Oracle® Banking Treasury Management

Weblogic Configuration

Release 14.8.0.0.0

2025

ORACLE

Oracle Banking Treasury Management Weblogic Configuration, Release 14.8.0.0.0

Contents

Preface

This topic contains the following sub-topics:

Purpose

This document helps the user with the configuration of the Weblogic.

Audience

This guide is intended for the central administrator of the Bank who controls the system and application parameters and ensures smooth functionality and flexibility of the banking application.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at Critical Patches, Security Alerts and Bulletins. All critical patches should be applied in a timely manner to ensure effective security, as strongly recommended by Oracle Software Security Assurance.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Resources

For more information on any related features, refer to the following documents

- Open Development Tool Installation
- Development Workbench - Administration

Conventions

The following text conventions are used in this document:

Table 1 Conventions

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

The acronyms and abbreviations are listed in this below table:

Table 2 Acronyms and Abbreviations

Abbreviations or Acronyms	Definition
DV	Derivatives
DSN	Data Source name
EAR	Enterprise archive
ETD	Exchange Traded Derivatives

Table 2 (Cont.) Acronyms and Abbreviations

Abbreviations or Acronyms	Definition
FX	Foreign Exchange
FCUBS	Oracle FLEXCUBE Universal Banking
LDAP	Lightweight Directory Access Protocol
JDBC	Java Database Connectivity
JVM	Java Virtual Machine
MM	Money Market
OBTR	Oracle Banking Treasury Management
OT	Over the Counter Options
SE	Securities
SR	Securities Repo
SMS	Security Management System
UI	User interface

1

Configure SSL on Oracle Weblogic

This topic explains the configurations for SSL on Oracle Weblogic Application Server.

This topic contains the following sub-topics.

- Set up SSL on Oracle Weblogic
  This topic explains the steps to set up the SSL on Oracle Weblogic.
- Certificates and Keypairs
  This topic explains the Certificates and Keypairs used for validating the authenticity of the server.

1.1 Set up SSL on Oracle Weblogic

This topic explains the steps to set up the SSL on Oracle Weblogic.

You need to perform the following steps to set up SSL on the Oracle Weblogic Application server:

1. Obtain an identity (private key and digital certificates) and trust (certificates of trusted certificate authorities) for the Oracle Weblogic Application server.
2. Store the identity and trust.

   Private keys and trust CA certificates are stored in keystores.
3. Configure the identity and trust keystores for the Oracle Weblogic Application server in the Administration console.
4. Set SSL attributes for the private key alias and password in the Oracle Weblogic Administration console.

1.2 Certificates and Keypairs

This topic explains the Certificates and Keypairs used for validating the authenticity of the server.

Certificates are used for validating the authenticity of the server. Certificates contain the name of the owner, certificate usage, duration of validity, resource location, or distinguished name (DN), which includes the common name (CN - website address or e-mail address depending on the usage) and the certificate ID of the person who certified (signs) this information. It also contains the public key and a hash to ensure that the certificate has not been tampered with. A certificate is insecure until it is signed. Signed certificates cannot be modified.

A certificate can be self-signed or obtained from a reputable certificate authority such as Verisign, Inc., Entrust.net, Thawte, GeoTrust, or InstantSSL.

SSL uses a pair of cryptographic keys - a **public key** and a **private key**. These keys are similar in nature and can be used alternatively. What one key encrypts can be decrypted by the other key of the pair. The private key is kept secret, while the public key is distributed using the certificate.

A key tool stores the keys and certificates in a keystore. The default keystore implementation implements it as a file. It protects private keys with a password. The different entities (key pairs and the certificates) are distinguished by a unique **alias**. Through its keystore, the Oracle Weblogic server can authenticate itself to other parties.

In Java, a keystore is a **java.security.KeyStore** instance that you can create and manipulate using the keytool utility provided with the Java Runtime.

There are two keystores to be managed by the Oracle Weblogic server to configure SSL.

- **Identity Keystore**: This contains the key pairs and the Digital certificate. This can also contain certificates of intermediate CAs.
- **Trust Keystore**: Contains the trusted CA certificates.

2

Choose the Identity and Trust Stores

This topic explains how to choose the identity and trust stores.

Oracle Financial Services Software recommends that the choice of Identity and Trust stores be made upfront. Oracle Weblogic Server supports the following combinations of Identity and Trust stores:

- Custom Identity and Command Line Trust
- Custom Identity and Custom Trust
- Custom Identity and Java Standard Trust
- Demo Identity and Demo Trust

Oracle Financial Services Software does not recommend choosing Demo Identity and Demo Trust for production environments.

It is recommended to separate the identity and trust stores since each Weblogic server tends to have its own identity but might have the same set of trust CA certificates. Trust stores are usually copied across Oracle Weblogic servers to standardize trust rules; it is acceptable to copy trust stores since they contain public keys and certificates of CAs. Unlike trust stores, identity stores contain private keys of the Oracle Weblogic server and hence should be protected against unauthorized access.

Command Line Trust, if chosen requires the trust store to be specified as a command-line argument in the Weblogic Server startup script. No additional configuration of the trust store is required in the Weblogic Server Administration Console.

Java Standard Trust would rely on the cacerts files provided by the Java Runtime. This file contains the list of trust CA certificates that ship with the Java Runtime and are located in the `JAVA_HOME/jre/lib/security` directory. It is highly recommended to change the default Java standard trust store password from **changeit** (without quotes), and the default access permission of the file. Certificates of most commercial CAs are already present in the Java Standard Trust store. Therefore, it is recommended to use the Java Standard Trust store whenever possible. The rest of the document will assume the use of Java Standard Trust since most CA certificates are already present in it.

One can also create custom trust stores containing the list of certificates of trusted CAs.

For further details on identity and trust stores, refer to the Oracle Weblogic Server documentation on Securing Oracle Weblogic Server.

3

Obtain the Identity Store

This topic explains the creation of Identity Stores.

This topic contains the following sub-topics.

- Create Identity Store with Self-Signed Certificates
  This topic explains the steps to create Identity Store with Self-Signed Certificates.
- Create Identity Store with Trusted Certificates Issued by CA
  This topic explains to create identity store with trusted certificates issued by CA.

3.1 Create Identity Store with Self-Signed Certificates

This topic explains the steps to create Identity Store with Self-Signed Certificates.

Create Identity Store with Self-Signed Certificates

Self-signed certificates are acceptable for use in a testing or development environment. Oracle Financial Services does not recommend the use of self-signed certificates in a production environment.

To create a self-signed certificate, the genkeypair option provided by the keytool utility of Sun Java 6 needs to be utilized.

Creation of Self-signed Certificate

Browse to the bin folder of JRE from the command prompt and type the following command.

```
keytool -genkeypair -alias alias -keyalg RSA -keysize 1024 -sigalg
SHA1withRSA -validity 365 -keystore keystore
```

In the above command,

1. ***alias*** is used to identify the public and private key pair created. This alias is required later when configuring the SSL attributes for the managed servers in Oracle Weblogic Server.
2. ***keystore*** is used to specify the location of the JKS file. If no JKS file is present in the path provided, one will be created.

The command will prompt for the following attributes of the certificate and keystore:

1. **Keystore Password:** Specify a password that will be used to access the keystore. This password needs to be specified later when configuring the identity store in Oracle Weblogic Server.
2. **Key Password:** Specify a password that will be used to access the private key stored in the keystore. This password needs to be specified later when configuring the SSL attributes of the managed server(s) in the Oracle Weblogic Server.
3. **First and Last Name (CN):** Enter the domain name of the machine used to access the application, for instance, www.example.com
4. **Name of your Organizational Unit:** The name of the department or unit making the request, for example, BPD. Use this field to identify the SSL Certificate you are creating, for example, by department or by the physical server.

5. **Name of your Organization:** The name of the organization making the certificate request, for example, Oracle Financial Services. It is recommended to use the company or organization's formal name, and this name entered here must match the name found in official records.
6. **Name of your City or Locality:** The city in which your organization is physically located, for example, Mumbai.
7. **Name of your State or Province:** The state/province in which your organization is physically located, for example, Maharashtra.
8. **Two-Letter Country Code for this Unit:** The country in which your organization is physically located, for example, US, UK, IN, etc.

Figure 3-1 Stop image

The key generation algorithm has been specified as RSA, the key size as 1024 bits, the signature algorithm as SHA1withRSA, and the validity days as 365. These can be changed to suitable values if the need arises. For further details, please refer to the documentation of the keytool utility in the JDK utilized by the Oracle Weblogic Server.

Listed below is the result of a sample execution of the command:

```
D:\Oracle\weblogic11g\jrockit_160_05_R27.6.2-20\bin>keytool -
genkeypair -alias selfcert -keyalg RSA -keysize 1024 -sigalg
SHA1withRSA -validity 365 -keystore D:\keystores\FCUBSKeyStore.jks

Enter keystore password:<Enter a password to protect the keystore>
Re-enter new password:<Confirm the password keyed above>
What is your first and last name?
[Unknown]:  cvrhp0729.i-flex.com
What is the name of your organizational unit?
  [Unknown]: BPD
What is the name of your organization?
  [Unknown]: Oracle Financial Services
What is the name of your City or Locality?
  [Unknown]:  Mumbai
What is the name of your State or Province?
  [Unknown]:  Maharashtra
What is the two-letter country code for this unit?
  [Unknown]:  IN
Is CN=cvrhp0729.i-flex.com, OU=BPD, O=Oracle Financial Services, L=Mumbai,
ST=Maharashtra, C=IN correct?
  [no]: yes
Enter key password for <selfcert>
(RETURN if same as keystore password):<Enter a password to protect the key>
Re-enter new password:<Confirm the password keyed above>
```

3.2 Create Identity Store with Trusted Certificates Issued by CA

This topic explains to create identity store with trusted certificates issued by CA.

Create Public and Private Key Pair

Browse to the bin folder of JRE from the command prompt and type the following command.

```
keytool -genkeypair -alias alias -keyalg keyalg -keysize keysize -
sigalg sigalg -validity valDays -keystore keystore
```

> **Note:**
>
> The placeholders should be replaced with suitable values when running the command.

In the above command,

1. **alias** is used to identify the public and private key pair created. This alias is required later when configuring the SSL attributes for the managed servers in Oracle Weblogic Server.
2. **keyalg** is the key algorithm used to generate the public and private key pair. The RSA key algorithm is recommended.
3. **keysize** is the size of the public and private key pairs generated. A key size of 1024 or more is recommended. Please consult with your CA on the key size support for different types of certificates.
4. **sigalg** is the algorithm used to generate the signature. This algorithm should be compatible with the key algorithm and should be one of the values specified in the Java Cryptography API Specification and Reference.
5. **valdays** is the number of days for which the certificate is to be considered valid. Please consult with your CA on this period.
6. **keystore** is used to specify the location of the JKS file. If no JKS file is present in the path provided, one will be created.

The command will prompt for the following attributes of the certificate and keystore:

1. **Keystore Password:** Specify a password that will be used to access the keystore. This password needs to be specified later when configuring the identity store in Oracle Weblogic Server.
2. **Key Password:** Specify a password that will be used to access the private key stored in the keystore. This password needs to be specified later when configuring the SSL attributes of the managed server(s) in the Oracle Weblogic Server.
3. **First and Last Name (CN):** Enter the domain name of the machine used to access FLEXCUBE UBS, for instance, www.example.com
4. **Name of your Organizational Unit:** The name of the department or unit making the request, for example, BPD. Use this field to identify the SSL Certificate you are creating, for example, by department or by the physical server.
5. **Name of your Organization:** The name of the organization making the certificate request, for example, Oracle Financial Services. It is recommended to use the company or organization's formal name, and this name entered here must match the name found in official records.
6. **Name of your City or Locality:** The city in which your organization is physically located, for example, Mumbai.
7. **Name of your State or Province:** The state/province in which your organization is physically located, for example, Maharashtra.
8. **Two-Letter Country Code for this Unit:** The country in which your organization is physically located, for example, US, UK, IN, etc.

Listed below is the result of a sample execution of the command:

```
D:\Oracle\weblogic11g\jrockit_160_05_R27.6.2-20\bin>keytool -
genkeypair -alias cvrhp0729 -keyalg RSA -keysize 1024 -sigalg
SHA1withRSA -validity 365 -keystore D:\keystores\FCUBSKeyStore.jks

Enter keystore password:<Enter a password to protect the keystore>
Re-enter new password:<Confirm the password keyed above>
What is your first and last name?
[Unknown]:  cvrhp0729.i-flex.com
What is the name of your organizational unit?
  [Unknown]: BPD
What is the name of your organization?
  [Unknown]: Oracle Financial Services
What is the name of your City or Locality?
  [Unknown]:  Mumbai
What is the name of your State or Province?
  [Unknown]:  Maharashtra
What is the two-letter country code for this unit?
  [Unknown]:  IN
Is CN=cvrhp0729.i-flex.com, OU=BPD, O=Oracle Financial Services, L=Mumbai,
ST=Maharashtra, C=IN correct?
  [no]: yes
Enter key password for <cvrhp0729>
(RETURN if same as keystore password):<Enter a password to protect the key>
Re-enter new password:<Confirm the password keyed above>
```

Generate CSR

To purchase an SSL certificate, one needs to generate a **Certificate Signing Request (CSR)** for the server where the certificate will be installed.

A CSR is generated from the server and is the server's unique **fingerprint**. The CSR includes the server's public key, which enables server authentication and secure communication.

> **Note:**
>
> If the keystore file or the password is lost and a new one is generated, the SSL certificate and the private key will no longer match. A new SSL Certificate will have to be requested.

The CSR is created by running the following command in the bin directory of the JRE:

```
keytool -certreq -alias alias –file certreq_file -keystore keystore
```

In the above command,

1. **alias** is used to identify the public and private key pair. The private key associated with the alias will be utilized to create the CSR. Specify the alias of the key pair created in the previous step.
2. **certreq_file** is the file in which the CSR will be stored.
3. **keystore** is the location of the keystore containing the public and private key pair.

Listed below is the result of a sample execution of the command.

```
D:\Oracle\Weblogic11g\jrockit_160_05_R27.6.2-20\bin>keytool -certreq -
alias cvrhp0729 -file D:\keystores\certreq.csr -
keystoreD:\keystores\FCUBSKeyStore.jks

Enter keystore password:[Enter the password used to access the keystore]
Enter key password for <cvrhp0729>
(RETURN if same as keystore password):[Enter the password used to access the
key in the keystore]
```

Obtain Trusted Certificate from CA

The processes of obtaining a trusted certificate vary from one CA to another. The CA might perform additional offline verification. Consult the CA issuing the certificate for details on the process to be followed for submission of the CSR and for obtaining the certificate.

Import Certificate into Identity Store

Store the certificate obtained from the CA in the previous step, in a file, preferably in PEM format. Other formats like the p7b file format would require conversion to the PEM format. Details on performing the conversion are not listed here. Refer to the Oracle Weblogic Server documentation on Securing Oracle Weblogic Server for details on converting a Microsoft **p7b** file to the **PEM** format.

The command to be executed for importing a certificate into the identity store depend on whether the trust store is chosen (in the earlier step). It is highly recommended to verify the trust path when importing a certificate into the identity store. The commands provided below assume the use of the Java Standard Trust store.

Import the Intermediate CA certificate
Most Certificate Authorities do not use the root CA certificates to issue identity certificates for use by customers. Instead, Intermediate CAs issue identity certificates in response to the submitted CSRs.

If the Intermediate CA certificate is absent in the Java Standard Trust store, the trust path for the certificate will be incomplete for the certificate, resulting in warnings issued by Weblogic Server during runtime. To avoid this, the intermediate CA certificate should be imported into the identity keystore. Although the intermediate CA certificate can be imported into the Java Standard Trust store, this is not recommended unless the intermediate CA can be trusted.

The following command should be executed to import the intermediate CA certificate into the keystore.

```
keytool -importcert -alias alias -file cert_file -trustcacerts -
keystore keystore
```

In the above command,

1. **alias** is used to identify the public and private key pair. Specify the alias of the key pair used to create the CSR in the earlier step.
2. **cert_file** is the location of the file containing the intermediate CA certificate in a PKCS#7 format (PEM or DER file).
3. **keystore** is the location of the keystore containing the public and private key pair.

The trustcacerts flag is used to consider other certificates (higher intermediaries and the root CA) in the chain of trust. If no chain of trust is established during verification, the certificate will be displayed and one would be prompted to verify it. It is recommended that due diligence be observed when the prompt is displayed to verify a certificate when a chain of trust is absent.

Listed below is a sample execution of the command.

```
keytool -importcert -alias verisigntrialintermediateca -
fileD:\keystores\VerisignIntermediateCA.cer -trustcacerts -
keystoreD:\keystoreworkarea\FCUBSKeyStore.jks

Enter keystore password:<Enter the password used to access the keystore>


Certificate was added to keystore.
```

Import the Identity Certificate

The following command should be executed to import the identity certificate into the keystore.

```
keytool -importcert -alias alias -file cert_file -trustcacerts -
keystore keystore
```

In the above command,

1. **alias** is used to identify the public and private key pair. Specify the alias of the key pair used to create the CSR in the earlier step.
2. **cert_file** is the location of the file containing the PKCS#7 formatted reply from the CA, containing the signed certificate.
3. **keystore** is the location of the keystore containing the public and private key pair.

The trustcacerts flag is used to consider other certificates (intermediate CAs and the root CA) in the chain of trust. If no chain of trust is established during verification, the certificate will be displayed and one would be prompted to verify it. It is recommended that due diligence be observed when the prompt is displayed to verify a certificate when a chain of trust is absent.

