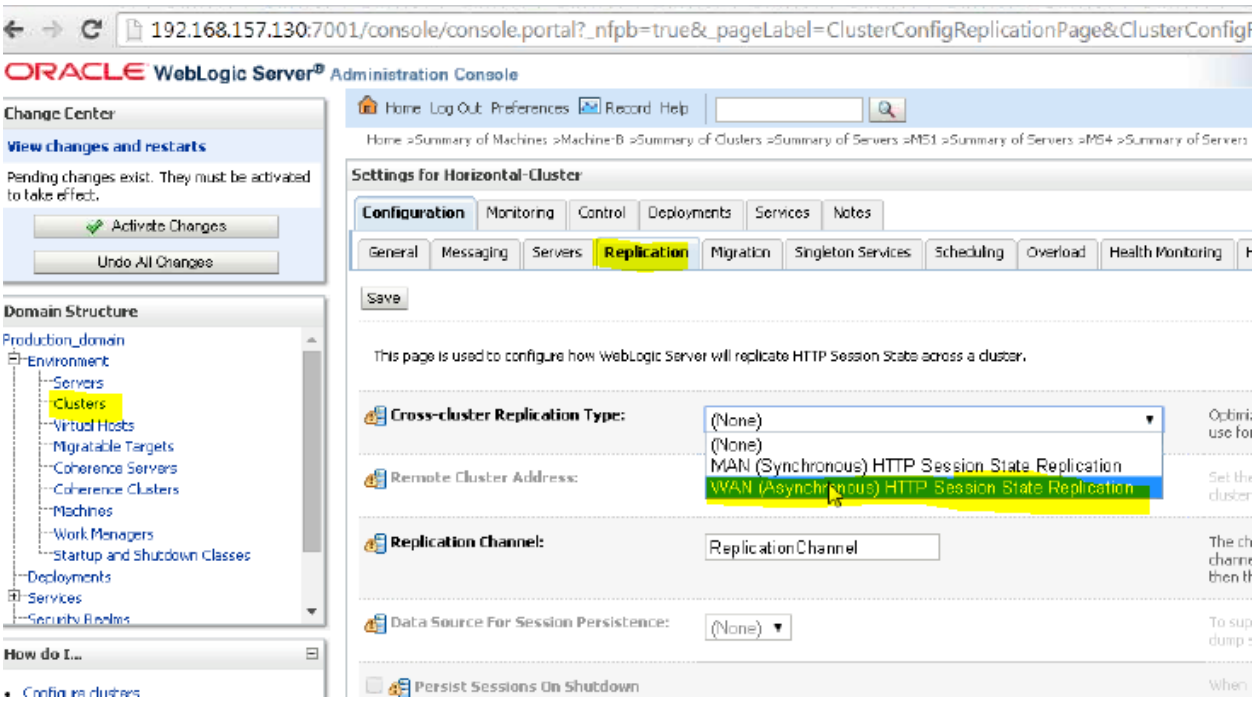
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



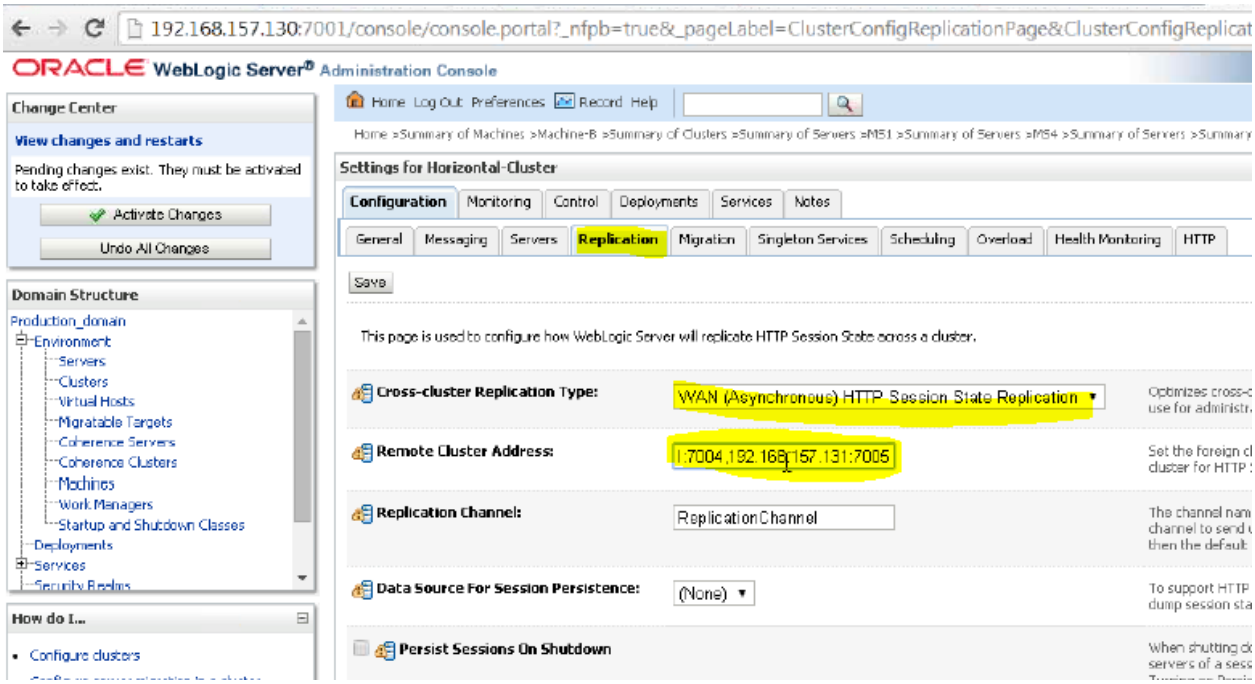
Select the Algorithm type: round-robin (default) ---select whatever you want
Cluster Address: 192.168.157.130:7003, 192.168.157.131:7004
Number of servers in cluster Address: 2

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

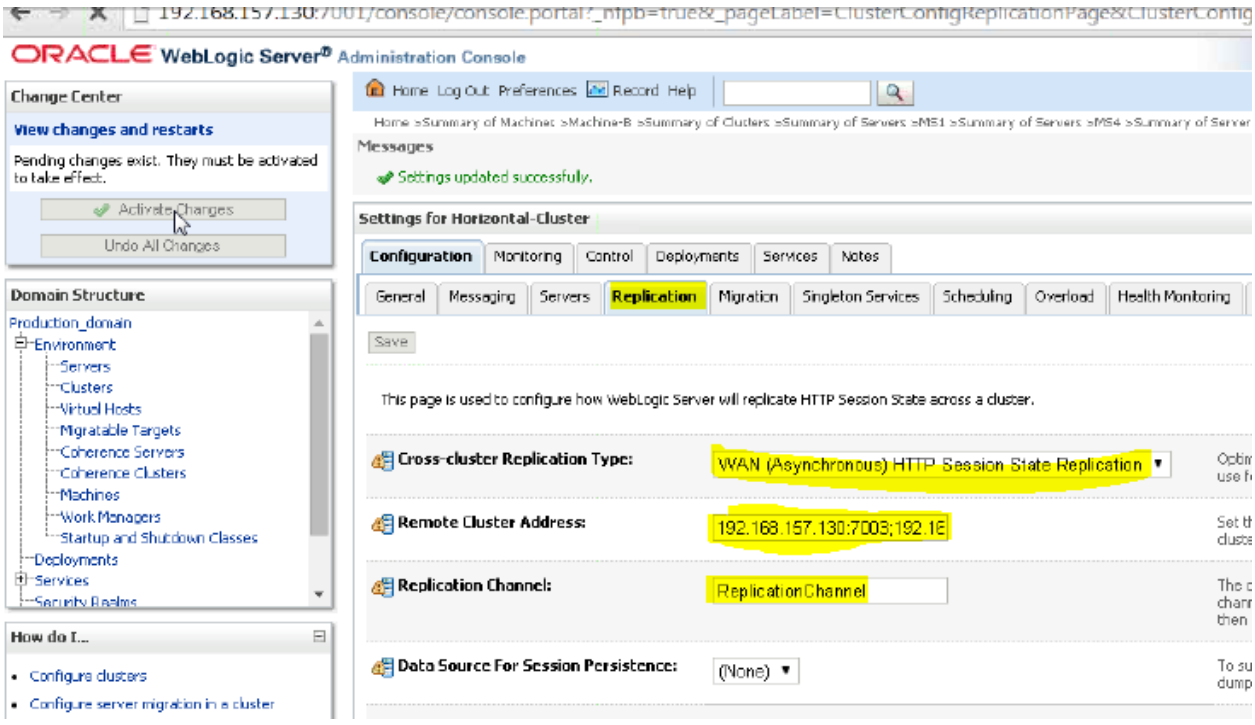




HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



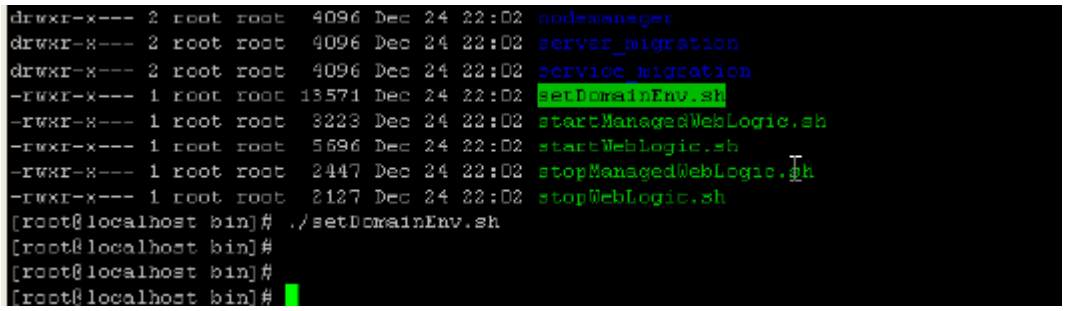
Cross-Cluster Replication Type: WAN(Asynchronous)HTTP Session State Replication
Remote Cluster Address: 192.168.157.130:7003, 192.168.157.131:7004



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Click saves.
Click on Active changes.
Now we have done all configurations in the console
1---Created manage servers
2---Created machines
3---Created cluster
Now I am going to pack this entire domain. Before packing the domain stop all server .

Step-8:
Prepare the pack and unpack commands:
Run pack command in machine-A(because we have domain in machine-A)
Run unpack command in machine-B(We want same domain in machine-B)
./pack.sh -domain=/root/Oracle/Middleware/user_projects/domains/Production_domain/ -template=/root/Oracle/Middleware/user_templates/Production_domain.jar -managed=true -template_name="Production domain for horizontal cluster "
./unpack.sh -template=/root/Oracle/Middleware/user_templates/Production_domain.jar -domain=/root/Oracle/Middleware/user_projects/domains/Production_domain



Run setDomainEnv.sh
Go to the root\Oracle\Middleware\wlserver-10.3\common\bin\
pack.sh

Run the pack.sh command

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

```
[root@localhost bin]# ./pack.sh -domain=/root/Oracle/Middleware/user_projects/domains/Production_domain/ -template=/root/Oracle/Middleware/user_t
ain.jar -managed=true -template_name="Production domain for horizontal Cluster "
<< read domain from "/root/Oracle/Middleware/user_projects/domains/Production_domain"
>>  succeed: read domain from "/root/Oracle/Middleware/user_projects/domains/Production_domain"
<< set config option Managed to "true"
>>  succeed: set config option Managed to "true"
<< write template to "/root/Oracle/Middleware/user_templates/Production_domain.jar"
.....................................................................................
>>  succeed: write template to "/root/Oracle/Middleware/user_templates/Production_domain.jar"
<< close template
>>  succeed: close template
[root@localhost bin]#
```

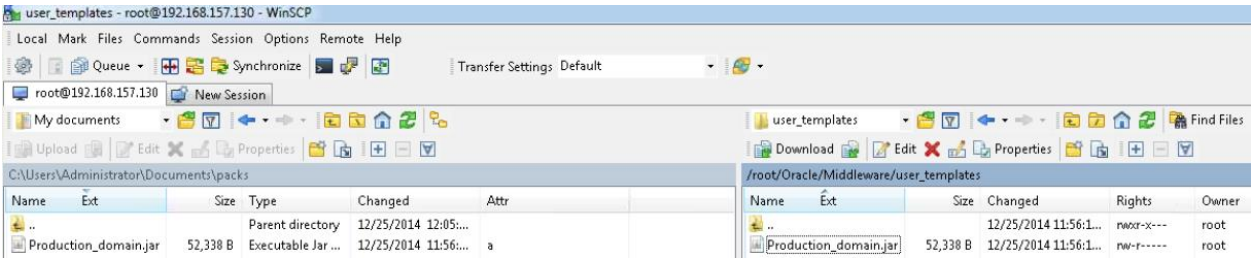
After packing the domain you will able to see above success message.

Domain is packed in to the user_templates folder with the Production.jar format.

Step-9:

Now I am going to send .jar file to machine-B location

Transfer the file over the scp or wincp



Step 10:

Now Go to the Machine-B and unpack the jar file by using unpack.sh.

Go to the /root/Oracle/Middleware/wlserver_10.3/Common/bin

Run the

```
[root@localhost user_templates]# cd /root/Oracle/Middleware/wlserver_10.3/common/bin
[root@localhost bin]# ll
total 164
-rwxr-x--- 1 root root 22460 Dec 24 22:32 commEnv.sh
-rwxr-x--- 1 root root  2120 Dec 24 22:32 config_builder.sh
-rwxr-x--- 1 root root  2272 Dec 24 22:32 config.sh
-rwxr-x--- 1 root root  2577 Dec 24 22:32 pack.sh
-rwxr-x--- 1 root root  1586 Dec 24 22:32 setPatchEnv.sh
-rwxr-x--- 1 root root  4080 Dec 24 22:32 startDerby.sh
-rwxr-x--- 1 root root  4086 Dec 24 22:32 startManagedWebLogic.sh
-rwxr-x--- 1 root root  1446 Dec 24 22:32 stopDerby.sh
-rwxr-x--- 1 root root  2085 Dec 24 22:32 unpack.sh
-rwxr-x--- 1 root root  3296 Dec 24 22:32 upgrade.sh
-rwxr-x--- 1 root root 31124 Dec 24 22:32 wlscontrol.sh
-rwxr-x--- 1 root root 13740 Dec 24 22:30 wlsifconfig.sh
-rwxr-x--- 1 root root   759 Dec 24 22:32 wlst.sh
[root@localhost bin]# ./unpack.sh -template=/root/Oracle/Middleware/user_templates/Production_domain.jar -domain=/root/Oracle/Middleware/user_pro
n_domain
```

```
[root@localhost bin]# ./unpack.sh -template=/root/Oracle/Middleware/user_templates/Production_domain.jar -domain=/root/Oracle/Middleware/user_pro
n_domain

<< read template from "/root/Oracle/Middleware/user_templates/Production_domain.jar"
>>  succeed: read template from "/root/Oracle/Middleware/user_templates/Production_domain.jar"
<< set config option DomainName to "Production_domain"
>>  succeed: set config option DomainName to "Production_domain"
<< write Domain to "/root/Oracle/Middleware/user_projects/domains/Production_domain"
.....................................................................................
>>  succeed: write Domain to "/root/Oracle/Middleware/user_projects/domains/Production_domain"
<< close template
>>  succeed: close template
[root@localhost bin]#
[root@localhost bin]#
```


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

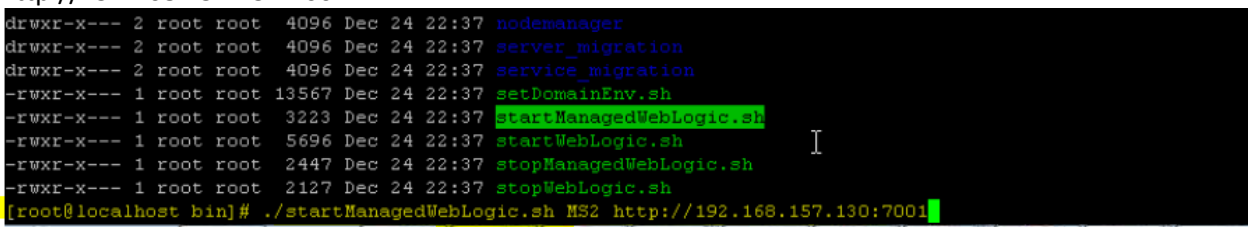
After unpacking the Domain you will able to see the above success message.
Now we have done pack the domain in machine-A and unpacking in machine-B.

Step11:

1) Now start the weblogic server in the machine-A(192.168.157.130) by using ./startweblogic.sh

2) After starting the Adminserver ,open the weblogic console http://192.168.157.130:7001/console

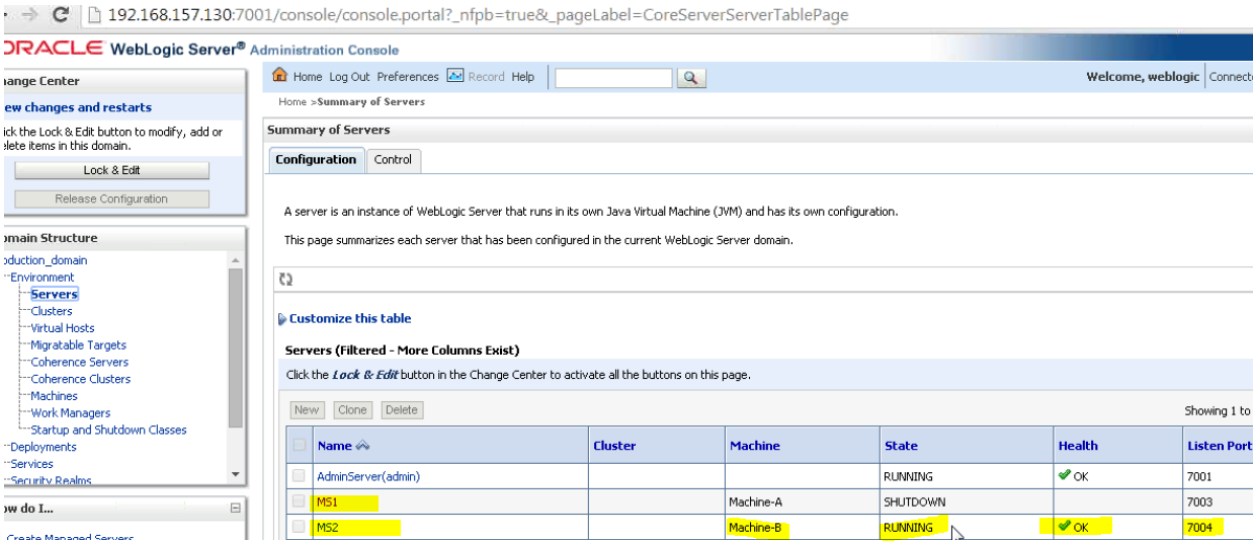
3) Run MS2(manage server) in machine –B(192.168.157.131) by using ./startmanagedweblogic.sh MS2 http://192.168.157.131:7004



Now Server is in running mode.

Step-12:

Now just refresh the Admin server.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

MS2 is going to running mode in the console.
This is the way we are going to configure horizontal cluster.

Configuring HA WebLogic cluster:

This recipe will cover other WebLogic cluster adjustments needed for high availability in production. The parameters are Failure Action, Panic Action, Cluster Address, and Number of Servers in Cluster Address for the cluster, and CrashRecoveryEnabled for Node Manager.

To change the Node Manager CrashRecoveryEnabled parameter, edit the configuration $WL_HOME/common/nodemanager/nodemanager.properties file in all machines. The cluster parameters are changed using the Administration Console or WLST.

To change the Node Manager's parameter:

1. Log in as a wls user to shell and shutdown Node Manager:

```
[wls@prod01]$ ps aux | grep weblogic.NodeManager | grep -v grep |awk '{print $2}'<PID>
[wls@prod01]$ kill -9 <PID>
```

2. Edit nodemanager.properties:

```
[wls@prod01]$ vi $WL_HOME/common/nodemanager/nodemanager.properties
```

3. Locate the CrashRecoveryEnabled parameter and change the line:
   From:
   CrashRecoveryEnabled=false
   To:
   CrashRecoveryEnabled=true
4. Type: wq! to save the file and exit.
5. Start Node Manager:

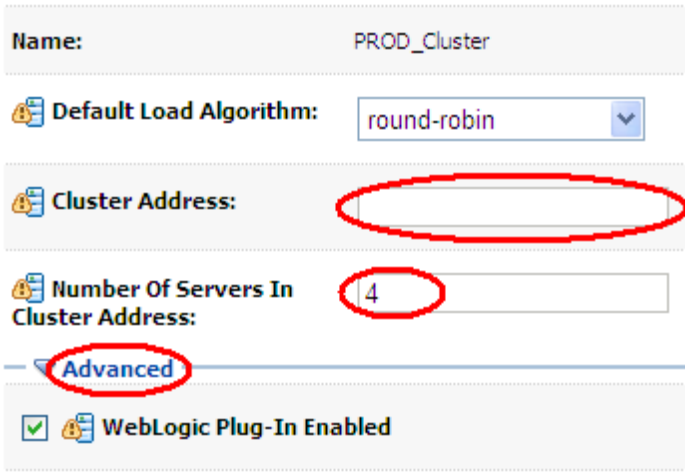
```
[wls@prod01]$ cd $WL_HOME/server/bin
[wls@prod01]$ nohup ./startNodeManager.sh &
```

Repeat the steps on every machine in the domain.

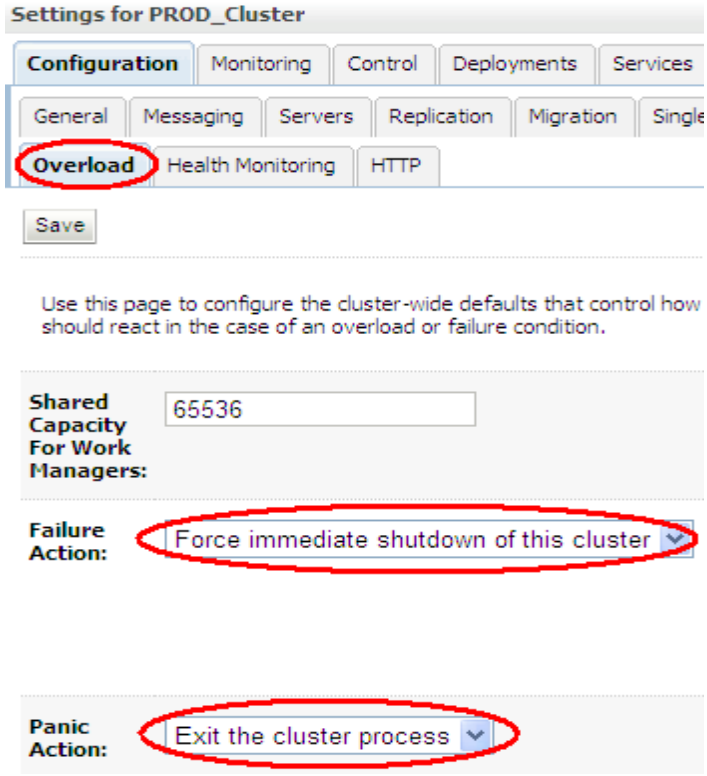
To change the cluster parameters:

1) Access the Administration Console with your web browser at http://prod01.domain.local:7001/console.
2) Click on the Lock & Edit button to start a new edit session.
3) Expand the Environment tree on the left and click on Clusters.
4) Click on the PROD_Cluster cluster to navigate to Configuration | General and type 4 in the Number of Servers in Cluster Address field. Leave the Cluster Address field Empty and click on the Save button.
5) Click on the Advanced link to display extra options and then select the WebLogic Plug-in Enabled checkbox, as shown in the following screenshot:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



6) Click on the Save button.
7) Click on the Overload tab to navigate to Configuration | Overload of PROD_Cluster.
8) Change the Failure Action drop-down menu to Force immediate shutdown of this Cluster and the Panic Action field to Exit the cluster process (as shown in the Following screenshot):



9) Click on the Save button and then the Activate Changes button.
10) Restart all Managed Servers of the PROD_Cluster cluster.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

Failure Action and **Panic Action** are the WebLogic overload settings that can shutdown aManaged Server when it reaches a FAIL state or when it throws an Out of Memory (OOM)PANIC error. Both errors would leave the affected Managed Server in an inconsistent state and would probably hang or return application errors to client requests. The Node Manager
monitors the Managed Servers and restarts the failed instance automatically. Unless there is an analysis of the root cause of these errors, it's recommended to enable both parameters.

CrashRecoveryEnabled is the Node Manager parameter that must be enabled so Node Manager can restart a crashed/failed WebLogic Managed Server instance automatically.

The **Cluster Address**, **Number of Servers in Cluster Address**, and **WebLogic Plug-in Enabled** are cluster configurations for the distribution of requests and load balancing.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

DEPLOYMENTS:

WebLogic Deployments:

Generally we get following types of files to deploy
1) .war --- Web Archive files
2) .ear --- Enterprise Archive files
4) .jar ---Java Archive files

Note: .ear = .war + .jar
In the above formats we cannot directly deploy .jar files in the server, but we can replace.jar which already deployed. Some important file we get in deployment bundle.

.war contain

—–> META-INF/MANIFEST.MF –> Version info etc
—–>WEB-INF/web.xml
—> weblogic.xml (optional)
—–> APP-INF/application.xml (optional)
—-> weblogic-application.xml (optional)
Web.xml is called DD **(deployment descriptor).** It contains details of **classes required and order of their deployment t etc.**
Note:
.war can be deployed anywhere like web-logic server, web-sphere, Glass-fish etc. For deployment on WL, there may be one more optional file
weblogic.xml
DD for .ear **is application.xml** (optional)
If a value is defined in both web.xml and weblogic.ml, then the value in web.xml will be taken into consideration.
An installation guide (IG) or deployment guide is provided by developer to tell if files have to be deployed in any specific order.

Modes of Deployments:

In WebLogic server we can have three modes

- **Stage**
- **NO-Stage**
- **External-Stage**

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Stage:

For the console deployments the default mode is STAGE when you deploy the application using stage mode, the admin server copies the deployment files from the original deployment location of the application on the admin server to each target server's stage directory.
For example, if you have a distributed environment, 3 managed servers in a cluster on 3 different machines, and when a web application is deployed on the cluster, the admin server copies the deployment files to the managed server/stage directory.
This mode is most commonly used. This is advantageous because when there is a network outage and the managed servers are not able to communicate with the admin server, the managed servers still have the deployment files so the application will be available.
Stage mode should only be used when the application to be deployed is not too large. It gives an overhead in the production environment to copy huge application on each machine.

Syntax Command line deployment:

java weblogic.Deployer -adminurl 192.168.1.104:8001 -user weblogic -password weblogic -name WorkManager1 -stage -targets Cluster1 -deploy WorkManager1

Admin Console deployment:

When you deploy an application to the cluster, stage mode is used by default. When you deploy the application to the cluster, a stage directory will be created in the domain/server/managed server directory. Inside the stage folder, the directory with the application name will be created. The whole application resides in this. This local copy will be used by the managed servers to run the application. Every time you redeploy the application, this local copy on every target server is updated with new changes. Using staging mode only makes sense when there is a distributed environment. When all the servers are in the same machine, the managed servers can use a shared copy of the application.
This recipe will cover the deployment of a J2EE Application archived file (EAR), but it also applies to Web Application (WAR), Resource Adapters(RAR), and other JAR archived files, such as libraries, EJBs, and Java classes.

No-Stage

When no-stage deployment is used, the admin server does not copy the application to the directory of every server. The servers have to access the application from a shared location. So if no stage option is used for deployment, the stage directory is ignored.

Syntax to deploy application in no-stage mode:

java weblogic.Deployer -adminurl 192.168.1.104:8001 -user weblogic -password weblogic -name mydeploymentname -targets MyCluster -nostage -deploy c:\localfiles\myapp.ear

If the application is too huge, it is better to use no-stage so that the overhead while deployment decreases.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Even if no-stage mode is used, it does not mean that the server does not have the application local copy with it. The application temporary copy is created and saved in the tmp folder of the server for easy and quick access of the application.

External stage:

External stage deployment is similar to stage mode deployment. But in this, admin server is not responsible to copy the deployment files to the target servers. This should be done manually by the user.
Steps:

1. Create a stage folder in each target server directory.
2. Inside the stage folder, create another folder by the name of the application that will be deployed. For e.g.: MyWebApp
3. Then copy the deployment files to the folder. Do this for each target server.
4. From the admin console, go to target server, -> Configuration -> Deployments. Change the Staging mode to external stage in the drop down list.

Enter the below command in the command prompt:

java weblogic.Deployer -adminurl 192.168.1.104:8001 –username weblogic -password weblogic -external_stage MyCluster -name web-app -deploy \home\Apps\web-app

External stage mode is the least common mode of application deployments. This mode is used when the application size is very huge and the time of application deployment needs to be saved.
When you need application local copies on every server and the application size is very huge, you cannot use no-stage mode and stage mode which takes lot of time. You can use external stage.

Deploying Applications:

The following steps will walk you through the process to deploy the application. The WebLogic administrator usually assumes the deployer role in a production environment, so make sure to define a well-structured procedure to deploy the applications and follow it.

How to do it :

Carry out the following steps:

1. Create a new directory to be the application installation directory using the syntax / oracle/applications/<environment>/<application>/<version>:
   [wls@prod01]$ mkdir -p /oracle/applications/prod/myApp/v1
   [wls@prod01]$ cd /oracle/applications/prod/myApp/v1

2. Create two directories:
   [wls@prod01]$ mkdir app
   [wls@prod01]$ mkdir plan

3. Copy the myApp.ear file to the app directory.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

4. Access the Administration Console with your web browser at http://prod01.domain.local:7001/console.

5. Click on the Lock & Edit button to start a new edit session.

6. Navigate to the Deployments page by clicking on the link in the domain structure.

7. Click on the Install button to install a new application.

8. Type the path /oracle/applications/prod/myApp/v1/app and click on Next.

9. Select myApp.ear from the list and click on Next.

10. Select Install this deployment as an application and click on Next.

11. Select the All servers from the cluster radio button from the PROD_Cluster cluster and click on Next.

12. Leave the default options and click on the Finish button.

13. A new deployment plan file will automatically be created in /oracle/applications/prod/myApp/v1/plan/Plan.xml.

14. Click on the Activate Changes button to apply the changes.
15. The application should be in a Prepared state. Start the application by selecting the myApp checkbox and clicking on the Start button with the Servicing all requests option.

How it works:

This procedure installs a simple enterprise application named myApp to the cluster PROD_Cluster in the WebLogic domain.

The application is distributed to the cluster using the default deployment option stage mode. In the stage mode deployment, the Administration Server prepares the myApp.ear file to be copied to the stages directory of each of the Managed Servers of the cluster PROD_Cluster. The directory is **$DOMAIN_HOME/servers/<servername>/stage/<application>.**
WebLogic will use this local copy until a new redeployment is made.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Deploying using the weblogic.Deployer tool:

You can use the command-line tool weblogic.Deployer to make deployment changesin a WebLogic domain.