Listed below is a sample execution of the command.

```
keytool - importcert -alias cvrhp0729 -file D:\keystores\cvrhp0729.cer
- trustcacerts -keystore D:\keystoreworkarea\FCUBSKeyStore.jks

Enter keystore password:<Enter the password used to access the keystore>
Enter key password for <cvrhp0729>:<Enter the password used to access the
private key>


Certificate reply was installed in keystore
```

The previous set of commands assumed the presence of the appropriate root CA certificate (in the chain of trust) in the Java Standard Trust store, i.e. in the cacerts file. If the CA issuing the identity certificate (for the Weblogic Server) does not have the root CA certificate in the Java Standard Trust store, one can opt to import the root CA certificate into cacerts, or the identity store, depending on factors including the trustworthiness of the CA, the necessity of transporting the trust store across the machine, among others.

4

Configure Identity and Trust Stores for Weblogic

This topic explains how to configure identity and trust stores for Weblogic.

- Enable SSL on Oracle Weblogic Server
  This topic provides the systematic instructions to enable SSL on Oracle Weblogic Server.
- Configure Identity and Trust Stores
  This topic provides the systematic instructions to configure identity and trust stores.

4.1 Enable SSL on Oracle Weblogic Server

This topic provides the systematic instructions to enable SSL on Oracle Weblogic Server.

To configure SSL on the Oracle Weblogic server, log in into the **Administration Console** and follow the steps given below:

1. Click the **Lock & Edit** button under **Change Center**.
2. Expand the **Servers** node.
3. Select the name of the server for which you want to enable SSL (example - exampleserver).
4. Navigate to **Configuration** and select the **General** tab.
5. Select the option **SSL Listen Port Enabled** and specify the SSL listen port.
6. Against **Listen Address**, specify the hostname of the machine in which the application server is installed.

4.2 Configure Identity and Trust Stores

This topic provides the systematic instructions to configure identity and trust stores.

To configure the Identity and Trust stores in Oracle Weblogic Server, log in to the **Administration Console** of Weblogic Server.

1. Click the **Lock & Edit** button under **Change Center**.
2. Expand the **Servers** node.
3. Select the name of the server for which you want to configure the keystores (example - exampleserver).
4. Navigate to **Configuration** and select the **Keystores** tab.
5. In the **Keystores** field, select the method for storing and managing private keys/digital certificate pairs and trusted CA certificates.

   This choice should match the one made in the topic *Choose the Identity and Trust Stores*.
6. In the **Identity** section, provide the following details:
   a. **Custom Identity Keystore File Name:** Fully qualified path to the Identity keystore.

b. **Custom Identity Keystore Type:** Set this attribute to JKS, the type of the keystore. If it is left blank, it defaults to **Java KeyStore (JKS)**.

c. **Custom Identity Keystore PassPhrase:** The password you enter when reading or writing to the keystore. This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. Oracle Weblogic Server only reads from the keystore. So whether or not you define this property depends on the requirements of the keystore.

7. In the **Trust** section, provide the following details:

   If you choose **Java Standard Trust**, specify the password used to access the trust store.

   If you choose **Custom Trust**, the following attributes have to be provided:

   a. **Custom Trust Keystore**: The fully qualified path to the trust keystore.

   b. **Custom Trust Keystore Type**: Set this attribute to JKS, the type of the keystore. If it is left blank, it defaults to **Java KeyStore (JKS)**.

   c. **Custom Trust Keystore Passphrase**: The password you enter when reading or writing to the keystore. This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. Oracle Weblogic Server only reads from the keystore. So, whether or not you define this property depends on the requirements of the keystore.

   > **Note:**
   >
   > When identity and trust stores are of the JKS format, the passphrases are not required.

5

Set SSL Attributes for Managed Servers

This topic explains how to set SSL attributes for managed servers.

- Set SSL Attributes for Private Key Alias and Password
  This topic provides the systematic instructions to set SSL attributes for private key alias and password.

5.1 Set SSL Attributes for Private Key Alias and Password

This topic provides the systematic instructions to set SSL attributes for private key alias and password.

To configure the private key alias and password, log in to the Oracle Weblogic Server **Administration Console**.

1. Click the **Lock & Edit** button under **Change Center**.
2. Expand the **Servers** node.
3. Select the name of the server for which you want to configure the keystores (example - exampleserver).
4. Navigate to **Configuration** and select the **SSL** tab.
5. Select **Keystores** from **Identity and Trust Locations**.
6. Under **Identity** section, specify the following details:
   a. **Private Key Alias:** Set this attribute to the alias name defined for the key pair when creating the key pair in the Identity keystore.
   b. **Private Key Passphrase:** The password defined for the key pair (alias_password) at the time of its creation. Confirm the password.
7. Click **Save**.
8. Click **Activate Changes** button under **Change Center**.
9. Go to the **controls** tab, check the appropriate server, and click **Restart SSL**. Confirm when it prompts.

6
Test Configuration

This topic explains to test the configuration

Once the Oracle Weblogic has been configured for SSL, deploy the application in the usual manner. The application can be tested in SSL mode after deployment.

To launch the application in SSL mode, enter the URL in the following format: `https://(Machine Name):(SSL_Listener_port_no)/(Context_root)`

We recommend to access the web application via the HTTPS channel instead of the HTTP channel.

7

Create Resources on Weblogic

This topic explains the steps to be executed to deploy the Oracle Banking Treasury application and Gateway applications in the Application Server.

- Resource Administration
  This topic deals with the process of Resource Administration on Oracle Weblogic.
- JMS Server Creation
  This topic explains the systematic instructions to create the JMS server in the Weblogic application server.
- JMS Modules Creation
  This topic explains the systematic instructions to create the JMS Modules in the Weblogic application server.
- Subdeployment Creation
  This topic explains the systematic instructions to create the subdeployment in the Weblogic application server.
- JMS Queue Creation
  This topic explains the systematic instructions to create the JMS Queue in the Weblogic application server.
- JMS Connection Factory Creation
  This topic explains the systematic instructions to create the JMS Connection Factory in the Weblogic application server.

7.1 Resource Administration

This topic deals with the process of Resource Administration on Oracle Weblogic.

All the resources mention in the topic *Resources To be Created* are need to be created before deployment. One example for each category is explained in the following sub-topics.

- Create Data Source
  This topic explains the methods to create data sources.
- XA Enabled Data Source
  This topic explains the systematic instructions to create the XA enabled data source in the Weblogic application server.
- Non-XA Enabled Data Source
  This topic explains the systematic instructions to create the Non-XA enabled data source in the Weblogic application server.
- Scheduler Data Source configuration
  This topic gives an overview to configure Scheduler Data Source.

7.1.1 Create Data Source

This topic explains the methods to create data sources.

The method for creating data sources is explained under the following headings.

Prerequisites

To create the data source, the OCI needs to be enabled.

For this, download Oracle Instant Client and install it. The details are given below:

Table 7-1 Oracle Instant Client

Package	Download Location	Remarks
Oracle Instant Client Package	http://www.oracle.com/technetwork/database/features/instant-client/index.html	Install Oracle Instant Client in a local directory. While configuring Weblogic for Windows or Unix/Linux box, the user needs to provide the directory path where Instant Client is installed.

The user needs to do the data source configuration with the OCI driver enabled. The configurations are given below.

- Oracle Weblogic on Windows Box:
  - Set **{ORACLE_HOME}** in the environment variable.
  - Update the Environment Variable Path as `{ORACLE_HOME}/Instance Client`. This is required to load all the **.dll** files.
  - Ensure that the **ojdbc*.jar** file in `{WL_HOME}/server/lib/ojdbc*.jar` is the same as the file `{ORACLE_HOME}/jdbc/lib/ojdbc*.jar`. This is required for ensuring compatibility.
  - Update PATH in **StartWebLogic.bat** or **setDomainEnv.bat**. This must be the directory path where Oracle Instant Client is installed.
- Oracle Weblogic on Unix/Linux Box:
  - Set **{ORACLE_HOME}** in the environment variable.
  - Update the environment variable LD_LIBRARY_PATH as `{ORACLE_HOME}/lib`. This is to load all the **.so** files.
  - Ensure that the **ojdbc*.jar** file in `{WL_HOME}/server/lib/ojdbc*.jar` is the same as the file `{ORACLE_HOME}/jdbc/lib/ojdbc*.jar`. This is to ensure compatibility.
  - Update LD_LIBRARY_PATH in **StartWeblogic.sh** or **setDomainEnv.sh**. This must be the directory path where Oracle Instant Client is installed.
  - If you are still not able to load the **.so** files, then you need to update the EXTRA_JAVA_PROPERTIES by setting `Djava.library.path` as `{ORACLE_HOME}/lib` in **StartWebLogic.sh** or **setDomainEnv.sh**.
  - If the target database is Autonomous Database then configure the TNS_ADMIN in the DB client of the Application server with the Autonomous Database Wallet given by the Database Administrator.

7.1.2 XA Enabled Data Source

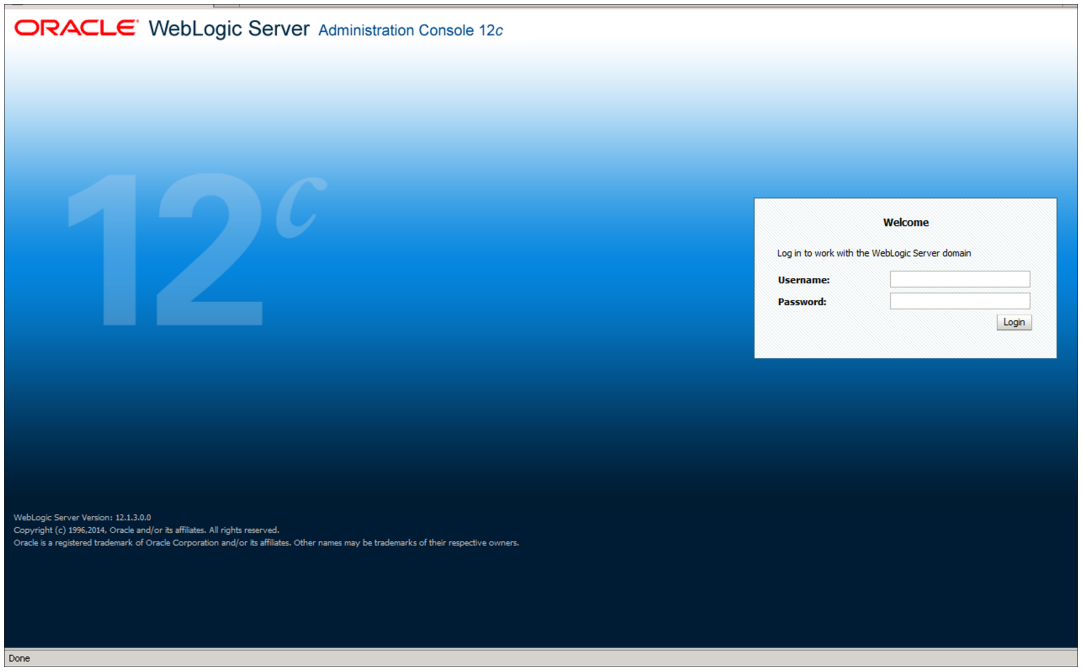
This topic explains the systematic instructions to create the XA enabled data source in the Weblogic application server.

Follow the steps given below to create the XA enabled data source for Gateway Application (MDB):

1. Start the **Administration Console** of the WebLogic Application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   The **Oracle WebLogic Server Login** screen is displayed.

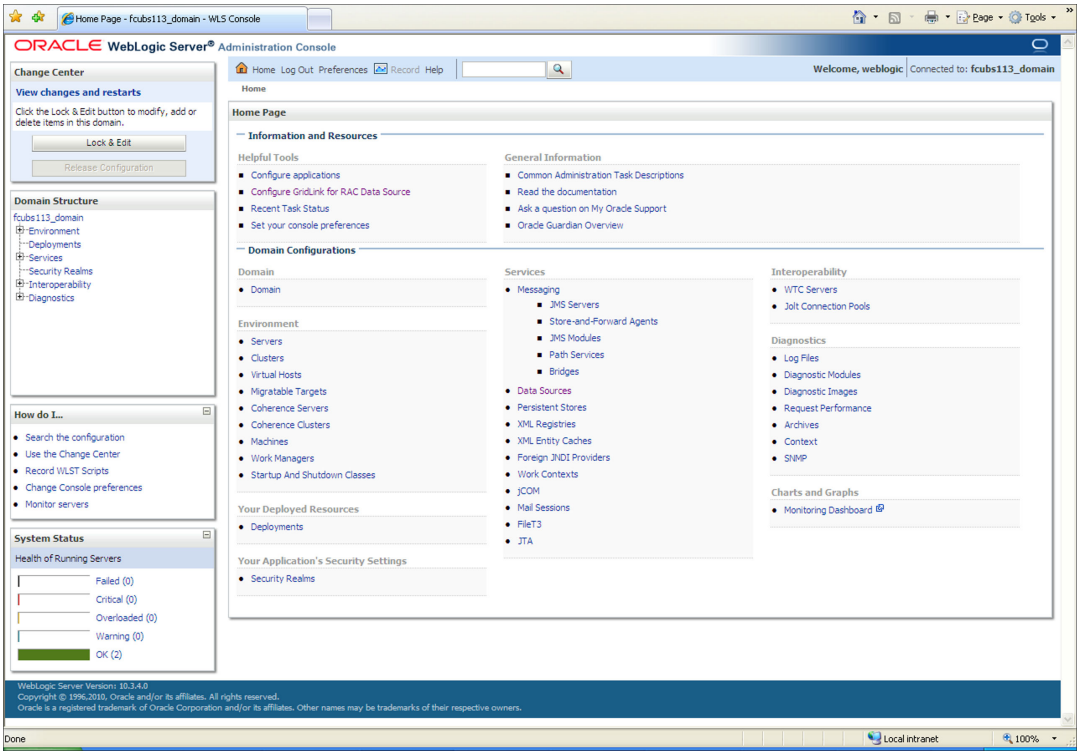
Figure 7-1 Weblogic Application Server Login



2. Specify the **Username** and **Password** in the WebLogic Server domain and click **Login**.

   The **Oracle Weblogic Server Home Page** screen is displayed.

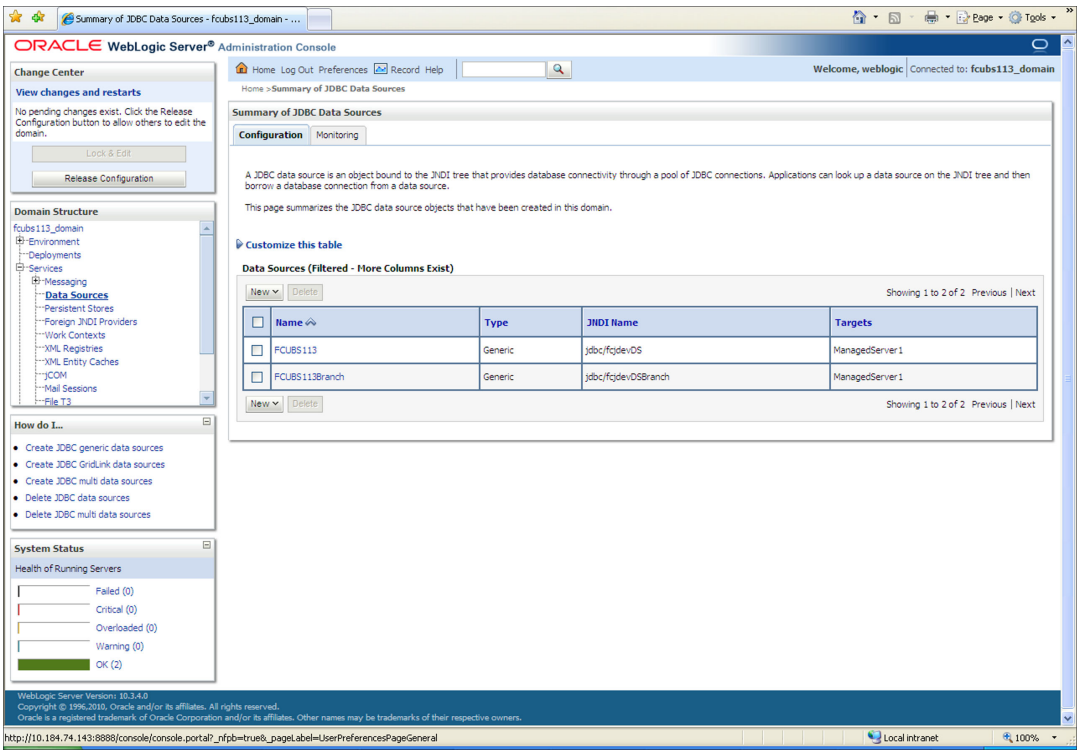
Figure 7-2 Oracle Weblogic Server Home Page



3. Click the **Lock & Edit** button under the **Change Center** section to add, modify or delete items.

   The **Summary of JDBC Data Sources** screen is displayed.