1. Go to the WebLogic domain's bin directory:

   [wls@prod01]$ cd $DOMAIN_HOME/bin

2. Set the environment variables:

   [wls@prod01**]$ . ./setDomainEnv.sh**

3. Run the weblogic.Deployer command line with the parameters:

[wls@prod01**]$**
java weblogic.Deployer -adminurl t3://prod01.domain.local:7001 -username wlsadmin -password <pwd> -deploy –targets PROD_Cluster/oracle/applications/prod/myApp/v1/app/myApp.ear

4. The following should be the output:

weblogic.Deployer invoked with options: -adminurl t3://prod01. domain.local:7001 -username wlsadmin -deploy -targets PROD_Cluster/oracle/applications/prod/myApp/v1/app/myApp.ear <Info><J2EE Deployment SPI><BEA-260121><Initiating deploy operation for application, myApp [archive: /oracle/applications/od/myApp/v1/app/myApp.ear], to PROD_Cluster .>

5. The myApp application should be deployed to the PROD_Cluster cluster.

Deploying applications using WLST:

Now let's deploy the application through WLST using the following steps:

1) Log in as a wls user to shell and start WLST:

   [wls@prod01]$ $WL_HOME/common/bin/wlst.sh

2) Connect to the Administration Server using wlsadmin as the user, <pwd> as the password, and t3://prod01.domain.local:7001 as the server URL:

   wls:/offline>**connect("wlsadmin","<pwd>","t3://prod01.domain.local:7001")**

3) Run the following WLST command to deploy the myApp.ear application to the PROD_Cluster cluster:deploy("myApp", /oracle/applications/prod/myApp/v1/app/myApp.ear","PROD_Cluster")

4) The following should be the output:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Deploying application from /oracle/applications/prod/myApp/v1/app/myApp.ear to targets PROD_Cluster (upload=false) ...<Nov, 2015 11:02:24 PM BRT><Info><J2EE Deployment SPI><BEA-260121><Initiating deploy operation for application, myApp[archive: /oracle/applications/prod/myApp/v1/app/myApp.ear], to PROD_Cluster .>.Completed the deployment of Application with status completed Current Status of your Deployment:Deployment command type: deploy Deployment State : completed

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

JDBC: JAVA DATA BASE CONNECTIVITY

Configuring JDBC Resources for High-Availability:

- In this chapter, we will cover the following recipes:
- Creating a JDBC data source
- Creating a multi data source
- Defining the multi data source HA Strategy
- Creating a Grid Link data source
- Managing JDBC data sources
- Tuning data sources for reliable connections
- Tuning multi data sources – surviving RAC node failures
- Updating the Oracle JDBC driver

Introduction:

JDBC API stands for Java database connectivity and allows Java applications to make calls to a database in the form of SQL statements. The connection to the database is encapsulated by the vendor's JDBC driver. WebLogic Server provides JDBC drivers for the most commonly used databases, such as DB2, Informix, Microsoft SQL Server, MySQL, Oracle, PostgreSQL,Sybase, and others.A third-party JDBC driver can also be added, such as the Teradata JDBC drivers.

The JDBC data source contains the parameters, such as the database host address, database port, instance, and service name, needed to connect to the database. The data source also includes transaction options and a pool for reusing the database connections, optimizing the time spent opening these connections.

Creating a JDBC data source:

Consider that the DBApp application is deployed on the PROD_Cluster cluster and requires a data source to connect to the database. The application looks for a non-XA data source with the Java Naming and Directory Interface (JNDI) as jdbc/ds-nonXA.The database is an Oracle database that is running in the dbhost hostname and listening to the port 1521. The listener is accepting requests to the service name dbservice.In this recipe, a new JDBC data source to connect to the Oracle database will be created and configured for the application DBApp.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Configure Generic data source:

Carry out the following steps to create a JDBC data source:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left, and then click on Data Sources.
4. Click on the New button and then click on Generic Data Source.
5. Type ds-nonXA in the Name field and jdbc/ds-nonXA in the JNDI Name field.
6. Select the Oracle option from the Database Type drop-down menu. Click on the Next button.
7. Choose *Oracle's Driver (Thin) for Service connections; Versions:9.0.1 and later from the Database Driver drop-down menu.
8. Click on the Next button.
9. Leave the default values for the Transaction options and click on the Next button.
10. On the Connection Properties page, type dbservice in the Database Name field, dbhost in the Host Name field, and 1521 in the Port field. Complete the Database
11. User Name, Password, and Confirm Password fields by typing dbuser and dbpwd as the username and password respectively. Click on the Next button.
12. Click on the Next button on the Test Database Connection page.
13. Select the All servers in the cluster radio button from the PROD_Cluster cluster.
14. Click on the Finish button.
15. Then, click on the Activate Changes button.

A new non-XA JDBC data source was created with the parameters required by the DBApp application. The non-XA Oracle JDBC driver is the thin version. All other parameters were left as their default values.

	Name	Type	JNDI Name	Targets
	ds-nonXA	Generic	jdbc/ds-nonXA	PROD_Cluster

Multi data source:

The multi data source is a data source abstraction that groups all the individual data sources that connect to each node of the Oracle RAC database.

Consider that the DBApp application requires an XA connection with a JNDI name jdbc/ds-XA added to a database.

The database is an Oracle RAC database with two nodes. The first node has an instance name instance-rac01 and runs in the dbhost-rac01 hostname and listens to the port 1521.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

The listener accepts requests to the service name dbservice-rac01. The second node is the instance instance-rac02, and it runs in the dbhost-rac02 hostname, listens to the port 1521, and has a service name dbservice-rac02. In this recipe, a new JDBC multi data source will be created and configured for the DBApp application.

Before creating the multi data source, the individual data sources pointing to each RAC node must be created. Two data sources will be created with the names ds-XA-rac01 and ds-XA-rac02 and with the JNDI names jdbc/ds-XA-rac01 and jdbc/ds-XA-rac02. The multi data source will be named ds-XA and includes both data sources.

Configure Multi Data source:

Carry out the following steps to create a multi data source:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left, and then click on Data Sources.
4. Click on the New button and then click on Generic Data Source.
5. Type ds-XA-rac01 in the Name field and jdbc/ds-XA-rac01 in the JNDI Name field. Leave the Database Type drop-down menu with the Oracle option selected. Click on the Next button.
6. Choose *Oracle's Driver (Thin XA) for RAC Service-Instance connections; Versions:10 and later from the Database Driver drop-down menu. Click on the Next button.
7. Then, click on the Next button in the Transaction Options page.
8. On the Connection Properties page, type dbservice-rac01 in the Service Name field, instance-rac01 in the Database Name field, dbhost-rac01 in the Host Name field, and 1521 in the Port field. Complete the Database User Name,Password, and Confirm Password fields by typing dbuser, and dbpwd as the username and password respectively. Click on the Next button.
9. Click on the Next button in the Test Database Connection page.
10. Select the All servers in the cluster radio button from the PROD_Cluster cluster.Click on the Finish button.
11. Repeat the previous steps and create another data source. Add ds-XA-rac02 as Name, jdbc/ds-XA-rac02 as JNDI Name, dbservice-rac02 as Service Name,instance-rac02 as Database Name, and dbhost-rac02 in the Host Name field.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

	Name	Type	JNDI Name	Targets
☐	ds-XA-rac01	Generic	jdbc/ds-XA-rac01	PROD_Cluster
☐	ds-XA-rac02	Generic	jdbc/ds-XA-rac02	PROD_Cluster

12. Create the multi data source by clicking on the New button then on the Multi Data Source link.

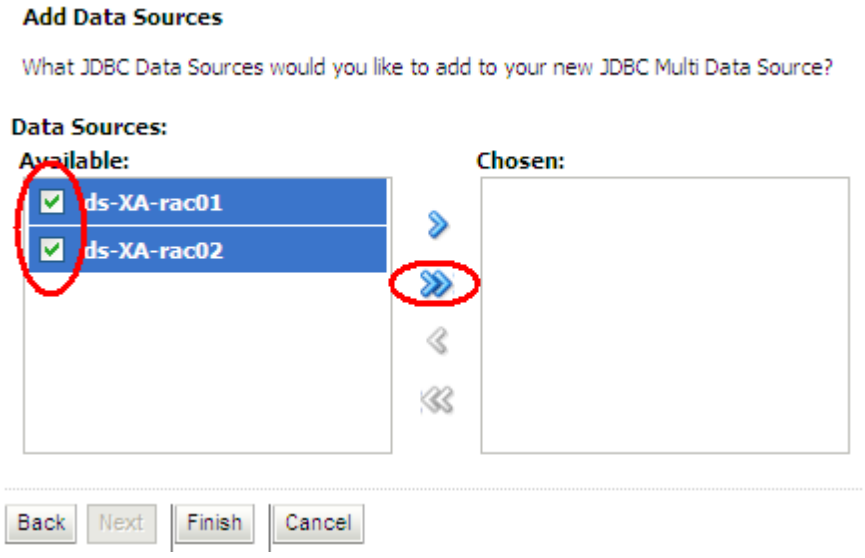
13. Type ds-XA in the Name field and jdbc/ds-XA in the JNDI Name field. Leave the other options as their default values. Click on the Next button.

14. Select the All servers in the cluster radio button from the PROD_Cluster cluster.

15. Click on the Next button.

16. Select the XA Driver option. Click on Next.

17. Select both data sources ds-XA-rac01 and ds-XA-rac02 from the left of the Add Data Source page. Click on the >> button in the center of both sides to move them to the right. Click on Finish.



18. Finally, click on the Activate Changes button. The multi data source ds-XA was created with the data sources ds-XA-rac01 and ds-XA-rac02 as members. The multi data source manages the application requests for a database connection and uses Algorithm Type to define the strategy for high availability.If using the Failover algorithm, be sure to enable the Failover Request if Busy checkbox of the multi data source.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

GridLink data source:

The GridLink data source is a new type of data source that has been available in WebLogic Server since **Version 10.3.4.** The GridLink is used to connect to Oracle RAC databases and is a recommended alternative to the multi data sources since it provides some useful features,**such as fast connection failover, runtime connection load balancing, graceful handling of Oracle RAC outages, GridLink affinity, and SCAN addresses.**

The same DBApp application requirement from the earlier recipe will be used, but in this recipe a GridLink data source will be used instead of a multi data source.Consider that the DBApp application requires an XA connection with a JNDI name jdbc/ds-GridLinkXA to a database. The database is an Oracle RAC database with two nodes.**The first node runs in the dbhost-rac01 hostname and listens to the port 1521. The secondnode runs in the dbhost-rac02 hostname and also listens to the port 1521. The database has a service name dbservice. The ONS service is running on the onshost hostname, port 6200.**

Note :

Create the GridLink data source with the JNDI name jdbc/ds-GridLinkXA in the Administration Console. Make sure the Administration Server is running.

GridLink Configuration:

1. Access the Administration Console with your web browser at http://adminhost. domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left, and then click on Data Sources.
4. Click on the New button and then click on GridLink Data Source.
5. Type ds-GridLinkXA in the Name field and jdbc/ds-GridLinkXA in the JNDI Name field. Choose *Oracle's Driver (Thin XA) for GridLink Connections Versions:11 and later from the Database Driver drop-down menu. Click on the Next button.
6. Click on the Next button on the Transaction Options page.
7. Leave the Enter individual listener information option selected on the GridLink data source connection Properties Options page and click on the Next button.
8. On the Connection Properties page, type dbservice in the Service Name field. Type dbhost-rac01:1521 in the Host and Port field and click on the Add button. Add the second host by typing dbhost-rac02:1521 in the Host and Port field and clicking on the Add button. Complete the database User Name, Password, and Confirm Password fields by typing dbuser and dbpwd as the username and password respectively. Leave the Protocol field with the TCP value. Click on the Next button.
9. Click on the Next button on the Test GridLink Database Connection page.
10. On the ONS Client Configuration page, type onshost:6200 in the ONS hostand port field and click on the Add button. Then click on Next.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

11. Click on the Next button on the Test ONS client configuration page.
12. Select the All servers in the cluster radio button from the PROD_Cluster cluster. Click on the Finish button.
13. Finally, click on the Activate Changes button.

How it works:

The Grid Link data source has some advantages over the multi data source.The multi data source uses the test connection feature to guarantee a reliable connection to the database. The GridLink data source on the other hand uses the call back **mechanism of Oracle RAC Fast Application Notification (FAN).** This means that the GridLink data source actively responds to events and notifications coming from the database, such as the fluctuation of RAC services.

Another improvement is the load balancing mechanism. The multi data source configured with the Load Balance algorithm distributes the application requests to borrow a connection from the data source members of the multi data source in a round-robin fashion. GridLink improves the distribution load by receiving load balancing events from the database. These events indicate the recommended connection distribution among the RAC nodes.

Managing JDBC data sources:

The WebLogic JDBC subsystem can be controlled when needed. A WebLogic Administrator can start, stop, shrink, pause, suspend, and reset the data source on demand. The statement cache can also be cleared if needed.

In this recipe, the ds-nonXA data source will be used as an example.

Configuration:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the plus sign to open the Services tree on the left, and then click on Data Sources.
3. Click on the ds-nonXA link to open the data source and then click on the Control tab.
4. Select the checkbox from ds-nonXA and click on the button of the desired operation:

Shrink, **Reset, Clear Statement Cache, Suspend, Resume, Shutdown, or Start**.

How the Mechanism works:

The data source control operations are useful in situations such as when an Oracle RAC node is down or during a database maintenance window.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

The **Shrink** operation closes the idle connections of the pool, freeing the database connections and resources in use. It reduces the opened connections until the greater value is between the **minimum capacity** parameter and the in-use connection.

The **Reset** operation resets the pool, closing and reopening all the database connections.

The **Clear** Statement Cache operation clears the callable and prepared statement caches.

Note :

The **Clear** Statement Cache operation is useful when there are changes in DBMS objects such as stored procedures. Some exceptions and errors can be caused by deprecated cached statements.

The **Suspend** operation disables the data source. Although it leaves the connection state unchanged in the pool, the applications cannot borrow connections. The **Suspend** operation can also be forced. In this case, all connections are closed.

The **Resume** operation resumes a suspended data source.

The **Shutdown** operation shuts down the data source. If there are connections in use by the application, the application operation, in the course of time, will fail and return an error. The **Shutdown** operation can be forced. In this case, if the connections are in use, WebLogic will forcibly close all of them and will shut down the data source.

The **Start** operation starts a stopped data source.

Tuning data sources for reliable connections:

In the previous recipes, some data sources required by the DBApp application were created.Some default parameter values are not the best option to use out of the box, so in this recipethe parameters of the GridLink ds-GridLinkXA data source will be tuned to avoid unreliable connections being delivered to the applications.

The changes are generic to the GridLink and Generic data source types and can be applied to all previously created data sources.

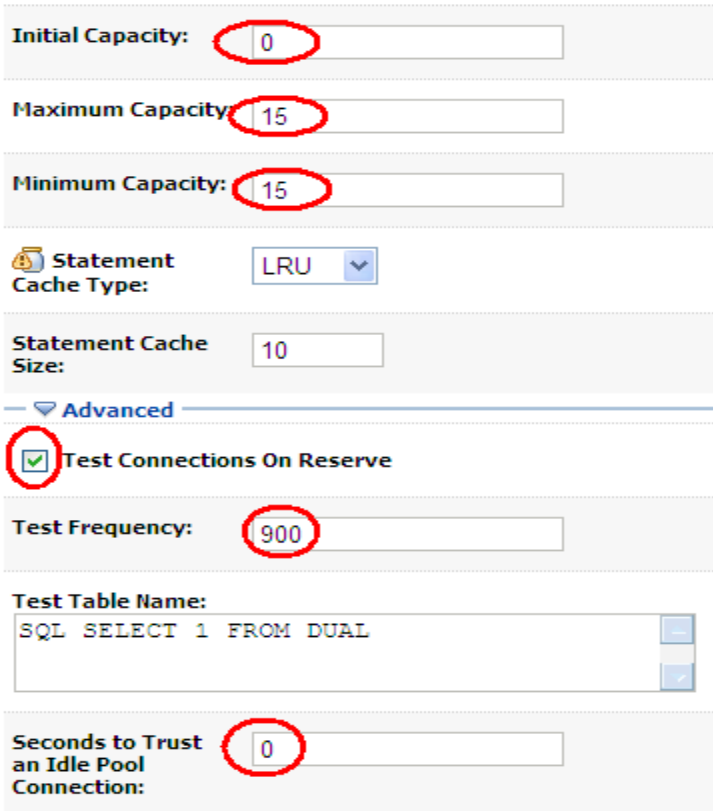
Note :

Access the Administration Console to tune the ds-GridLinkXA data source parameter.Make sure the ds-GridLinkXA GridLink was created in the previous recipe and that the Administration Server is running.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Configuration:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the **Lock & Edit** button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left, and then click on **Data Sources.**
4. Click on the **ds-GridLinkXA** link to open the data source, and then click on the **Connection Pool** tab under the main Configuration tab.
5. **Type 0 in the Initial Capacity field and 15 in the Maximum Capacity and Minimum Capacity fields.** Click on the **Save** button and then click on the Advanced link to open the advanced options.
6. Check the Test Connection on Reserve checkbox. **Type 900** in the Test Frequency field and type **0 in the Seconds to Trust an Idle Connection field**. Leave the other options as their default values (as shown in the following screenshot) and click on the Save button.



7. Finally, click on the Activate Changes button.

How the Mechanism works:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

During the WebLogic Server startup process, the data sources are deployed and the connections to the databases are opened according to the **Initial Capacity** parameter.

If a data source connection cannot be established with the database during startup for some reason, the Managed Server starts in the ADMIN state instead of in the RUNNING state.The commonly used procedure in this situation is to click on the Resume button; the server instance resumes to the RUNNING state and starts accepting and processing the application
requests. However, the data source remains undeployed and uninitialized, and errors will occur with the applications that use this data source. Even if the database goes back online,the data source will not start automatically.

Note :

To avoid the ADMIN state on startup, set the data source Initial Capacity to **0 so it won't open any connection to the database during the server startup process.** The Managed Server instance will start in the RUNNING state, and as soon as the database goes back online, the data source will reconnect to it without intervention.

The drawback is that since WebLogic will start in the RUNNING state, even with the database out, the application requests that use the data source will return errors. Therefore, it is a good idea to use the option that best suits the application's requirements.

It's also a common recommendation to set **Initial Capacity and Maximum Capacity** to the same value so all connections would already be open when needed. **The value of 15 for Maximum Capacity was used as an example.**

Note :

In WebLogic Server 12c, **a better recommendation for production environments would be to set Initial Capacity to 0 and Minimum Capacity and Maximum Capacity to the same value.** The Minimum **Capacity parameter was added in WebLogic Server 10.3.6.**

With the **Test Connection on Reserve option enabled**, WebLogic tests the connection with the database using the query specified in the Test Table Name field, just before lending the connection to the application. The value of 0 seconds in the Seconds to Trust an Idle Connection field forces the test to be made in every request. The Test Frequency field was also increased to **900 seconds instead of the default value of 120 seconds.**
The values used in this recipe are based on an online application that receives requests during a 24 x 7 period. A batch application that runs once a day for a few hours should use some different settings, such as a Minimum Capacity of 0, so the connections to the database are closed when the application is idle. Tune the parameters according to your
application requirements.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

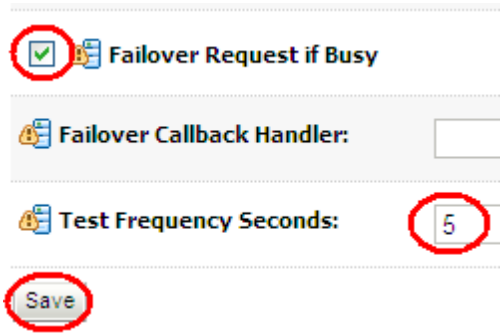
Tuning multi data sources – surviving RAC node failures:

The multi data source default configuration doesn't provide the best settings for surviving an Oracle RAC node failure properly.
This recipe tunes the ds-XA-rac01 and ds-XA-rac02 data sources and the ds-XA multidata source.

Access the Administration Console to tune the data sources parameters. Make sure the ds-XA-rac01 and ds-XA-rac02 data sources and the ds-XA multi data source were created in the previous recipe and that the Administration Server is running.

Configuration:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left, and then click on Data Sources.
4. Click on the ds-XA link to open the multi data source configuration page.
5. Check the Failover Request if Busy checkbox and type 5 in the Test Frequency Seconds field. Click on the Save button, as shown in the following screenshot:



6. Click on the Data Sources link on the left navigation tree again. Click on the ds-XA-rac01 data source link and then on the Connection Pool tab.
7. In the URL field, add the following parameters to the text value: (ENABLE=BROKEN)(LOAD_BALANCE=OFF)(FAILOVER=OFF). The final URL should be jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=BROKEN)(LOAD_BALANCE=OFF)(FAILOVER=OFF)(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=dbhost-rac01)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=dbservice-rac01)(INSTANCE_NAME=instance-rac01))).
8. Add the line oracle.net.CONNECT_TIMEOUT=10000 to the Properties field.
9. If not already tuned from the previous recipe, type 0 in the Initial Capacity field and 15 in the Maximum Capacity and Minimum Capacity fields. Click on the Advanced link to open the advanced options.
10. Check the Test Connection on Reserve checkbox. Type 900 in the Test Frequency field and type 0 in the Seconds to Trust an Idle Connection field.
11. Leave the other options as their default values and click on the Save button.
12. Repeat steps 6 to 10 with the ds-XA-rac02 data source.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

13. Finally, click on the Activate Changes button.

How it works:

The **Failover Request if Busy** option is not enabled by default and is recommended to be enabled when using multi data sources with the Failover algorithm. It works by providing a connection from the next data source member on the list when all the connections from the primary data source member are in use (overloaded).
WebLogic Server closes all the connections of a data source and disables it if the connection to the database fails to be created two consecutive times. By default, the multi data source rechecks the database every 120 seconds based on the multi data source **Test Frequency Seconds** parameter. Changing the **Test Frequency Seconds parameter** to 5 seconds is recommended.

Updating the Oracle JDBC driver:

Download the updated JDBC drivers from the Oracle website.The Oracle JDBC driver filename is ojdbc6.jar. Create a temporary folder, such as ~/jdbctemp, and download the file to it. The procedure is done manually, so make sure that none of the WebLogic Server instances are running, including the Administration Server.

Configuration:

1. Log in as a wls user to the prod01 shell and run the following commands:

   [wls@prod01]$ cd $WL_HOME/server/lib

   [wls@prod01]$ mv ojdbc6.jar ojdbc6.original

   [wls@prod01]$ cp ~/jdbctemp/ojdbc6.jar .

2. Repeat the step in prod02 and all machines of the cluster.
3. Start the Administration Server and all the Managed Servers.

How it works:

Updating the Oracle JDBC drivers is a straightforward process.Just make a backup copy of the JDBC driver file ojdbc6.jar, replace it with the new file version, and start the WebLogic Server instances.

Verify the driver version:

1. Log in as a wls user to shell and navigate to the folder of the downloaded file:
   [wls@prod01]$ cd ~/jdbctemp
2. Run the following Java command to display the driver version:
   [wls@prod01]$ /oracle/jvm/bin/java -jar ojdbc6.jar
3. The JDBC driver Version will be displayed on the screen:
   Oracle 11.2.0.3.0 JDBC 4.0 compiled with JDK6 on Fri_Nov_04_08:05:20_PDT_2011
   #Default Connection Properties Resource
   #Sun Apr 07 04:08:20 BRT 2013

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

JMS:

In this chapter we will cover the following recipes:

- Creating the file stores
- Creating the JDBC stores
- Creating the JMS servers
- Creating the JMS module
- Configuring the subdeployment target
- Creating the distributed queue destination and the connection factory
- Starting/stopping consumers for a JMS destination
- Using the Server affinity to tune the distributed destinations' load balance
- Creating a pinned queue with clustering and HA with service migration
- Configuring the messaging bridge with source and target distributed destinations
- Relying on SAF to transfer JMS messages to another WebLogic domain

The Java Message Service (JMS) is a standard Java API that enables an enterprise Application to communicate asynchronously with other applications by sending and receiving Messages.

Managed Server	Persistence Store
PROD_Server01	FileStore01
PROD_Server02	FileStore02
PROD_Server03	FileStore03
PROD_Server04	FileStore04

The file stores will be saved in the $DOMAIN_HOME/filestores directory, so make sure the Directory is created before creating the file stores.

How to do it:

Create the **$DOMAIN_HOME/filestores** directory in all machines, as follows:

1. Log in as a wls user to the prod01 shell and create the directory.
   [wls@prod01]$ cd $DOMAIN_HOME
   [wls@prod01]$ **mkdir filestores**
2. Repeat this step for prod02.

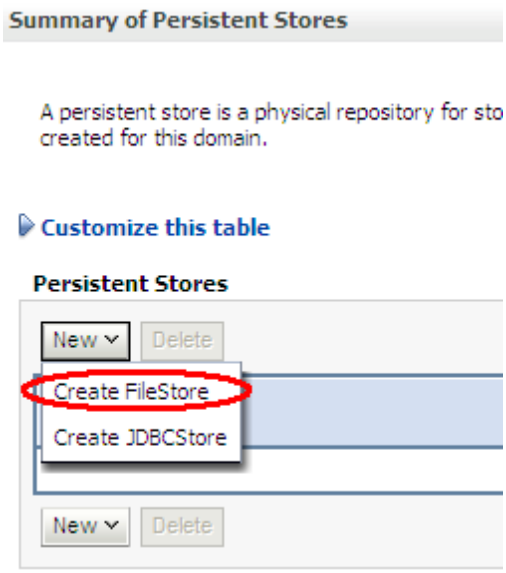
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Create the file stores using the Administration Console, as follows:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.

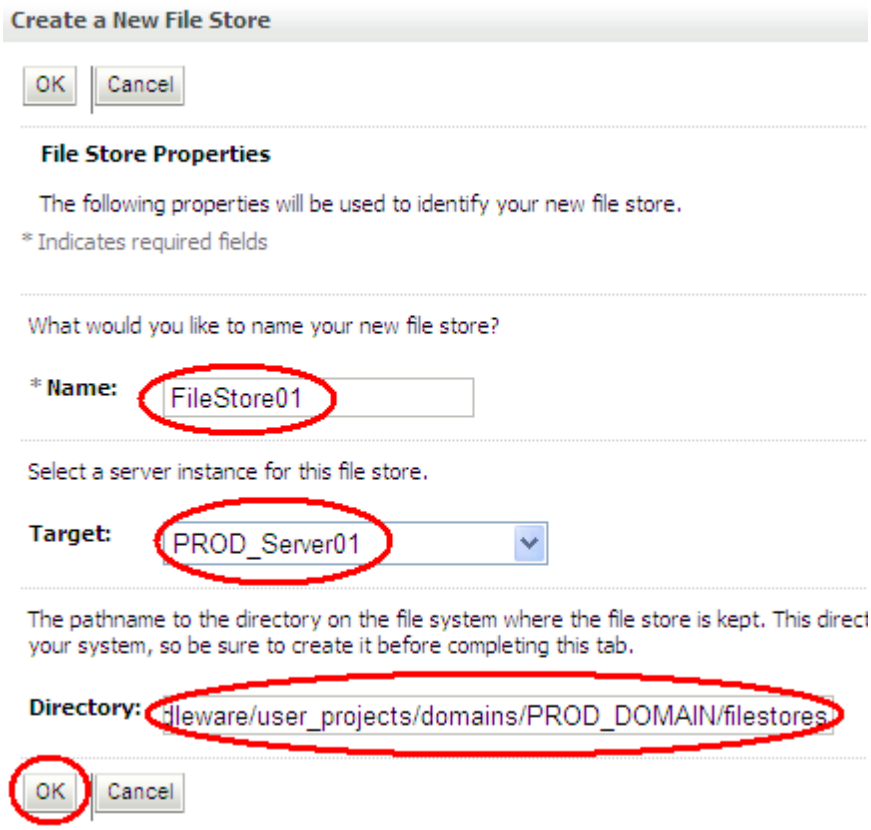
2. Click on the Lock & Edit button to start a new edit session.

3. Expand the Services tree on the left and click on Persistent Stores.



5. Type FileStore01 in the Name field. Click on the Target drop-down menu and select the PROD_Server01 option.
6. Type /oracle/Middleware/user_projects/domains/PROD_DOMAIN/ filestores in the Directory field to point to the newly created directory and Click on the OK button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



Repeat the previous steps and create the remaining file stores, FileStore02,FileStore03, and FileStore04 targeting the corresponding Managed Servers,PROD_Server02, PROD_Server03, and PROD_Server04.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

8. Click on the Activate Changes button to finish.

How it Works :

The file stores were created in all Managed Servers of the cluster and will be used as persistent stores for the JMS servers.

Note :

Although it is possible to use the default file store for the JMS servers, creating a separate file store is recommended to decouple and isolate the configuration for the JMSApp application. Using the default store also eliminates the possibility of using a migratable target.

Creating the JDBC Source:

The persistent store can also persist the data in the database by using the JDBC store. In this recipe a new JDBC store will be created in all Managed Servers of the PROD_Cluster cluster. The database that will host the stores is an Oracle RAC database with two nodes. The first node has an instance name instance-rac01, runs in the dbhost-rac01 hostname, and listens to the port 1521. The listener accepts requests to the service name dbservicerac01.The second node is the instance instance-rac02, runs in the dbhost-rac02 hostname, listens to the port 1521, and has a service name dbservice-rac02.

For the cluster PROD_Cluster, we will consider the JDBC stores JDBCStore01, JDBCStore02, JDBCStore03, and JDBCStore04 for the instances PROD_Server01,PROD_Server02, PROD_Server03, and PROD_Server04 respectively.

Managed Server	Persistence Store
PROD_Server01	JDBCStore01
PROD_Server02	JDBCStore02
PROD_Server03	JDBCStore03
PROD_Server04	JDBCStore04

How it works:

Create the data sources and the multi data source:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the Services tree on the left and then click on Data Sources.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

4. Click on the New button and click on Generic Data Source.
5. Type ds-store-rac01 in the Name field and jdbc/ds-store-rac01 in the JNDI Name. Leave the Database Type drop-down menu with the Oracle option selected.Click on the Next button.
6. Choose *Oracle's Driver (Thin) for RAC Service-Instance connections; Versions:10 and later from the Database driver drop-down menu. Click on the Next button.
7. Disable the Supports Global Transactions checkbox and click on the Next button.
8. In the Connection Properties page, type dbservice-rac01 in the Service Name field, instance-rac01 in the Database Name field, dbhost-rac01 in the Host
9. Name field, and 1521 in the Port field. Fill in the Database User Name, Password,and Confirm Password fields with dbuser and dbpwd. Leave the Protocol field with
10. the default TCP value. Click on the Next button.
11. Click on the Next button in the Test Database Connection page.
12. Click on the All servers in the cluster radio button from the PROD_Cluster cluster.Click on the Finish button.
13. Repeat the previous steps and create another data source. Use ds-store-rac02 as Name, jdbc/ds-store-rac02 as JNDI Name, dbservice-rac02 as Service Name, instance-rac02 as Database Name, and dbhost-rac02 as Host Name.
14. Create the multi data source by clicking on the New button and then the Multi Data Source link.
15. Type ds-store in the Name field and jdbc/ds-store in the JNDI Name field. Leave the Algorithm Type option in the default Failover option. Click on
16. the Next Button.
17. Click on the All servers in the cluster radio button from the PROD_Cluster cluster.Click on the Next button.
18. Click on the Non-XA Driver option in the Select Data Source Type page. Click on Next.
19. Select both data sources ds-store-rac01 and ds-store-rac02 from the left of the Add Data Source page. Click on the >button in the center of both sides to move them to the right. Click on Finish.
20. Click on the Activate Changes button.

Create the JDBC stores using the Administration Console:

1. Access the Administration Console again with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left and click on Persistent Stores.
4. Click on the New button and click the Create JDBCStore link to start creating a new persistent store.
5. Type JDBCStore01 in the Name field. Click on the Target drop-down menu and select the PROD_Server01 option. Select the ds-store value in the Data Source field. Type JDBCStore01 again in the Prefix Name field and click on the OK button.
6. Repeat the previous steps and create the remaining JDBC stores, JDBCStore02,JDBCStore03, and JDBCStore04 targeting the corresponding Managed Servers,PROD_Server02, PROD_Server03, and PROD_Server04. Use the same dsstore data source for all JDBC stores.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

7. Click on the Activate Changes button to finish.

How it works:

The JDBC stores were created in all Managed Servers of the cluster and can be used as persistent stores for the JMS servers.
The JDBC store uses a multi data source pointing to an Oracle RAC database with two RAC nodes.

Note :

It's mandatory to use non-XA data sources and a multi data source with the Failover algorithm with the JDBC store. Make sure to tune all JDBC parameters according to how they were tuned in the previous chapter .

Creating the JMS Servers:

The JMS server is a WebLogic resource that provides a container for the JMS queues and JMS topics' destinations. A JMS server can manage several destinations at a time and it uses the specified persistent store to persist the messages. The persistent store of the JMS server can be the default persistent store of the WebLogic Server instance, a custom file store or a JDBC store.

Following the roadmap configuration for the JMSApp application, a JMS server will be created for each of the Managed Servers of the cluster PROD_Cluster, and each one will be configured to use the file stores created before.

Managed Server	JMS Server	Persistent Store
PROD_Server01	JMSServer02	FileStore01
PROD_Server01	JMSServer02	FileStore02
PROD_Server01	JMSServer03	FileStore03
PROD_Server01	JMSServer04	FileStore04

How to do it:

To create the JMS servers, carry out the following steps:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left; click on Messaging and then on JMS Servers.
4. Click on the New button to open the Create a New JMS Server page.
5. Type JMSServer01 in the Name field and choose FileStore01 from the Persistent Store drop-down menu. Click on the Next button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

6. Choose PROD_Server01 from the Target drop-down menu. Click on the Finish button.
7. Repeat the previous steps and create JMSServer02, JMSServer03, JMSServer04 using the file stores FileStore01, FileStore02, FileStore03, and FileStore04.

i.

Name	Persistent Store	Target
JMSServer01	FileStore01	PROD_Server01
JMSServer02	FileStore02	PROD_Server02
JMSServer03	FileStore03	PROD_Server03
JMSServer04	FileStore04	PROD_Server04

8. Click on the Activate Changes button to finish.

How it Works:

The JMS servers were created pointing to their specific file stores. The JMS servers are still working as independent units and the configuration for clustering will be achieved with the creation of the JMS module and the JMS destinations and resources.
The JDBC stores created in the previous recipe can also be used as persistent stores to the JMS servers.

Creating the JMS Module:

The JMS module is a WebLogic global system resource that aggregates and stores JMS resources and JMS-related configurations such as queues, topics, connection factories, quotas, distributed queues, and distributed topics.

There are two types of JMS modules: the JMS application module and the JMS system module. In this recipe, the JMS system module will be covered since it is the module WebLogic administrators use for creation and configuration. Although it has the same functions as those of a system module, the JMS application module should be handled by the developer and has to be included and packaged inside the application's EAR file.

The system module is also preferred over the application module because the application module can be handled only by manually editing the XML. The system module, on the other hand, can be managed through the Administration Console, WLST, or JMX.

Continuing the setup process for the JMS resources needed by the **JMSApp application**, create a new JMS module called **JMSAppModule** in PROD_DOMAIN.

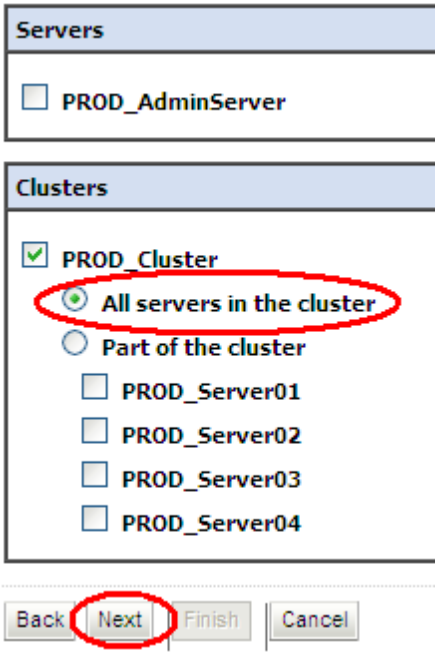
The JMS module will be created using the Administration Console, so make sure the Administration Server is running.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How to do it:

Carry out the following steps to create the JMS module:

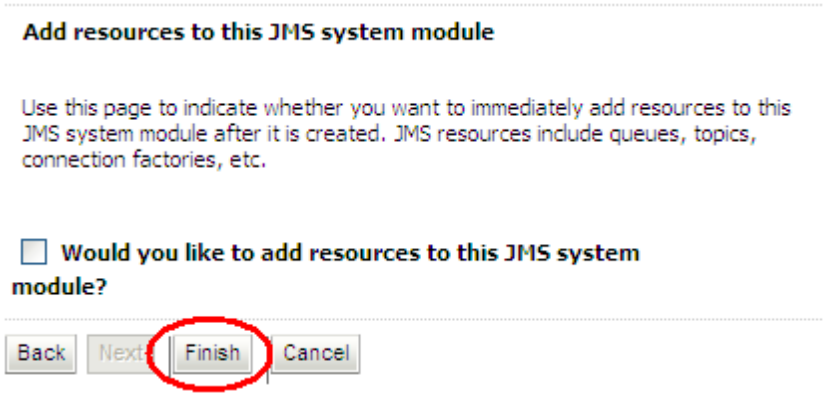
1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left; click on Messaging and then on JMS Modules.
4. Click on the New button to open the Create JMS System Module page.
5. Type JMSAppModule in the Name field. Leave the Descriptor File Name and Location in Domain text fields blank. Click on the Next button.
6. Check the All Servers in the cluster radio button to target the JMSAppModule to the cluster PROD_Cluster. Click on the Next button.



a.

7. Click on the Finish button, leaving the checkbox Would you like to add resources to this JMS system module? unchecked.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



8. Click on the Activate Changes button to finish.

How it works:

The JMS module JMSAppModule was created in the PROD_DOMAIN domain and is ready to be configured and used by the JMSApp application.
The module configuration is saved at the $DOMAIN_HOME/config/jms directory and uses the filename convention <module-name>-jms.xml. In this case, the filename created is jmsappmodule-jms.xml.

Configuring the subdeployment target:

The JMS module JMSAppModule is targeted to the cluster PROD_Cluster, meaning the JMS resources added to the module will use the PROD_Cluster cluster as the default target. Because of the JMS resource, such as a distributed queue, the default target is not the best option for production.

WebLogic Server has a feature called subdeployment targeting to handle this special targeting. The subdeployment allows a JMS resource to use a different target from the default JMS module target, using a single or multiple WebLogic Server instances, WebLogic clusters, or JMS servers as target.

Note :

Subdeployment is a not a precise term and confuses WebLogic administrators. Subdeployment is better referred to as advanced targeting.

The subdeployment JMSAppSub will be created using the Administration Console.Make sure the Administration Server is running and the JMSAppModule module is created.

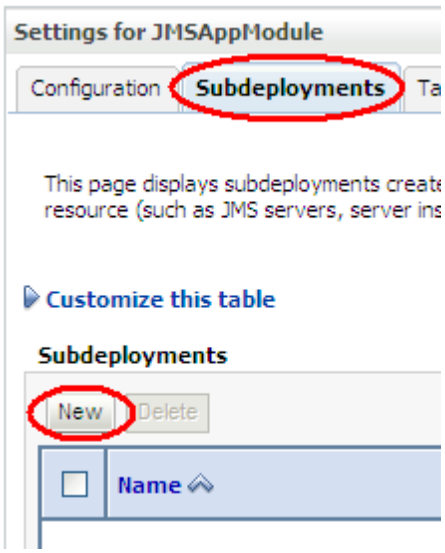
How to do it:

Carry out the following steps to configure the JMS subdeployment:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

3. Click on the plus sign to open the Services tree on the left; click on Messaging and then on JMS Modules.
4. Click on the JMSAppModule link and then the Subdeployments tab. Now click on the New button.



5. Type JMSAppSub in the Name text field under Subdeployments and click on the Next button.
6. Click and enable only the checkboxes JMSServer01, JMSServer02, JMSServer03,and JMSServer04 and click on the Finish button.

7. Click on the Activate Changes button to finish.

How it works:

The subdeployment **JMSAppSub** was created for the JMS module **JMSAppModule**, pointing to the JMS servers JMSServer01, JMSServer02, JMSServer03, and JMSServer04,meaning that any JMS resource targeting to the subdeployment will be restricted for use by the selected JMS servers only.

In the next recipe, the JMS resources required by the **JMSApp** application will be configured

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

to target the subdeployment **JMSAppModule**.

Creating the distributed queue destination and the connection factory:

Oracle WebLogic Server 12c has two types of queues:
The queue and the distributed queue.The queue is a JMS resource targeted to a single Managed Server.A distributed queue must be used when working with WebLogic clustering. The distributedqueue is a logical entity that groups single queues distributed across the JMS servers andManaged Servers of the cluster but is accessible as a single and transparent JNDI name.JMS applications require that all resource and JNDI names be unique across the entire applicationenvironment, including the WebLogic domains involved, the clusters, Managed Servers, and
JMS resources.

The members (queues) of the distributed queue can use weighted or uniform distribution.

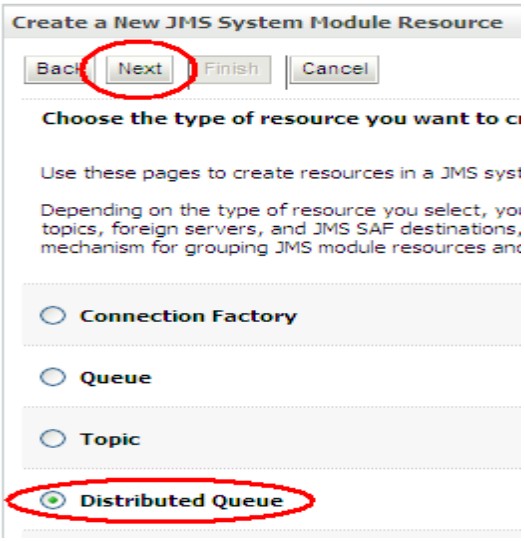
A JMS queue must be created to handle the JMSApp application's messages. In this recipe,the uniformly distributed queue called JMSAppQueue and the connection factory called JMSAppConnectionFactory will be created and added to the JMSAppModule module.

The queue will be added using the Administration Console. The persistent stores, JMS servers,JMS module, and subdeployment of the previous recipes must already be created.

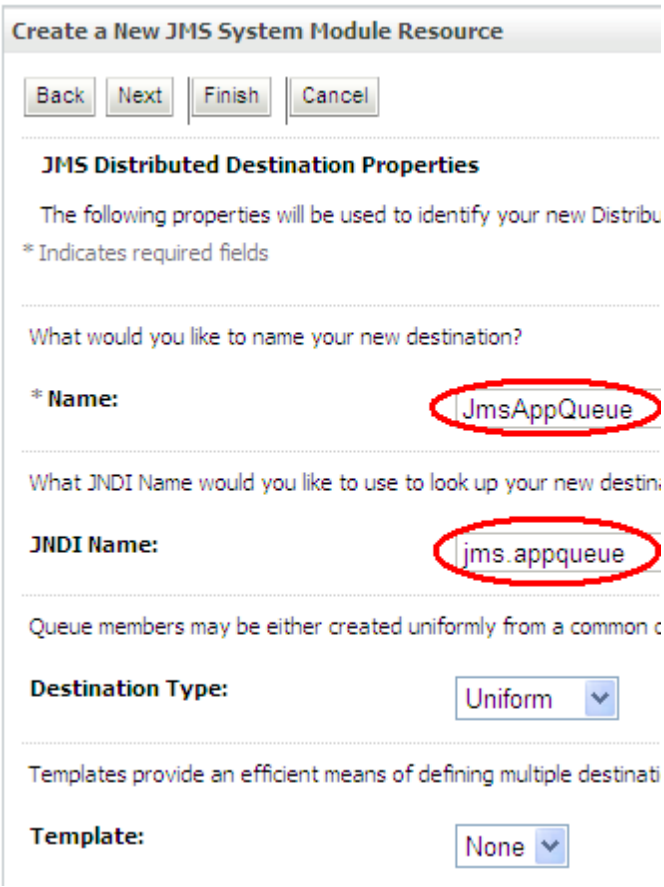
How to do it:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left; click on Messaging and then on JMS Modules.
4. Click on the JMSAppModule link to open the Configuration page. Click on the New button.
5. Select the Distributed Queue option from the Create a New JMS System Module Resource page and click on the Next button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



6. Type JMSAppQueue in the Name field and jms.appqueue in the JNDI Name field.Leave the default values Uniform in the Destination Type field and None in the Template drop-down menus. Click on the Next button.



1.

7. Click on the Advanced Targeting button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

8. From the Subdeployments drop-down menu, select the JMSAppSub option and click on the Finish button.
9. Click on the New button again and choose the Connection Factory radio button.Click on the Next button.
10. Type JMSAppConnectionFactory in the Name field and jms.appcf in the JNDI Name field. Leave all other fields at their default values and click the Next button.
11. Click on the Finish button to confirm the PROD_Cluster cluster as target.
12. Click on the Activate Changes button to finish.

How it works:

The distributed queue JMSAppQueue and the connection factory JMSAppConnectionFactory were created and added to the JMS module.

Name	Type	JNDI Name	Subdeployment	Targets
JMSAppConnectionFactory	Connection Factory	jms.appcf	Default Targetting	PROD_Cluster
JmsAppQueue	Uniform Distributed Queue	jms.appqueue	JMSAppSub	JMSServer01, JMSServer02, JMSServer03, JMSServer04

The uniformly distributed queue is targeted to the JMSAppSub subdeployment. The uniform distribution means that WebLogic create a queue member of the distributed queue in each of the JMS servers of the subdeployment automatically.

Note :

Avoid targeting a uniformly distributed queue to a cluster, otherwise WebLogic will create a queue member in every JMS server found in the cluster. Use the subdeployment targeting with specific JMS servers for a more controlled configuration.

WebLogic has two connection factories enabled by default that can be used by the application—**weblogic.jms.ConnectionFactory** and **weblogic.jms.XAConnectionFactory**.

It's recommended that an application use a custom connection factory since the two default connection factories cannot be tuned.

The custom connection factory **JMSAPPConnectionFactory** was created with the JNDI name jms.**appcf** and is ready to be used by the **JMSApp**.

Starting /stopping consumers for JMS Destination:

The WebLogic JMS subsystem allows a JMS destination queue to have its message operations ontrolled when needed. A queue destination can have its consumers paused and resumed on demand.

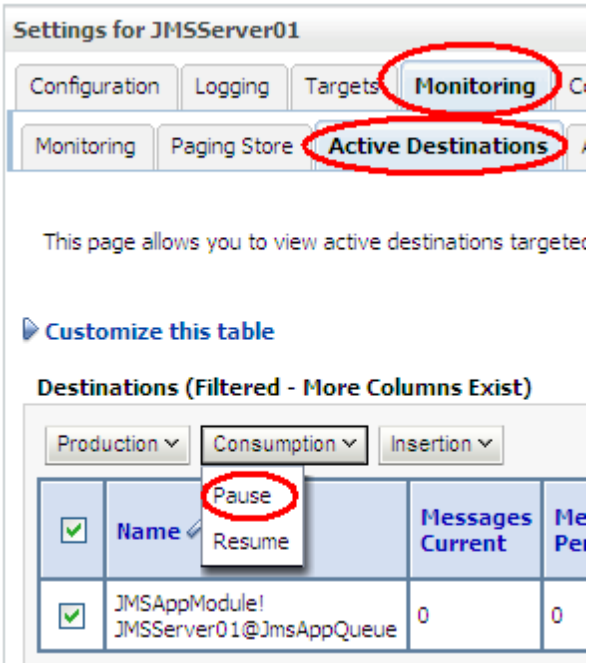
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

In this recipe, the JMSAppQueue queue will be used as an example. The message operations will use the Administration Console, so make sure the Administration Server is up and running.

How to do it:

To pause the consumers operations for the JMSAppQueue queue, do the following:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left; click on Messaging and then JMS Servers.
4. Click on the JMSServer01 link to open the Configuration page. Click on the Monitoring tab and then on the Active Destinations tab.
5. Click on the checkbox from the JMSAppModule!JMSServer01@JmsAppQueue queue and click on the Consumption button. Then click on the Pause link.



6. Click the Yes button to confirm.
7. Repeat these steps for the JMSServer02, JMSServer03, and JMSServer04 JMS servers.

To resume the consumers operations of the JMSAppQueue queue, do the following:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Services tree on the left; click on Messaging and then on JMS Servers.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

4. Click on the JMSServer01 link to open the Configuration page. Click on the Monitoring tab and then on the Active Destinations tab.
5. Click on the checkbox from the JMSAppModule!JMSServer01@JmsAppQueue queue and click on the Consumption button. Then click on the Resume link.
6. Click on the Yes button to confirm.
   Repeat these steps for the JMSServer02, JMSServer03, and JMSServer04 JMS servers.

How it works:

The consumer operations can be temporarily paused through the Administration Console when the JMS messages dequeue must stop in a situation of a maintenance window or a troubleshooting process.

Using the Server affinity to tune the distributed destinations' load balance:

The connection factory is the JMS resource used by the client application to control the behavior of the load balance algorithm used when publishing JMS messages to the distributed destination and its queue or topic members.
An application deployed in a production environment should preferably use a custom connection factory that can be tuned to fit the application requirements so as to avoid using the default WebLogic connection factories. The JMS message-publishing and load-balancing behavior can be tuned by changing the server affinity configuration of the connection factory. By default, the load-balancing and server-affinity options of the connection factory are enabled.

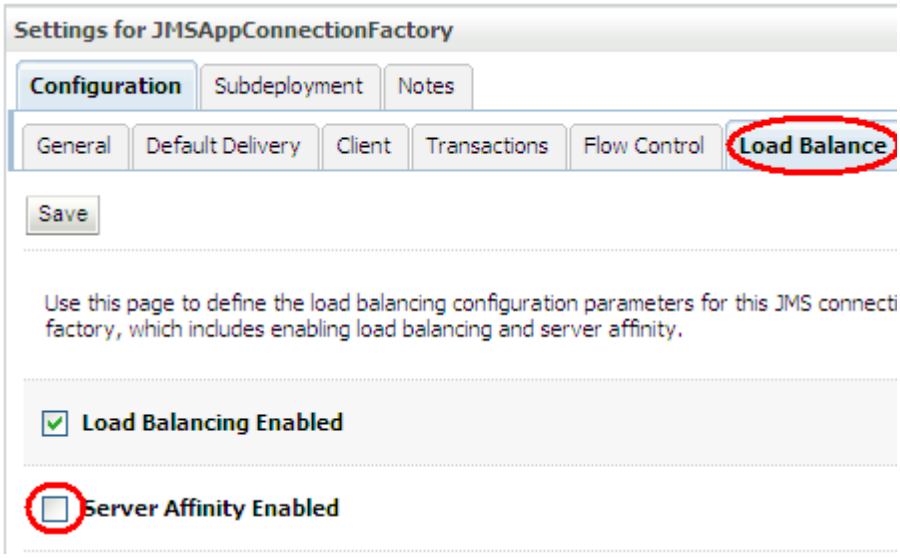
The JMSApp application uses the connection factory JMSAppCF created in the previous recipe to connect and publish JMS messages to the distributed queue JMSAppQueue.

Disabling the server affinity of the connection factory will force the load balance when publishing the JMS messages to the distributed queue. There is the drawback of the overhead of an additional TCP connection from the source Managed Server where the application request is running to the target Managed Server where the queue member of the distributed queue is hosted.

How to do:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Services tree on the left; click on Messaging and then on JMS Modules.
4. Click on the JMSAppModule link to open its Configuration page.
5. Click on the JMSAppConnectionFactory link and then click on the Load Balance tab to open the load-balancing options.
6. Disable the Server Affinity Enabled checkbox and click on the Save button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



Click on the Activate Changes button to finish.

How it works:

To illustrate the flow of the process, suppose an application request is being processed in the first anaged Server, PROD_Server01. The application will use the connection factory JMSAppCF to publish a JMS message to the JMSAppQueue distributed queue. Thanks to the server affinity option being enabled, the JMS message will be published to the local queue member of the JMSAppQueue distributed queue that is running in the same Managed Server, PROD_Server01.

In order to prioritize the delivery of a JMS message to a queue member of the distributed queue, the balancing algorithm verifies some characteristics such as if the queue member is local to the Managed Server, if it has a consumer, or if it has a persistent store. The disabled server affinity removes the influence of being a local queue. So, with the server affinity disabled, the same request running in PROD_Server01 will balance the load and publish the JMS messages to all queue members of the distributed queue, JMSAppQueue, running in all Managed Servers, PROD_Server01, PROD_Server02, PROD_Server03, and PROD_Server04.

Configuring messaging bridge with source and target distributed destinations :

The messaging bridge is used to forward JMS messages from **one source queue to another target ueue.**In this recipe, a bridge will be created to forward the JMS messages from the JMSAppQueue distributed queue to a hypothetic distributed queue with a JNDI name jms.remotequeue, hosted by a separate WebLogic domain named REMOTE_DOMAIN. The REMOTE_DOMAIN domain is configured with a cluster with two Managed Servers instances running at the addresses t3://remote01.domain.local:9001 and t3://remote02.domain.local:9002. A remote connection factory is available under the JNDI name jms.remoteappcf. Both local and remote queues are distributed destinations.

Getting ready:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

The Administrative Console is used to configure the messaging bridge, so make sure the Administration Server is up and running.The bridge destinations use the jms-xa-adp (XA) or jms-notran-adp (non-XA) resource adapters to connect to the destinations. They are not deployed by default so you have to deploy them. In this recipe, since the bridge is non-XA, the resource adapter jms-notranadp should be deployed. WebLogic can automatically deploy the resource adapter when you
create the bridge, but it's recommended you deploy it manually.

How to do it :

Create the directory to host the deployment plan to all machines, as follows:

1. Login as a wls user to the prod01 shell and create the directory.

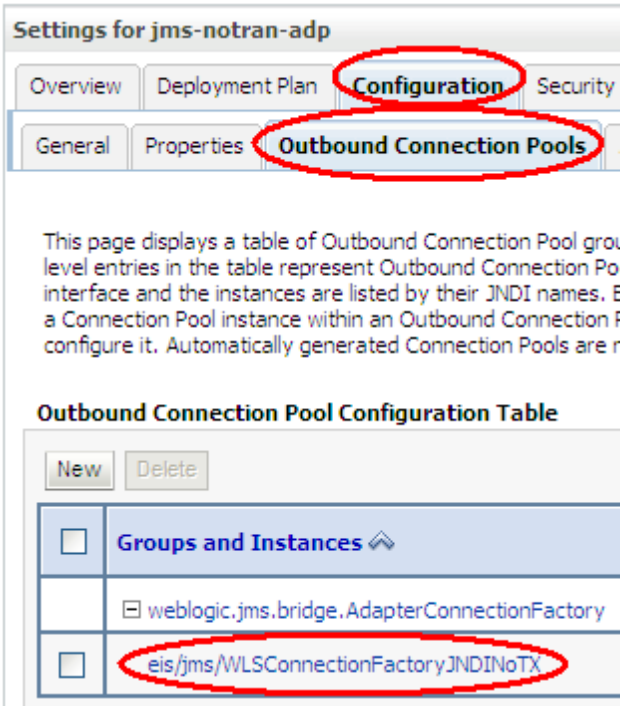
   ```
   [wls@prod01]$ cd $DOMAIN_HOME
   [wls@prod01]$ mkdir plans
   ```

2. Repeat this step for prod02.

3. Deploy the jms-notran-adp resource adapter, as follows:

4. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
5. Click on the Lock & Edit button to start a new edit session.
6. Click on the Deployments link on the navigation tree to the left.
7. Click on the Install button.
8. Type /oracle/Middleware/wlserver_12.1/server/lib in the Path field and click on Next.
9. Select the jms-notran-adp.rar resource adapter and click on the Next button.
10. Select the Install this deployment as an application radio button and click on Next.
11. Select the All servers in the cluster radio button from the PROD_Cluster target and click on the Next button.
12. Select the I will make the deployment accessible from the following location radio button; leave the other options in their default values and click on Next.
13. Click on the Finish button to open the Deployment's Configuration screen.
14. Click on the Configuration and then the Outbound Connection Pools tabs.
15. Click on the + button from the weblogic.jms.bridge.AdapterConnectionFactory link and click on the eis/jms/WLSConnectionFactoryJNDINoTX link.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

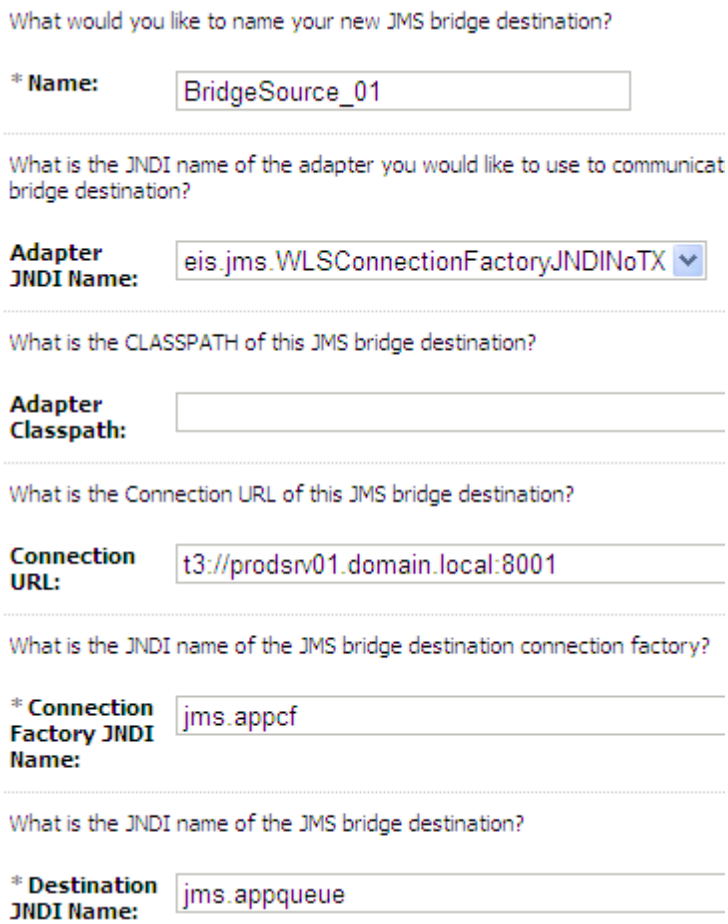


16. Click on the Connection Pool tab and change the Max Capacity field to the desired value (use 2x the number of bridges). Click on the Save button.
17. Type the full path,/oracle/Middleware/user_projects/domains/PROD_DOMAIN/plans/jms-notran-adp-Plan.xml, in the Path field. Click on the OK button.
18. Replicate the file /oracle/Middleware/user_projects/domains/PROD_DOMAIN/plans/jms-notran-adp-Plan.xml to all the machines of the cluster.
19. Click on the Activate Changes button to finish.
20. Go to the Deployments page again, select the checkbox to the left of the jms-notran-adp deployment, and click on the Start button and then the Servicing all request link.

Create the JMS bridge destinations as follows:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Services tree on the left and then click on +Messaging,+Bridges, and JMS Bridge Destinations.
4. Click on the New button to open the Create a New JMS Bridge Destination page.
5. Type BridgeSource_01 in the Name field and select the eis.jms. WLSConnectionFactoryJNDINoTX value from the Adapter JNDI Name drop-down menu. Leave the default blank value for the Adapter Classpath field and type t3://prodsrv01.domain.local:8001 on the Connection URL field, jms.appcf on the Connection Factory JNDI Name field, and jms.appqueue on the Destination
6. JNDI Name field. Click on the OK button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



7. Repeat the previous steps for creating the BridgeSource_02, BridgeSource_03,and BridgeSource_04 bridge destinations, using t3://prodsrv02.domain.local:8002, t3://prodsrv03.domain.local:8003, and t3://prodsrv04.
8. domain.local:8004 as the values to the Connection URL field.
9. Click on the New button again to create the remote destination target.
10. Type BridgeTarget in the Name field, select the eis.jms.WLSConnectionFactoryJNDINoTX from the Adapter JNDI Name field, and type t3://remote01.domain.local:9001,remote02.domain.local:9002
11. in the Connection URL field. Type jms.remoteappcf on the Connection Factory JNDI Name field and jms.remotequeue on the Destination JNDI Name field. Click on the OK button.
12. Click on the Activate Changes button to finish.

Create the messaging bridges as follows:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Services tree on the left; click on Messaging and then on Bridges.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

4. Click on the New button to open the Create a New Bridge page.
5. Type Bridge_01 in the Name field, leave the Selector field blank and select the Quality of Service field as the Atmost-once option. Enable the checkbox Started and click on the Next button.
6. In the Select an Existing Source Destination drop-down menu, choose BridgeSource_01 and click on the Next button.
7. Leave the Messaging Provider option in the default WebLogic Server 7.0 or higher and click on Next.
8. Select the BridgeTarget option from the Select an Existing Target Destination drop-down menu. Click on Next.
9. Again, leave the Messaging Provider option in the default WebLogic Server 7.0 or higher and click on Next.
10. On the target screen, enable only the PROD_Server01 checkbox and click on Next.
11. Click on the Finish button.
12. Repeat the previous steps to create the remaining bridges according to the following table:

Bridge	Source Destination	Target destination	Target
Bridge_01	BridgeSource_01	BridgeTarget	PROD_Server01
Bridge_02	BridgeSource_02	BridgeTarget	PROD_Server01
Bridge_03	BridgeSource_03	BridgeTarget	PROD_Server01
Bridge_04	BridgeSource_04	BridgeTarget	PROD_Server01

13. Click on the Activate Changes button to finish.

How it works:

One message bridge per queue member must be created when using a distributed destination like the JMSAppQueue queue to guarantee every message from every member is forwarded.

The target destination, BridgeTarget, uses a remote connection factory, jms. remoteappcf, to connect to the remote queue. Since the remote queue jms.remotequeue is also distributed, disable the server ffinity option of the remote connection factory jms.remoteappcf so the messages are load-balanced to all members of the distributed queue jms.remotequeue.

Depending on the number of bridges created in the WebLogic domain, you should tune the max connections settings of the resource adapter. The recommendation is to set the max connections value as 2 times the number of bridges. If you have 20 bridges, change the max connections to at least 40.

Using the Server affinity to tune the distributed destinations' load balance:

Relying on SAF to transfer JMS messages to another WebLogic domain:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Store-and-Forward (SAF) is another mechanism WebLogic provides to transfer messages from one source to a target destination.

The WebLogic messaging bridge and the SAF work in a similar way although the SAF is recommended over the messaging bridge when used to transfer messages between domains with WebLogic Version 9.0 or later. JMS SAF also simplifies the configuration since there is no
need to configure a local JMS queue destination as the SAF creates a local representation of the remote queue.

In this recipe, a SAF agent will be created to forward the JMS messages to a hypothetic distributed queue with a JNDI name jms.remotequeue, hosted by a separate WebLogic domain REMOTE_DOMAIN. The REMOTE_DOMAIN domain is configured with a cluster with
the two Managed Servers instances running at the t3://remote01.domain.local:9001 and t3://remote02.domain.local:9002 addresses. The local representation of the remote queue in the PROD_DOMAIN domain will be configured under the JNDI name jms.
remotequeue-saf.

Getting ready:

This recipe assumes the REMOTE_DOMAIN domain is up and running and the SAF agent can connect to the remote destinations. The configuration is changed by accessing the Administration Console, so make sure the Administration Server is running.