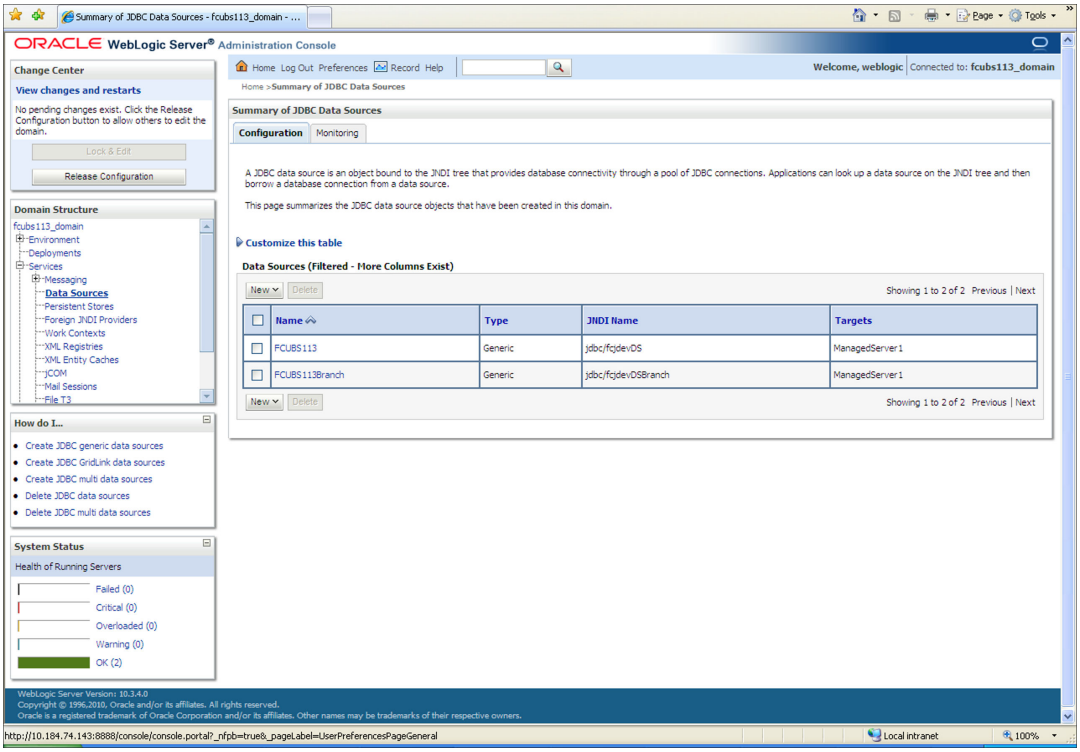
Figure 7-3 Summary of JDBC Data Sources



4. On the left pane, under **Domain Structure**, expand the node **Services** and click **Data Sources** from the list.

   The **Summary of JDBC Data Sources_Configuration** screen is displayed.

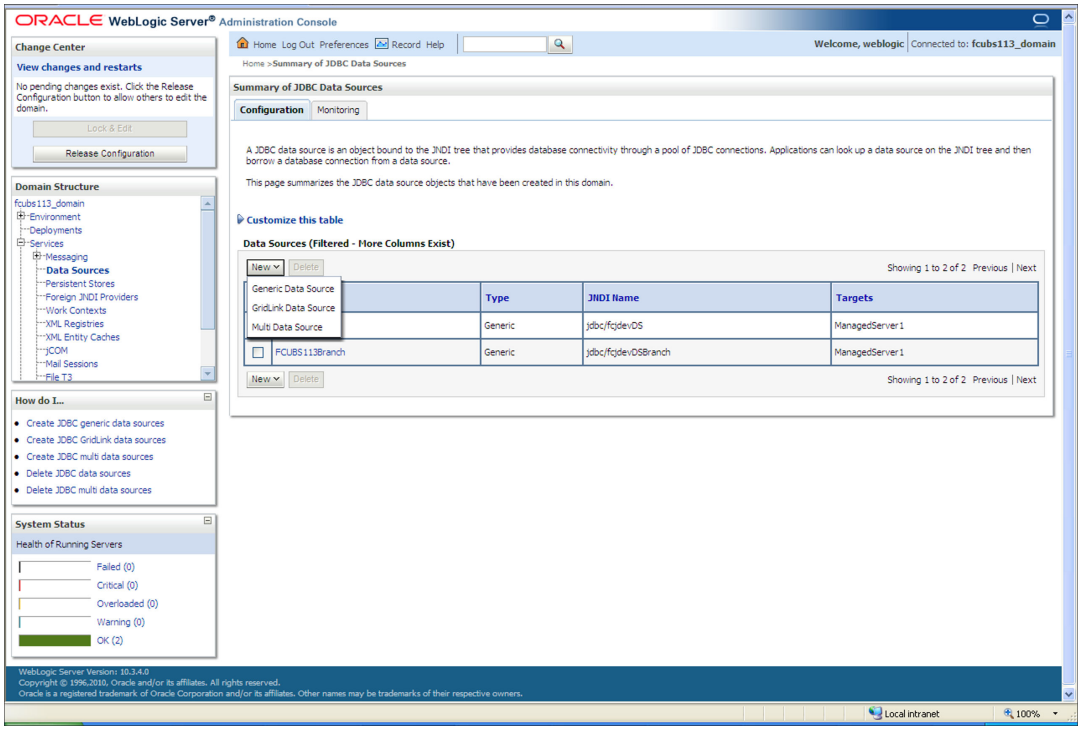
Figure 7-4 Summary of JDBC Data Sources



5. Navigate to **Data Sources** section.

   The **Summary of JDBC Data Sources_Data Sources** screen is displayed.

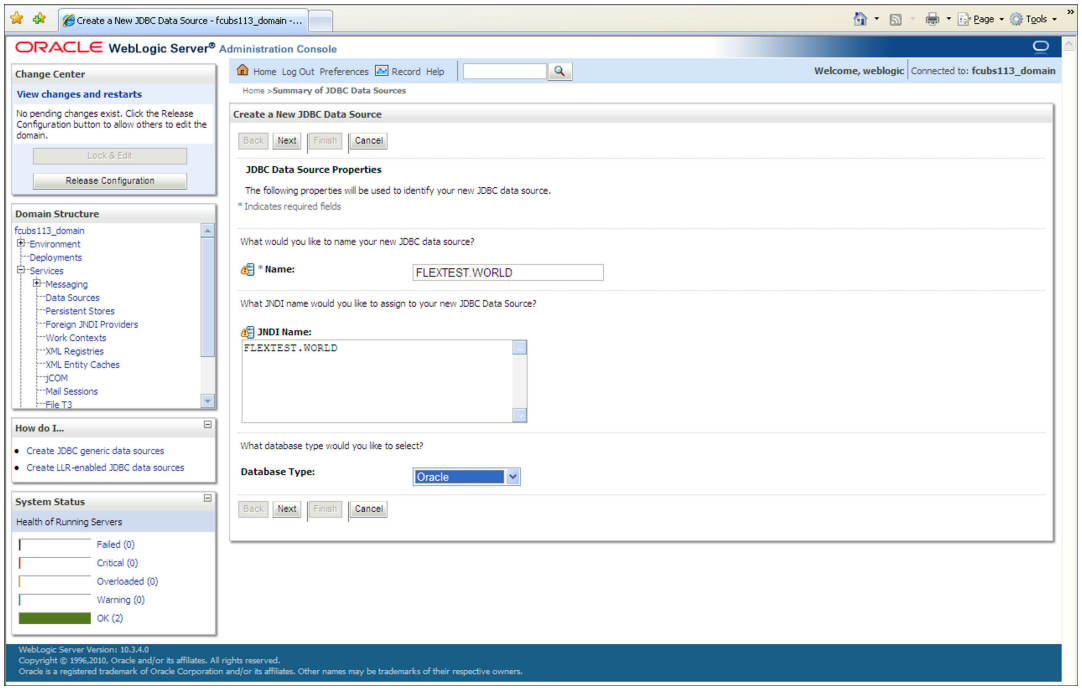
Figure 7-5 Data Sources_New_Generic Data Source



6. Click **New** to create a new data source and select **Generic Data Source** from the drop-down.

   Create a New JDBC Data Source_JDBC Data Source Properties screen is displayed.

Figure 7-6 Create a New JDBC Data Source_JDBC Data Source Properties



7. On the **Create a New JDBC Data Source** screen, specify the fields.

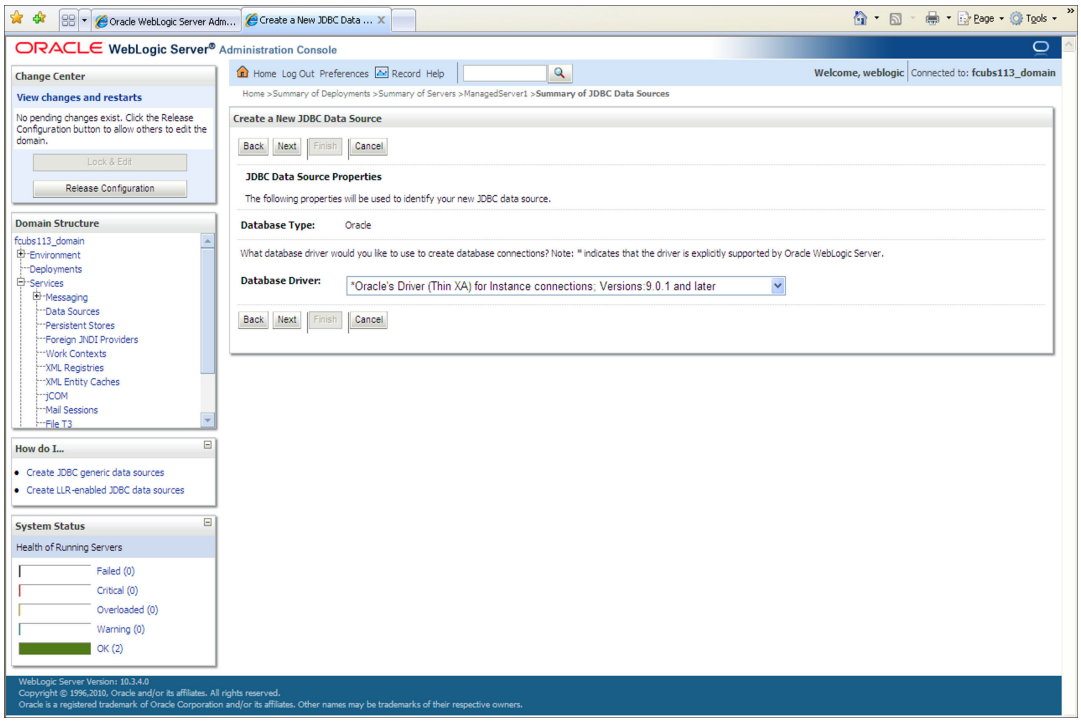
Table 7-2 Create a New JDBC Data Source

Field	Description
JDBC Datasource Name	Name of the data source.
JNDI Name	JNDI name which will be used for lookup.
Database Type	Specify the database type as Oracle from the drop-down list.

8. Click **Next** to specify **Database Driver**.

   Create a New JDBC Data Source_Database Driver screen is displayed.

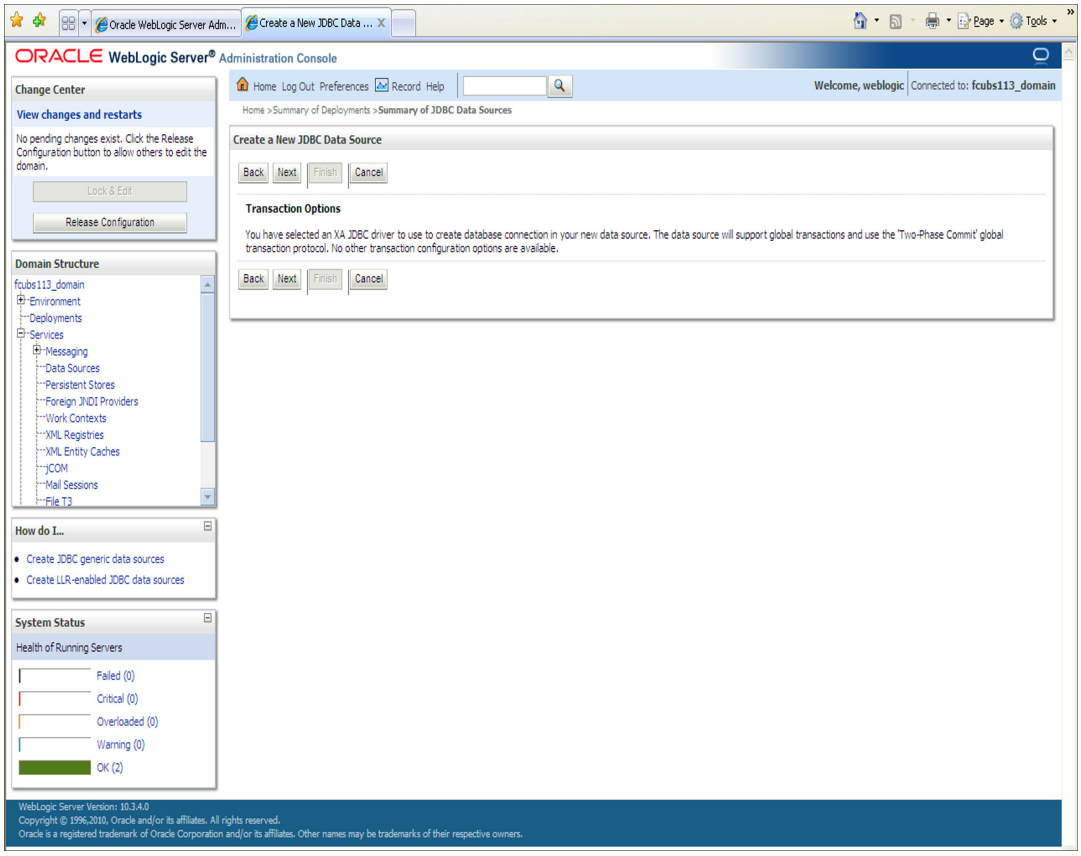
Figure 7-7 Create a New JDBC Data Source_Database Driver



9. Select the XA database driver from the drop-down list and click **Next** to specify the transaction options.

   Create a New JDBC Data Source_Transaction Options screen is displayed.

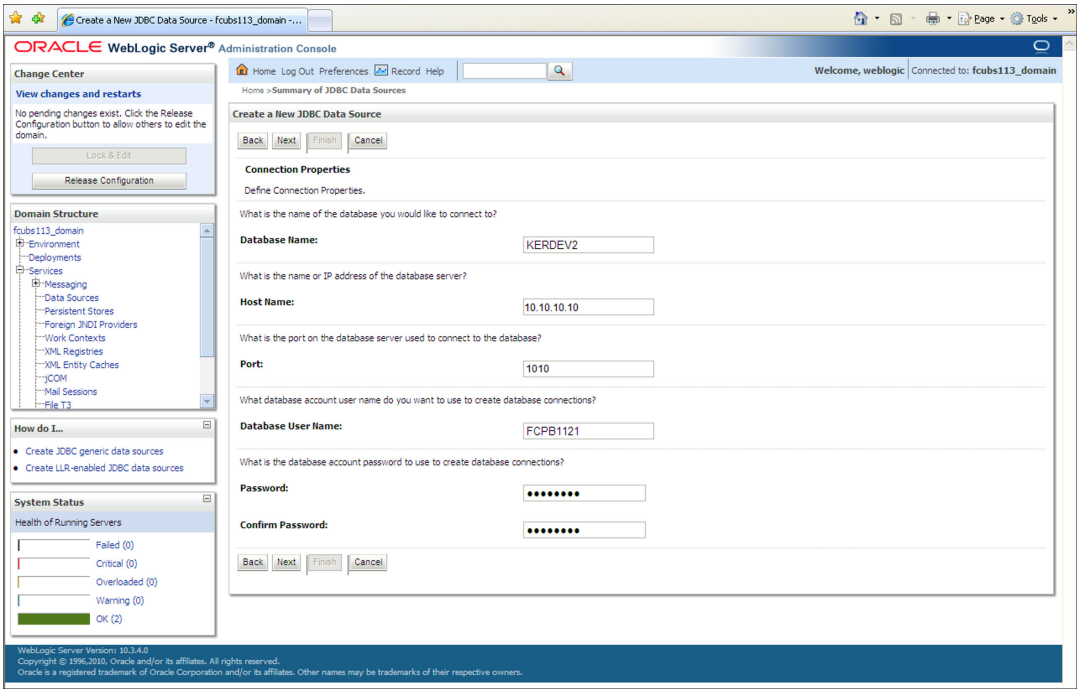
Figure 7-8 Create a New JDBC Data Source_Transaction Options



10. Click **Next** to define the connection properties. On the **Create a New JDBC Data Source_Connection Properties** screen, specify the Database Name, Host Name, Port of the database server to connect to the Database User Name, Password, and Confirm Password.

    Create a New JDBC Data Source_Connection Properties screen is displayed.