How to do it:

Create the SAF agents as follows:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left; click on +Messaging and then on Store-and-Forward Agents.
4. Click on the New button to open the Create a New Store-and-Forward Agentpage. Type SAFAgent_01 in the Name field and select the FileStore01 option in the Persistent Store drop-down menu. Change the Agent Type option to Sendingonly and click on the Next button.
5. Select the PROD_Server01 server as Target. Click on the Finish button.
6. Repeat the previous steps and create the SAFAgent_02, SAFAgent_03, and SAFAgent_04 agents. Use FileStore02, FileStore03, and FileStore04 as the values for the Persistent Store drop-down menu and PROD_Server02,PROD_Server03, and PROD_Server04 as the values for the Target drop-down menu.
7. Click on the Activate Changes button to finish.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Create the SAF resources as follows:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Services tree on the left; click on Messaging and then on JMS Modules.
4. Click on the JMSAppModule link to open the Configuration page. Click on the New button and select the Remote SAF Context radio button and then click on the Next button.
5. Type RemoteSAFContextApp in the Name field and t3://remote01.domain.local:9001,remote02.domain.local:9002 in the URL field. Type in the username and password of an already registered user of REMOTE_DOMAIN in the fields User Name, Password, and Confirm Password. Click on the OK button.
6. Click on the New button again and select the SAF Imported Destinations radio button and click on Next.
7. Type SAFImportedDestinationsApp in the Name field, choose the RemoteSAFContextApp value from the Remote SAF Context drop-down menu,and click on the Next button.
8. Click on the Next button to confirm the target to All servers in the cluster of PROD_Cluster.
   Click on the SAFImportedDestinationsApp value from the JMS resource list of the JMSAppModule module; then click on the Queues tab.
   Click on the New button and type SAFRemoteQueue in the Name field. Type jms.remotequeue in the Remote JNDI Name field and click on the OK button.
   Click on the created SAF queue, SAFRemoteQueue, and type jms.remotequeuesaf in the Local JNDI Name field. Click on the Save button.
   Click on the Activate Changes button to finish.

How it works:

The SAF agent and SAF resources were configured in PROD_DOMAIN to forward messages to the jms.remotequeue queue hosted by the REMOTE_DOMAIN domain. The JNDI representation of the jms.remotequeue queue in the PROD_DOMAIN domain is jms. remotequeue-saf.

The SAF agent was configured to use the persistent stores FileStore01, FileStore02,FileStore03, and FileStore04 to persist the messages.

With this configuration, every message posted to the jms.remotequeue-saf agent from the PROD_DOMAIN domain is forwarded to the jms.remotequeue queue of the REMOTE_DOMAIN domain.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

MONITORING WEBLOGIC SERVER:

In this chapter, we will cover the following recipes:

- Customizing the Administration Console tables
- Using the JRockit Mission Control Management Console
- Monitoring Linux with SAR
- Sending e-mail notifications with WLDF
- Generating an SMNP trap
- Creating a Monitoring Dashboard custom view
- Viewing historical data in the monitoring dashboard using a database

Introduction:

Oracle WebLogic Server provides some out of the box monitoring tools that will help maintain and support WebLogic production environments.

Although there are several other vendors offering WebLogic Server monitoring applications,it's important for a WebLogic Administrator to learn how to use the embedded tools provided,such as the Administration Console, the Monitoring Dashboard, and other common tools such as SAR, WLDF, and SNMP.

This chapter focuses on showing the system administrators how to use the tools and features to monitor WebLogic. It does not intend to provide the metrics and thresholds that should be used.

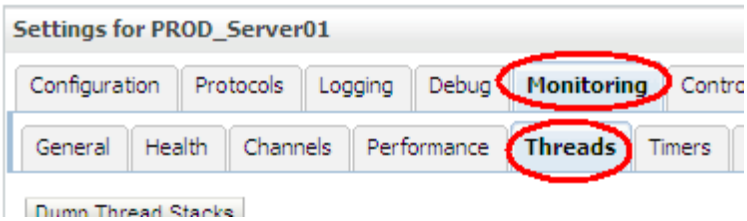
Access the Administration Console. The following procedure customizes the threads'monitoring table of the Managed Server Self-Tuning's thread pool.

How to do:

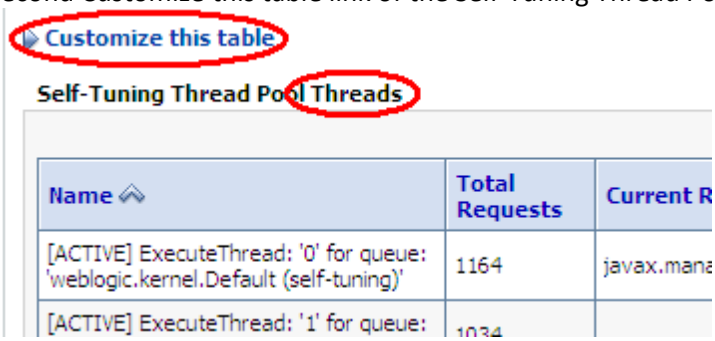
Carry out the following steps to customize the Administration Console tables:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the plus sign to open the Environment tree on the left and then click on the Servers link.
3. Click on any server, such as the PROD_Server01 link. Click on the Monitoring tab and then on the Threads tab to open the Threads page.

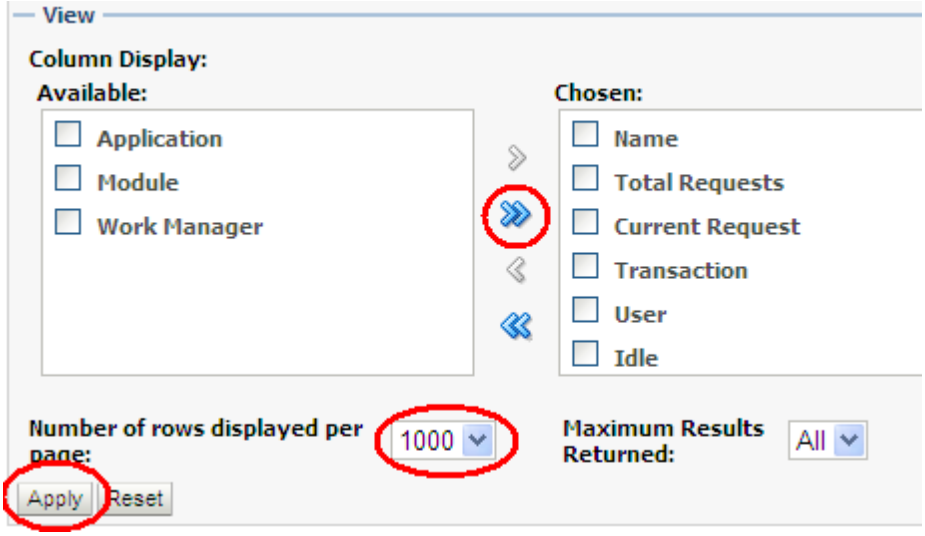
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



4. Click on the second Customize this table link of the Self-Tuning Thread Pool Threads table.



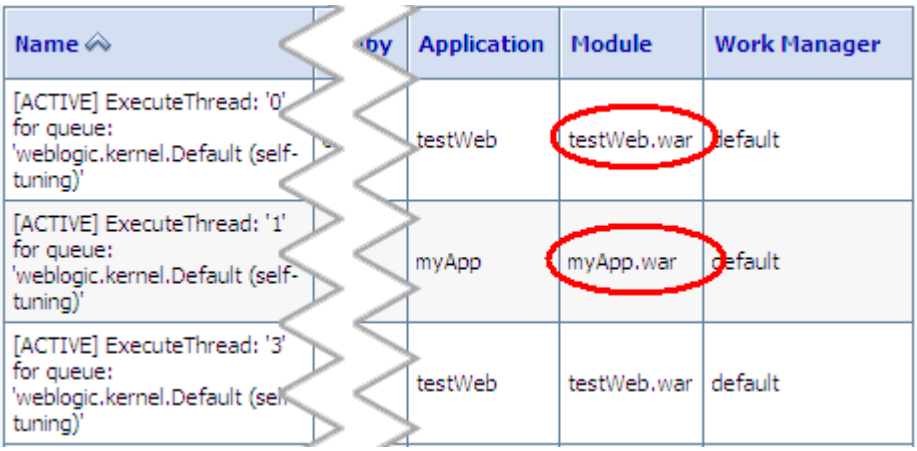
5. Click on the >> button to add the columns Application, Module, and Work Manager. Change the Number of rows displayed per page value to 1000.Click on the Apply button.



How it works:

The Administration Console allows the user to customize and add more columns to the monitoring tables. In this recipe, the Application, Module, and Work Manager columns are added to the Self-Tuning Thread Pool Threads table.The added columns are very useful to monitor the application requests being processed.The following table displays thread 0 processing a request of the testWeb applicationand thread 1 processing a request of the myApp application.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



Name	by	Application	Module	Work Manager
[ACTIVE] ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)'		testWeb	testWeb.war	default
[ACTIVE] ExecuteThread: '1' for queue: 'weblogic.kernel.Default (self-tuning)'		myApp	myApp.war	default
[ACTIVE] ExecuteThread: '3' for queue: 'weblogic.kernel.Default (self-tuning)'		testWeb	testWeb.war	default

Customizing the Administration Console monitoring tables is a common task and can be applied in a variety of tables, such as the data sources, the JMS queues, and transactions.

Using the JRockit Mission Control Management Console:

Mission Control is a monitoring and troubleshooting application provided with Oracle JRockit.From a monitoring point of view, Mission Control provides a Management Console to monitor the garbage-collection behavior, processor utilization by the JVM, memory allocation, thread utilization, and some other useful monitoring metrics.

Mission Control is a standalone application, so it must be started either locally from the same machine that WebLogic is running on or from a remote workstation.

This recipe will run Mission Control in a Microsoft Windows desktop and will remotely connect and monitor the PROD_Server01 Managed Server.

Getting Ready:

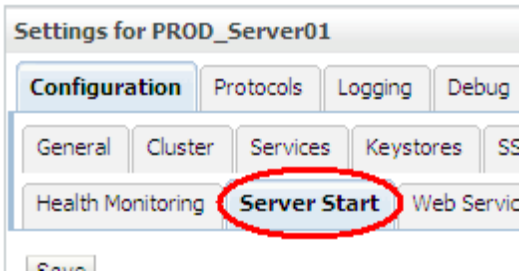
Oracle JRockit must be downloaded and installed in the Windows desktop. Download Oracle JRockit 6 for Microsoft Windows at http://www.oracle.com/technetwork/middleware/jrockit/downloads. The filename is jrockit-jdk1.6.0_XXX-windowsiaYY.exe, where XXX stands for the JRockit release and JDK version and YY stands for32 bits or 64 bits. Choose the version that matches your desktop and accept the licenseagreement to download it.

How to do it:

Enable the PROD_Server01 Managed Server to accept JMX connections from Mission Control:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Environment tree on the left and then click Servers.
4. Click on the PROD_Server01 link and then click the Server Start tab.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

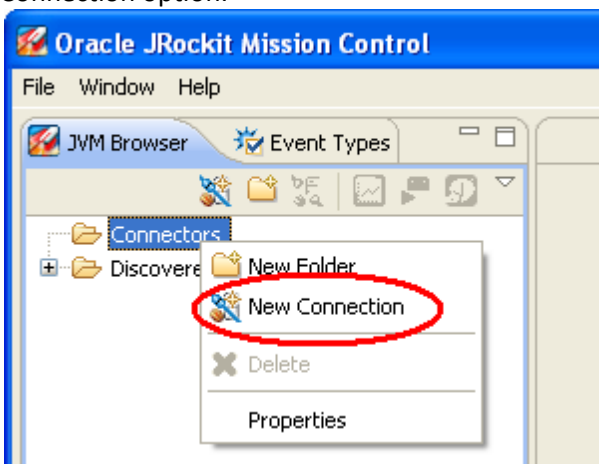
i. 

5. Add the following to the Arguments field and click on the Save button: Xmanagement:autodiscovery=false,authenticate=false,ssl=false,interface=prodsrv01.domain.local ,port=8081 -Djava.rmi.
6. server.hostname=prodsrv01.domain.local -Djavax.management.builder.initial=weblogic.management.jmx.mbeanserver.
7. WLSMBeanServerBuilder

8. Click on the Activate Changes button.
9. Restart PROD_Server01.

Start Mission Control on the desktop, as follows:

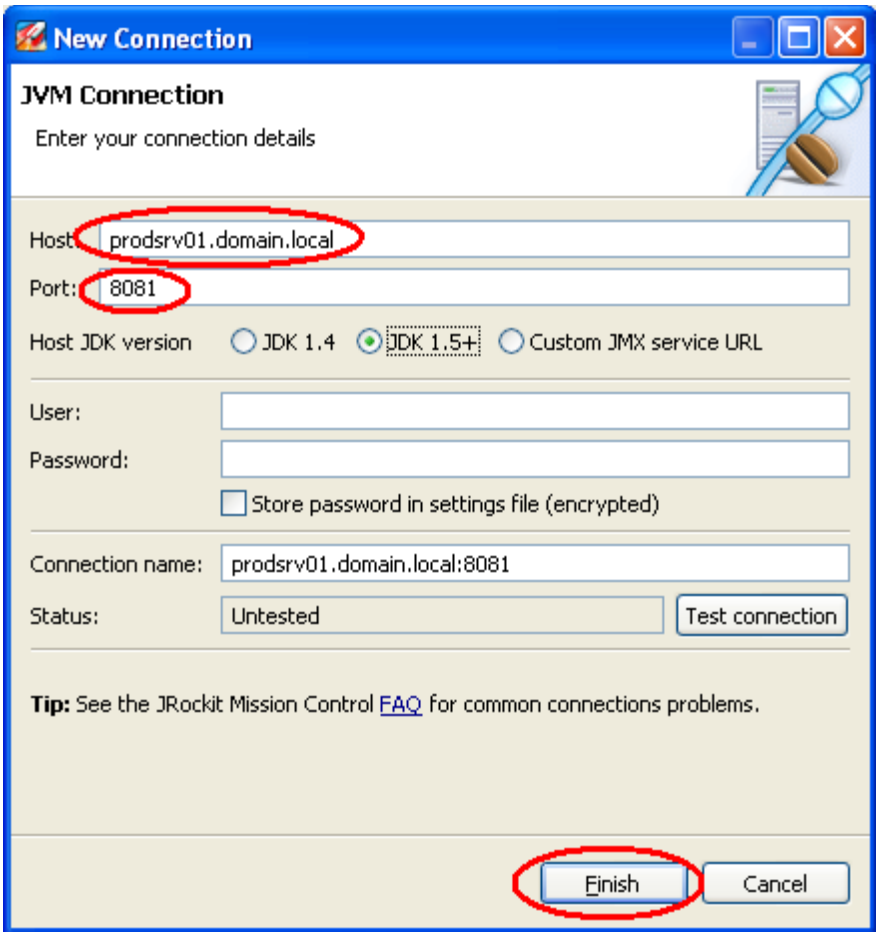
1. Start Mission Control by double-clicking on the Oracle JRockit Mission Control icon.
2. On the JVM Browser panel to the left, right-click on the Connectors folder and click On the New Connection option.



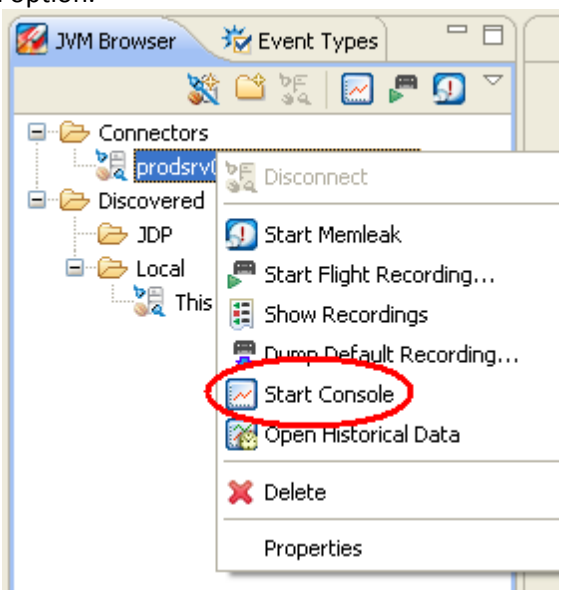
3. Type prodsrv01.domain.local in the Host field and 8081 in the Port field and click on the Finish button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

ii.



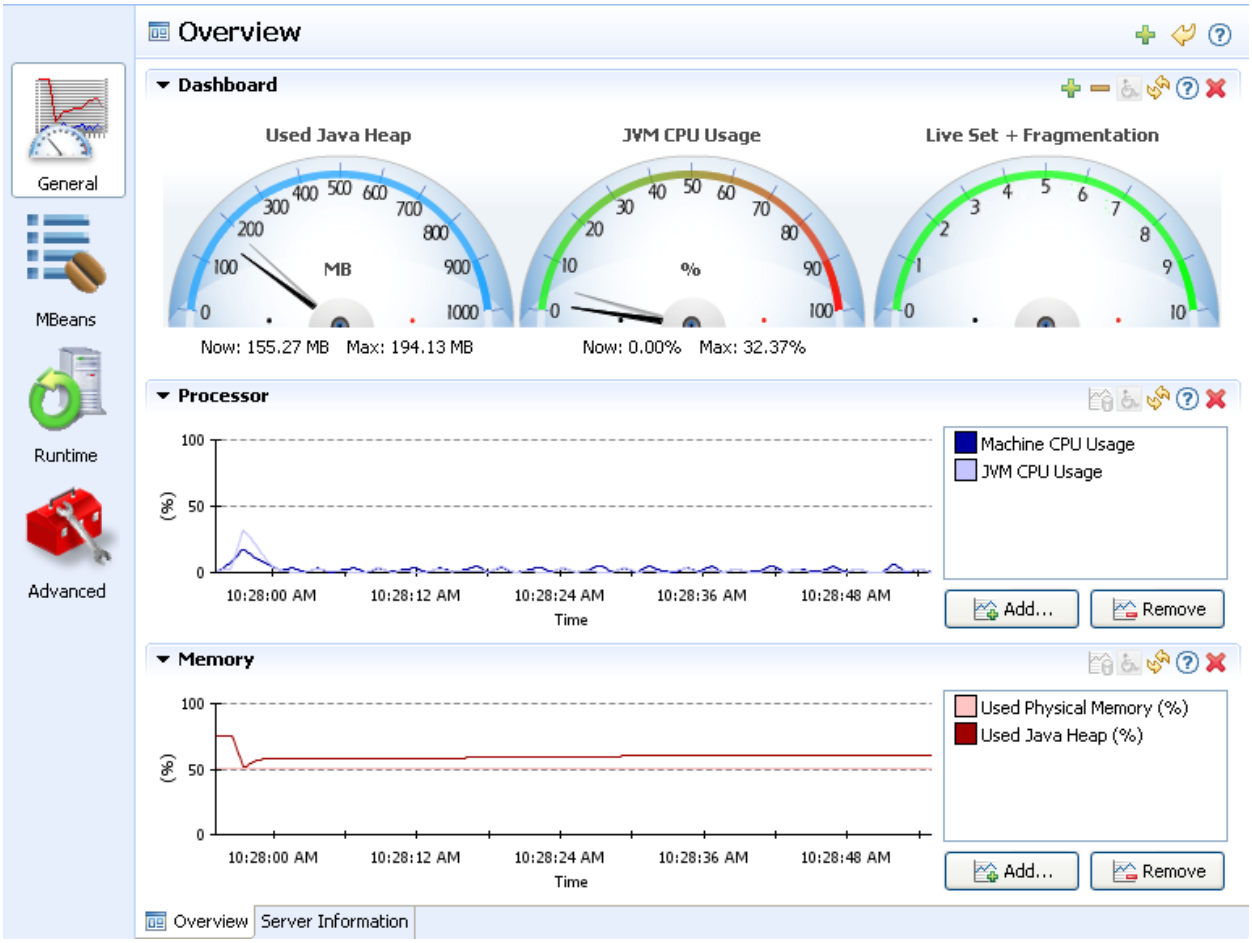
4. Right-click on the newly created connection, prodsrv01.domain.local:8081,and click on the Start Console menu option.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

Mission Control connects to the specified host and port defined in the Xmanagement start-up argument and starts monitoring the PROD_Server01 JVM.



Note :

Add the start-up arguments to monitor the other WebLogic instances. Mission Control can connect to any running JRockit JVM.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

MONITORING LINUX WITH SAR:

A Linux host with Red Hat Enterprise Linux or Oracle Linux can be monitored using the SAR command-line utility. The SAR is included in the SYSSTAT package bundle and is usually included with these Linux distributions.
SAR retrieves activity counters of the operational system, such as CPU, memory, disk, and network I/O usage. By default, it keeps a history of seven days, so it is a very useful tool to retrieve past reports and quickly search for behavioral patterns.
This recipe will retrieve some statistics from the prod01 machine using the SAR command-line utility.
Getting Ready:
SAR should already be installed in a Red Hat or Oracle Linux distribution. If it is not installed, you can use the yum package management utility to install the SYSSTAT package, which includes SAR.
As root user, execute the yum command and follow the onscreen instructions to install SYSSTAT:
[root@prod01]$ yum install sysstat

How to do:

To retrieve the queue length and load averages from the current day, as a wls user execute the following command:
[wls@prod01]$ sar –q

This will display the following result:

	runq-sz	plist-sz	ldavg-1	ldavg-5	ldavg-15
08:20:01 AM					
08:30:01 AM	4	1896	0.31	0.93	0.91
08:40:01 AM	3	1903	0.10	0.35	0.61
08:50:01 AM	5	1908	0.70	1.02	0.91
09:00:01 AM	9	1920	0.14	0.32	0.60
09:10:01 AM	4	1900	0.89	0.58	0.61
09:20:01 AM	4	1902	0.49	0.95	0.97
09:30:01 AM	10	1931	0.18	0.47	0.74
09:40:01 AM	4	1900	1.66	1.51	1.13
09:50:01 AM	5	1907	0.82	0.79	0.91
10:00:01 AM	10	1914	1.25	1.02	0.97
10:10:01 AM	5	1901	1.00	1.84	1.59

To retrieve a past CPU usage from the 21st day of the month, as a wls user execute the following **command:**
[wls@prod01]$ sar –u –f /var/log/sa/sa21

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

This will display the following result:

08:20:01 AM	CPU	%user	%nice	%system	%iowait	%steal	%idle
08:30:01 AM	all	1.77	1.37	0.26	0.31	0.00	96.29
08:40:01 AM	all	2.39	1.37	0.29	0.37	0.00	95.58
08:50:01 AM	all	1.78	1.36	0.27	0.29	0.00	96.30
09:00:01 AM	all	1.88	1.36	0.26	0.28	0.00	96.21
09:10:01 AM	all	2.87	1.37	0.31	1.01	0.00	94.44
09:20:01 AM	all	2.42	1.37	0.29	0.30	0.00	95.63
09:30:01 AM	all	2.17	1.36	0.27	0.50	0.00	95.70
09:40:01 AM	all	2.38	1.36	0.30	0.37	0.00	95.59
09:50:01 AM	all	2.53	1.37	0.32	0.35	0.00	95.44
10:00:01 AM	all	2.83	1.37	0.31	0.43	0.00	95.06
10:10:01 AM	all	2.71	1.37	0.34	0.60	0.00	94.98

How it works:

The SAR utility is very flexible and provides a quick way to watch the host behavior for the current day and for the past seven days. SAR runs every 10 minutes and saves a summary at the end of the day.
These are the default values in the crontab and can be adjusted.

Apart from the options discussed earlier, we can view more fine-grained data as well.

Collecting SAR data every minute
SAR can store statistical data in a more fine-grained time interval.
Log in as a root user to the shell and execute the following command:
[root@prod01]$ vi /etc/cron.d/sysstat
Locate the following lines:
run system activity accounting tool every 10 minutes
*/10 * * * * root /usr/lib64/sa/sa1 1 1
Change these lines to the following:
run system activity accounting tool every 1 minute
*/1 * * * * root /usr/lib64/sa/sa1 1 1
Type :wq! to save and close the file.

Sending e-mail notifications with WLDF

The WebLogic Diagnostic Framework (WLDF) is a set of functionalities to monitor,collect, and analyze runtime counters, metrics, and statistical and diagnostic data fromvarious WebLogic Server components. The metrics can be gathered from the JRockit JVM,the WebLogic domain, the WebLogic clusters, Managed Servers, applications, and everycomponent that exposes data through MBeans.

WebLogic Administrators can include active monitoring in production environments with WLDF. It is possible to configure WebLogic to send e-mail alerts when certain conditions and metrics are reached.

In this recipe, the WebLogic domain will be configured to send e-mail alerts when the WebLogic thread pool from a Managed Server has queued requests waiting to be processed.

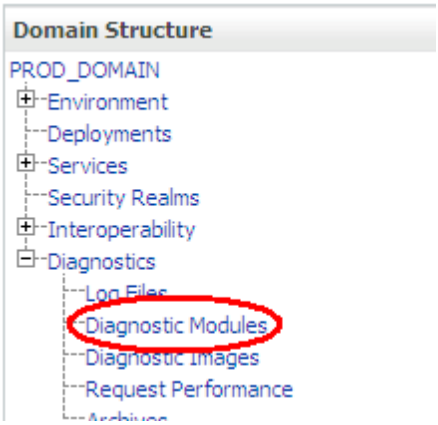
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Follow these steps to create the EmailAlertMailSession mail session:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left and then click on Mail Sessions.
4. Click on the New button and type EmailAlertMailSession in the Name field.Type mail/emailAlertMailSession in the JNDI Name field.
5. Add to JavaMail the properties needed according to the SMTP Server used:
   mail.smtp.host=<smtp-host>
   mail.smtp.port=<smtp-port>
6. Click on the Next button and select the All servers in the cluster target. Click on the Finish button.
7. Click on the Activate Changes button to finish.

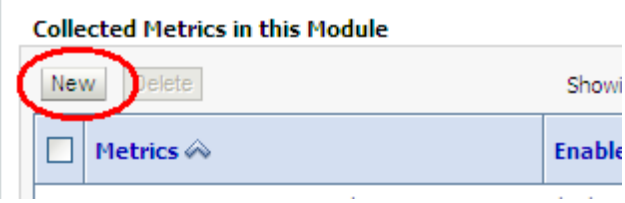
Follow these steps to create the EmailAlertModule WLDF module:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Diagnostics tree on the left and then click on Diagnostic Modules

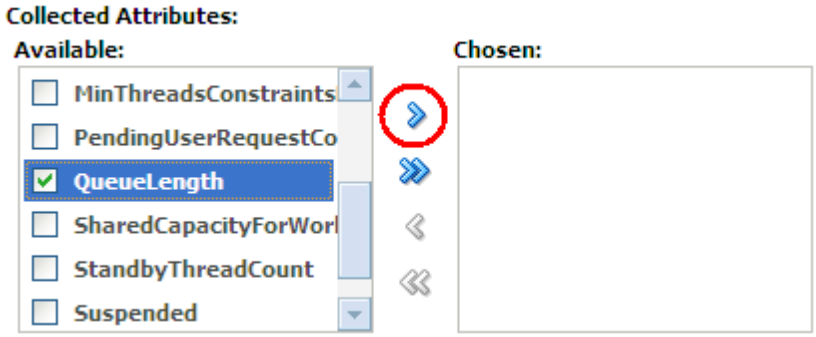


4. In the Summary of Diagnostic Modules page, click on the New button.
5. Type EmailAlertModule in the Name field and click on the OK button.
6. Click on the newly created EmailAlertModule WLDF module and then on the Targets tab. Select the All servers in the cluster radio button and click on the Save button.
7. Click on the Configuration tab and then the Collected Metrics tab. Change the Sampling Period value to 60000 and click on the Save Button.
8. Click on the New button in the Collected Metrics in this Module table to create a new Harvester.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

ii. 

10. Choose ServerRuntime from the MBean Server location drop-down menu and click on the Next button.
11. Select weblogic.management.runtime.ThreadPoolRuntimeMBean from the MBean Type drop-down menu and click on the Next button.
12. Select the QueueLength item from the Collected Attributes table on the left and click on the > icon to move it to the Chosen table on the right. Click on the Next button.

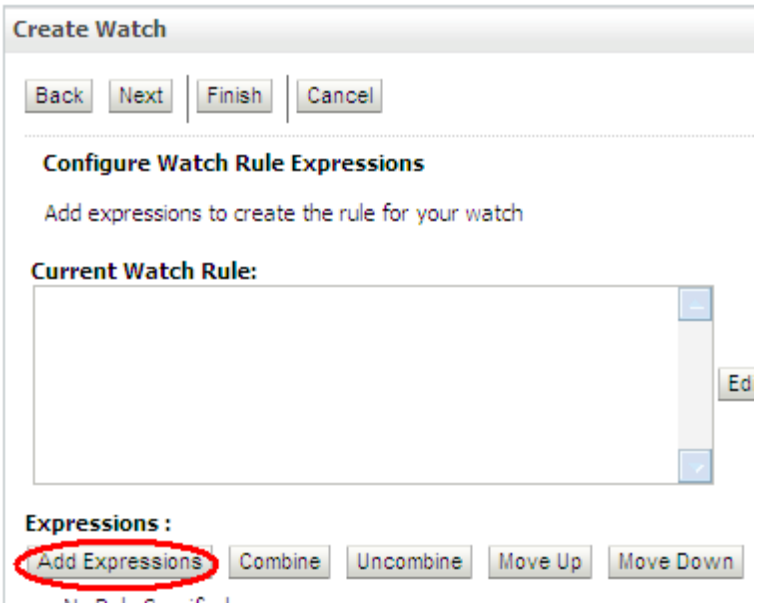
i. 

13. In the Select Instances page, leave the Chosen Collected Instances and Instance Expressions textboxes empty and then click on the Finish button.

Follow these steps to create the watches and notifications for the WLDF module:

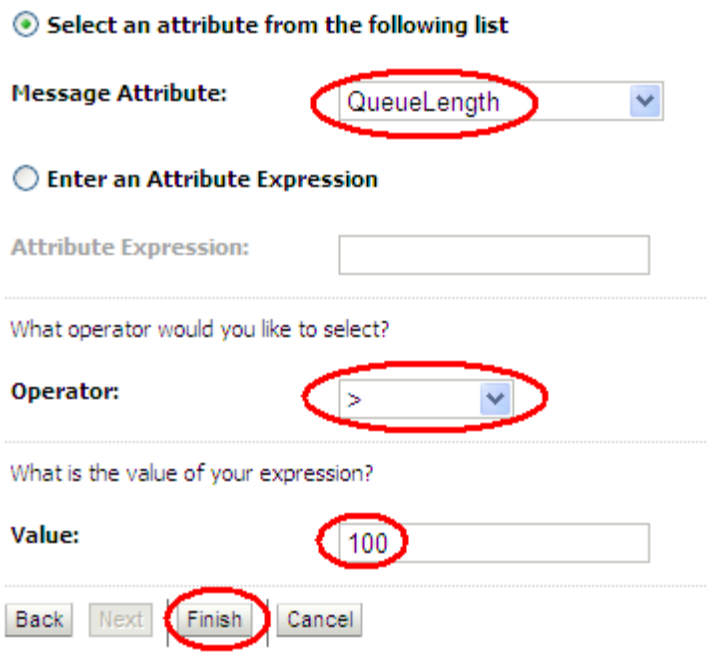
1. Click on the Watches and Notifications tab and then on the Notifications tab below.Click on the New button to create a new notification.
2. Select the SMTP (E-Mail) option in the Type drop-down menu and click on the Next button.
3. Type EmailAlertNotification in the Notification Name field and click on the Next button.
4. Select the EmailAlertMailSession option from the Mail Session Name drop-down menu. Type an e-mail address in the E-Mail Recipients textbox and click on the Finish button. This e-mail address will receive the alerts.
5. Now click on the Watches tab and click on the New button to create a new watch.
6. Type EmailAlertWatch in the Watch Name field. Leave the Collected Metrics option selected in the Watch Type field and click on the Next button.
7. Click on the Add Expressions button to add a new expression rule.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



8. Select the Server Runtime option from the MBean Server location drop-down menu and then click on the Next button.
9. Select **weblogic.management.runtime.ThreadPoolRuntimeMBean** from the MBean Type drop-down menu and click on the Next button.
10. Select the Enter a custom Instance radio button and type com.bea:Name=Thread PoolRuntime,ServerRuntime=PROD_Server*Type=ThreadPoolRuntime in the Custom Instance field. Click on the Next button.
11. Select the **QueueLength** option from the Message Attribute drop-down menu and select the > option from the Operator drop-down menu and type 100 in the Value field. Click on the Finish button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



12. On the following screen, click on the Finish button again.
13. Click on the EmailAlertWatch to assign the notification. Click the Notifications tab.
14. Select EmailAlertNotification from the Available table to the left and click on the >icon to move it to the Chosen table to the right. Click on the Save button.
15. Click on the Activate Changes button to finish.

How it works:

WLDF is a very powerful framework and helps WebLogic Administrators to properly monitor a WebLogic domain.
The diagnostic module was created with a WLDF Harvester for the QueueLength attribute of the ThreadPoolRuntimeBean MBean and a sampling period of 60000. The sampling period is the interval between metric-collection cycles, in milliseconds. The data collected by the harvester is stored in a WLDF archive.

The default archive is stored in a file-based format at $DOMAIN_HOME/servers/<serverName>/data/store/diagnostics.

The scenario of 100 queued requests waiting to be processed can indicate a potential problem because all threads of the WebLogic Server thread pool could be busy and the requests will not be processed fast enough, causing the incoming requests to pile up. The EmailAlertWatch watch was created for such situations to observe when the QueueLength value is greater than 100. When this condition is reached, the watch triggers an event to the EmailAlertNotification notification and an e-mail is sent to the specified recipients.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

The notification can also be configured as an SNMP trap, a JMS message, a JMX operation, or a Diagnostic Image.

Generating an SNMP trap

WebLogic provides Simple Network Management Protocol (SNMP) standards to monitor the domain, Managed Servers, and applications the same way the exposed MBean's attributes were monitored in the last recipe using WLDF.

In this recipe, an SNMP agent will be added to monitor all Managed Servers of the PROD_ Domain, watching for the same QueueLength attribute of the ThreadPoolRuntime MBean.

An SNMP trap will be triggered by adding a gauge monitor to verify if the QueueLength value is above 100.

Getting ready :

The Server SNMP agent called PROD_SNMPAgent will be created using the Administration Console, so make sure the Administration Server is running.
This recipe assumes there is an SNMP Manager listening for an SNMP trap at the hostname snmphost on port 162.

How to do it:

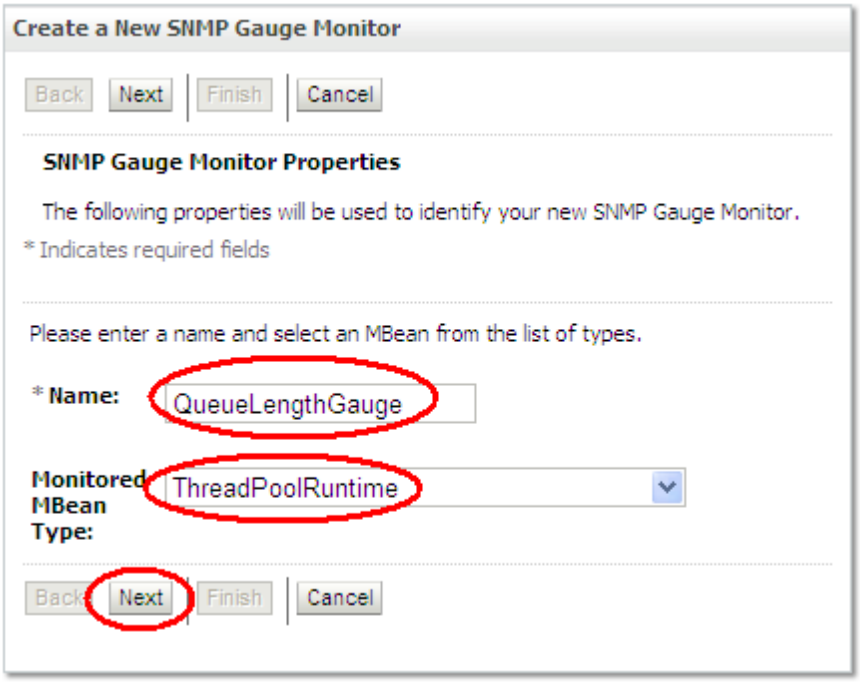
Follow these steps to create the Server SNMP Agent:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Diagnostics tree on the left and then click on SNMP.
4. Click on the New button from the Server SNMP Agent. Type PROD_SNMPAgent in the Name field and click on the OK button.
5. Click on the newly created SNMP Agent, PROD_SNMPAgent.
6. Click on the Enabled checkbox to enable it. Type 1610 in the SNMP UDP Port field and 7050 in the Master AgentX Port field. Click on the Save button.
7. Click on the Targets tab and select the All servers in the cluster option from the PROD_Cluster target. Click on the Save button.

Follow these steps to create the gauge monitor and the SNMP trap:

1. Click on the Configuration tab and then on the Gauge Monitors tab. Click on the New button to create a new SNMP Gauge Monitor.
2. Type QueueLengthGauge in the Name field. Select the ThreadPoolRuntime option from the Monitored MBean Type drop-down menu. Click on the Next button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



2. Select the QueueLength option from the Monitored Attribute Name drop-down menu and click on the Finish button.
3. Click on the newly created QueueLengthGauge gauge and type 100 in the Threshold High field. Click on the Save button.
4. Click on the Servers tab and enable the PROD_Server01, PROD_Server02,PROD_Server03, and PROD_Server04 checkboxes. Click on the Save button.
5. Go to Diagnostics | SNMP on the left tree again and click on the PROD_SNMPAgent link.
6. Click on the Trap Destinations tab and click on the New button.
7. Type QueueLengthTrap in the Name field. Type snmphost in the Host field and 162 in the Port field. Click on the OK button.
8. Click on the Activate Changes button to finish.

How it works:

same notification from the previous recipe was created, but in this recipe instead of sending an e-mail alert using WLDF, a SNMP trap is generated to a hypothetical SNMP Manager running at snmphost on port 162.
The threshold condition is the same. The SNMP trap is sent when the QueueLength attribute from the WebLogic thread pool from a Managed Server of the PROD_Cluster value is above 100.
The trap generated has the following format:
--- Snmp Trap Received ---
Version : v1
Source : UdpEntity:<source_ip>:1610
Community : public
Enterprise : enterprises.140.625

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

AgentAddr : <source_ip>
TrapOID : enterprises.140.625.0.75
RawTrapOID : 1.3.6.1.4.1.140.625.0.75
Trap Objects : {
{ enterprises.140.625.100.5=Sun Nov 11 15:50:55 BRST 2012 }
{ enterprises.140.625.100.10=PROD_Server02 }
{ enterprises.140.625.100.55=jmx.monitor.gauge.high }
{ enterprises.140.625.100.60=100 }
{ enterprises.140.625.100.65=435 }
{ enterprises.140.625.100.70=com.bea:Name=ThreadPoolRuntime,Server
Runtime=PROD_Server02,Type=ThreadPoolRuntime }
{ enterprises.140.625.100.75=ThreadPoolRuntime }
{ enterprises.140.625.100.80=QueueLength }
}

CREATING A MONITORING DASHBOARD CUSTOM VIEW:

The Monitoring Dashboard is embedded with the Oracle WebLogic Server 12c. The dashboard is an application deployed in the Administration Server and is accessible at the URL http://adminhost.domain.local:7001/console/dashboard.

The Monitoring Dashboard can graphically display the metrics collected from the WLDF and exposed by the MBeans.

In this recipe, a Dashboard custom view will be created to display the Queue Length attribute from the ThreadPoolRuntime MBean of the PROD_Server01 Managed Server. This is the same attribute monitored in the previous recipes.

Getting Ready:

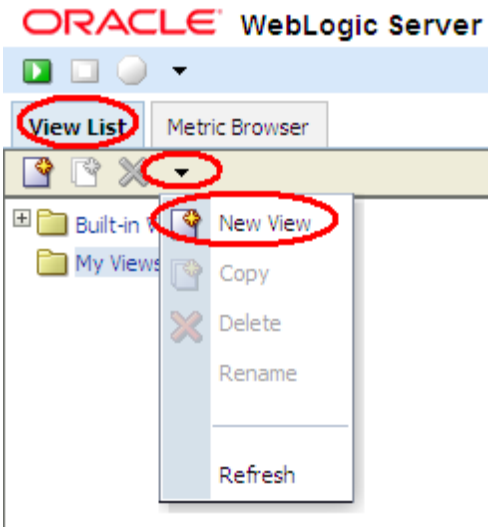
The Monitoring Dashboard is a functionality of the Administration Console. Make sure the Administration Server is running.

How to do it:

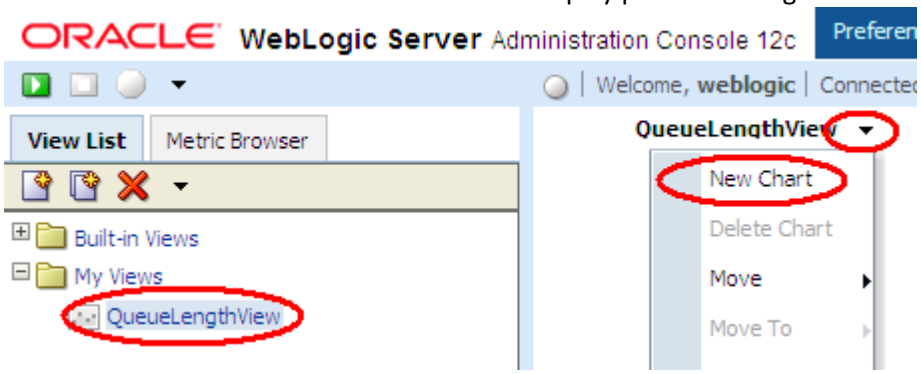
Carry out the following steps to create a custom view:

1. Access the Monitoring Dashboard with your web browser at http://adminhost.domain.local:7001/console/dashboard.
2. From the View List tab, click on My Views and then on the down arrow to display the drop-down menu. Click on the New View menu option

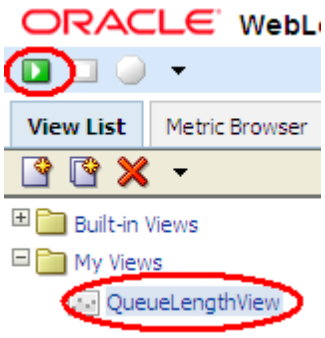
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



3. Type QueueLengthView to set the View name.
4. Click on the arrow down on the chart of the display pane to the right. Click on New Chart.



5. 5. Click on the Metric Browser tab, select the PROD_Server01 option from theServers drop-down menu, and click on the GO button.
6. 6. Click on the ThreadPool item from the Types selection list and then on the ThreadPoolRuntime from the Instances selection list. Click on the QueueLength (int) option and drag it to the QueueLengthView chart on the right display panel.
7. Click on the green start button on the left to start collecting.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

A custom view was created with a new chart to display the QueueLength counter metrics. The QueueLength attribute is collected directly from the exposed ThreadPoolRuntime MBean.

Other charts can be created and any other exposed runtime MBean attribute can be added. The QueueLength attribute from the other Managed Servers of the PROD_Cluster cluster can also be added to the same chart.

Viewing historical data in the monitoring dashboard using a database:

In the previous recipe, a Dashboard custom view was created to display the QueueLength attribute of the ThreadPoolRuntime MBean of the Managed Servers of the PROD_Cluster cluster.

The Dashboard can collect the data from two sources. One is from the exposed MBeans, displaying polled runtime statistics like the previous recipe. The other form is from the WLDF archive data collected from a WLDF harverster. This stored data is called collected metrics by the Monitoring Dashboard.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

In this recipe, the PROD_Server01, PROD_Server02, PROD_Server03, and PROD_Server04 WLDF archives will be configured to use a database instead of the default file-based archive to hold the data collected from a WLDF Harvester. A new custom view will be created to display this data and it will be able to display all the collected historical data instead of the default 30 minutes collected from the polled collector. The QueueLength attribute will be used.

The database is an Oracle database, running in the dbhost hostname and listening to the port 1521. The listener is accepting requests to the service named dbservice.
Getting ready:

The recipe will use the database dbhost, so create a new JDBC data source named ds-wldf-archive, with a JNDI name jdbc/ds-wldf-archive targeted to the PROD_Cluster cluster.

The database tables WLS_EVENTS and WLS_HVST have to be created. WebLogic uses these tables to store the WLDF archive data.

How to do it:

Follow these steps to create tables in the Oracle dbhost database:

1. Connect to the database at dbhost, port 1521, and service name dbservice.
2. Run the following SQL script:

```
DROP TABLE WLS_EVENTS;
CREATE TABLE WLS_EVENTS (
RECORDID number NOT NULL,
TIMESTAMP number default NULL,
CONTEXTID varchar2(128) default NULL,
TXID varchar2(32) default NULL,
USERID varchar2(32) default NULL,
TYPE varchar2(64) default NULL,
DOMAIN varchar2(64) default NULL,
SERVER varchar2(64) default NULL,
SCOPE varchar2(64) default NULL,
MODULE varchar2(64) default NULL,
MONITOR varchar2(64) default NULL,
FILENAME varchar2(64) default NULL,
LINENUM number default NULL,
CLASSNAME varchar2(250) default NULL,
METHODNAME varchar2(64) default NULL,
METHODDSC varchar2(4000) default NULL,
ARGUMENTS clob default NULL,
RETVAL varchar2(4000) default NULL,
PAYLOAD blob default NULL,
CTXPAYLOAD varchar2(4000),
```

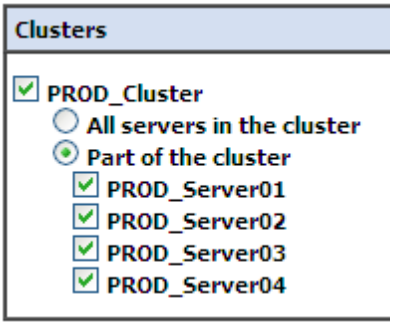

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

```
DYES timestamp default NULL,
THREADNAME varchar2(128) default NULL
);
DROP TABLE WLS_HVST;
CREATE TABLE WLS_HVST (
RECORDID number NOT NULL,
TIMESTAMP number default NULL,
DOMAIN varchar2(64) default NULL,
SERVER varchar2(64) default NULL,
TYPE varchar2(64) default NULL,
NAME varchar2(250) default NULL,
ATTRNAME varchar2(64) default NULL,
ATTRTYPE number default NULL,
ATTRVALUE varchar2(4000)
);
DROP SEQUENCE MON_EVENTS;
CREATE SEQUENCE MON_EVENTS
START WITH 1
INCREMENT BY 1
NOMAXVALUE;
DROP TRIGGER TRIG_EVENTS;
CREATE TRIGGER TRIG_EVENTS
BEFORE INSERT ON WLS_EVENTS
FOR EACH ROW
BEGIN
SELECT MON_EVENTS.NEXTVAL INTO :NEW.RECORDID FROM DUAL;
END;
/
DROP SEQUENCE MON_HVST;
CREATE SEQUENCE MON_HVST
START WITH 1
INCREMENT BY 1
NOMAXVALUE;
DROP TRIGGER TRIG_HVST;
CREATE TRIGGER TRIG_HVST
BEFORE INSERT ON WLS_HVST
FOR EACH ROW
BEGIN
SELECT MON_HVST.NEXTVAL INTO :NEW.RECORDID FROM DUAL;
END;
```


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Follow these steps to create the JDBC data source:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Services tree on the left and then click Data Sources.
4. Click on the New button and click on Generic Data Source.
5. Type ds-wldf-archive in the Name field and jdbc/ds-wldf-archive in the JNDI Name. Leave the Database Type drop-down menu with the Oracle option selected. Click on the Next button.
6. Choose *Oracle's Driver (Thin) for Service connections; Versions:9.0.1 and later from the Database Driver drop-down menu.Click on the Next button.
7. Leave the Transaction options with the default values and click on the Next button.
8. In the Connection Properties page, type dbservice in the Database Name field,dbhost in the Host Name field, and 1521 in the Port field. Complete the Database User Name, Password, and Confirm Password fields with dbuser and dbpwd. Click on the Next button.
9. Click on the Next button in the Test Database Connection page.
10. Click on each Managed Server checkbox of the PROD_Cluster cluster. Don't click on the All servers in the cluster radio button. Click on the Finish button.



11. Click on the Activate Changes button to finish.

Follow these steps to configure the WLDF archive to store the data to a JDBC based store:

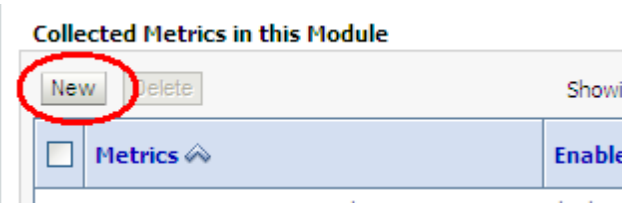
1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Diagnostics tree on the left and then click on Archive.
4. Click on the PROD_Server01 Managed Server.
5. Select the JDBC option from the Type drop-down menu. Select the ds-wldfarchive option from the Data Source drop-down menu. Click on the Save button.
6. Repeat the previous steps for the PROD_Server02, PROD_Server03, and PROD_Server04 Managed ervers.
7. Click on the Activate Changes button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Follow these steps to create a new WLDF module and a new WLDF Harvester:

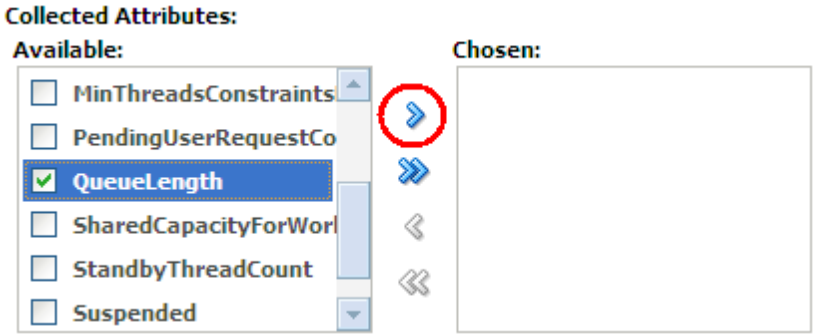
1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the plus sign to open the Diagnostics tree on the left and then click on Diagnostic Modules.
4. In the Summary of Diagnostic Modules page, click on the New button.
5. Type QueueLengthModule in the Name field and click on the OK button.
6. Click on the newly created QueueLengthModule module and click on the Targets tab. Select the All servers in the cluster radio button and click on the Save button.
7. Click on the Configuration tab and then the Collected Metrics tab. Change the Sampling Period to 60000 and click on the Save Button.
8. Click on the New button in the Collected Metrics in this Module table to create a new WLDF harvester.

ii.



8. Choose ServerRuntime from the MBean Server location drop-down menu and click on the Next button.
9. Select weblogic.management.runtime.ThreadPoolRuntimeMBean from the MBean Type drop-down menu and click on the Next button.
10. Select the QueueLength item from the Collected Attributes table on the left and click the > icon to move it to the Chosen table on the right. Click on the Finish button.

i.

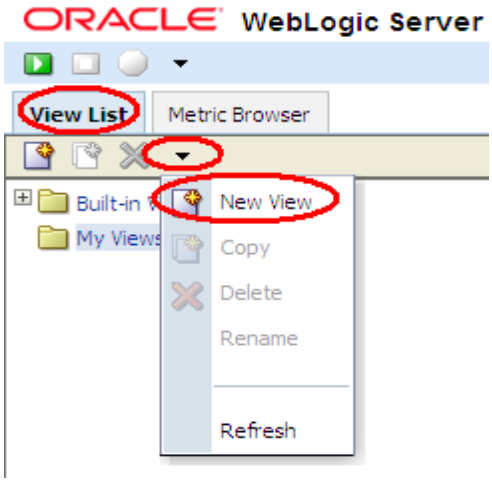


11. Click on the Activate Changes button and restart all Managed Servers.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Follow these steps to create a new custom view in the Monitoring Dashboard:

1. Access the Monitoring Dashboard with your web browser at http://adminhost.domain.local:7001/console/dashboard.
2. From the View List tab, click on My Views and then click on the down arrow to displaythe drop-down menu. Click on the New View menu option.

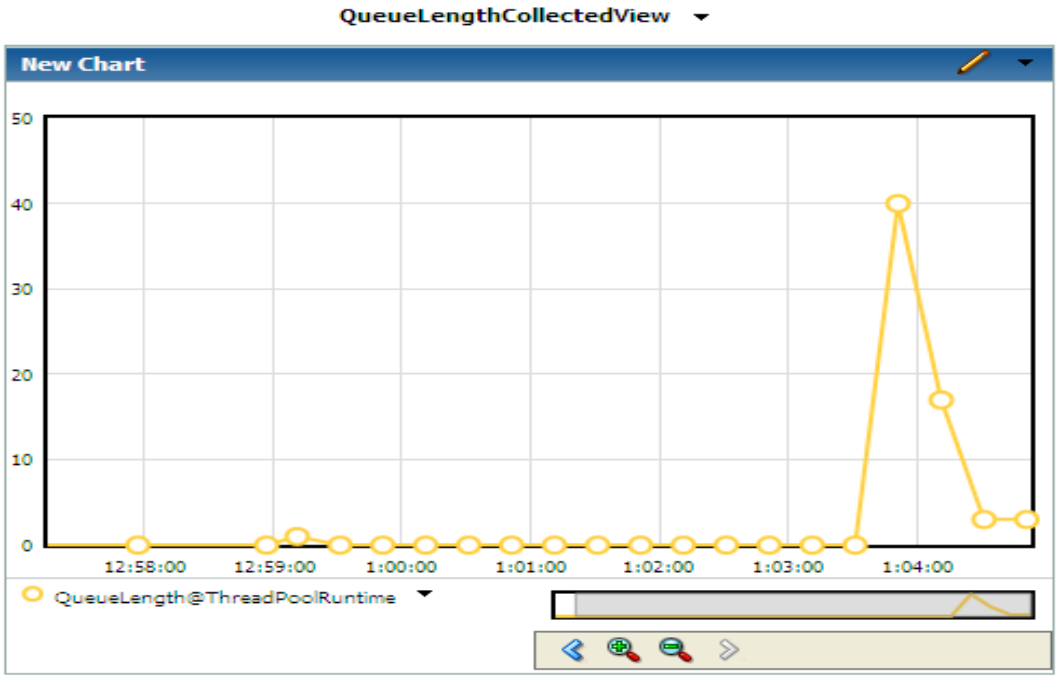


3. Type in QueueLengthCollectedView to set the View name.
4. Click on the down arrow on the chart of the display pane to the right. Click on New Chart.
5. Click on the Metric Browser tab and then enable the Collected Metrics Only checkbox. Select the PROD_Server01 option from the Servers drop-down menu and click on the OK button.
6. Click on the ThreadPool item from the Types selection list and then click on ThreadPoolRuntime from the Instances selection list. Click on the QueueLength (int) option and drag it to the QueueLengthCollectView chart on to the display panel to the right.
7. Repeat steps 5 and 6 and add the metrics from the PROD_Server02, PROD_Server03, and PROD_Server04 Managed Servers.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

A custom view was created with a new chart to display the QueueLength counter metrics of the WLDF archive stored in the database.



This chart can be used to display historical data and uses the time range stored in the database's WLDF archive. This is an important tool to understand the Managed Servers'behavioral data and past events in production environments.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

TROUBLESHOOTING WEBLOGIC SERVER

In this chapter, we will cover the following recipes:

- Changing log levels to debug
- Including the time taken field in access.log
- Enabling verbose garbage collection logging
- Taking thread dumps
- Enabling the JRockit Mission Control Flight Recorder
- Analyzing a heap dump
- Recovering the WebLogic admin password
- Recovering the data source password

Introduction:

Problems in the WebLogic environment can be exemplified as application errors, a JVM crash, when the application hangs and stops responding or when the client starts to feel the application is responding slower than usual.

All these examples are common problems in a production environment and they are usually solved by simply bouncing/restarting the affected Managed Server.

If the same problem becomes more and more frequent, a troubleshooting process must be started. Troubleshooting is the analysis process to solve a specific problem.

The purpose of all troubleshooting processes is to find the source and root cause of the problem. The source of a problem can reside either in the application code or on the host machine or the issue may only occur because of a bad WebLogic configuration or a network issue. The problem can also occur because of a WebLogic bug. A product bug must always be considered in software of this size and complexity.
A correct WebLogic analysis and diagnosis of the troubleshooting process is not an easy task.It is complex, complicated, and involves a deep knowledge of WebLogic core mechanisms and Java architecture. Incorrect diagnoses can lead to an imprecise root cause that in turn will generate wasted efforts in both development and infrastructure.

Changing log levels to debug:

The level of logged information is essential during a troubleshooting analysis. Any additional log can give a hint that will help reveal the root cause of the problem in a production environment.

This recipe shows a hypothetical scenario to diagnose transaction problems. The log level setting of the PROD_Server01 Managed Server will be changed to add more debugging information.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

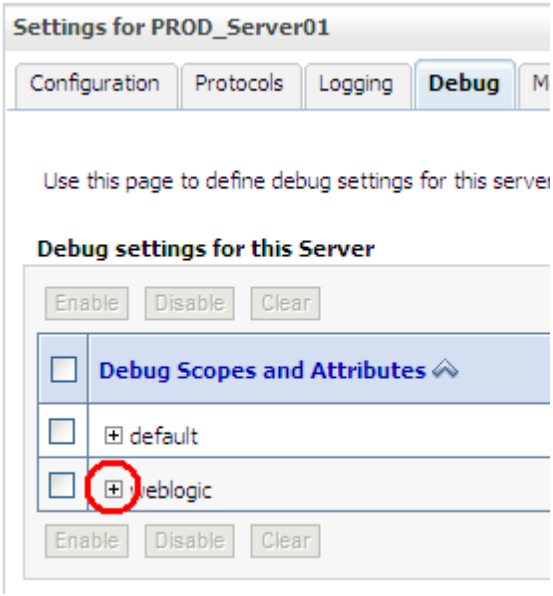
Getting ready:

The log levels will be changed using the Administration Console, so make sure the Administration Server is running.

How to do it:

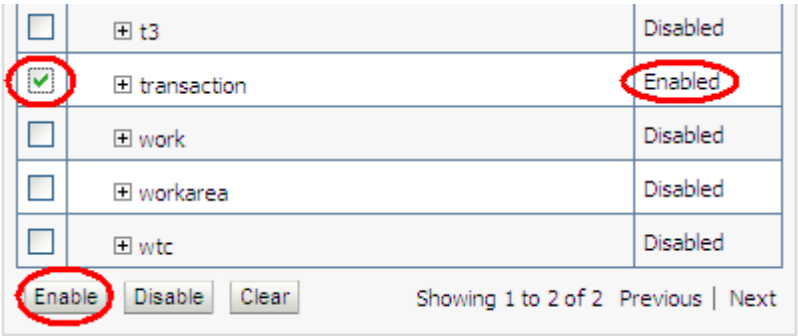
Carry out the following steps to change the log level for debugging:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree on the left, and then click on the Servers link.
4. Click on the PROD_Server01 Managed Server link and then click on the Logging tab.
5. Click on the Advanced link to open the advanced options.
6. Change the Minimum severity to log drop-down menu to Trace.
7. Change the Severity Level value for the Log file value to Trace.
8. Change the Severity Level for the Standard out value to Debug.
9. Type -1 in the stdout Stack Trace Depth field.
10. Click on the Save button.
11. Click on the Debug tab.
12. Click on the [+] sign in the weblogic scope.



13. Check the transaction checkbox and click on the Enable button:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



14. Finally, click on the Activate Changes button.

How it works:

Changing the log levels to display more information is the first common and useful task WebLogic administrators should set in production environments.

The PROD_Server01 logfiles are located by default at $DOMAIN_HOME/servers/PROD_Server01/logs/PROD_Server01.log and $DOMAIN_HOME/servers/PROD_Server01/logs/PROD_Server01.out.

This recipe raises the level of logging and enables only the transaction debug. WebLogic has several other debug options that can be enabled, such as security, protocol, JMS, and deploy among others. Enable the option that best suits your troubleshooting needs.

Including the time taken field in access.log:

The access.log file registers every HTTP request received by the WebLogic Server.It contains valuable information for the analysis in the production environments.
In this recipe, all Managed Servers of the PROD_Cluster cluster will be configured to add the time taken field to the access.log logfile. The default configuration does not include the time taken to process an HTTP request in the access.log file.

Getting ready:

The access.log file has been configured using the Administration Console, so make sure the Administration Server is running.

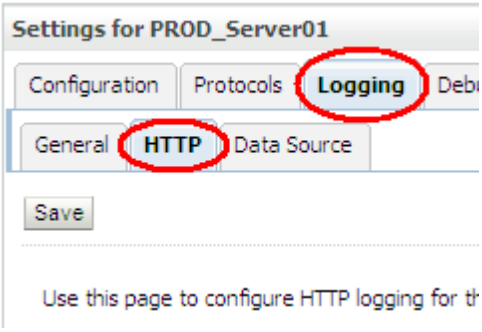
How to do it:

Carry out the following steps to add the time taken field:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree on the left and then click on Servers.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

4. Click on the PROD_Server01 link. Click on the Logging tab and then on the HTTP tab.



5. Click on the Advanced link to open the advanced options.
6. Select the Extended option from the Format drop-down menu.
7. Type c-ip date time cs-method cs-uri sc-status bytes time-taken in Extended Logging Format Fields and click on the Save button.
8. Click on the Activate Changes button to finish.
9. Repeat the preceding steps for the PROD_Server02, PROD_Server03, and PROD_Server04 Managed Servers. Restart all the Managed Servers.

How it works:

The access.log file of the PROD_Server01 Managed Server is located at $DOMAIN_HOME/servers/PROD_Server01/logs/access.log.The extended logging format adds the last field with the time taken for an HTTP request in seconds. The field is very helpful to watch how much time the HTTP requests are taking to execute.

```
10.2.8.2 2012-11-14 11:04:02 GET  /myApp/get.jsp  200 321  3.602
10.1.8.2 2012-11-14 11:04:02 POST /myApp/go.jsp   200 432  3.577
10.1.8.2 2012-11-14 11:04:02 GET  /myApp/map.jsp  200 754  3.599
10.5.8.4 2012-11-14 11:04:02 POST /myApp/put.jsp  200 100  3.602
10.6.8.6 2012-11-14 11:04:02 POST /myApp/put.jsp  200 184  3.588
10.3.8.9 2012-11-14 11:04:02 POST /myApp/put.jsp  200 5433 3.591
10.2.8.9 2012-11-14 11:04:02 GET  /myApp/home.jsp 200 4031 0.039
```

Always compare a problematic scenario to an error-free scenario when troubleshooting a WebLogic Managed Server. Compare the heap usage, count the number of requests, and watch the time taken.

Enabling verbose garbage collection logging:

Enabling garbage collection (GC) logging can provide rich information about the behavior of the JVM. Enabling the verbose GC is an easy change and will help diagnose some possible issues in the WebLogic Server.

One of the steps in the troubleshooting process is to find out if the JVM heap utilization is adequate.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

This recipe enables Oracle JRockit verbose GC logging in the WebLogic standard output log file for the PROD_Server01 Managed Server.

It also shows an example of a troubleshooting scenario and the analysis made with the verbose GC.

Getting ready:

The verbose GC has been changed using the Administration Console, so make sure the Administration Server is running.