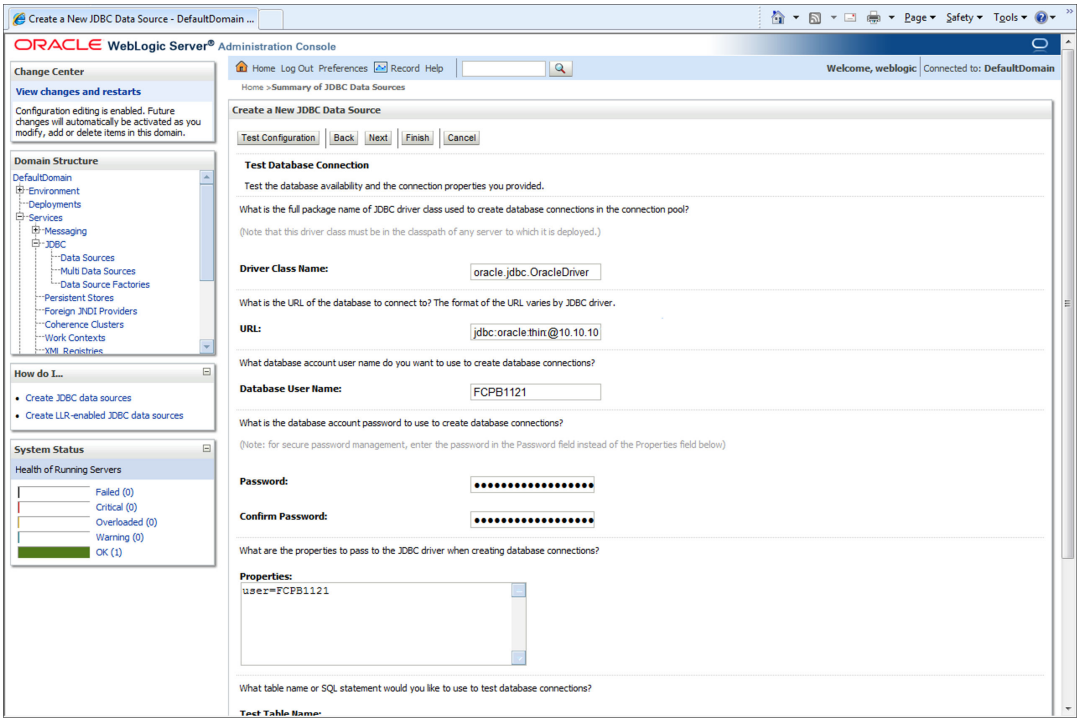
Figure 7-9 Create a New JDBC Data Source_Connection Properties



11. Click **Next**.

Create a New JDBC Data Source_Test Database Connection screen is displayed.

Figure 7-10 Create a New JDBC Data Source_Test Database Connection



12. Specify the **Driver Class Name**.

For Example: oracle.jdbc.OracleDriver.

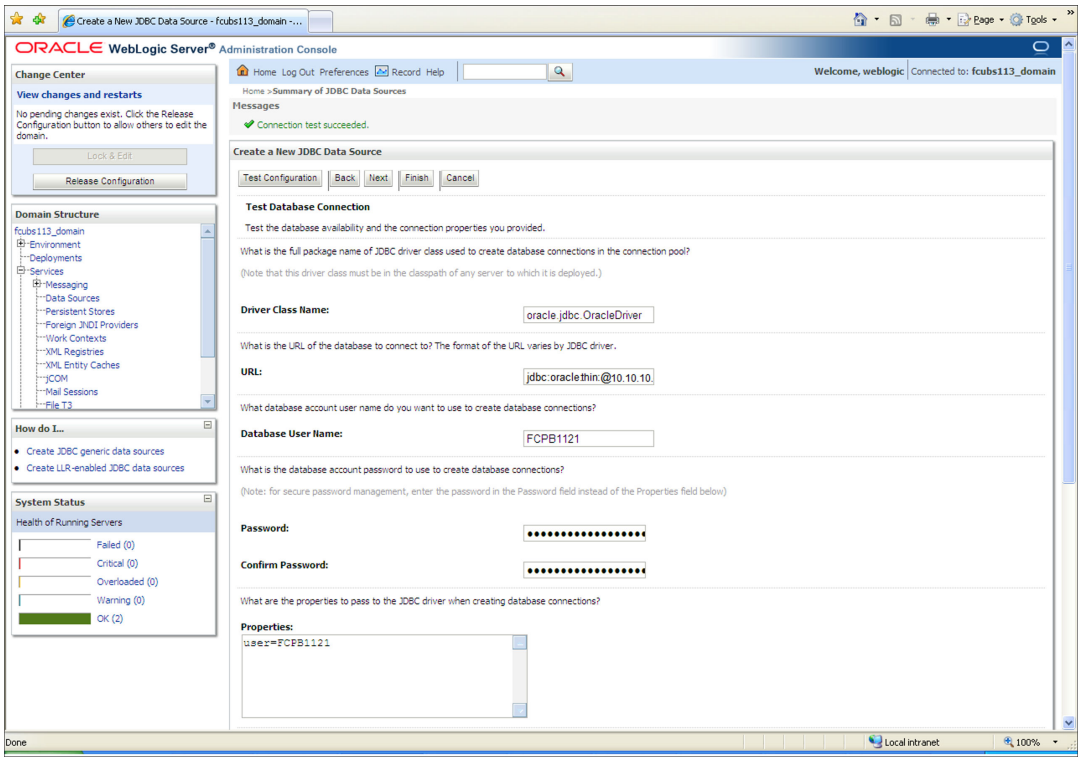
13. Specify the URL as `jdbc:oracle:thin:@10.10.10.10:1001<INSTANCE_NAME>`.
14. Specify the Database Username.

    For Example: FCPB1121
15. Specify password and confirm the password.
16. Click **Test Configuration** tab in the **Create a New JDBC Data Source** screen.

    If the connection is established successfully, the message `Connection test succeeded` is displayed.

    Create a New JDBC Data Source_Messages screen is displayed.

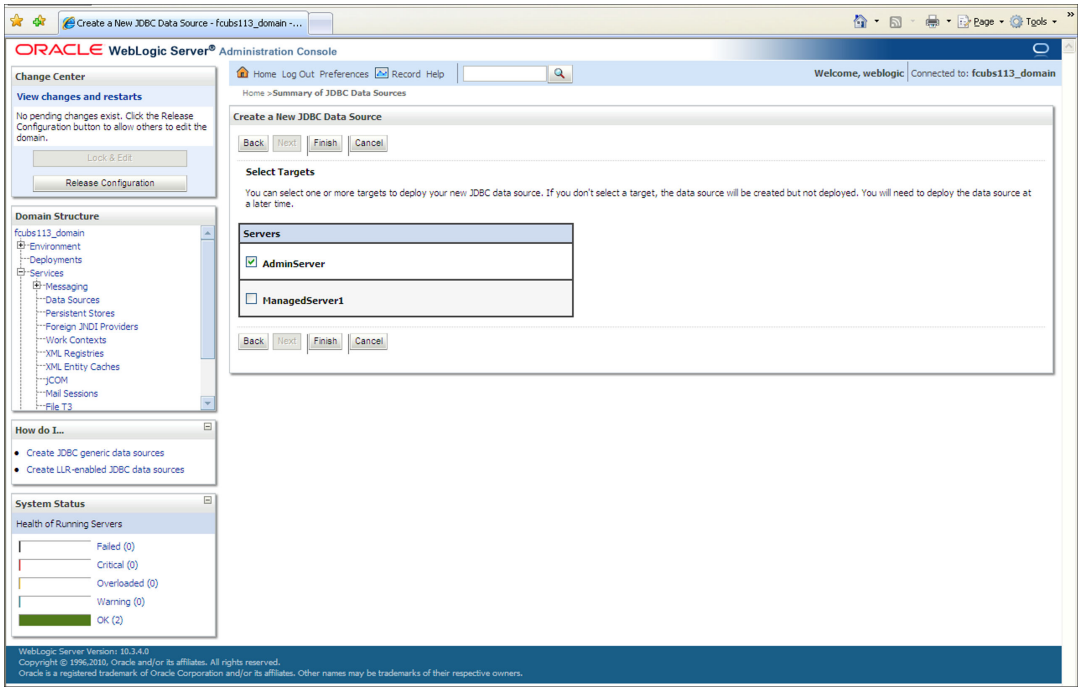
Figure 7-11 Create a New JDBC Data Source_Messages



17. Click **Next** to select targets.

    Create a New JDBC Data Source_Select Targets screen is displayed.

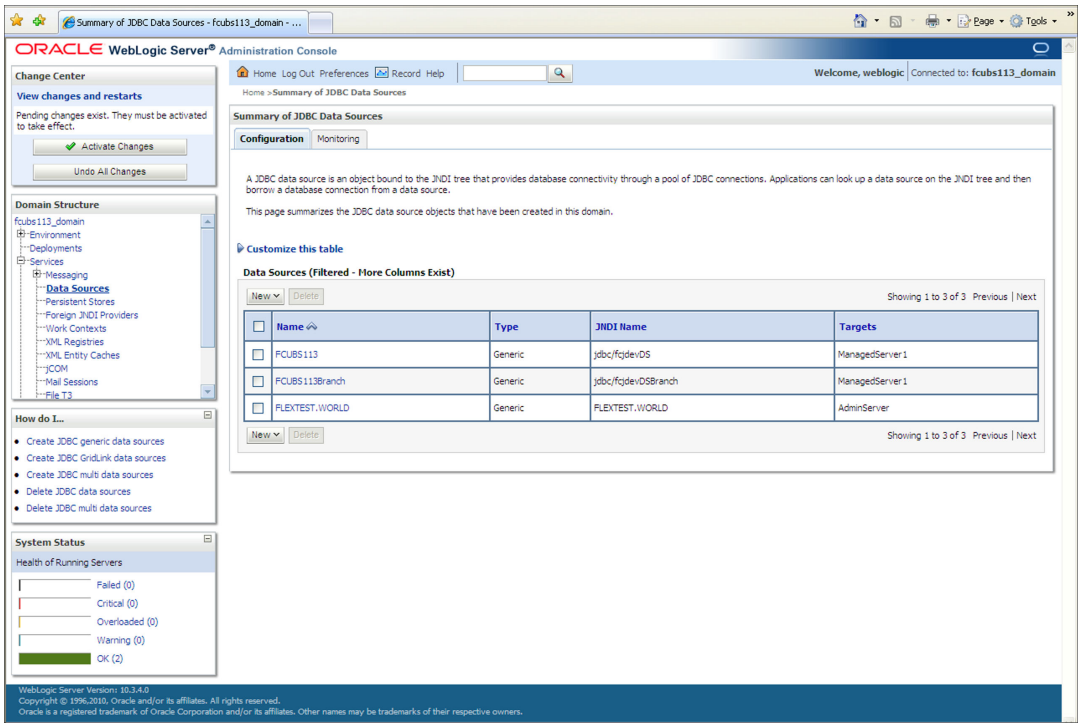
Figure 7-12 Create a New JDBC Data Source_Select Targets



18. Select the boxes against the required servers and click **Finish**.

    Summary of JDBC Data Sources_New Data Source screen is displayed.

Figure 7-13 Summary of JDBC Data Sources_New Data Source

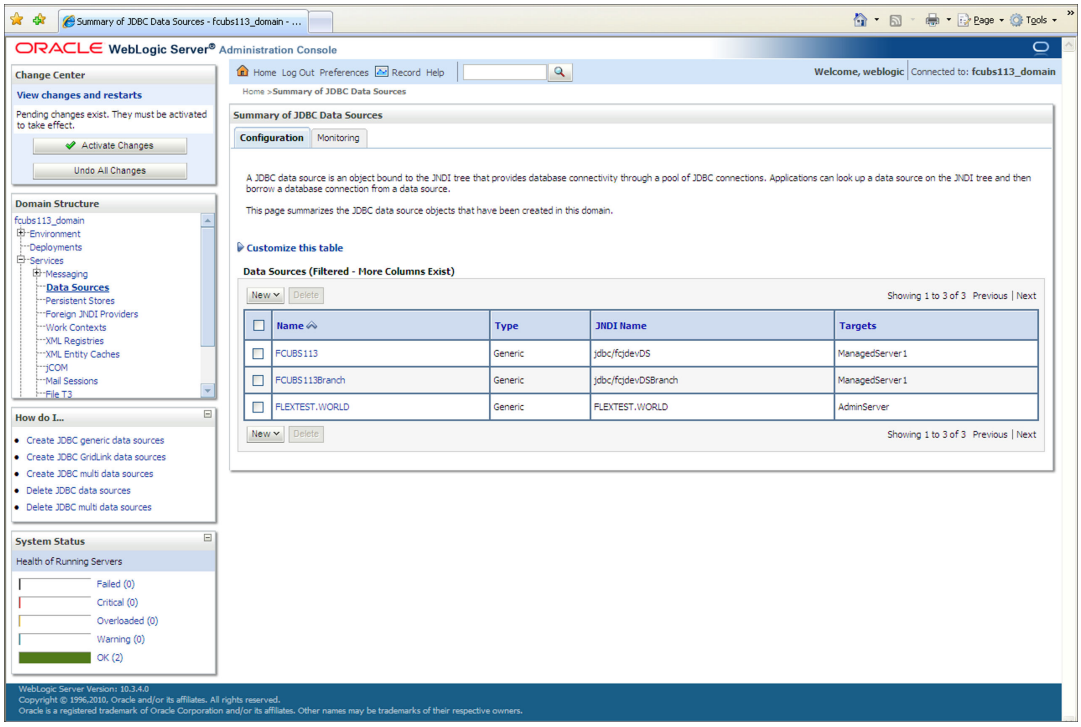


19. Click the **Activate Changes** button under the **Change Center** section of the screen.

    The message `All the changes have been activated. No restarts are necessary.` is displayed.

The **Summary of JDBC Data Sources** screen is displayed.

Figure 7-14 Change Center_Activate Changes



20. On the **Summary of JDBC Data Sources** screen, you can view the new data source created in the **Data Sources** section.

    Refer to #unique_45 for the list of XA data sources to be created.

    The new **Data Source** is created. For Example: `FLEXTEST.WORLD`

7.1.3 Non-XA Enabled Data Source

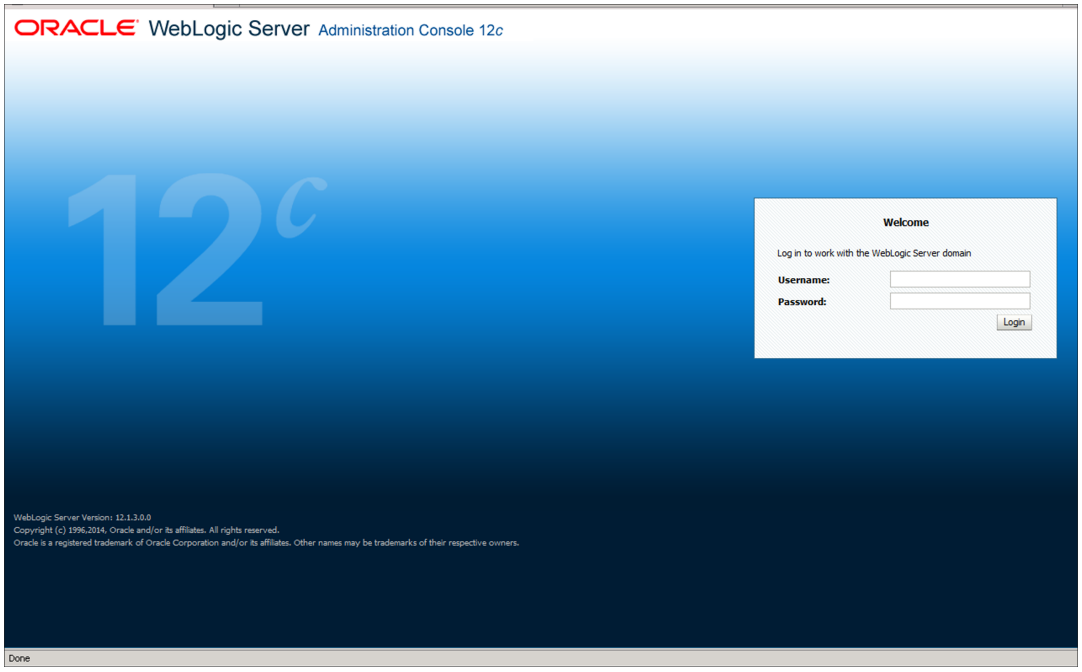
This topic explains the systematic instructions to create the Non-XA enabled data source in the Weblogic application server.

Follow the steps given below to create the XA enabled data source for Gateway Application (MDB):

1. Start the **Administration Console** of the WebLogic Application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   The **Oracle WebLogic Server Login** screen is displayed.

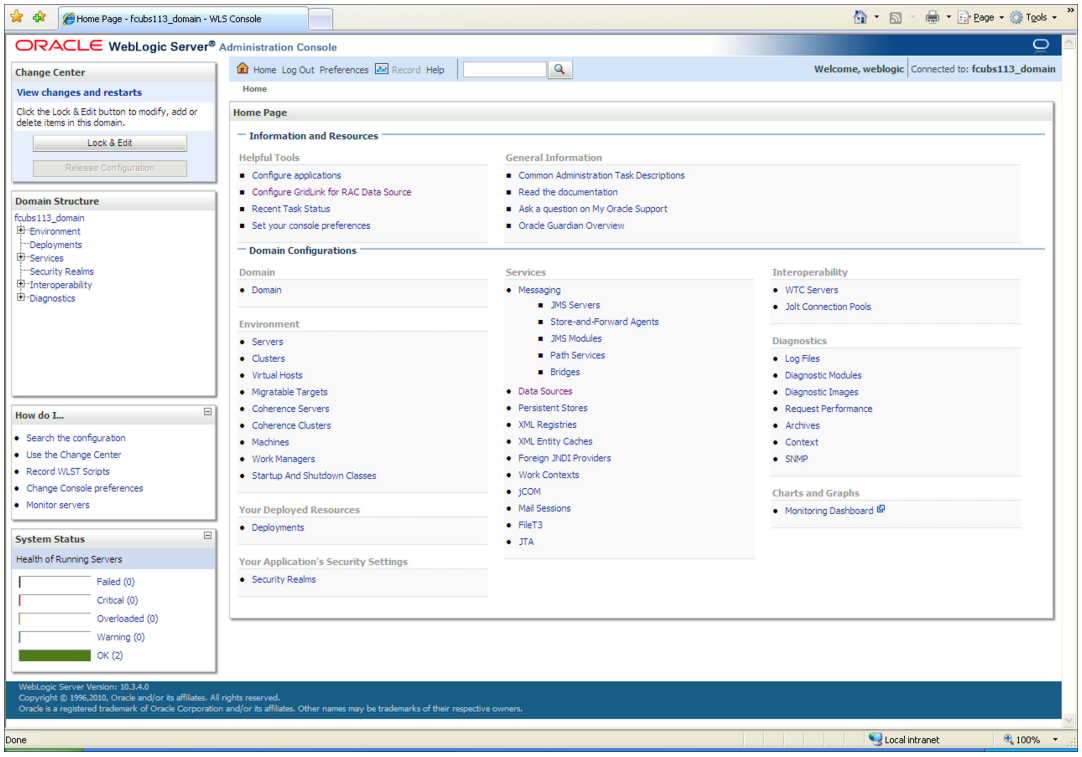
Figure 7-15 Weblogic Application Server Login



2. Specify the **Username** and **Password** in the WebLogic Server domain and click **Login**.

   The **Oracle Weblogic Server Home Page** screen is displayed.