How to do it:

To enable the PROD_Server01 Managed Server to log the GC times in the standard output log, follow these steps:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree on the left and then click on Servers.
4. Click on the PROD_Server01 link and then click on the Server Start tab.
5. Add the following to the Arguments field and click on the Save button:
   -Xverbose:memory,gc -XverboseTimeStamp
6. Click on the Activate Changes button.
7. Restart PROD_Server01.

How it works:

The -Xverbose:memory,gc argument enables the verbose GC and heap memory logging.
The -XverboseTimeStamp attribute adds the timestamp of each entry.

The verbose GC is logged to the PROD_Server01 standard output logfile PROD_Server01.out under $DOMAIN_HOME/servers/PROD_Server01/logs.

Each entry has the following format:

[memory][<date>][<collection type - YC or OC> #<collection counter>]<start>-<end>: <type> <used heap before>KB-><used heap after>KB (<heaptotal>KB), <time> s, sum of pauses <pause time> ms, longest pause<pause time> ms.

In this troubleshooting scenario, the verbose GC log has to be analyzed to find out if the heap was an issue during a period when the CPU and load of the host machine were very high.

The pattern shown in the following screenshot is an example of where the SAR tool displays a very high load in the host prod01 between 08:50 A.M. and 10:50 A.M.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

```
[weblogic@prod01 ~]$ sar -q
08:30:01 AM   runq-sz  plist-sz   ldavg-1   ldavg-5  ldavg-15
08:30:01 AM         4      3325      6.57      6.64      5.79
08:40:01 AM         8      3336      6.12      6.43      6.07
08:50:01 AM        17      3385     13.22     10.67      8.19
09:00:01 AM        11      3528     13.53     13.30     10.98
09:10:01 AM        17      3578     16.60     15.86     13.16
09:20:01 AM        22      3803     30.02     25.14     18.87
09:30:01 AM        29      4034     29.82     30.17     24.95
09:40:01 AM        18      4150     34.60     34.38     29.52
09:50:01 AM        19      4255     29.01     31.19     30.15
10:00:01 AM        53      4103     40.70     37.95     33.72
10:10:01 AM        32      4189     30.77     32.10     32.44
10:20:01 AM        37      4217     29.91     30.44     31.47
10:30:01 AM        26      4206     32.47     34.16     32.96
10:40:01 AM        26      4211     27.81     30.40     31.77
10:50:01 AM         1      4204      6.30      9.81     20.03
11:00:01 AM         3      4191      6.34      6.86     13.54
```

Checking the verbose GC log during the period reveals that the heap usage was a possible cause of the problem.

A verbose output pattern at 8:07 AM with a regular load in the host is shown as follows:

[memory][Wed Nov 14 08:07:23 2012][YC#812] 14913.268-14913.541: YC 7444470KB->6687193KB (8388608KB), 0.273 s, sum of pauses 272.307 ms,longest pause 272.307 ms.
[memory][Wed Nov 14 08:07:48 2012][YC#813] 14937.672-14938.002: YC7581758KB->6861173KB (8388608KB), 0.330 s, sum of pauses 329.867 ms,longest pause 329.867 ms.
[memory][Wed Nov 14 08:08:17 2012][YC#814] 14967.114-14967.327: YC7793098KB->7028158KB (8388608KB), 0.213 s, sum of pauses 212.877 ms,longest pause 212.877 ms.
[memory][Wed Nov 14 08:08:40 2012][YC#815] 14990.600-14990.720: YC7956848KB->7183819KB (8388608KB), 0.120 s, sum of pauses 119.871 ms,longest pause 119.871 ms.
There are six Young Collections (YC) before an Old Collection (OC) or full GC. There is aninterval of about three minutes between full GCs.
The following log snippet is the verbose output pattern during the high load at 9:45 AM. Note the difference from the first snippet. There are several full GCs in the sequence (OC) that can indicate an excessive utilization of the heap:

[memory][Wed Nov 14 09:45:40 2012][YC#5842] 20810.279-20810.310: YC 7864695KB->7858455KB (8388608KB), 0.030 s, sum of pauses 29.888 ms, longest pause 29.888 ms.
[memory][Wed Nov 14 09:45:40 2012][YC#5843] 20810.331-20810.380: YC 7867299KB->7863260KB (8388608KB), 0.049 s, sum of pauses 41.743 ms,longest pause 41.743 ms.

The heap free is at 15 percent after the first full GC at 09:45:46 AM (OC #422). The time spent is about 5.5 seconds. Just after it, another full GC is triggered at 09:45:51 AM (OC #423). Again, the heap free is about 15 percent and the time spent is 5.7 seconds, and another full GC is triggered. This pattern continues, indicating that the JVM is spending almost all the CPU time on garbage collections; so, the machine high

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

load is probably a consequence of the GC threads working in a non-stop fashion, trying to free the heap memory.

Excessive utilization of the heap can be the consequence of a peak load in the WebLogic Managed Server. Too many threads are executing at the same time, so too many objects are live in the heap. The problem may be the consequence of a heap size being too small or the concurrency being too high. Raise the heap size according to the footprint needed or limit the threads' concurrency. Adding more Managed Servers to the cluster is also an option.

The high heap usage can also be the result of a memory leak. Memory leaks are usually caused by application bugs that can lead to a continuous rise of the heap utilization until an Out of Memory (OOM) occurs.
The GC analysis can also be made using a graphical tool, such as the Monitoring Dashboard or the JRockit Mission Control. For troubleshooting purposes, the verbose GC is very precise and does not need the JVM to be up and running since the information is already written in the logfile.

Enabling the verbose GC with jrcmd :

The verbose GC can be enabled without restarting the JVM using the jrcmd command-line tool.

1. Find the <PID> value of the PROD_Server01 Managed Server process:
   [wls@prod01]$ ps aux | grep PROD_Server01 | grep -v grep | awk'{print $2} '
   <PID>
2. Issue a jrcmd command using <PID> as an argument:
   [wls@prod01]$ /oracle/jvm/bin/jrcmd <PID> verbosity set=gc,memory

Taking thread dumps:

The thread dump is a well-known feature of the Java Virtual Machine. It is a snapshot of the JVM that helps the WebLogic administrator analyze which Java method the Java threads were executing in at the exact moment the snapshot was taken. As in the rest of the book, the JVM used in this recipe is Oracle JRockit JVM.

In a hypothetic scenario, the WebLogic Administrator received several e-mails from the configured WLDF alerts from the previous chapter, indicating that the QueueLength value is growing up in the PROD_Server01 Managed Server. The requests are probably piling up.

In this recipe, thread dumps of PROD_Server01 will be taken to analyze the problematic scenario.

Getting ready:

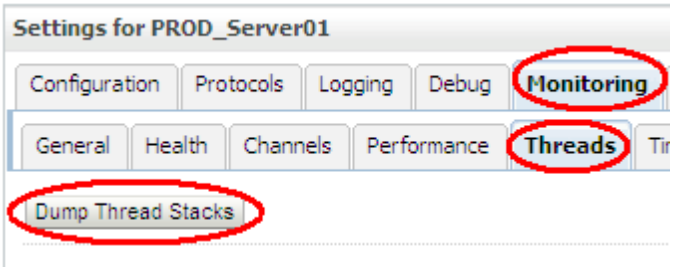
The thread dump can be taken from the Administration Console from the shell using the jrcmd command-line tool or from the JRockit Mission Control.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How to do it:

Get the thread dump from the Administration Console:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Expand the Environment tree on the left and then click on Servers.
3. Click on the PROD_Server01 link; click on the Monitoring tab and then click on the Threads tab.
4. Click on the Dump Thread Stacks button.



The thread dump will be displayed on the screen.

The thread dump can also be taken from the shell by using the jrcmd command-line tool or by issuing the command kill -3 to <PID>.

1. Find the <PID> value of the PROD_Server01 Managed Server process:
   [wls@prod01]$ ps aux | grep PROD_Server01 | grep -v grep | awk '{print $2} '<PID>
2. Issue a jrcmd command using <PID> as an argument:
   [wls@prod01]$ /oracle/jvm/bin/jrcmd <PID> print_threads > some_log_file
3. Alternatively, you can also run the kill -3 command using <PID> as an argument:
   [wls@prod01]$ kill -3 <PID>

How it works:

The jrcmd command-line tool is redirecting the output to some_log_file defined in the command line, so the thread dump will be written there. The thread dump taken by using the kill command is written to the standard output logfile of the PROD_Server01 Managed Server.
The thread dump is one of the most guaranteed methods of troubleshooting a Managed Server. Normally, you need to observe only the threads from the self-tuning thread pool. The thread stack trace from the self-tuning thread pool can have the [ACTIVE], [STANDBY], and [STUCK] flags, and it has the following pattern:
"[ACTIVE] ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)'" id=39 idx=0x80 tid=4952 prio=5 alive, waiting, native_blocked, daemon-- Waiting for notification on: weblogic/work/ExecuteThread@0xf06a43b0 [...] at weblogic/work/ExecuteThread.run()

Take some thread dumps in a sequence with a short interval between them. There is no specific formula for how many thread dumps to take or which interval to use. Sometimes, only one thread dump is sufficient.
In this scenario, the sequence of thread dumps reveals that almost all threads are stopped at the following stack:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

"[ACTIVE] ExecuteThread: '10' for queue: 'weblogic.kernel.Default (self-tuning)'" id=166 idx=0x2a4 tid=25647 prio=1 alive, in native,daemon
at jrockit/net/SocketNativeIO.readBytesPinned()
[...]
at oracle/jdbc/driver/OraclePreparedStatement.execute()
[...]
at oracle/jdbc/driver/OraclePreparedStatementWrapper.execute()
at weblogic/jdbc/wrapper/PreparedStatement.execute()
at com/myapp/db/DBUtil.executeQuery()
[...]
at weblogic/work/ExecuteThread.execute()
at weblogic/work/ExecuteThread.run()

The important Java methods to observe are com.myapp.db.DBUtil.executeQuery() and oracle.jdbc.driver.OraclePreparedStatement.execute(). The application is probably executing some function in the database, and all the threads are waiting for the response, piling up the requests. In this specific case, it is possible that the problem is coming from a slow database or from a large query that is not using any indexes, or even from some sort of database lock. The thread dump can point to the right direction and, in this scenario, further investigation will be needed in the application code and the database.

Enabling the JRockit Mission Control Flight Recorder:

Mission Control is a monitoring and troubleshooting application provided with Oracle JRockit.Besides the Management Console covered in *Chapter 5*, *Monitoring WebLogic Server 12c*, Oracle JRockit also provides an important feature called the Flight Recorder. The Flight Recorder registers the JVM's metrics and events, garbage collection behavior, thread contention, locking, and other information, according to the configuration template you use.The default template has a very low overhead and can be used in production environments.This recipe will enable the Flight Recorder to store a flight recorder file (.jfr extension) for the Managed Servers PROD_Server01, PROD_Server02, PROD_Server03, and PROD_Server04 with the default template configuration. The recording will be configured to store the last 24 hours of events or the maximum of a 100 MB file.

Getting ready:

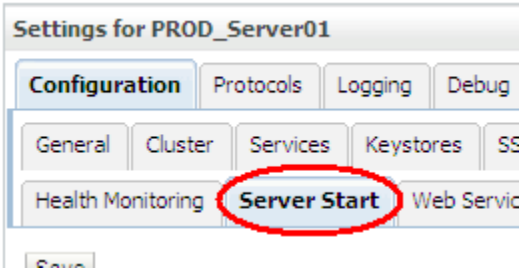
There is no need to start Oracle JRockit Mission Control to configure the Flight Recorder.It is enabled by changing parameters using the Administration Console. So make sure the Administration Server is running.

How to do it :

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

To enable the Managed Servers to record the Flight Recorder's events, follow these steps:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree on the left and then click on Servers.
4. Click on the PROD_Server01 link and then click on the Server Start tab.



5. Add the following line of code to the Arguments field and click on the Save button: -XX:FlightRecorderOptions=defaultrecording=true,disk=true, repository=./jfr/PROD_Server01,maxage=1440m,size=100m
6. Click on the Activate Changes button.
7. Restart PROD_Server01.
8. Repeat the previous steps for the Managed Servers PROD_Server02, PROD_ Server03, and PROD_Server04, and define the corresponding path for the repository=./jfr/PROD_ServerXX repository parameter of each server.

How it works:

repository=./jfr/PROD_ServerXX parameter sets the file location of therecordings. In this case, it will save the recording to the $DOMAIN_HOME/jfr/PROD_ServerXX directory. The directory will be created automatically if it does not exist.
The Flight Recorder is enabled and configured to store the last 24 hours of events, or until the file size reaches 100 MB. The objective is to record the previous events like an airplane's black box.
There are many settings you can change for the Flight Recorder.
The parameters can be altered to save the events during the entire life cycle of the JVM. Change the maxage and size parameters to 0 as shown in the following line of code:

-XX:FlightRecorderOptions=defaultrecording=true,disk=true,repository=./jfr/PROD_Server0X,maxage=0, size=0

For every new recording, a new directory with the pattern $DOMAIN_HOME/jrf/<SERVER_NAME>/YYYY_MM_DD_HH_MM_SS_PID is created. YYYY stands for the year, MM stands forthe month, DD for the day, HH for the hour, SS for the second, and PID is the process ID of theWebLogic Managed Server.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

The file has a similar pattern YYYY_MM_DD_HH_MM_SS_PID.jfr and it must be opened in the JRockit Mission Control to be analyzed. The content of the Mission Control Management Console is mostly the same, but it can include more information depending on the configuration template used to save the .jfr file.

Analyzing a heap dump:

A heap dump is another important feature of the Java Virtual Machine. It contains a memory dump of all the current live Java objects of the heap.

The heap dump is useful in problematic situations, such as in a memory leak condition in a WebLogic Server where the heap usage grows until the JVM crashes by Out of Memory (OOM),or when the heap utilization is so high that the Managed Server hangs and stops responding.Both scenarios can use a heap dump to discover the offender objects in the heap.
This recipe will display how to take a heap dump from a Managed Server in the HPROF format.HPROF is a binary file that stores the heap and CPU profiles of the JVM. For demonstration purposes, the heap dump was taken from a JVM configured to a small heap with a maximum size of 256 MB (-Xmx256mb).

Getting ready;

The heap dump can be taken from the shell using the jrcmd command-line tool or from the Mission Control.
The heap dump will be analyzed with the Eclipse Memory Analyzer (MAT), so download and install it from http://www.eclipse.com/mat.

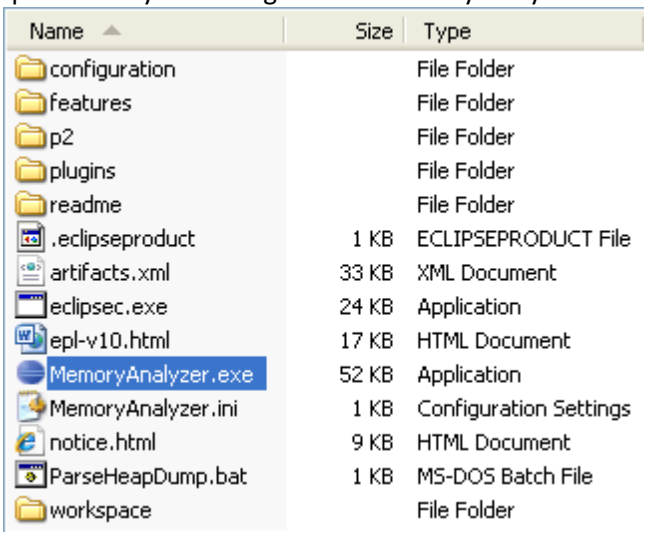
How to do it:

1. Log in to the shell and use the JRCMD tool.
2. Find the <PID> value of the PROD_Server01 Managed Server process:
   [wls@prod01]$ ps aux | grep PROD_Server01 | grep -v grep | awk '{print $2} '
   <PID>
3. Issue a jrcmd command using <PID> as an argument:
   [wls@prod01]$ /oracle/jvm/bin/jrcmd <PID> hprofdump
   <PID>:
   Wrote dump to heapdump_<date>.hprof

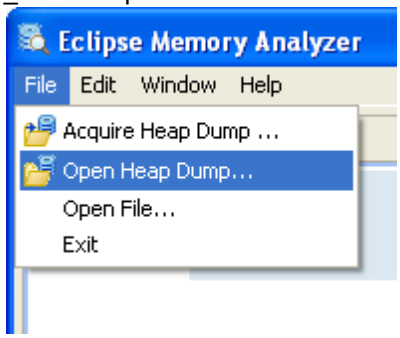
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Analyze the heap dump with MAT:

1. Open MAT by executing the file MemoryAnalyzer.exe.

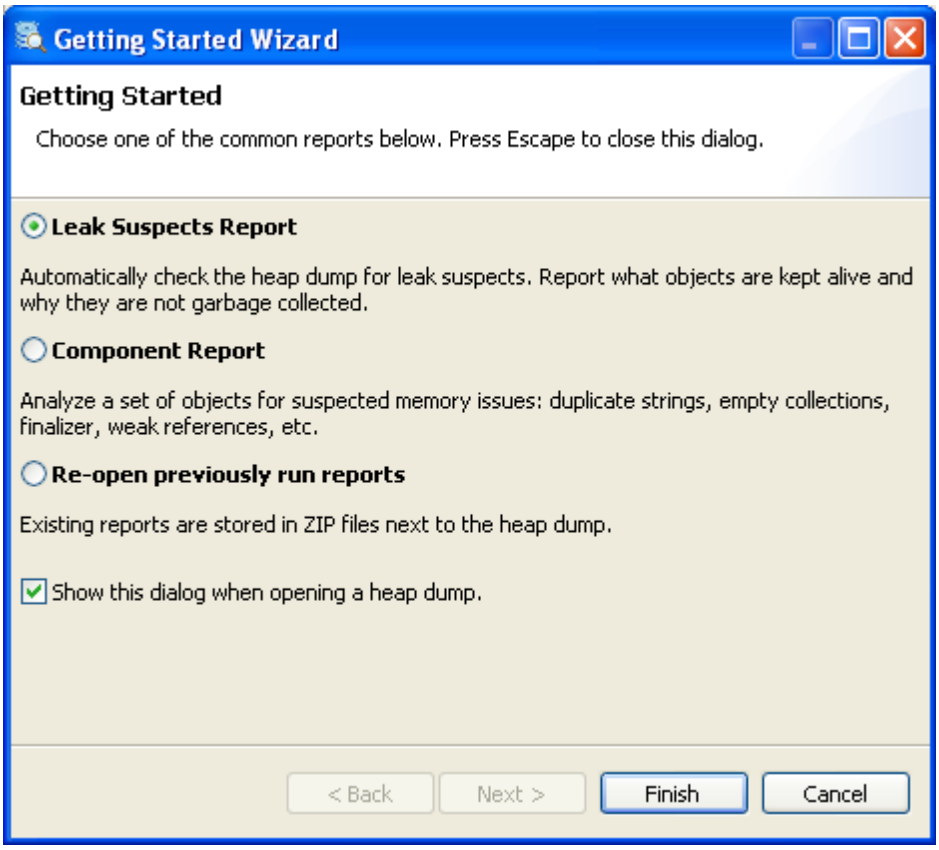


2. Open the File menu and select the Open Heap Dump menu option. Choose the heap dump file heapdump_<date>.hprof.

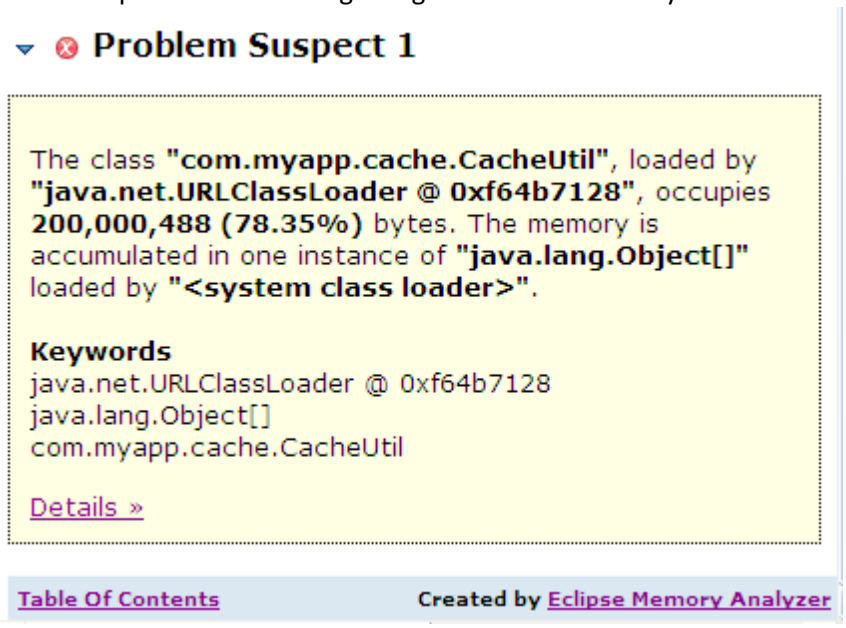


3. Select the Leak Suspects Report option from the Getting Started Wizard screen and click on the Finish button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



4. Watch out for the suspects that are using a large amount of memory.



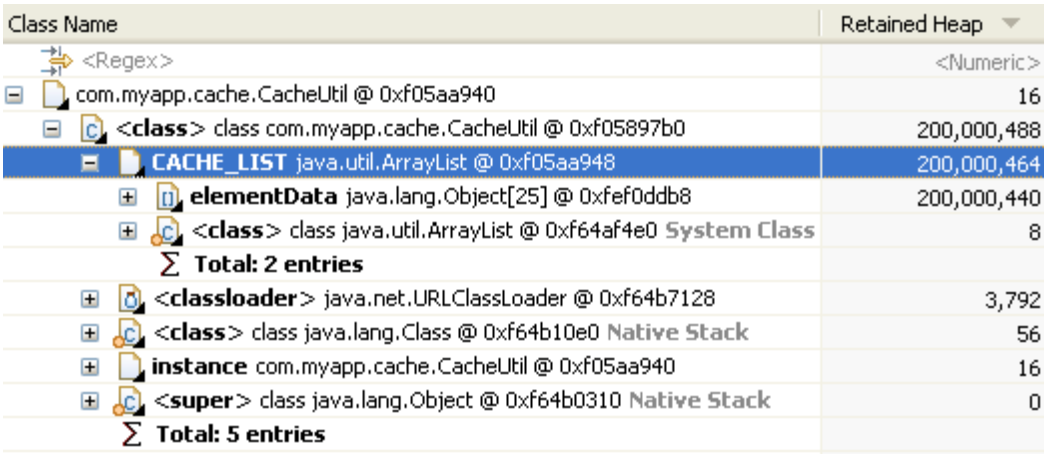
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

The jrcmd command-line tool issues a command to the JVM to generate the heap dump.The heap dump file is written to the $DOMAIN_HOME root directory with a filename of thepattern, heapdump_<date>.hprof.

The JVM runs a full GC before saving the heap dump, so the size of the file is the size of the live Java objects. If a heap is reaching 3 GB of utilization after a full GC, the size of the file will be 3 GB.

In this recipe, the heap dump was opened and analyzed by using the Eclipse Memory Analyzer application. The analysis indicated the com.myapp.cache.CacheUtil class is an offender, occupying about 80 percent of the total heap size. The following screenshot displays an ArrayList named CACHE_LIST; it is the variable of the com.myapp.cache.CacheUtil class and consumes 200 MB of the 256 MB configured for the heap.



The heap dump analysis indicates which part of the Java code may be the offender.
In this case, the Java code must be revised to fix the problem or the heap size must be adjusted to fit the number of objects needed in this array. Use the solution that meets the application requirements.

Note :

There are several profiling tools to analyze a heap dump. The Eclipse Memory Analyzer—MAT (http://www.eclipse.org/mat) and the YourKit Java Profiler (http://www.yourkit.com) are excellent known options.

The heap dump can be configured to be generated automatically in certain conditions:

Generating the heap dump automatically on OOM conditions

The heap dump can be generated automatically when an out of memory condition is reached.

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

3. Expand the Environment tree on the left and then click on Servers.
4. Click on the PROD_Server01 link then click on the Server Start tab.
5. Add the following to the Arguments field and click on the Save button:
   -XX:+HeapDumpOnOutOfMemoryError
6. Click on the Activate Changes button.
7. Restart PROD_Server01.
8. Repeat the previous steps for the Managed Servers PROD_Server02, PROD_ Server03, and PROD_Server04.
9. The heap dump file that is generated follows the same pattern when generated from the jrcmd tool.

Recovering the WebLogic admin password:

The WebLogic Administrator username and password are used to start up the WebLogic Server instances. They are stored encrypted in the boot.properties file.
This recipe will provide the steps to recover the username and password from the boot. properties file of the PROD_DOMAIN domain.

Getting ready:

The recovery will use WLST to decrypt the boot.properties file.

How to do it:

Carry out the following steps to recover the WebLogic Admin password:

1. Log in as the wls user to shell and set the domain environment variables for the domain you want to recover:
   [wls@prod01]$ cd $DOMAIN_HOME/bin
2. [wls@prod01]$ . ./setDomainEnv.sh
   Start WLST:
   [wls@prod01]$ $WL_HOME/common/bin/wlst.sh
3. Run the following WLST commands to display the username and password:

```
from weblogic.security.internal import BootProperties
BootProperties.load("/oracle/Middleware/user_projects/domains/
PROD_DOMAIN/servers/PROD_AdminServer/security/boot.properties",
false)
prop = BootProperties.getBootProperties()
print "username: " + prop.getOneClient()
print "password: " + prop.getTwoClient()
```

The username and password will be displayed on the screen.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

The script reads the boot.properties file, decrypts it, and displays the username and password provided.

The script points to the boot.properties file located in the security folder of the PROD_ AdminServer. You can point to any security folder that contains the boot.propertiesfile of the other Managed Servers.

It is important to set the domain environments first, otherwise the script will not be able to find the SerializedSystemIni.dat file, which is the seed used by the domain to encrypt and decrypt.

Recovering the data source password:

The same way that the WebLogic Administrator password is recoverable, the data sourcepassword can be retrieved as well.

In this recipe, the ds-nonXA data source with the JNDI name jdbc/non-XA will be used to retrieve the password.

Getting ready:

The encrypted password must be retrieved from the JDBC configuration files in the $DOMAIN_HOME/config/jdbc directory. To decrypt the password, use WLST.

How to do it:

Carry out the following steps to recover the data source password:

1. Log in as a wls user to shell and open the $DOMAIN_HOME/config/config.xml file to get the JDBC configuration filename.
   [wls@prod01]$ cd $DOMAIN_HOME/config
   [wls@prod01]$ vi config.xml
2. Locate the <jdbc-system-resource> tag of the ds-nonXA data source and get the descriptor filename.
   <jdbc-system-resource>
   <name>ds-nonXA</name>
   <target></target>
   <descriptor-file-name>jdbc/ds-nonXA-jdbc.xml</descriptorfile-name>
   </jdbc-system-resource>
3. Open the JDBC file:
   [wls@prod01]$ vi jdbc/ds-nonXA-jdbc.xml

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

4. Locate the <password-encrypted> tag and copy the password.
   <password-encrypted>{AES}PASSWORD_ENCRYPTED</passwordencrypted>
5. Start WLST using the following command:
   [wls@prod01]$ $WL_HOME/common/bin/wlst.sh
6. Set the copied password to the passwd variable, set the full path of the $DOMAIN_ HOME/security in the secPath variable, and run the following WLST commands to display the password:
   from weblogic.security.internal import *
   from weblogic.security.internal.encryption import *
   passwd = "{AES}PASSWORD_ENCRYPTED"
   secPath = "/oracle/Middleware/user_projects/domains/PROD_DOMAIN/ security"
   encService = SerializedSystemIni.getEncryptionService(secPath)
   coeService = ClearOrEncryptedService(encService)
   print "password: " + coeService.decrypt(passwd)
7. The password will be displayed on the screen:
   wls:/offline> password: dbpwd

How it works:

This recipe displayed a simple procedure to get the encrypted password stored in the JDBC configuration file and use it in the WLST script file.

There is no need to set the domain environment variables this time since the script receives the full path to the SerializedSystemIni.dat file.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

STABILITY AND PERFORMANCE:

In this chapter we will cover the following recipes:

- Limiting the log disk usage
- Rotating the STDOUT logfile
- Turning off the domain logging
- Enabling Linux HugePages
- Configuring the transaction (JTA) timeouts
- Choosing the JRockit garbage collection mode
- Tuning thread concurrency with default work managers
- Tuning the application thread concurrency with custom work managers
- Limiting the JMS Queue consumers

Introduction:

This chapter is about WebLogic Server stability and performance, but most of the recipes are focused on stability.

For the WebLogic Administrator, stability should come first in the priority list of objectives. It is improbable that an unstable system achieves sufficient performance.

This chapter provides simple but effective small tunings, targeting a stable and predictable production environment.

Limiting the log disk usage:

WebLogic Server 12*c* fixes a file size limit of 5 MB for the domain, server, and HTTP logs. It also sets 100 as the maximum number of rotated files for the domain and server logs. The HTTP access.log file does not have a default limit of rotated files. This is the default configuration for WebLogic Server 12*c* domains running in the production mode.

In this recipe, the logging subsystem will be configured, so that all the PROD_DOMAIN domain logfile usage never surpasses a known disk size usage.

Getting ready:

The logging subsystem is configured by using the Administration Console, so make sure the Administration Server is running.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How to do it:

Carry out the following steps to limit the log disk usage:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree to the left and then click on Servers.
4. Click on the PROD_Server01 link. Click on the Logging tab and then click on the General tab.
5. Type 50000 in the Rotation file size text field.
6. Make sure that the Limit number of retained files checkbox is checked.
7. Type 20 in the Files to retain text field.
8. Check the Rotate log file on startup checkbox.
9. Click on the Save button and then click on the Advanced link to open theadvanced options.
10. Check the Redirect stdout logging enabled checkbox.
11. Check the Redirect stderr logging enabled checkbox.
12. Change the Severity Level of the Standard out drop-down menu to Off.
13. Click on the Save button.
14. Click on the HTTP tab.
15. Type 50000 in the Rotation file size text field.
16. Make sure that the Limit number of retained files checkbox is checked.
17. Type 20 in the Files to retain text field.
18. Check the Rotate log file on startup checkbox.
19. Click on the Save button.
20. Repeat the previous steps for the PROD_Server02, PROD_Server03 and PROD_ Server04 Managed Servers and the PROD_AdminServer Administration Server.
21. Click on the PROD_DOMAIN link on the left-hand side navigation tree and then click on the Logging tab.
22. Type 50000 in the Rotation file size text field.
23. Make sure that the Limit number of retained files checkbox is checked.
24. Type 20 in the Files to retain text field.
25. Check the Rotate log file on startup checkbox and click on the Save button.
26. Click on the Activate Changes button to finish.
27. Restart the entire PROD_DOMAIN domain.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

In production environments, predictability must be a priority when setting up WebLogic.
This configuration brings a fixed known value to the logging disk usage of the entire domain.

WebLogic Instance/DOMAIN	Server log	HTTP log
PROD_AdminServer	10 MB * 50 MB = 500MB	10 MB * 50 MB = 500 MB
PROD_Server01	500MB	500MB
PROD_Server01	500MB	500MB
PROD_Server01	500MB	500MB
PROD_Server01	500MB	500MB
PROD_DOMAIN log	500MB	
Total	6 MB * 500 MB = 3GB	5 MB * 500 MB = 2.5 GB

WebLogic Server does not provide an out of the box configuration to rotate the standard output log ($DOMAIN_HOME/servers/PROD_ServerXX/PROD_ServerXX.out). As a workaround, STDOUT and STDERR from the PROD_ServerXX.out file were redirected to the server log and disabled. This configuration maintains the information of the PROD_ServerXX.out file in the PROD_ServerXX.log file.

Rotating the STDOUT log will be explained further on this chapter.