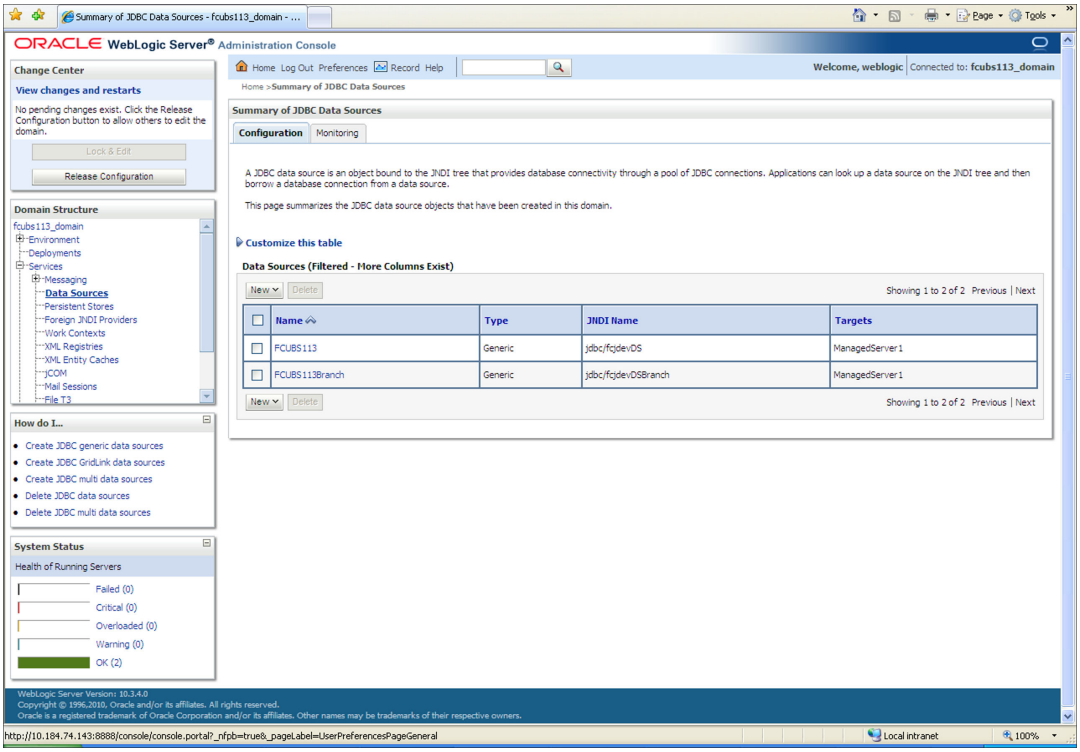
Figure 7-16 Oracle Weblogic Server Home Page



3. Click the **Lock & Edit** button under the **Change Center** section to add, modify or delete items.

The **Summary of JDBC Data Sources** screen is displayed.

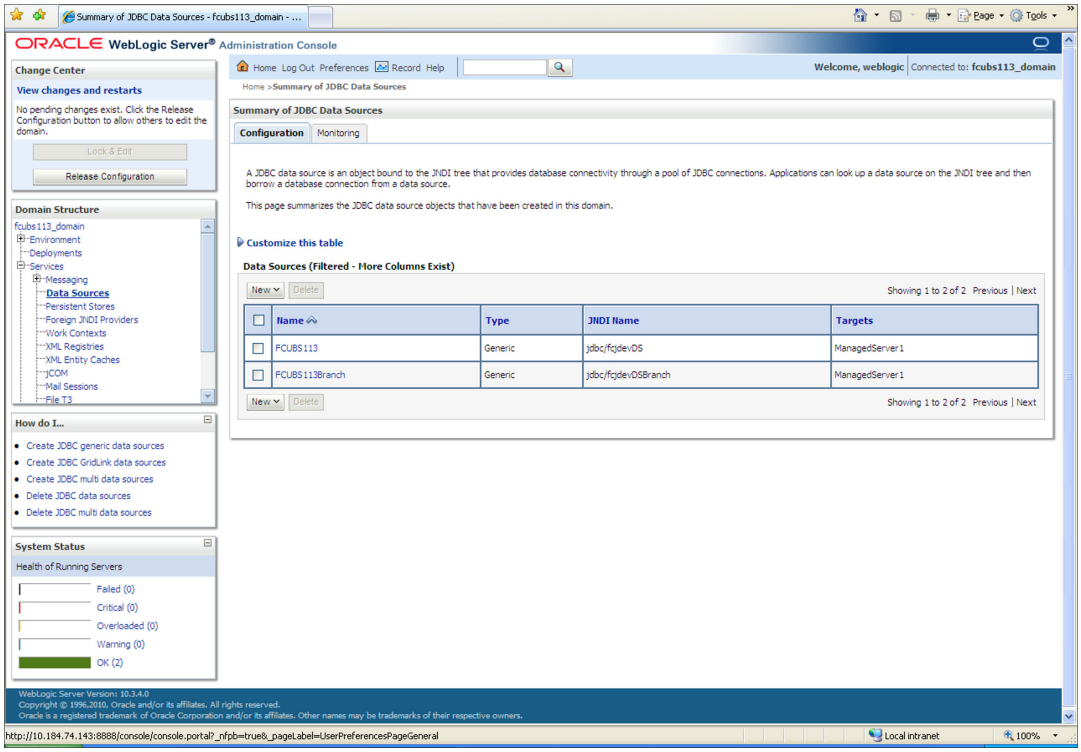
Figure 7-17 Summary of JDBC Data Sources



4. On the left pane, under **Domain Structure**, expand the node **Services** and click **Data Sources** from the list.

   The **Summary of JDBC Data Sources_Configuration** screen is displayed.

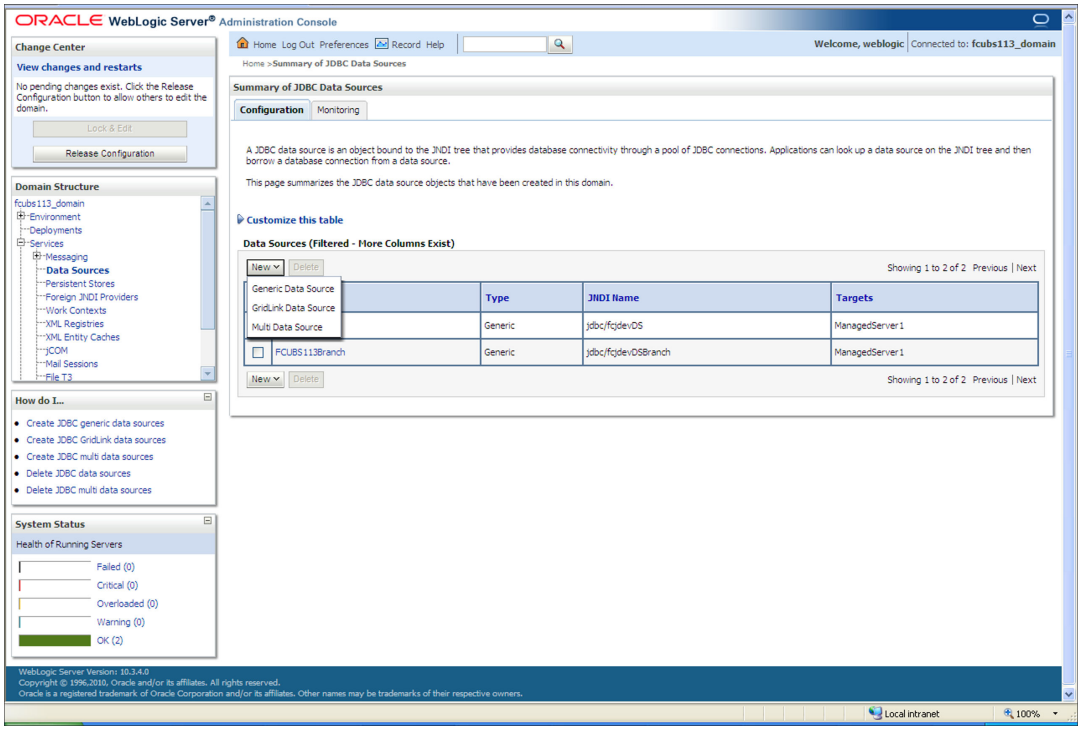
Figure 7-18 Summary of JDBC Data Sources



5. Navigate to **Data Sources** section.

   The **Summary of JDBC Data Sources_Data Sources** screen is displayed.

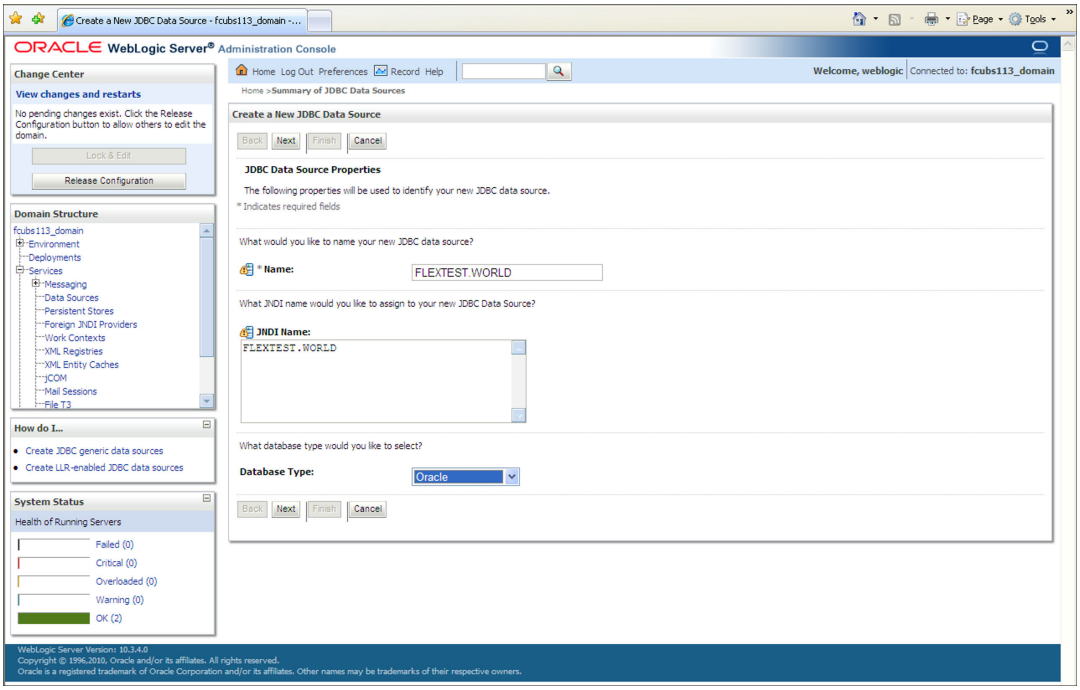
Figure 7-19 Data Sources_New_Generic Data Source



6. Click **New** to create a new data source and select **Generic Data Source** from the drop-down.

   Create a New JDBC Data Source_JDBC Data Source Properties screen is displayed.

Figure 7-20 Create a New JDBC Data Source_JDBC Data Source Properties



7. On the **Create a New JDBC Data Source** screen, specify the fields.

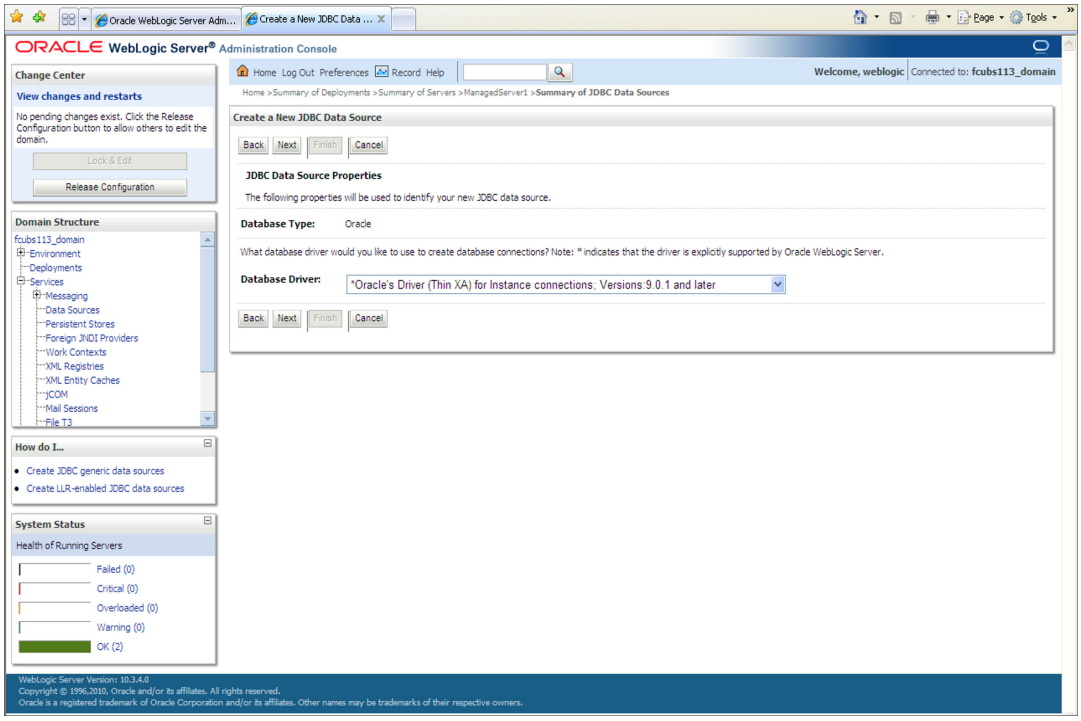
Table 7-3 Create a New JDBC Data Source

Field	Description
JDBC Datasource Name	Name of the data source.
JNDI Name	JNDI name which will be used for lookup.
Database Type	Specify the database type as Oracle from the drop-down list.

8. Click **Next** to specify **Database Driver**.

   Create a New JDBC Data Source_Database Driver screen is displayed.

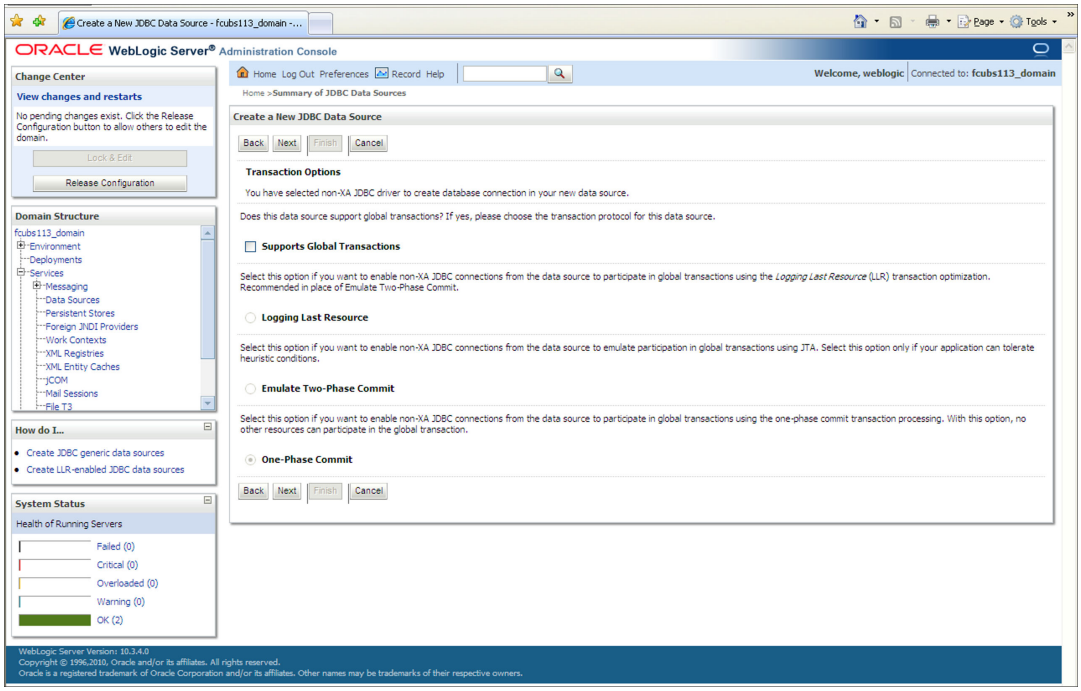
Figure 7-21 Create a New JDBC Data Source_Database Driver



9. Select the Non-XA database driver from the drop-down list and click **Next** to specify the transaction options.

   Create a New JDBC Data Source_Transaction Options screen is displayed.

Figure 7-22 Create a New JDBC Data Source_Transaction Options for Non-XA

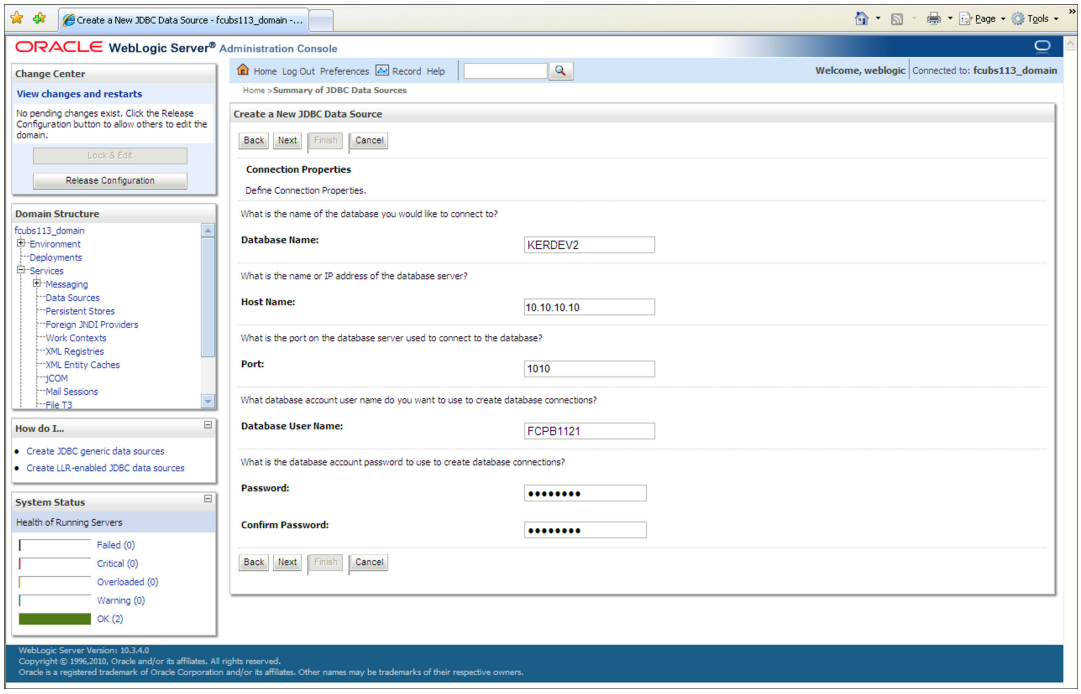


10. Select the option **Logging Last Resource** if you want to enable non-XA JDBC connections from the data source to participate in global transactions.

11. Select **Logging Last Resource** then uncheck **Support Global Transactions**.
12. Click **Next** to define the connection properties. On the **Create a New JDBC Data Source_Connection Properties** screen, specify the Database Name, Host Name, Port of the database server to connect to the Database User Name, Password, and Confirm Password.

    Create a New JDBC Data Source_Connection Properties screen is displayed.

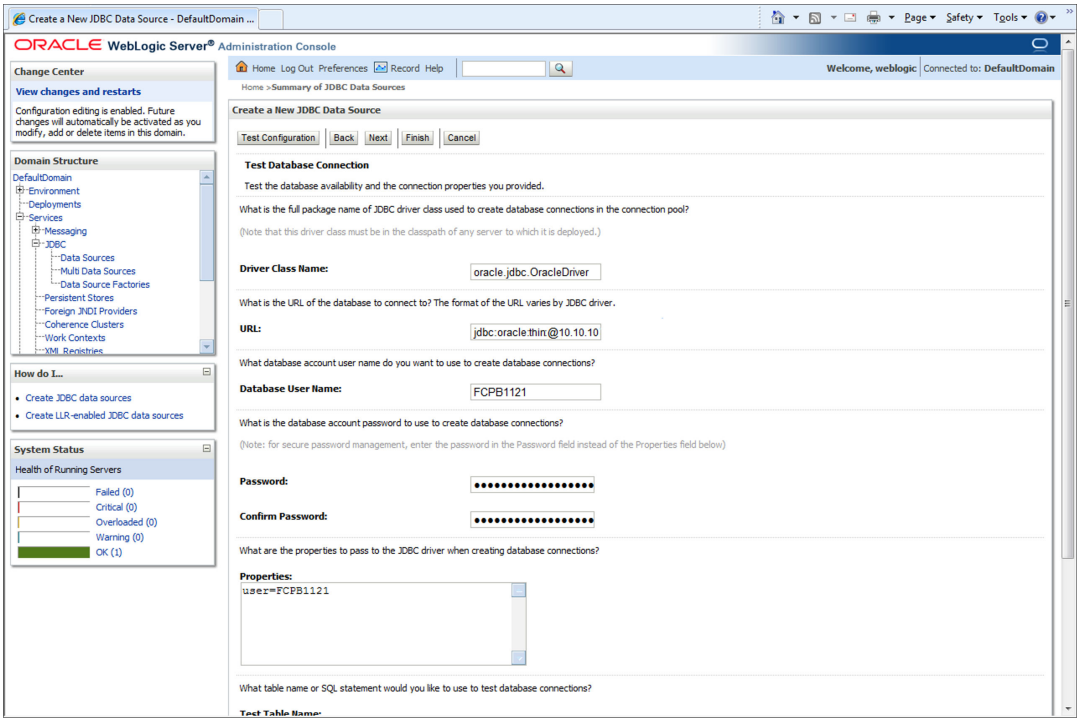
Figure 7-23 Create a New JDBC Data Source_Connection Properties



13. Click **Next**.

    Create a New JDBC Data Source_Test Database Connection screen is displayed.

Figure 7-24 Create a New JDBC Data Source_Test Database Connection



14. Specify the **Driver Class Name**.

    For Example: oracle.jdbc.OracleDriver.

15. Specify the URL.

    The default URL is `jdbc:oracle:thin:@10.10.10.10:1001<INSTANCE_NAME>` and change the default URL to `jdbc:oracle:oci:@10.10.10.10:1010:<INSTANCE_NAME>`.

16. Specify the Database Username.

    For Example: testdb

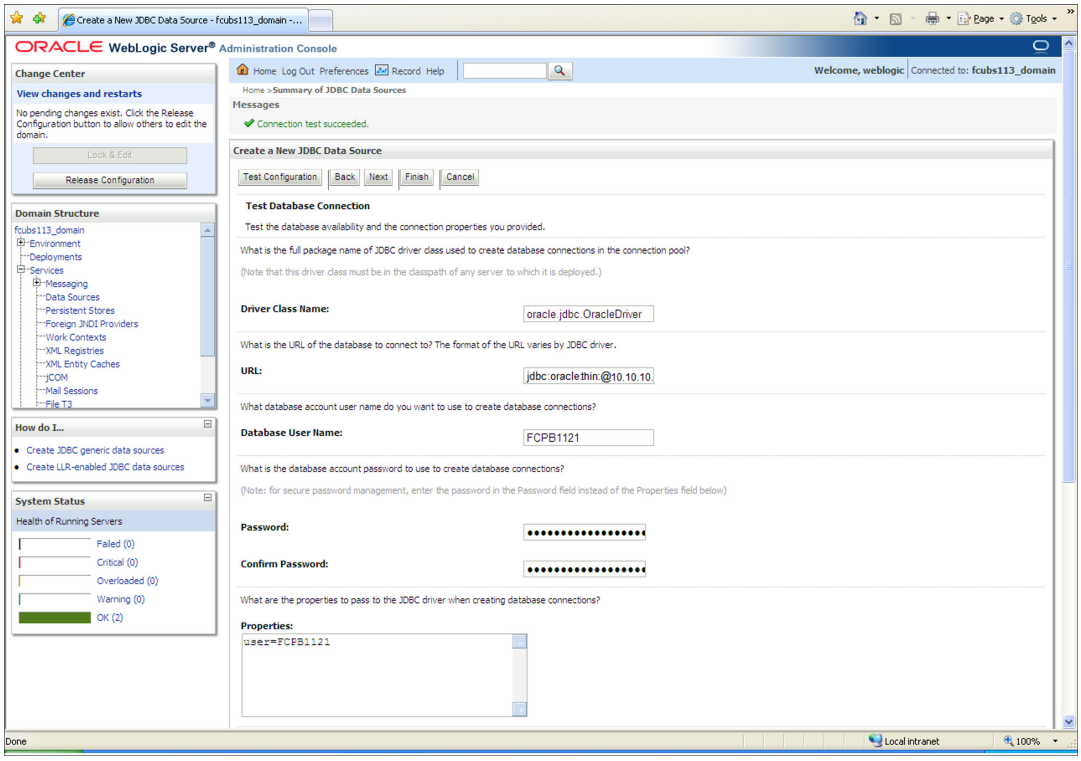
17. Specify password and confirm the password.

18. Click **Test Configuration** tab in the **Create a New JDBC Data Source** screen.

    If the connection is established successfully, the message `Connection test succeeded` is displayed.

    Create a New JDBC Data Source_Messages screen is displayed.

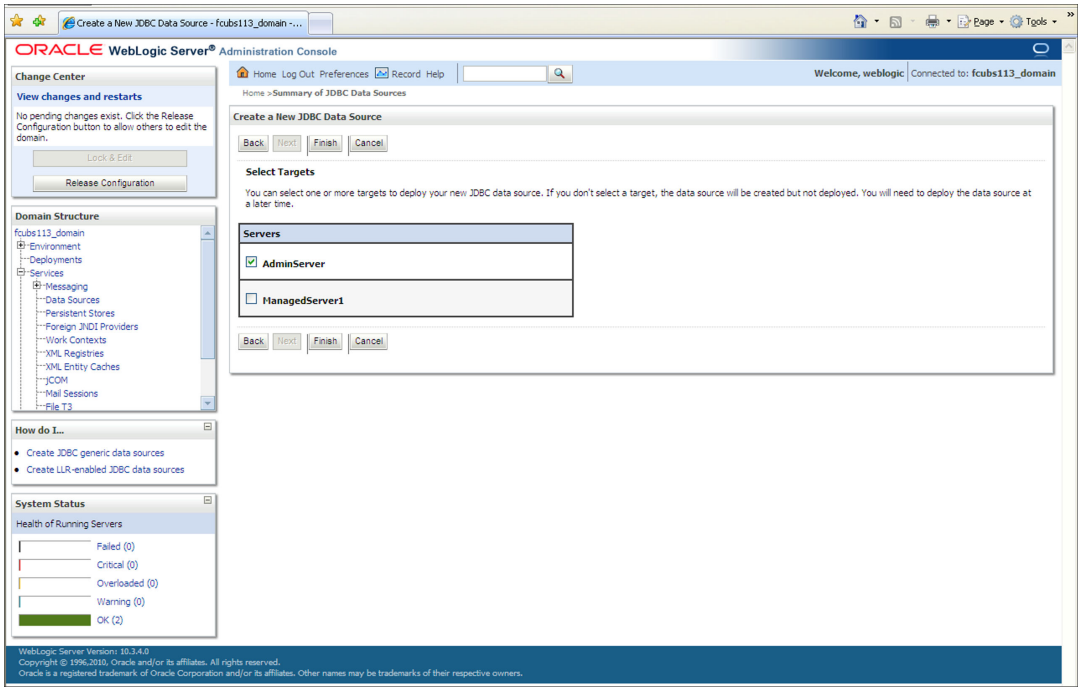
Figure 7-25 Create a New JDBC Data Source_Messages



19. Click **Next** to select targets.

    Create a New JDBC Data Source_Select Targets screen is displayed.

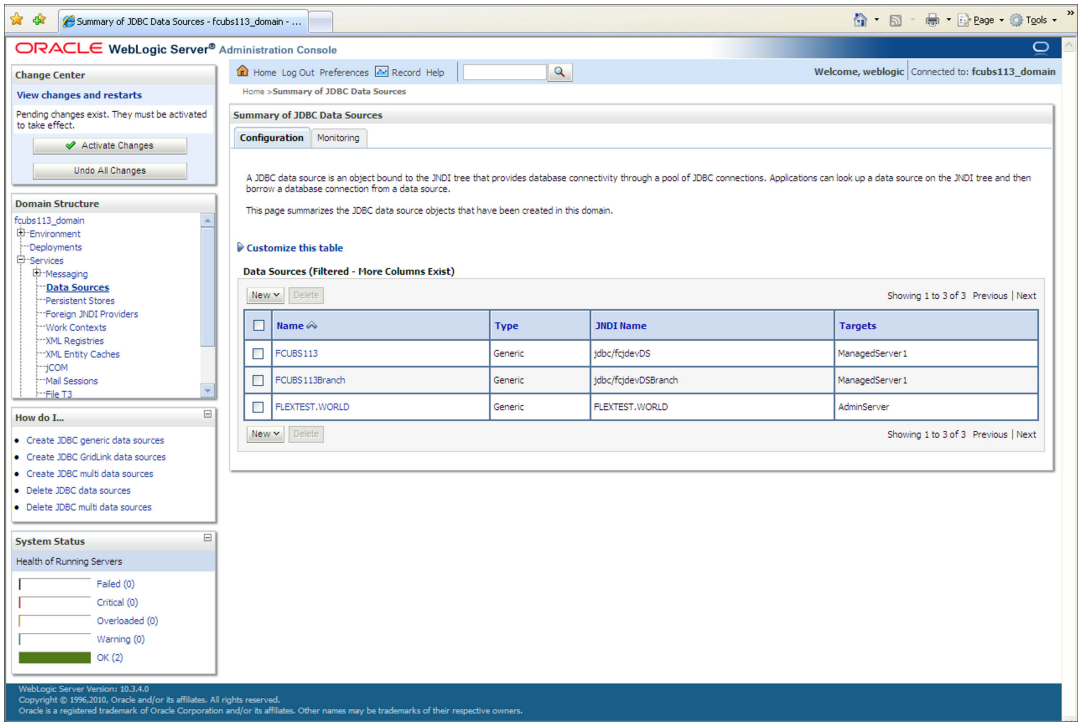
Figure 7-26 Create a New JDBC Data Source_Select Targets



20. Select the boxes against the required servers and click **Finish**.

    Summary of JDBC Data Sources_New Data Source screen is displayed.

Figure 7-27 Summary of JDBC Data Sources_New Data Source

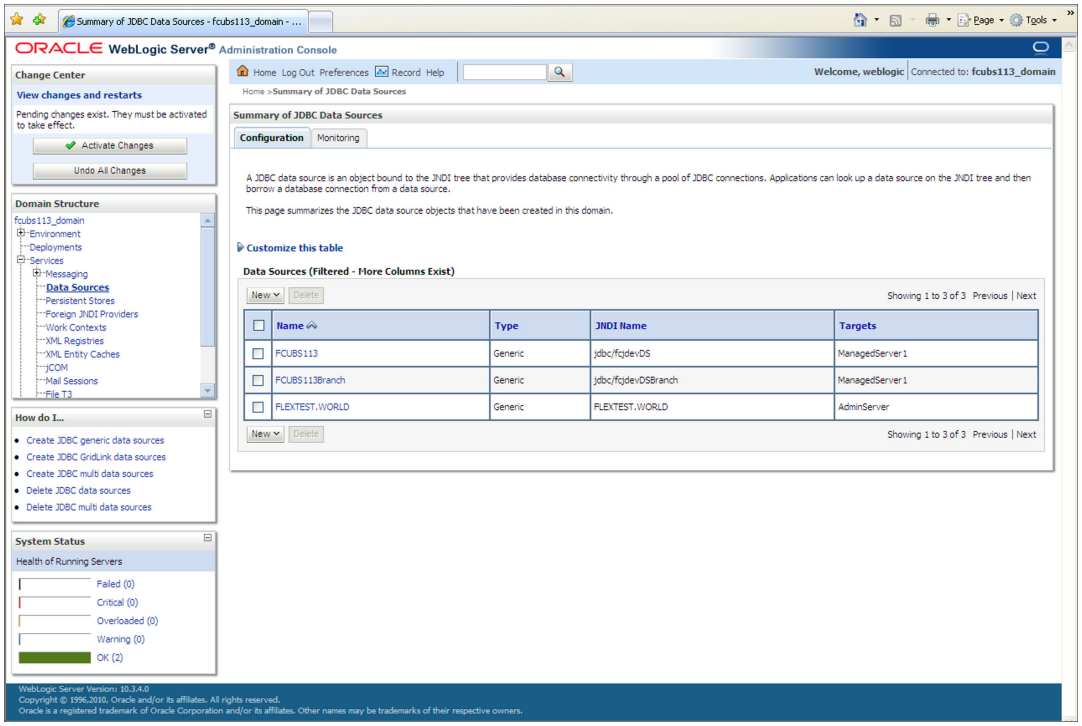


21. Click the **Activate Changes** button under the **Change Center** section of the screen.

    The message `All the changes have been activated. No restarts are necessary.` is displayed.

    The **Summary of JDBC Data Sources** screen is displayed.