Rotating the STDOUT logfile:

Oracle WebLogic Server 12*c* does not provide an out of the box configuration to rotate the STDOUT and STDERR logfiles in runtime (.out file). In the previous recipe, as a workaround, it was redirected to the server logfile and disabled.

Fortunately in Linux, the included logrotate command-line tool can do the job.

In this recipe, the logrotate will be configured to run hourly and rotate all .out files from the PROD_DOMAIN WebLogic instances larger than 50 MB. It will rotate to the limit of 20 files, repeating the configuration of the server logfiles in the previous recipe.

Getting ready:

The log rotate settings must be configured in the machine shell. To do this log in to the prod01 machine.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How to do it:

Carry out the following steps to rotate the .out file:

1. Log in as a wls user to shell and create a new file wls-stdout-logrotate.conf and save it to the $DOMAIN_HOME folder:

```
[wls@prod01]$ cd /oracle/Middleware/user_projects/domains/PROD_
DOMAIN
[wls@prod01]$ vi wls-stdout-logrotate.conf
```

2. Add the following lines to the file:

```
# WebLogic STDOUT logrotate for PROD_DOMAIN
/oracle/Middleware/user_projects/domains/PROD_DOMAIN/servers/
PROD_*/logs/*.out {
rotate 20
size 50M
copytruncate
nocompress
missingok
nodateext
noolddir
create 0640 wls wls
    }
```

3. Type :ws! to save the file and exit.
4. Change to root user and create the wls-stdout-logrotate file to add new entry to the hourly crontab:
   [root@prod01]# vi /etc/cron.hourly/wls-stdout-logrotate

5. Add the following lines to the script:

```
    #!/bin/sh
   sudo /usr/sbin/logrotate /oracle/Middleware/user_projects/domains/PROD_DOMAIN/wls-
stdout-logrotate.conf
    RETCODE=$?

    if [ $RETCODE != 0 ]; then

    /usr/bin/logger -t logrotate "WLS STDOUT logrotate finished
    abnormally with code [$RETCODE]"
    fi
    exit 0
```

6. Type :ws! to save the file and exit.
7. Change the file permissions:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

[root@prod01]# chmod 755 /etc/cron.hourly/wls-stdout-logrotate

8. Repeat the previous steps for all machines in the PROD_DOMAIN.

How it works:

The logrotate settings will be read from the wls-stdout-logrotate.conf configuration file. The file was saved in the root directory of the WebLogic domain, but it can be saved anywhere.

A crontab job was added to run hourly in the form of a script. This configuration must be executed by the root user.

When the crontab job runs, it checks for all .out files that matches the path $OMAIN_HOME/servers/PROD_*/logs/*.out. If the file size is over 50 MB, the job will copy the file content to a new one (PROD_Server01.out.1) and then truncate the original file.

The copytruncate parameter guarantees that the rotation works properly since WebLogic keeps the stream to the .out file open.

Turning off domain logging:

The WebLogic Domain's log concentrates all the WebLogic Managed Server loggings in one single log managed by the Administration Server.

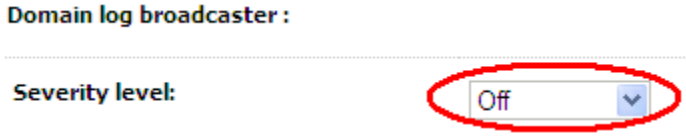
The domain log can become a bottleneck and affect the performance in certain scenarios with a very large domain with several Managed Servers.

In this recipe, the log will be turned off for the PROD_DOMAIN domain.

Getting ready:

Carry out the following steps to turn off the domain logging:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the PROD_DOMAIN link on the navigation tree to the left and then click on the Logging tab.
4. Change Severity level of the Domain log broadcaster drop-down menu to Off, as shown in the following screenshot:



5. Click on the Save button.
6. Repeat the preceding steps for all Managed Servers in the PROD_DOMAIN domain.
7. Click on the Activate Changes button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

Disabling the domain log can lighten the communication between the Managed Servers and the Administration Server.

It is a recommended setting in very busy WebLogic environments since the information logged already exists on each of the Managed Servers.

Enabling Linux Huge Pages:

Enabling Linux HugePages can improve the performance of the Oracle JRockit JVM and have several advantages over the default page size of 4 KB in Linux.

In this recipe, the HugePages will be enabled on all machines in the PROD_DOMAIN domain. A new JVM parameter will be added to the Managed Servers so the JVM makes use of the HugePages.

prod01 hosts the PROD_Server01 and PROD_Server02 Managed Servers. Since each instance is configured with a heap size of 2 GB, reserving 4 GB for HugePages in prod01 should be enough.

Getting ready:

Before configuring the Linux HugePages, shut down all WebLogic Servers in the domain.Also stop the Administration Console instance and the Node Manager.

How to do it:

1. Log in as root user to the shell and execute the following commands to check the HugePages configuration:
   [root@prod01]$ cat /proc/meminfo | grep Huge
   HugePages_Total: 0
   HugePages_Free: 0
   HugePages_Rsvd: 0
   Hugepagesize: 2048 kB
2. Create a mount point of type hugetlbfs:
   [root@prod01]$ mkdir /mnt/hugepages
   [root@prod01]$ chmod -R 777 /mnt/hugepages
3. Add the following mount point to the /etc/fstab:
   [root@prod01]$ mount -t hugetlbfs nodev /mnt/hugepages
4. Add the following line:
   hugetlbfs /mnt/hugepages hugetlbfs rw,mode=0777 0 0

5. Enter :ws! to save the file and exit.
6. Edit the sysctl.conf file under /etc/:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

[root@prod01]$ vi /etc/sysctl.conf

7. Reserve 2048 HugePages by adding the following line:
   vm.nr_hugepages = 2048
8. Enter :ws! to save the file and exit.
9. Execute the following command to make the changes effective:
   [root@prod01]# sysctl -p
10. Verify the change:

    [root@prod01]$ cat /proc/meminfo | grep Huge
    HugePages_Total: 2048
    HugePages_Free: 2048
    HugePages_Rsvd: 0
    Hugepagesize: 2048 kB

11. Restart the host if the HugePages could not be allocated.
12. Repeat the preceding steps on all machines in the PROD_DOMAIN.

To enable the Managed Servers to make use of the Huge Pages, follow the ensuing steps:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree to the left and then click on Servers.
4. Click on the PROD_Server01 link and then click on the Server Start tab.
5. Add the following to the Arguments field and click on the Save button:
   -XX:+UseLargePagesForHeap
6. Click on the Activate Changes button.
7. Restart PROD_Server01.
8. Repeat the preceding steps for PROD_Server02, PROD_Server03, and PROD_Server04.

How it works:

The first thing to do is to estimate how much memory should be reserved as HugePages.Sum all heap sizes of the JVMs in the machine. In our case, we have 2 JVMs with 2 GB each so a total of 4 GB should be reserved in HugePages.

To calculate how many pages, get the memory needed and divide by the page size. The formula is *sum of JVM heaps / page size*.

The memory needed is 4 GB (4096 MB) and the page size is 2 MB (Hugepagesize is 2048 KB). The number of HugePages is *4096/2 = 2048*.

This 4 GB of reserved HugePages will be pre-allocated and cannot be used by other applications, even when the JVMs are not running, so be sure to do the sizing properly.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

If you have the -Xverbose:gc,memory enabled, you can check if the JVM is using the HugePages properly in the STDOUT logfile.

[INFO][memory] Using 2MB pages for Java heap.

With the PROD_Server01 started, checking the /proc/meminfo should also reveal that HugePages are in use.

```
[wls@prod01]$ cat /proc/meminfo | grep Huge
HugePages_Total: 2048
HugePages_Free: 1024
HugePages_Rsvd: 0
Hugepagesize: 2048 kB
```

Using Huge Pages provides some significant advantages:

- Because of the larger page size, the page table will be smaller in size. With HugePages, a 10 GB heap size should use only 5120 page entries despite the 2621440 page entries present when using the default 4 KB page size. This minimizes the CPU cost and the page table memory usage.
- A performance boost in memory operations because the TLB cache works more efficiently, with more cache hits.
- The memory reserved for HugePages will never swap to disk.
- It forces a more controlled memory usage of the heap sizes

.

Configuring the transaction (JTA) timeouts:

In production environments, slowness in a legacy or external system (a database or webservice for example) can lead to a scenario where all WebLogic threads and resources become busy waiting for response. The slowness can pile up the requests, generating a hang scenario in WebLogic.

In this recipe, the timeouts will be configured for the PROD_DOMAIN, including the domain JTA timeout and a hypothetic XA resource, such as the XA data source ds-XA, that points to an Oracle Database. The recipe will use a timeout of 600 seconds as base.

Getting ready:

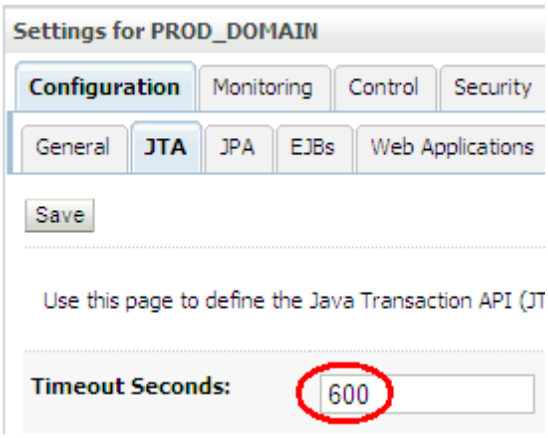
The timeouts will be configured by using the Administration Console, so make sure the Administration Server is running.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

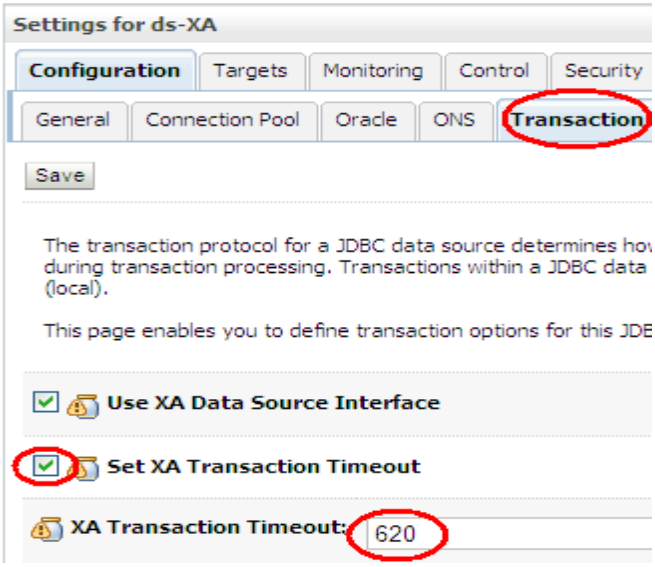
How to do it:

Carry out the following steps to configure the timeouts:

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the PROD_DOMAIN link on the left-hand side navigation menu, and then click on the JTA tab to open the JTA page under Domain | Configuration.
4. Type 600 in the Timeout seconds text field (as shown in the following screenshot) and click on the Save button.



5. Expand the Services tree to the left and then click on Data Sources.
6. Click on the ds-XA data source then the Transaction tab to navigate to Configuration | Transaction.
7. Check the Set XA Transaction Timeout checkbox and enter 620 in the XA Transaction Timeout text field, as shown in the following screenshot:



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

8. Click on the Save button.
9. Click on the Activate Changes button to finish.

How it works:

The timeout settings are important configurations to prevent a hang scenario, freeing WebLogic resources that are busy for a longer time than expected. The inverse situation can also occur where a request can naturally take a long time to process and a timeout error is thrown unnecessarily.

The WebLogic applications usually make use of the **Java Transaction API (JTA)** standards to control the transaction processes. It allows multiple resources (XA) to participate in the same transaction. The timeout configurations must be set correctly; otherwise, the timeout exception will not be properly handled by the application when invoked.

As a rule of thumb, the **Transaction Manager (TM)** should have a smaller timeout value than the XA resources it calls. When the timeout is reached for the TM, the TM should properly handle all resources, rolling back all resource transactions. If a resource times out by itself, the TM probably won't handle the error properly.

In this recipe, the JTA is configured with 600 and the XA data source is configured with 620. In this case, the parameter DISTRIBUTED_LOCK_TIMEOUT of the Oracle Database should also be set with a higher value, such as 1000 seconds.

Choosing the JRockit garbage collection mode:

Some WebLogic Administrators consider JVM tuning as the top tuning recommendation for WebLogic. But nowadays, with the newer JVM releases, just setting -Xms and –Xmx arguments and leaving the GC with default throughput GC mode, should be fine for most of the WebLogic applications. Since R28, the Oracle JRockit behaves very well out of the box and, in the majority of the cases, more advanced JVM tunings are not necessary.

There are some situations where a specific WebLogic application could not wait for a full GC that lasts more than a few seconds. In this recipe, the GC mode will be changed to the pausetime mode for this particular application.

Getting ready:

A JVM startup argument will be added, by using the Administration Console, so make sure the Administration Server is running.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

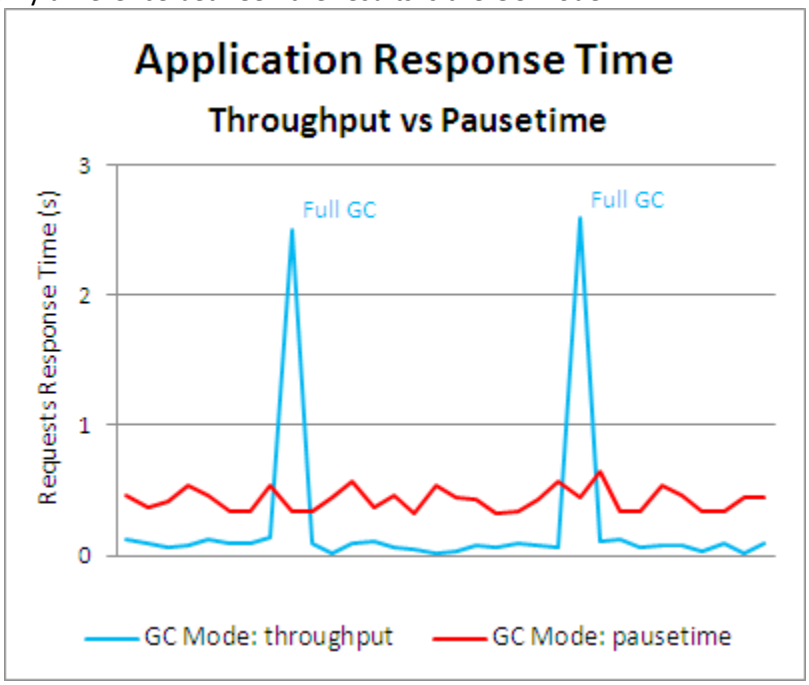
How to do it:

To change the JVM garbage collection mode:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree to the left and then click on Servers.
4. Click on the PROD_Server01 link and then click on the Server Start tab.
5. Add the following to the Arguments field and click on the Save button:
   -Xgc:pausetime
6. Click on the Activate Changes button.
7. Restart PROD_Server01.
8. Repeat the preceding steps for Managed Servers PROD_Server02, PROD_Server03, and PROD_Server04.

How it works:

The following graph displays the results of two different tests with the response times of the exactly same application. The only difference between the results is the GC mode:



The blue line shows the default throughput mode. It has very low application response times for most of the test (average of 100 ms), but returns a higher response time when the GC is triggered. Note that the graph is from the application response time and not from the GC pause time. The throughput mode tries to leave the JVM running without any interference. Then, when needed, execute the GC as fast as

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

possible by running it in parallel and using all the CPUs available; on the start up, the JVM checks how many CPU cores the machine has and starts the same number of GC threads. Although the JVM lets the application run at full speed when the GC is not needed, it will "stop the world" when full GC is invoked.

The red line shows the pausetime GC mode behavior. The application is not sensitive to full GC as the throughput mode, keeping the time consistent during all times. On the other hand, the responses get a much higher average time (400 ms) in comparison to the blue line. Since the JVM focuses on keeping the GC pausetime low, it keeps the GC running concurrent to the application, sharing CPU cycles and increasing the response time.

Tuning thread concurrency with the default work manager:

In production environments, keeping the computer resources such as memory, processes, connections, and threads under control is essential to maintain a stable and predictable system.

For a WebLogic Administrator focused on stability, the top tuning recommendation for WebLogic should be controlling the thread concurrency and resources usage.

In this recipe, the default work manager will be created to override the default settings with a new maximum thread constraint named defaultMaxThreads and configured with 50,targeting it to the cluster, PROD_Cluster.

Getting ready:

The work manager will be created in the Administration Console.

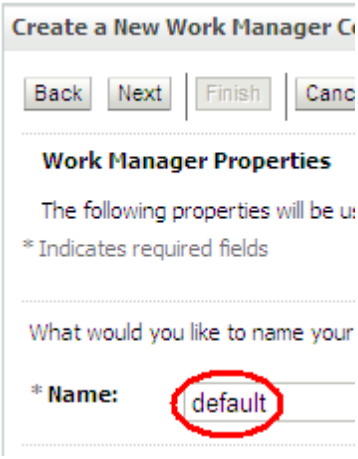
How to do it:

Create the defaultMaxThreads maximum threads constraint and the default work manager in the Administration Console by following the ensuing steps:

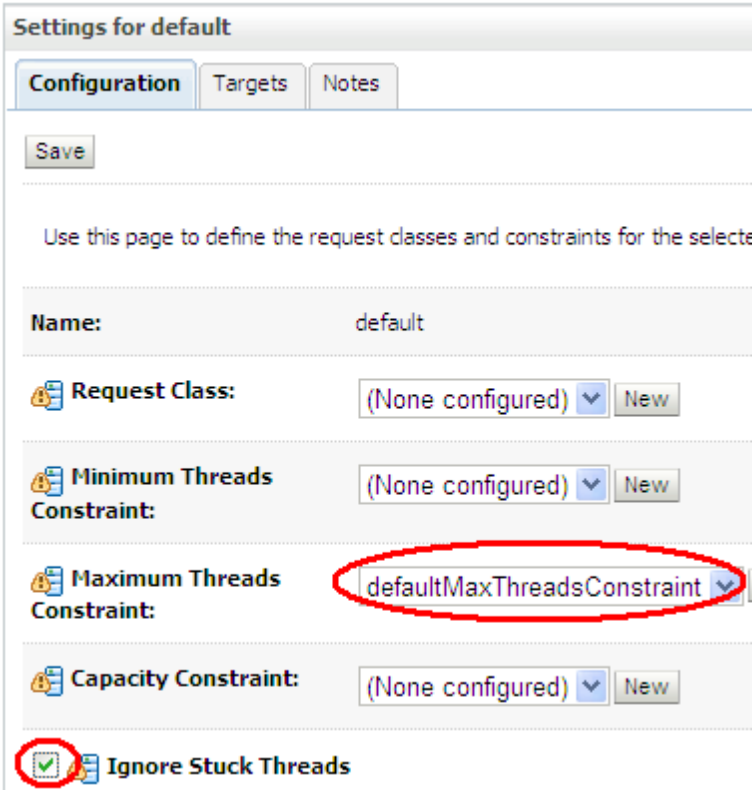
1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree to the left and then click on Work Managers.
4. Click on the New button then select the Maximum Threads Constraint radio button. Click on the Next button.
5. Enter defaultMaxThreads in the Name text field and 50 in the Count text field. Click on the Next button.
6. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
7. Click on the New button again and select the Work Manager radio button. Click on the Next button.
8. Enter default in the Name text field (as shown in the following screenshot) and click

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

on the Next button.



9. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
10. Click on the newly created default work manager and select the defaultMaxThreads option from the Maximum Threads Constraint drop-down menu. Check the Ignore Stuck Threads checkbox (as shown in the following screenshot) and click on the Save button.



11. Click on the Activate Changes button to finish.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

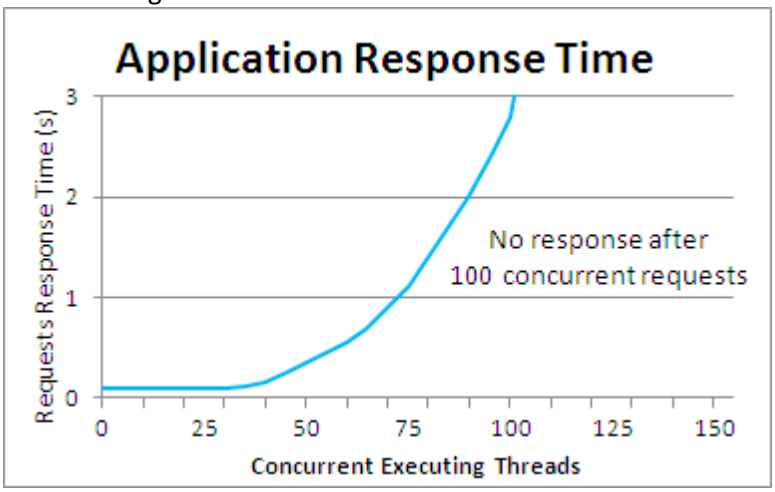
How it works:

The Oracle WebLogic Server is a multithreaded application server. In a very simplistic way, it is a thread pool that receives and dispatches the incoming requests to be processed by the execution threads.

The WebLogic thread pool is called Self-tuning Thread Pool. It adjusts itself automatically trying to maximize the throughput by increasing and decreasing the number of active threads in the pool. The Self-tuning Thread Pool feature was introduced in WebLogic Server Version 9 and is, at the same time, one of its greatest features and one of the greatest weaknesses of the WebLogic Server.

In production environments, some scenarios of peak time or large bursts of incoming requests can fool the Self-tuning Thread Pool to grow too much, reaching some hundred threads. The default maximum size the Self-tuning Thread Pool can have is 400 threads and, with this size,it is unlikely that the WebLogic Server would handle all the concurrent incoming requests,probably leading the system to a hang situation.

The following graph illustrates a scenario where a WebLogic Managed Server hangs after it reaches a concurrency of 100 simultaneous requests being processed. The response time increases until the WebLogic Managed Server hangs:



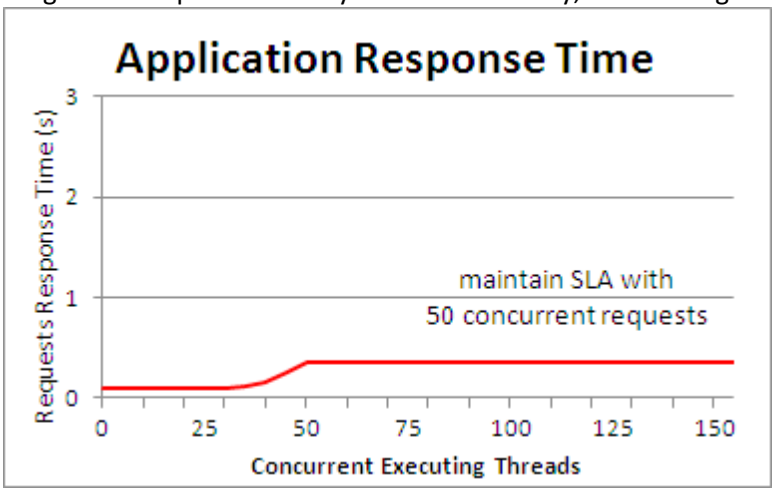
Designing a clustered architecture of small boxes, with low concurrency and low number of threads running in each Managed Server will normally get better stability and performance than a single, larger instance with a lot of threads.

In this recipe, the default work manager was created to override the default configuration,limiting the concurrency of all applications deployed. The default work manager handles the application requests when no other work manager is defined. The work manager constraints are valid per WebLogic Server, so this configuration limits the **PROD_Server01, PROD_Server02, PROD_Server03, and PROD_Server04** to execute a maximum of 50 concurrent and simultaneous threads in each; or 200 concurrent threads, if counting the whole **PROD_Cluster**(50 threads * 4 instances)**.**

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

At peak time, when the default work manager already has 50 threads executing on a Managed Server, the next request will be queued in the priority queue and wait until a thread is free. This configuration avoids the Self-tuning Thread Pool increasing the number of threads to a point that the WebLogic cannot handle the concurrency. On the other side, it also adds the possibility of queuing some of the requests, with a possible impact in the response time.

The following graph illustrates the protected WebLogic. Even though it's receiving more than 50 incoming requests, the Managed Server processes only 50 simultaneously, maintaining the SLA.



Tuning the application thread concurrency with custom work managers:

In the previous recipe, the default work manager was created to limit the concurrency of all applications deployed in the PROD_Cluster cluster.

In this recipe, a new custom work manager myWebServiceWM will be created, with a maximum thread constraint named myWebServiceMaxThreads, configured with 20,and a capacity constraint of 20 as well.

The myWebServiceWM will be associated with the myWebService.war application so only the concurrency for the requests of this application will be limited to 20 threads per Managed Server.

Getting ready

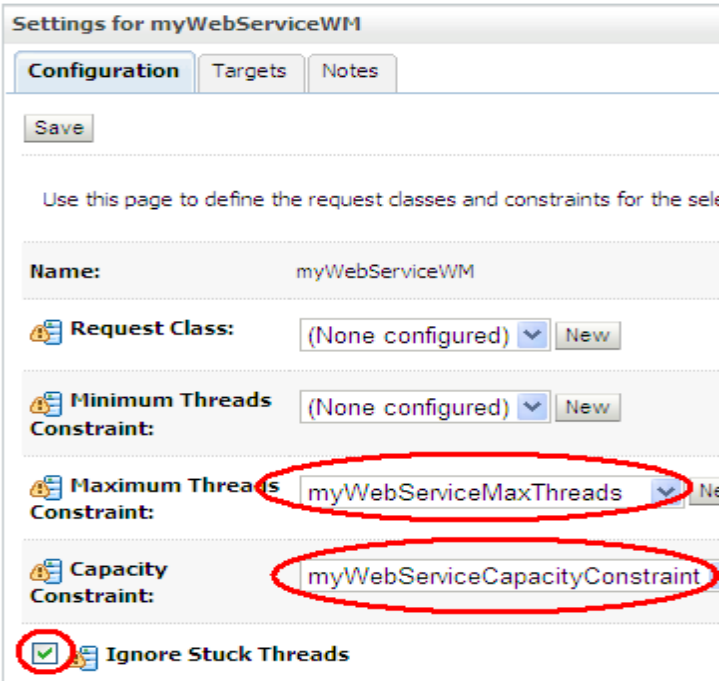
The work manager will be defined in the Administration Console. The application also must update a descriptor in order to be associated with the new work manager.

How to do it:

First create the myWebServiceMaxThreads maximum threads constraint and the myWebServiceWM work manager in the Administration Console:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

1. Access the Administration Console with your web browser at http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree on the left and then click on Work Managers.
4. Click on the New button then select the Maximum Threads Constraint radio button. Click on the Next button.
5. Type myWebServiceMaxThreads in the Name text field and 20 in the Count text field. Click on the Next button.
6. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
7. Click on the New button and select the Capacity Constraint radio button.
8. Type myWebServiceCapacityConstraint in the Name field and 20 in the Count field. Click on the Next button.
9. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
10. Click on the New button again and select the Work Manager radio button. Click on the Next button.
11. Type myWebServiceWM in the Name text field and click on the Next button.
12. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
13. Click on the newly created myWebServiceWM work manager. Select the myWebServiceMaxThreads option from the Maximum Threads Constraint drop-down menu and the myWebServiceCapacityConstraint option from the Capacity Constraint drop-down menu. Check the Ignore Stuck Threads checkbox (as shown in the following screenshot) and click on the Save button.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

14. Click on the Activate Changes button to finish.Now associate the myWebServiceWM work manager to be used by the myWebService.warapplication, by editing the file descriptor, WEB-INF/weblogic.xml:

1. Edit the file WEB-INF/weblogic.xml and add the tag <wl-dispatch-policy>:

```
<weblogic-web-app>
<session-descriptor>
   ...
</session-descriptor>
<wl-dispatch-policy>myWebServiceWM</wl-dispatch-policy>
</weblogic-web-app>
```

2. Save and repack the myWebService.war application.
3. Redeploy the myWebService.war application.

How it works:

The custom work manager works the same way as the default work manager, but is limited to handle only the myWebService.war application requests.

In this recipe, a capacity constraint was also added to the work manager. The capacity constraint is the sum of the concurrent executing threads (limited by max threads constraint) and the queued requests. Since the myWebServiceWM is configured with the capacity constraint and max threads constraint with the same value of 20, the work manager won't allow any requests to be queued in the instance.

This application exposes only stateless web services, so there is no need for session affinity. The requests can be load balanced in a round-robin fashion. When one WebLogic Server instance reaches 20 concurrent requests, the next request will fail over to the next WebLogic instance of the cluster. In practice, when the instance is overloaded with the defined max threads running, it returns the HTTP Code 503. The recognizes the 503 code and fails over the request to the next instance, which is transparent to the caller.

Note :

The HTTP request failover is controlled by the WebLogic plug-in, so it's mandatory to use the Apache/OHS with the plug-in when configuring work managers.

Limiting the JMS Queue consumers:

By default, an MDB (message driven bean) uses up to 16 threads per WebLogic Server instance to consume a JMS Queue destination.

In this recipe, an MDB QueueMDB from the hypothetical JMSApp application will be configured to use only one thread to consume the JMS Queue. This will be done by creating a new work manager

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

jmsAppWM with a maximum thread constraint jmsAppMaxThreads with the value 1 and update the MDB descriptor to associate the new work manager.

Getting ready:

The work manager will be defined in the Administration Console. The MDB descriptor must also update a descriptor in order to be associated with the new work manager.

How to do it:

First, create the jmsAppMaxThreads maximum threads constraint and the jmsAppWM work manager in the Administration Console:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Expand the Environment tree to the left and then click on Work Managers.
4. Click on the New button and then select the Maximum Threads Constraint radio button. Click on the Next button.
5. Type jmsAppMaxThreads in the Name text field and 1 in the Count text field. Click on the Next button.
6. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
7. Click on the New button again and select the Work Manager radio button. Click on the Next button.
8. Type jmsAppWM in the Name text field and click on the Next button.
9. Select the All servers in the cluster radio button from the PROD_Cluster option and click on the Finish button.
10. Click on the newly created jmsAppWM work manager. Select the jmsAppMaxThreads option from the Maximum Threads Constraint drop-down menu. Check the Ignore Stuck Threads checkbox and click on the Save button.
11. Click on the Activate Changes button to finish.

Now associate the jmsAppWM work manager to be used by the JMSApp application, by editing the file descriptor META-INF/weblogic-ejb-jar.xml:

1. Edit the file META-INF/weblogic-ejb-jar.xml and add the tag <dispatchpolicy>:

```
<weblogic-enterprise-bean>
<ejb-name>QueueMDB</ejb-name>
<message-driven-descriptor>
...
</message-driven-descriptor>
<dispatch-policy>jmsAppWM</dispatch-policy>
```


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

</weblogic-enterprise-bean>

2. Save and repack the JMSApp application.
3. Redeploy the JMSApp application.

How it works.

The QueueMDB now consumes the JMS Queue with a maximum of 1 thread instead of 16.

Controlling the JMS Queue consumers is important to prevent too much concurrency in the environments with a lot of JMS Queues. In a scenario where a hypothetic application uses 20 JMS Queues and there is a large backlog of messages in them, WebLogic can use up to 320 threads (20 Queues * 16 consumer threads).

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

SECURITY

In this chapter, we will cover the following recipes:

- Setting up SSL for production environments
- Creating a new SQL authentication provider
- Assigning a user to a group
- Securing a web application with basic authentication
- Enabling the Administration Port

Introduction:

To properly secure a production WebLogic domain, the hardware and host machines must be physically safe, and operating system and filesystem access must be restricted. Network access must be protected from unwanted traffic by means of a firewall, and communication must be encrypted to protect information.

The security subject in Oracle WebLogic Server 12*c* includes so many aspects and features that a whole book could be written about it.