Figure 7-28 Change Center_Activate Changes



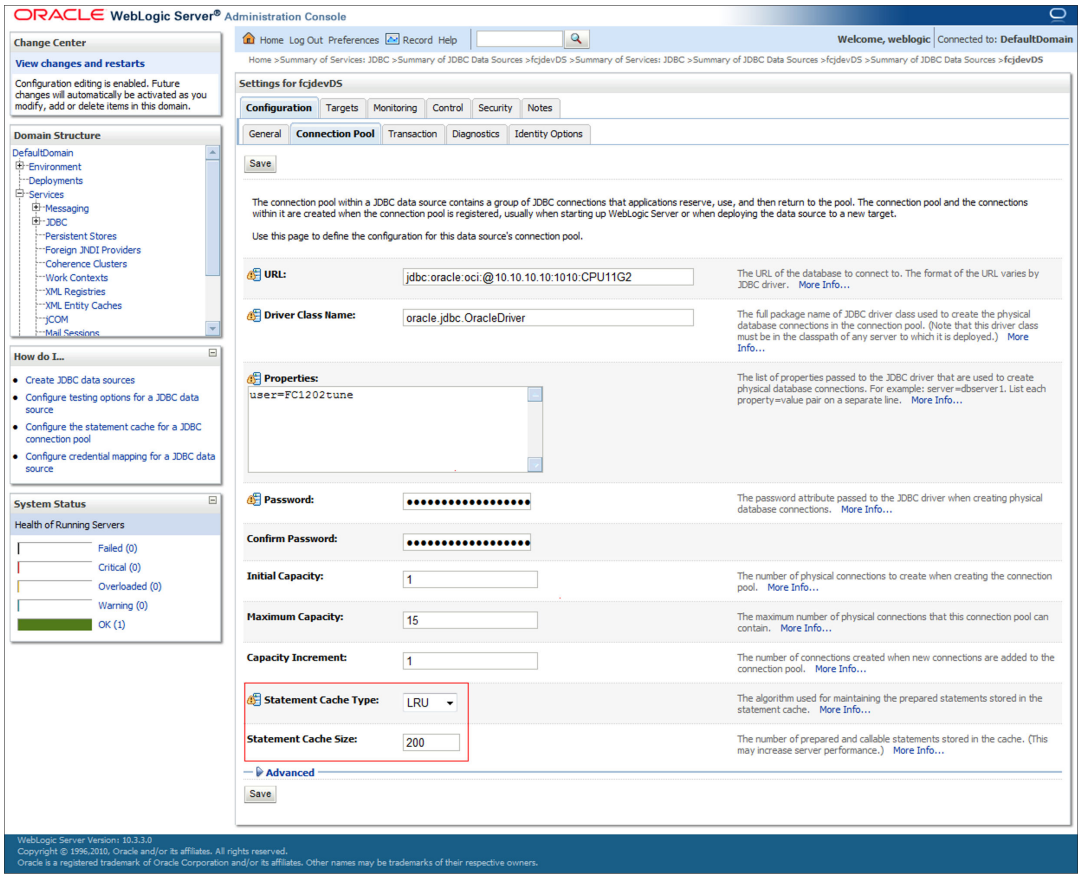
22. On the **Summary of JDBC Data Sources** screen, you can view the new data source created in the **Data Sources** section.

    The new **Data Source** is created. **For Example**: `FCISDS`

23. Click on any of the Data Sources(fcjdevDS) created, and then click the **Connection Pool** tab.

    Settings for fcjdevDS_Connection Pool screen is displayed.

Figure 7-29 Settings for fcjdevDS_Connection Pool



24. On the **Settings for fcjdevDS_Connection Pool** screen, select the **Statement Cache Type** as **LRU** from the drop-down list.
25. Specify the statement cache size as **200**.
26. Click **Save**.

Note:

- You need to create another data source for Oracle FCIS with the JNDI name **<Non-XA FCIS HOST JNDI name>_ASYNC**. **For Example**: if the Oracle FCIS HOST Non XA data source JNDI name is **jdbc/fcjdevDS**, then you need to create another data source for FCIS with the JNDI name **jdbc/fcjdevDS_ASYNC**.
- While creating a branch using the **Branch Parameters Maintenance (STDBRANC)** screen, if you have created a data source for the branch, then you need to create a corresponding ASYNC data source with the JNDI name **<Non-XA FCIS BRANCH JNDI name>_ASYNC**.

7.1.4 Scheduler Data Source configuration

This topic gives an overview to configure Scheduler Data Source.

Scheduler Data Source configuration

For all the LOB and SMS schema created for **FCIS**, equivalent XA data sources are required for Scheduler with Jndi name as **jndi name of LOB/SMS schema+_XA** (Standard naming convention).

Example 7-1 FCIS And Scheduler Data Source configuration

If there are three LOB schema's for FCIS with below jndi names,

- jdbc/BR1204R1
- jdbc/EN1204R1
- jdbc/AMC1204R1

Refer the table for the equivalent XA data sources for Scheduler.

Table 7-4 FCIS And Scheduler Data Source configuration

LOB schemas for FCIS	XA Data Sources for Scheduler	Jndi Name for Scheduler
jdbc/BR1204R1	BR1204R1_XA	jdbc/BR1204R1_XA
jdbc/EN1204R1	EN1204R1_XA	jdbc/EN1204R1_XA
jdbc/AMC1204R1	AMC1204R1_XA	jdbc/AMC1204R1_XA

7.2 JMS Server Creation

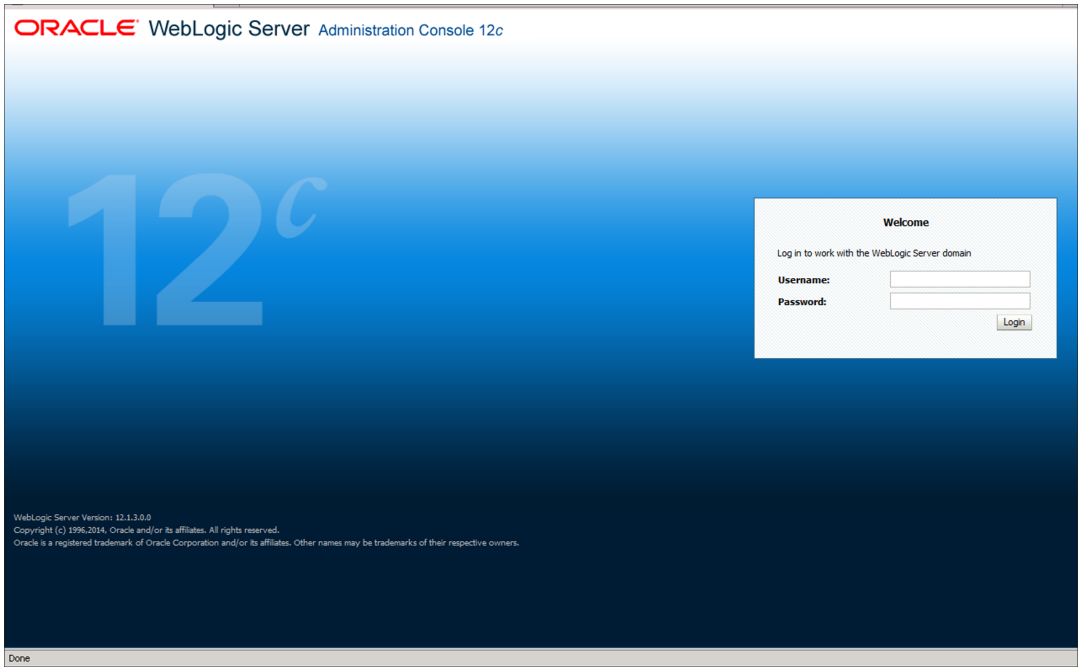
This topic explains the systematic instructions to create the JMS server in the Weblogic application server.

To create the JMS server, follow the steps given below:

1. Start the **Administration Console** of the WebLogic Application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   The **Oracle WebLogic Server Login** screen is displayed.

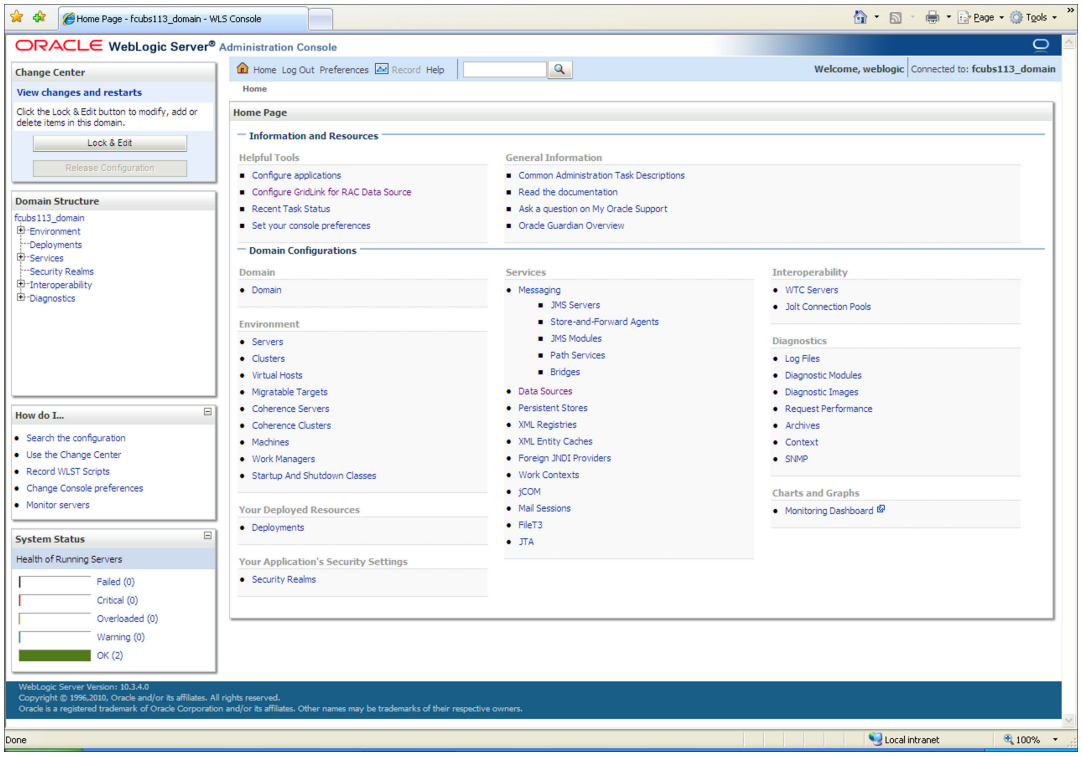
Figure 7-30 Weblogic Application Server Login



2. Specify the **Username** and **Password** in the WebLogic Server domain and click **Login**.

   The **Oracle Weblogic Server Home Page** screen is displayed.

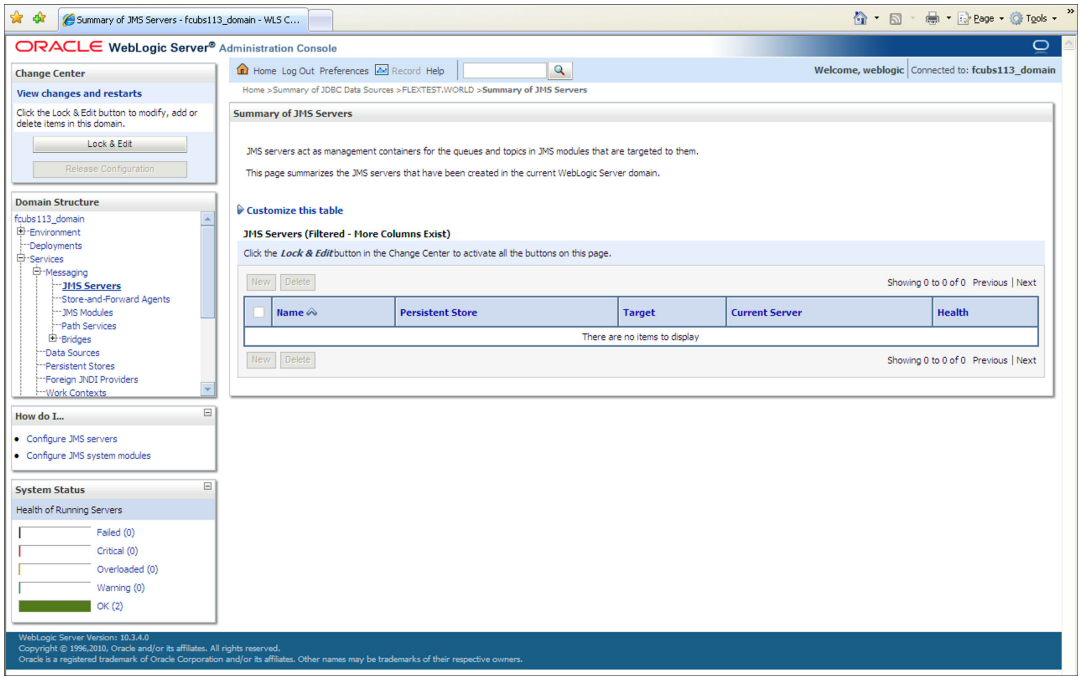
Figure 7-31 Oracle Weblogic Server Home Page



3. In the **Domain Structure**, expand the node **Services** and **Messaging**, and click **JMS Servers** from the list.

The **Summary of JMS Servers** screen is displayed.

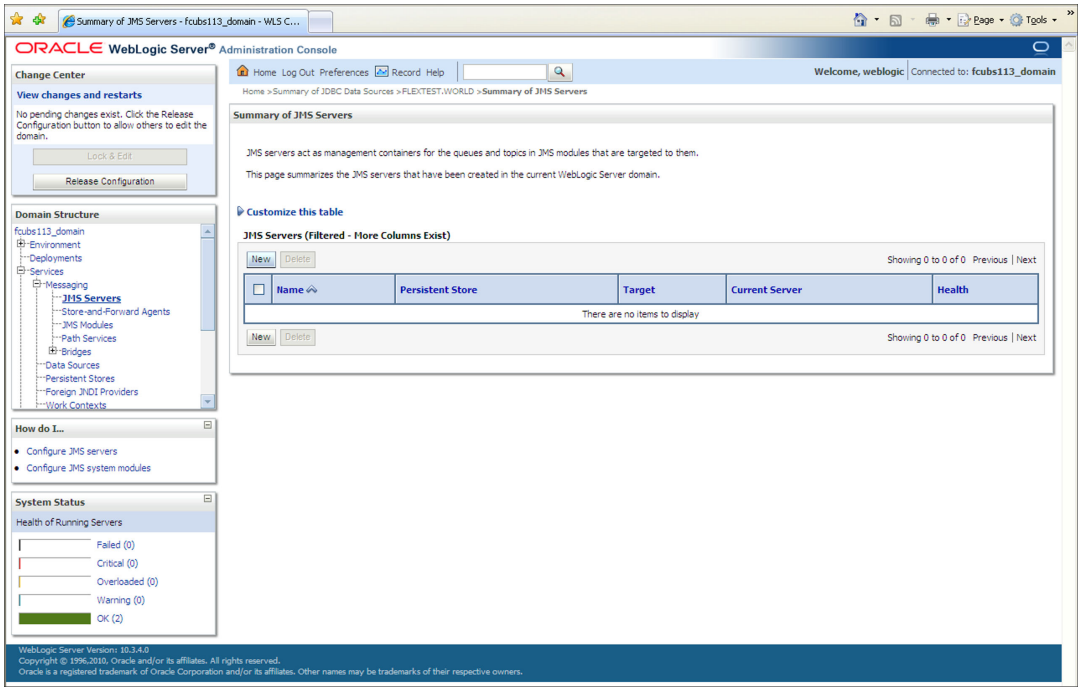
Figure 7-32 Domain Structure_Services_Messaging_JMS Servers



4. Click the **Lock & Edit** button in the **Change Center** to add, modify or delete items by acivating all the buttons on this screen.

   The **Summary of JMS Servers** screen is displayed with all the buttons enabled to edit.

Figure 7-33 Click Lock and Edit_Summary of JMS Servers



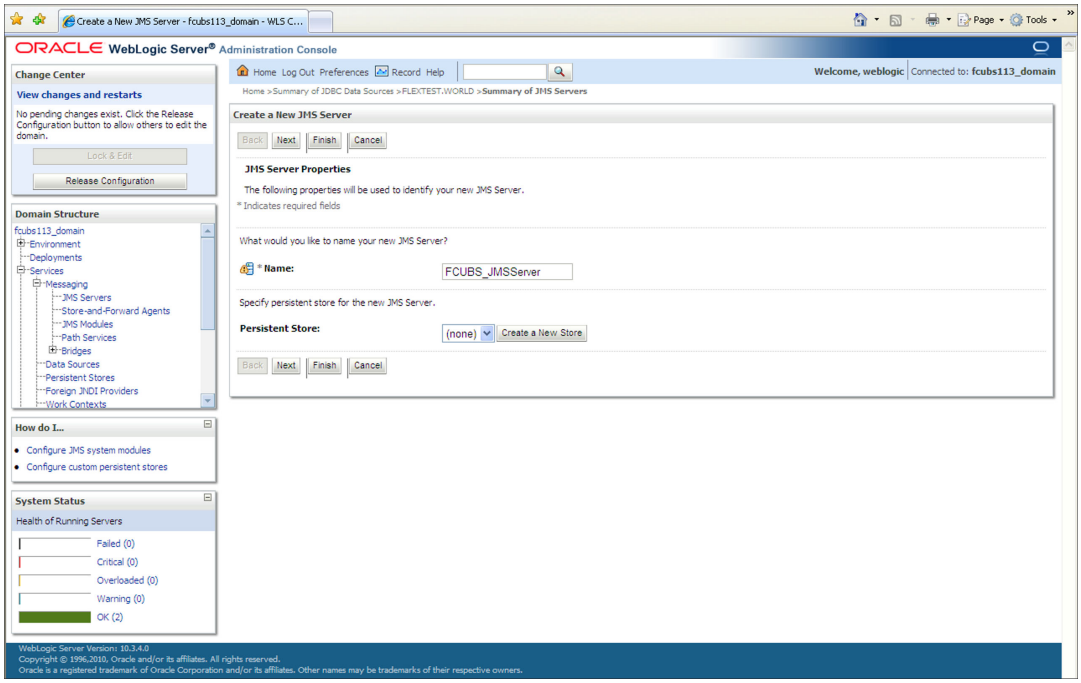
5. Navigate to **JMS Servers** section.