WebLogic Server includes the WebLogic Security Service, a set of configurations and tools to secure the WebLogic domain and its resources. As this is an administration cookbook, this chapter focuses on some security administration tasks, such as setting up an authentication provider and enabling **Secure Sockets Layer (SSL).**

Setting up SSL for production environments

WebLogic Server 12*c* supports SSL to add security and encryption to the data transmitted over the network.

In this recipe, SSL will be enabled in the PROD_AdminServer instance of the PROD_DOMAIN domain.

A new identity keystore and a new trusted keystore will be created to store the new certificate.The WebLogic Server instances and the Node Manager will be configured to enable the SSL protocol and use the custom keystores.

Getting ready:

The keystores are created with the keytool command-line utility, and we will demonstrate signing a certificate with the CertGen Java utility. keytool comes as standard with the Java distribution, and CertGen is part of the WebLogic Server. Both utilities run from the command line, so log in to the Linux shell.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How to do it:

Create the identity keystore PRODIdentity.jks on the prod01 machine:

3. Log in to shell as the user wls, and create a new folder named /oracle/Middleware/user_projects/domains/PROD_DOMAIN/keystores:

[wls@prod01]$ mkdir /oracle/Middleware/user_projects/domains/PROD_DOMAIN/keystores

2. Set the PROD_DOMAIN environment variables with the setDomainEnv.sh script and create the keystore:

[wls@prod01]$ cd /oracle/Middleware/user_projects/domains/PROD_ DOMAIN/bin
[wls@prod01]$ . ./setDomainEnv.sh
[wls@prod01]$ cd keystores
[wls@prod01]$ keytool -genkeypair -alias prodcert -keyalg RSA -keysize 1024 -dname "CN=*.domain.local,OU=MyOrganization,O=MyComp any,L=MyCity,S=MyState,C=US" -keystore PRODIdentity.jks

3. Type and confirm the password for the keystore, and then type <ENTER> to use the same password for prodcert:

Enter keystore password: <Type a new password>
Re-enter new password: <Re-type the password>
Enter key password for <prodcert>

Generate a new CSR using PRODIdentity.jks:

1. Execute the keytool utility to generate the CSR.

[wls@prod01]$ keytool -certreq -v -alias prodcert -file PRODCert.csr -keystore PRODIdentity.jks

2. Type the password when required:

Enter keystore password: <Type the password> Certification request stored in file <PRODCert.csr>
Submit this to your CA

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Sign the CSR and import it into the identity keystore:

1. Submit PRODCert.csr to the Certificate Authority of your choice to get the digital certificate and its private key. For demonstration purposes, this recipe will use the CertGen utility to create and sign the certificate from the CSR. CertGen uses the WebLogic Demo CA (CertGenCA.der):

[wls@prod01]$ java utils.CertGen -keyfile PRODCertPrivateKey -keyfilepass password -certfile PRODCert -cn "*.domain.local" Generating a certificate with common name *.domain.local and key strength 1024 /oracle/Middleware/wlserver_12.1/server/lib/CertGenCA.der file and key from /oracle/Middleware/wlserver_12.1/server/lib/CertGenCAKey.der file

2. Import the server certificate and private keys to the PRODIdentity.jks keystore:

[wls@prod01]$

java utils.ImportPrivateKey -keystore PRODIdentity.jks -keyfile PRODCertPrivateKey.pem -keyfilepass password -certfile PRODCert.pem -storepass password -alias prodcert

Create the custom trust keystore PRODTrust.jks on the prod01 machine:

1. Create the PRODTrust.jks keystore by making a copy from the Standard Java Trust.

[wls@prod01]$ cp /oracle/jvm/jre/lib/security/cacerts ./PRODTrust.jks

2. Change the default cacerts password. The default is changeit. Change it to a new one:

[wls@prod01]$ keytool -storepasswd -keystore PRODTrust.jks Enter keystore password: changeit New keystore password: <Type the new password>Re-enter new keystore password: <Re-type the new password>

3. In previous steps, the WebLogic Demo CA (CertGenCA.der) was used to sign the certificate, so it will be imported to the trust keystore. In production, import the CA certificate from your trusted CA vendor.

[wls@prod01]$ keytool -import -v -trustcacerts -alias rootCA-file /oracle/Middleware/wlserver_12.1/server/lib/CertGenCA.der -keystore PRODTrust.jks

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Distribute the keystore folder to all machines on the PROD_DOMAIN domain:

1. Copy the keystore folder to the prod02 machine:

[wls@prod01]$ scp –r /oracle/Middleware/user_projects/domains/PROD_DOMAIN/keystores prod02:/oracle/Middleware/user_projects/domains/PROD_DOMAIN/

Change the Node Manager in the prod01 and prod02 machines to use the custom keystores and the new certificate:

2. Edit the nodemanager.properties file:

[wls@prod01]$ vi $WL_HOME/common/nodemanager/nodemanager.properties

3. Add the following lines to the file:

KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=/oracle/Middleware/user_projects/
domains/PROD_DOMAIN/keystores/PRODIdentity.jks
CustomIdentityKeyStorePassPhrase=password
CustomIdentityAlias=prodcert
CustomIdentityPrivateKeyPassPhrase=password

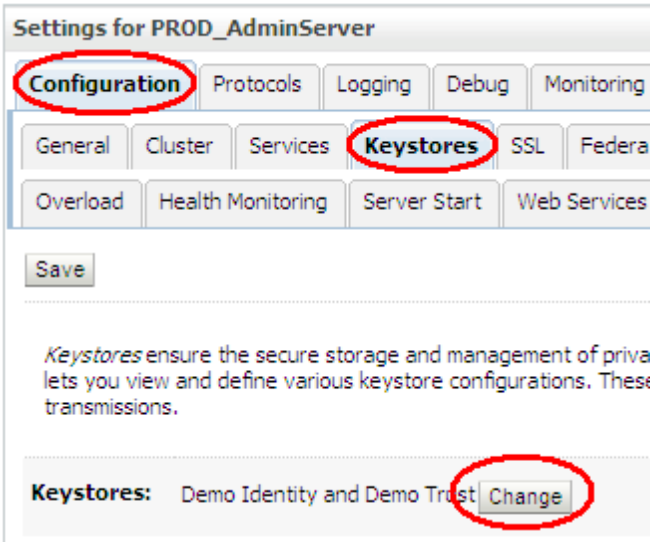
4. Enter :**ws!** to save and exit.
5. Repeat the **nodemanager.properties** configurations for the prod02 machine.
6. Restart the Node Manager.

Assign WebLogic Server instances to use the custom keystores and the certificate:

4. Access the Administration Console by pointing your web browser to **http://adminhost.domain.local:7001/console.**
2. Click on the [+] Environment sign from the navigation box to the left and then click on the Servers link.
3. Click on the PROD_AdminServer link.

4. Click on the Keystores tab and then click on the Change button from the Keystores option As shown in the following screenshot:

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**



5. Select the Custom Identity and Custom Trust option from the Keystores drop-down menu and click on the Save button.

6. Enter ./keystores/PRODIdentity.jks in the Custom Identity Keystore text field. Then, enter jks in the Custom Identity Keystore Type text field. Enter the password chosen earlier in Custom Identity Keystore Passphrase and Confirm Custom Identity Keystore Passphrase.

7. Enter ./keystores/PRODTrust.jks in the Custom Trust Keystore text field.Then, enter jks in the Custom Trust Keystore Type text field. Enter the password chosen earlier in Custom Trust Keystore Passphrase and Confirm Custom Trust Keystore Passphrase. Click on the Save button.
8. Click on the SSL tab and type prodcert in the Private Key Alias text field. Enter the password chosen earlier in Private Key Passphrase and Confirm Private Key Passphrase. Select the Custom Hostname Verifier option from the Hostname Verification drop-down menu and enter weblogic.security.utils. SSLWLSWildcardHostnameVerifier in the Custom Hostname Verifier text field. Click on the Save button.

9. Click on the General tab and check the SSL Listen Port Enabled checkbox As shown in the following screenshot:

10. Enter 7002 in the SSL Listen Port text field and click on the Save button.

11. Repeat the preceding steps for the PROD_Server01, PROD_Server02, PROD_Server03, and PROD_Server04 instances, using 9001, 9002, 9003, and 9004 in the SSL Listen Port text fields, respectively.
12. Click on the Activate Changes button.

13. Restart the Administration Server and the Managed Servers.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works:

Two new custom keystores were created. The **identity keystore**, **PRODIdentity.jks**,was created to store the certificate and its private key. The trust keystore, PRODTrust.jks,was created to store the root CA certificate.

All WebLogic Server instances were configured to use the custom identity and trust keystores and stop using the default **DemoIdentity.jks** and **DemoTrust.jks keystores**. The Node Manager was also configured to use the custom keystores and the new certificate.

The SSL protocol was then enabled in the PROD_AdminServer and PROD_Server01,PROD_Server02, PROD_Server03, and PROD_Server04 Managed Servers.

This recipe used only one certificate for the WebLogic server instances and the Node Managers. The certificate was signed with CN=*.domain.local, meaning it should be valid to any host with the domain.local address. This is possible by enabling the weblogic.security.utils.SSLWLSWildcardHostnameVerifier class of the Custom HostName Verification namespace.

Creating a new SQL authentication provider:

New domains in WebLogic Server 12*c* are created with the default authentication provider called DefaultAuthenticator. DefaultAuthenticator authenticates the users and groups stored in the internal LDAP mechanism on the WebLogic Server. The internal LDAP runs embedded with the WebLogic Server Instance. The Administration Server runs the master LDAP and the Managed Servers run the LDAP as replicas.
It is possible to use the internal LDAP to store and authenticate the users and groups in production, but the WebLogic administrator can add more robust types of authentication providers such as, a database, Active Directory, and external LDAP, among many others.

In this recipe, a new SQL authentication provider named PRODSQLProvider will be configured and added to the PROD_DOMAIN domain to store and handle the users and groups in an Oracle database.

A new data source, ds-Provider, will be created. The database runs at the dbhost hostname and listens to the port 1521. The listener accepts requests to the service name dbservice. The database username is dbuser, and the password is dbpwd.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Getting ready

Unfortunately, WebLogic Server 12*c* does not provide an out of the box database script to create the tables needed to store the users and groups, so you must run the script provided to create the tables and insert the default groups.

How to do it :

1. Run the following script to create the tables in your Oracle database:

```
CREATE TABLE USERS
(
U_NAME VARCHAR(200) NOT NULL,
U_PASSWORD VARCHAR(50) NOT NULL,
U_DESCRIPTION VARCHAR(1000)
);
ALTER TABLE USERS
ADD CONSTRAINT PK_USERS PRIMARY KEY (U_NAME);
CREATE TABLE GROUPS
(
G_NAME VARCHAR(200) NOT NULL,
G_DESCRIPTION VARCHAR(1000) NULL
);
ALTER TABLE GROUPS
ADD CONSTRAINT PK_GROUPS PRIMARY KEY (G_NAME);
CREATE TABLE GROUPMEMBERS
(
G_NAME VARCHAR(200) NOT NULL,
G_MEMBER VARCHAR(200) NOT NULL
);
```

ALTER TABLE GROUPMEMBERS

```
ADD CONSTRAINT PK_GROUPMEMS PRIMARY KEY ( G_NAME, G_MEMBER );
ALTER TABLE GROUPMEMBERS
ADD CONSTRAINT FK1_GROUPMEMBERS FOREIGN KEY ( G_NAME )
REFERENCES GROUPS (
    G_NAME) ON DELETE CASCADE;
```

2. Populate the tables with the default WebLogic groups:

```
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES ('AdminChannelUse
rs','AdminChannelUsers can access the admin channel.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES ('Administrators
```


HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

```
','Administrators can view and modify all resource attributes and
start and stop servers.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES
('AppTesters','AppTesters group.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES ('CrossDomainCon
nectors','CrossDomainConnectors can make inter-domain calls from
foreign domains.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES
('Deployers','Deployers can view all resource attributes and
deploy applications.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES
('Monitors','Monitors can view and modify all resource attributes
and perform operations not restricted by roles.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES
('Operators','Operators can view and modify all resource
attributes and perform server lifecycle operations.');
INSERT INTO GROUPS (G_NAME,G_DESCRIPTION) VALUES
('OracleSystemGroup','Oracle application software system group.');

COMMIT;
```

Access the Administration Console to create the new data source ds-Provider:

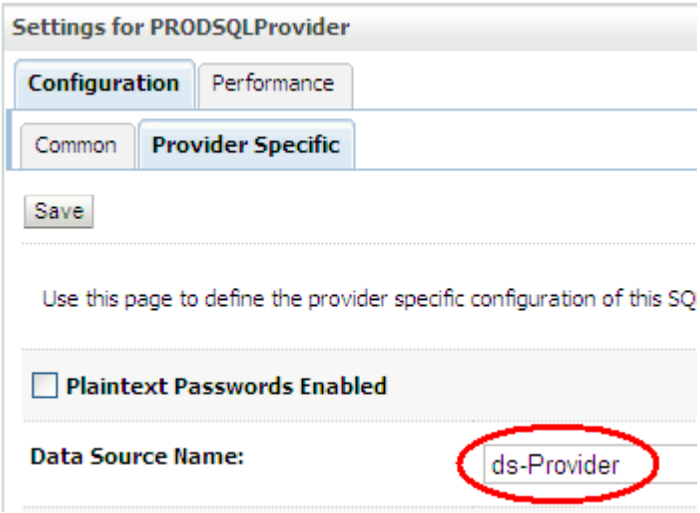
7. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
8. Click on the Lock & Edit button to start a new edit session.
9. Expand the Services tree to the left, and then click on Data Sources.
10. Click on the New button and then click on Generic Data Source.
11. Enter ds-Provider in the Name field and jdbc/ds-Provider in the JNDI Name field. Leave the Database Type drop-down menu with the Oracle option selected.Click on the Next button.
12. Choose *Oracle's Driver (Thin) for Service connections; Versions:9.0.1 and later from the Database Driver drop-down menu. Click on the Next button.
13. Leave Transaction Options with the default values and click on the Next button.
14. On the Connection Properties page, enter dbservice in the Database Name field,dbhost in the Host Name field, and 1521 in the Port field. Fill the Database User Name, Password, and Confirm Password fields with dbuser and dbpwd. Click on the Next button.
15. Click on the Next button on the Test Database Connection page.
16. Select the PROD_AdminServer checkbox and the All servers in the cluster radio button from the PROD_Cluster cluster. Click on the Finish button.Click on the Activate Changes button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Create a new security provider, PRODSQLProvider:

1. Click on the Lock & Edit button to start a new edit session.
2. Click on the Security Realms option (shown in the following screenshot)in the left-hand navigation box and then click on the myrealm link.

3. On the Settings for myrealm page, click on the Providers tab.
4. Click on the New button on the Authentication Providers page.
5. Enter PRODSQLProvider in the Name text field and choose SQLAuthenticator in the Type drop-down menu. Click on the OK button.
6. Click on PRODSQLProvider and then click on the Provider Specific tab.
7. Enter ds-Provider in the Data Source Name text field (as shown in the following screenshot) and click on the Save button. Leave all other options at their default values.



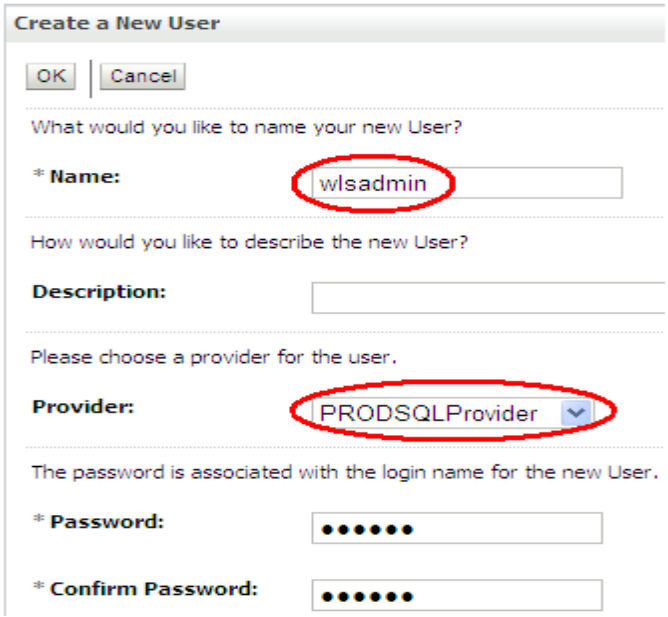
8. Click on the Activate Changes button.
9. Restart all instances of PROD_DOMAIN.

Create a new user, wlsadmin, for your new provider:

1. Access the Administration Console again by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Security Realms option in the left-hand navigation box, and then click on the myrealm link.

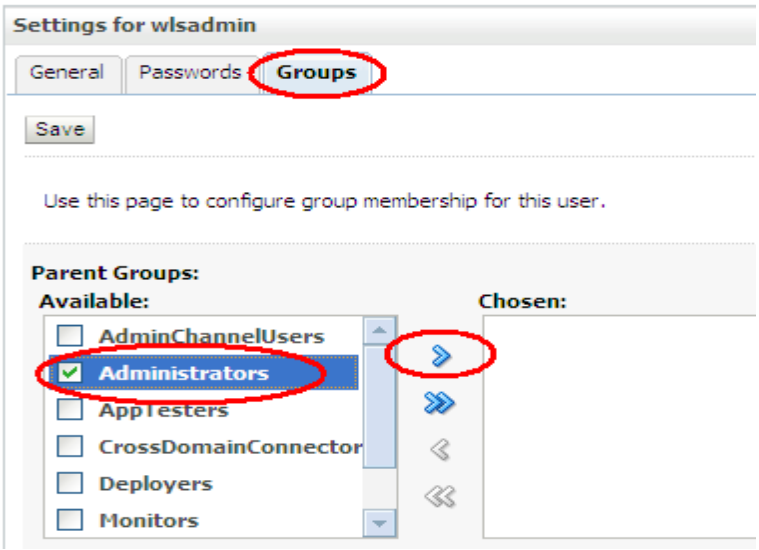
HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

3. Click on the Users and Groups tab.
4. On the Users page, click on the New button.
5. Enter **wlsadmin** in the Name text field, choose the PRODSQLProvider from the **Provider** drop-down menu, and enter wlspwd123 in the **Password** and **Confirm Password** text fields. Click on the OK button, as shown in the following screenshot:



6. Click on the previously created wlsadmin user for PRODSQLProvider and click on the Groups tab.

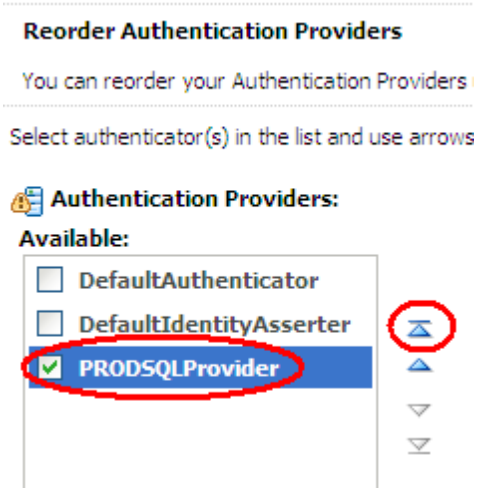
7. Associate the Administrators group with the user by selecting the Administrators checkbox in the Available: table and then clicking on the > button (as shown in the following screenshot). Click on the Save button.



HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

Assign PRODSQLProvider as the first provider and leave Default Authenticator as the second provider. To do this, follow the steps mentioned below:

1. Click on the Lock & Edit button to start a new edit session.
2. Click on the Security Realms option in the left-hand navigati on box and then click on the myrealm link.
3. On the Settings for myrealm page, click on the Providers tab.
4. Click on the Reorder button.
5. Select the PRODSQLProvider checkbox in the Available table and click on the upper arrow on the right to move PRODSQLProvider to the top of the list (as shown in the following screenshot). Click on the OK button.



6. Click on PRODSLQProvider again. Change the Control Flag drop-down menu to SUFFICENT. Click on the Save button.
7. Go back to the Providers page and click on DefaultAuthenticator. Change the Control Flag drop-down menu selection to SUFFICENT. Click on the Save button.
8. Click on the Activate Changes button.
9. Shut down the Administration Server and all instances of the PROD_DOMAIN.

Change the boot.properties file of the Administration Server to look up for the user PRODSQLProvider wlsadmin by following these steps:

1. Go to the Administration Server root folder:

[wls@prod01]$ cd **$DOMAIN_HOME/servers/PROD_AdminServer/security**

2. Recreate the boot.properties file to match the wlsadmin user created:

[wls@prod01]$ echo -ne "username=wlsadmin\npassword=wlspwd123" > boot.properties
[wls@prod01]$ cat boot.properties

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

username=wlsadmin
password=wlspwd123

3. Start the Administration Server.

How it works:

The first step created the tables in the database to store the user and group data.

We created the provider PRODSQLProvider with the default values, including the use of encrypted passwords only. We created the data source ds-Provider and associated it with the provider to handle the database connection.

We changed the order of the providers such that PRODSQLProvider is placed before the efaultAuthenticator provider. With both providers configured with the SUFFICIENT control flag, WebLogic will try to authenticate first on PRODSQLProvider; if authentication fails, it will try on DefaultAuthenticator. A possible alternative configuration is to delete DefaultAuthenticator and leave only PRODSQLProvider with the REQUIRED control flag.

Assigning a user to a group:

In this recipe, a new group called myAuthGroup will be created and a new user, authUser, will be created and assigned to this group. PRODSQLProvider will be used.

Getting ready:

The change will be made by using the Administration Console, so make sure the Administration Server is running.

How to do it:

Create a new group, myAuthGroup, and a new user, authUser, for PRODSQLProvider:

1. Access the Administration Console again by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Security Realms option in the left-hand navigation box, and then click on the myrealm link.
3. Click on the Users and Groups tab.
4. Click on the Groups tab and click on the New button.
5. Enter myAuthGroup in the Name text field and choose PRODSQLProvider from the Provider drop-down menu. Click on the OK button.
6. Click on the Users tab and then on the New button.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

7. Enter authUser in the Name text field, choose PRODSQLProvider from the Provider drop-down menu, and enter authpwd123 in the Password and Confirm Password text fields. Click on the OK button.
8. Click on the authUser user for PRODSQLProvider and click on the Groups tab.
9. Associate the myAuthGroup group with the user by checking the myAuthGroup checkbox in the Available table and then clicking on the > button. Click on the Save button.

Securing a web application with basic authentication:

WebLogic Security services allow the WebLogic Administrator to add declarative security roles and policies to WebLogic resources such as web applications, EJBs, and other resources without making changes to the source code or to the file descriptors of the application.

In this recipe, a hypothetical myAuthApp.war web application will be deployed and configured to be accessed only by the users from the PRODSQLProvider that are members of the group myAuthGroup.

Getting ready:

Deploy myAuthApp.war to PROD_Cluster:

1. Create a new application installation directory using the syntax: /oracle/applications/<environment>/<application>/<version>:

   [wls@prod01]$ mkdir -p /oracle/applications/prod/myAuthApp/v1
   [wls@prod01]$ cd /oracle/applications/prod/myAuthApp/v1

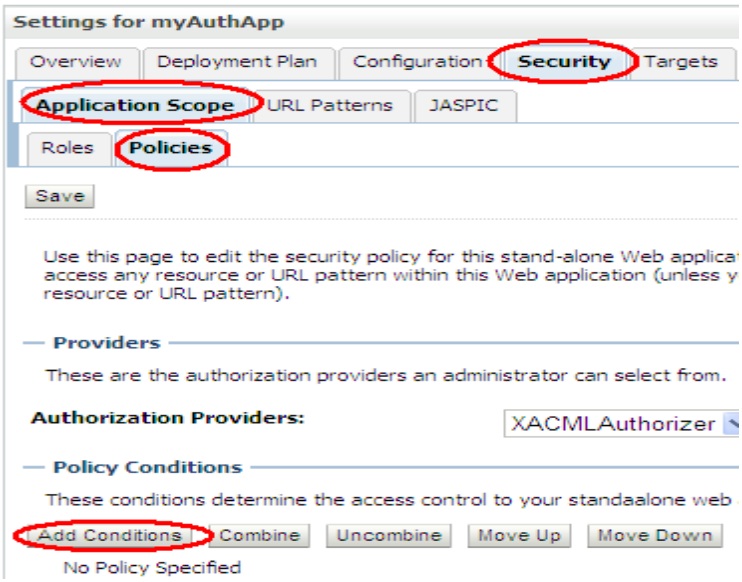
2. Create two directories using the following commands:
   [wls@prod01]$ mkdir app
   [wls@prod01]$ mkdir plan

3. Copy the myAuthApp.war file to the app directory.
4. Access the Administration Console at http://adminhost.domain.local:7001/console.
5. Click on the Lock & Edit button to start a new edit session.
6. Navigate to the Deployments page by clicking on the link in the Domain Structure on the left-hand side navigation table.
7. Click on the Install button to install a new application.
8. Type the path /oracle/applications/prod/myAuthApp/v1/app and click on Next.
9. Select myAuthApp.war from the list and click on Next.
10. Select Install this deployment as an application and click on Next.
11. Select the All servers from the cluster radio button from the PROD_Cluster cluster and click on Next.
12. Select the Custom Roles and Policies: Use only roles and policies that are defined in the Administration Console radio button from the Security tab and click on the Finish button.
15. A new deployment plan file called Plan.xml will automatically be created in /oracle/applications/prod/myAuthApp/v1/plan.
16. Click on the Activate Changes button to apply the changes.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

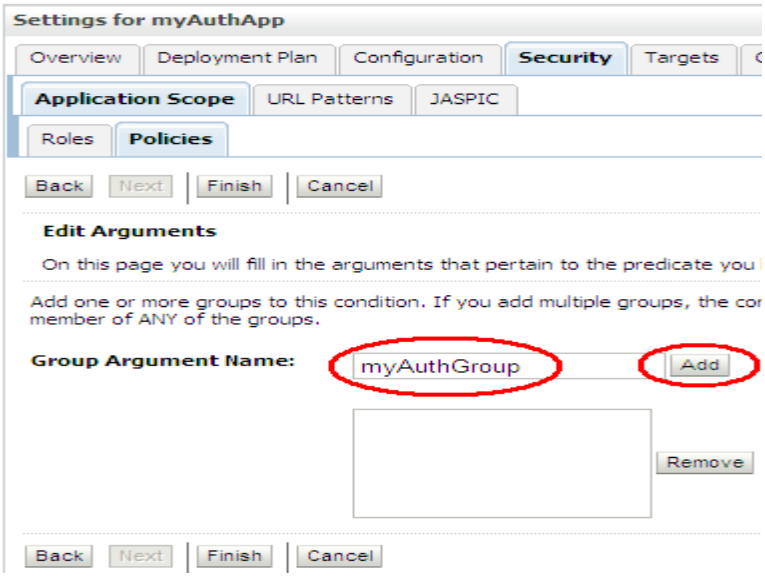
Apply security to myAuthApp.war by following these steps:

1. Click on the myAuthApp link from the Deployments page.
2. Open the Security tab and then click on the Policies tab from the Application Scope.Click on the Add Condition button, as shown in the following screenshot:



3. Choose the Group option from the Predicate List drop-down menu and click on the Next button.

4. Enter myAuthGroup in the Group Argument Name text field and click on the Add button (see the following screenshot) to add it to the list below. Click on the Finish button.



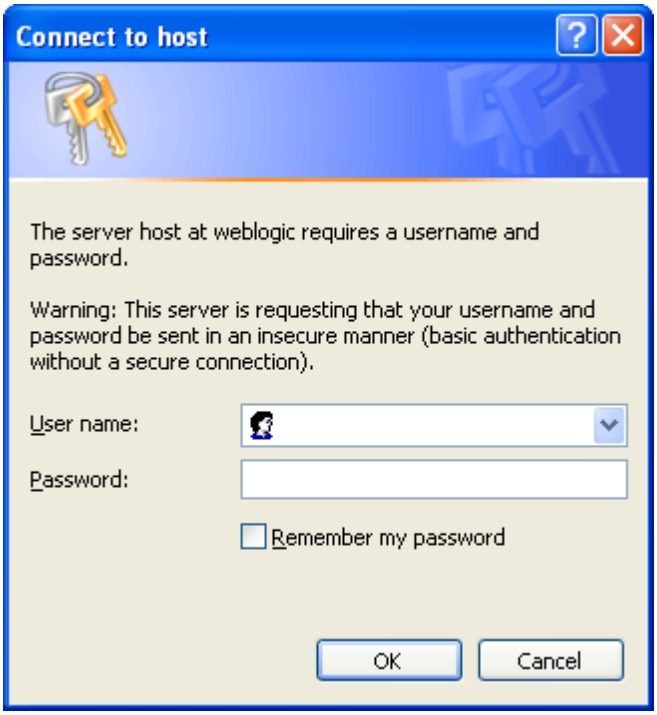
5. Click on the Save button on the Policies page to finish.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How it works

The myAuthApp.war application can now be accessed only by users that match the security policy. In this case, the security policy checks whether the user belongs to the group myAuthGroup.

When a user tries to access the application URI at /myAuthApp, the browser returns a basic authentication window:



To access the application, enter a username, such as authUser, which we created in the previous recipe. authUser is a member of the myAuthGroup group, which is permitted to access the application. Try accessing the application as a user that does not match the policy, and the application will return a HTTP 401 Unauthorized error.

Enabling the Administration Port:

The Administration Port is a domain wide configuration that segregates all administrative traffic from the application traffic.

In this recipe, the Administration Port will be enabled in the PROD_Domain domain.

Getting ready

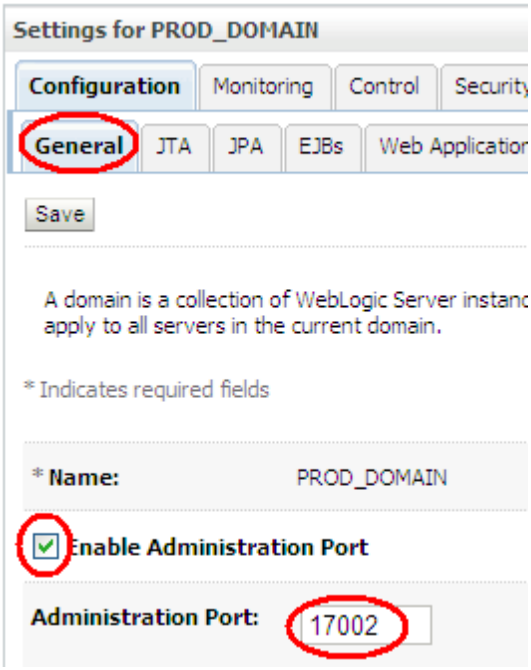
The Administration Port requires all WebLogic Server instances, including the Administration Server and the Managed Server, to already be configured to use SSL.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**

How to do it.

To enable the Administration Port, access the Administration Console:

1. Access the Administration Console by pointing your web browser to http://adminhost.domain.local:7001/console.
2. Click on the Lock & Edit button to start a new edit session.
3. Click on the PROD_DOMAIN link on the left-hand side navigation tree.
4. Check the Enable Administration Port checkbox and enter 17002 inthe Administrative Port text field (as shown in the following screenshot).Click on the Save button.



5. Click on Activate Changes to finish.
6. The Administration Console now is accessible only from the URL https://adminhost.domain.local:17002/console.

How it works...

With the Administration Port enabled, the WebLogic Server creates a new internal network administrative channel that is now used to transfer administrative data between the Administration Server and the Managed Servers.

The Administration Port also allows the segregation of application and administrative traffic through different channels.

HI N Technologies
Flat #402/20,16th A Main, 13th Cross, Maruthinagar, Madivala, Bangalore-560068
PH: 080-42288565 Mob: +91-7676333847
www.hiNtechnologies.com **www.weblogic4you.blogspot.com**