   The **Summary of JMS Servers_JMS Servers** screen is displayed.

6. Click **New**.

   Create a New JMS Server screen is displayed.

Figure 7-34 Create a New JMS Server



7. On **Create a New JMS Server** screen, specify the fields.

   For more information on fields, refer to the field description table.

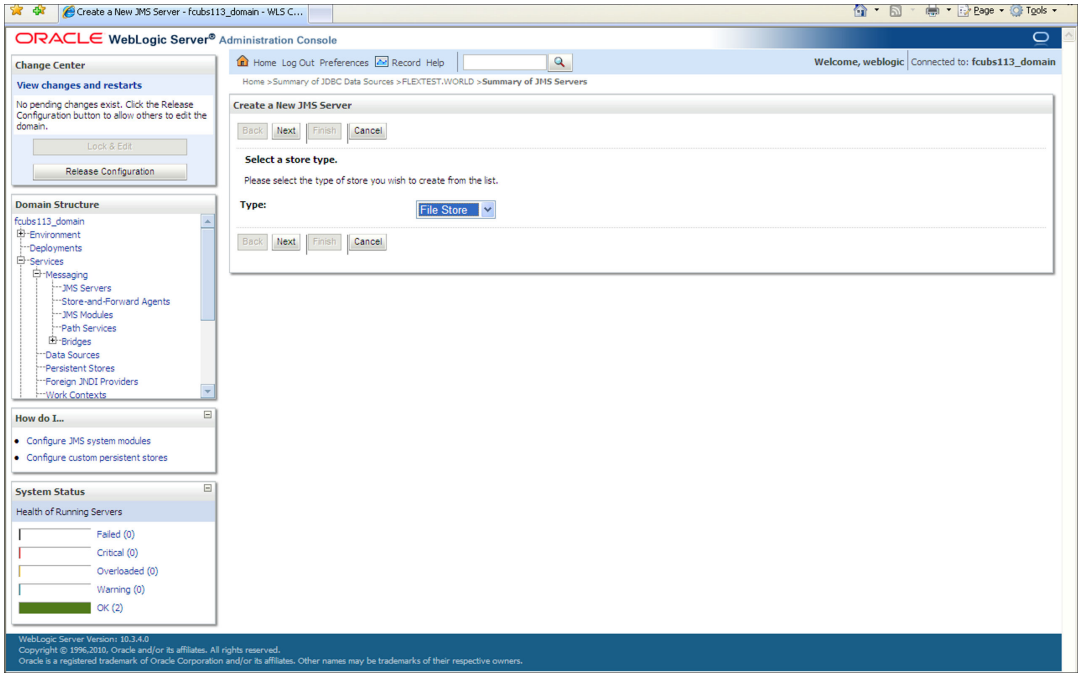
Table 7-5 Create a New JMS Server

Field	Description
JMS Server Name	Specify the name of JMS Server.

8. Click **Create a New Store** button.

   Create a New JMS Server_Store Type screen is displayed.

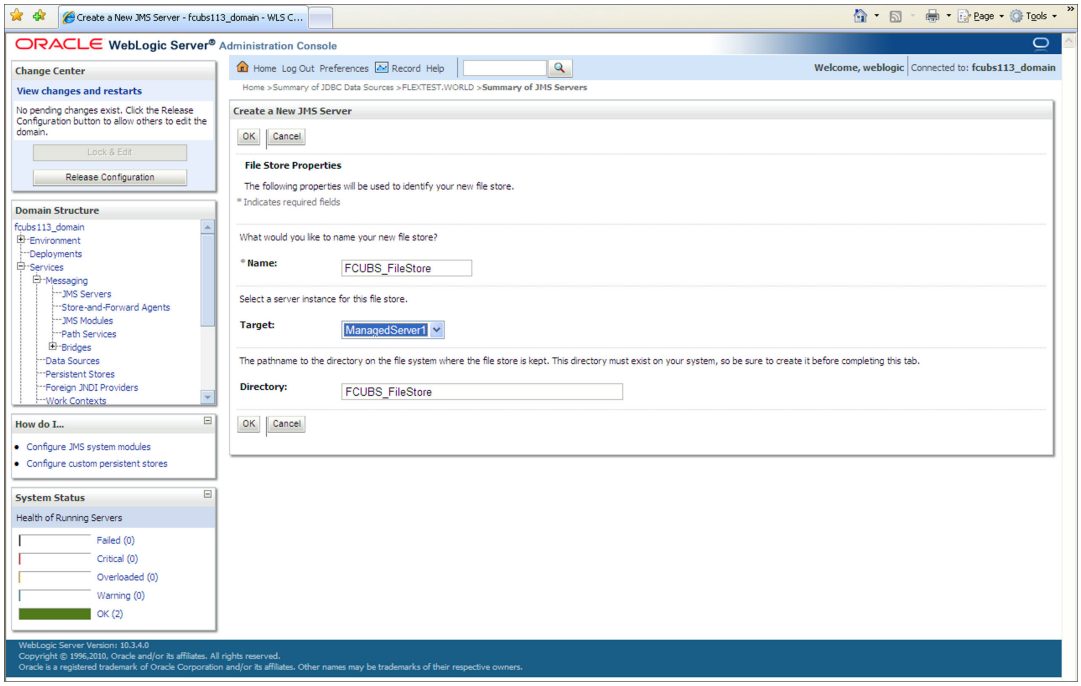
Figure 7-35 Create a New JMS Server_Store Type



9. Select the **Type** as **File Store** from the drop-down.
10. Click **Next** to specify the file store property.

    Create a New JMS Server_File Store Properties screen is displayed.

Figure 7-36 Create a New JMS Server_File Store Properties



11. Specify the following properties to identify the new **File Store**.

    For more information on fields, refer to the field description table.

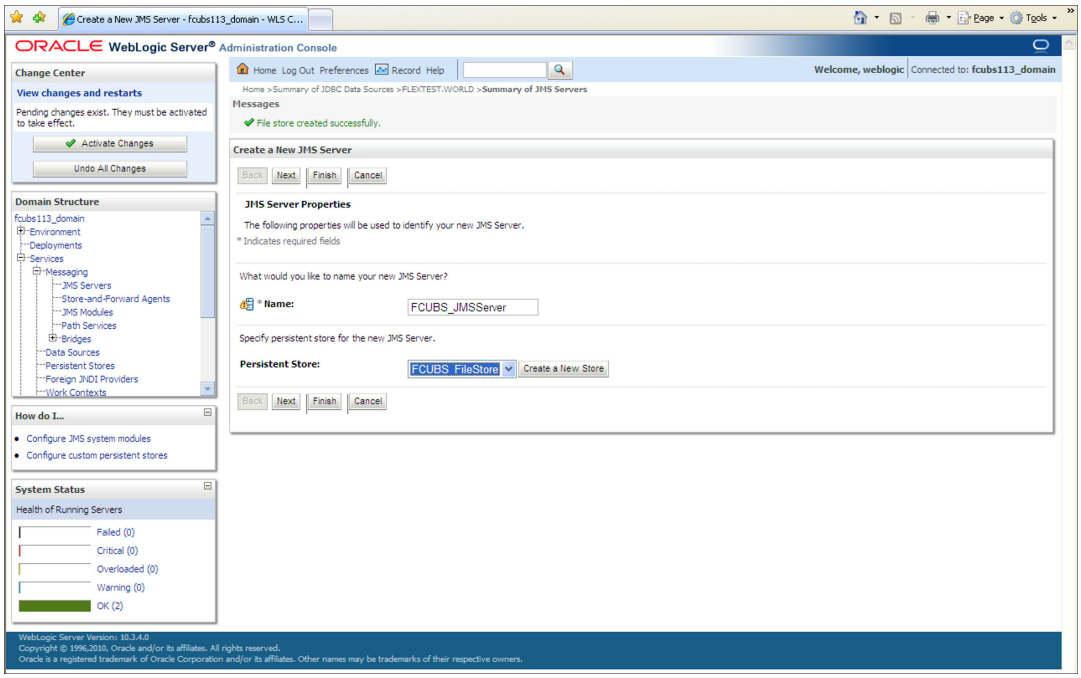
Table 7-6 Create a New JMS Server

Field	Description
Name	Specify the file store name. For Example: **FCIS_FileStore**
Target	Select a server instance for the file store in the **Target** field. You may select **ManagedServer1** (created by the user).
Directory	Specify the path name to the directory on the system where the file store is kept. Directory path as `C:/FCIS_FileStore`.

12. Click **Ok**.

    The following screen **Create a New JMS Server_File Store Created Message** is displayed with message `File store created successfully`.

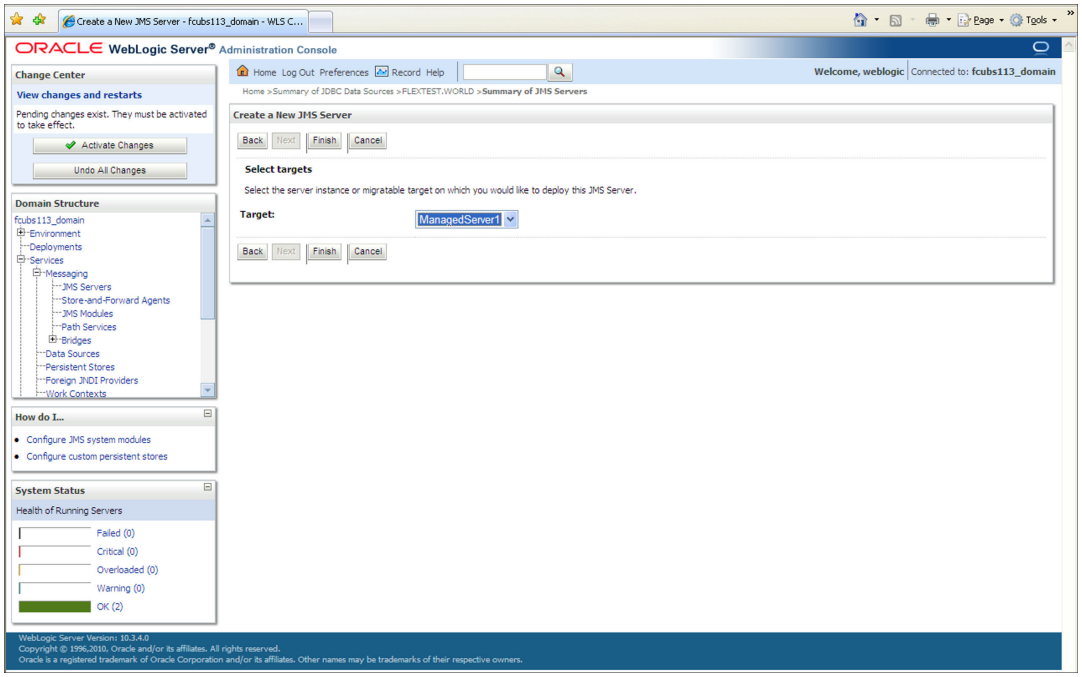
Figure 7-37 Create a New JMS Server_File Store Created Message



13. Click **Next** to select the target.

    Create a New JMS Server_Select Targets screen is displayed.

Figure 7-38 Create a New JMS Server_Select Targets



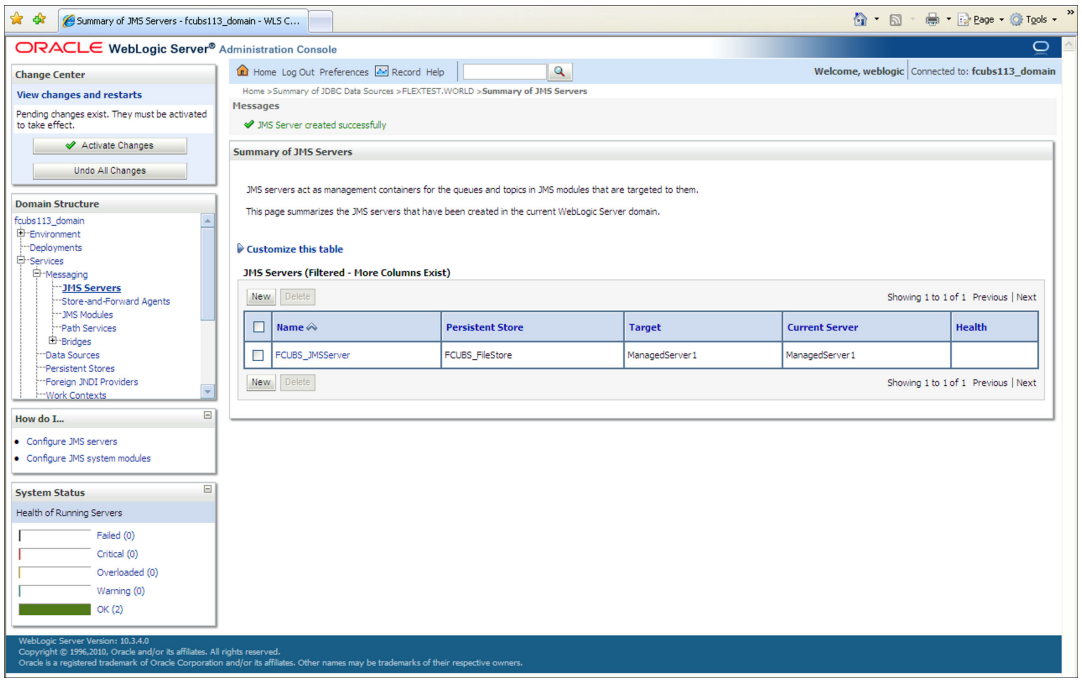
14. Select the server instance in the **Target** field where you would like to deploy the JMS server.

    Select the target as **ManagedServer1**.

15. Click **Finish** to creare a new JMS server.

    On successful creation of a new JMS server, the message `JMS Server created successfully` is displayed.

Figure 7-39 Summary of JMS Servers_JMS Server created



16. Click the **Activate Changes** button in the **Change Center** section of the screen.

    The message `All the changes have been activated. No restarts are necessary.` is displayed.

17. On the **Summary of JMS Servers** screen, you can view the new JMS Server created in the **JMS Servers** section.

    The new **JMS Server** is created.

7.3 JMS Modules Creation

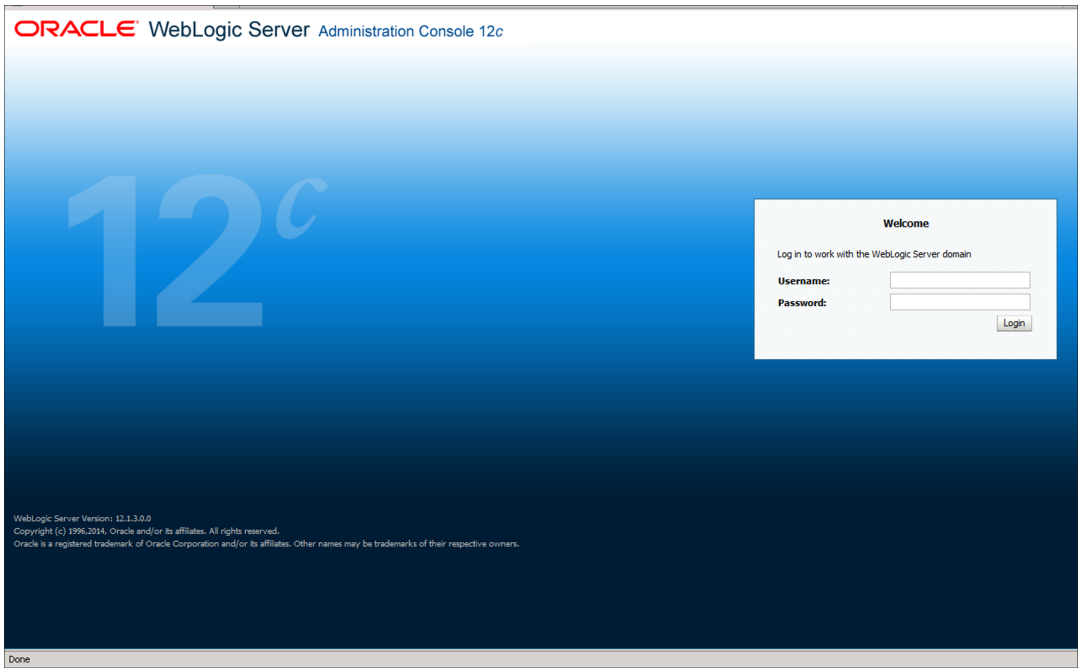
This topic explains the systematic instructions to create the JMS Modules in the Weblogic application server.

To create the JMS Modules, follow the steps given below:

1. Start the **Administration Console** of the WebLogic Application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   The **Oracle WebLogic Server Login** screen is displayed.

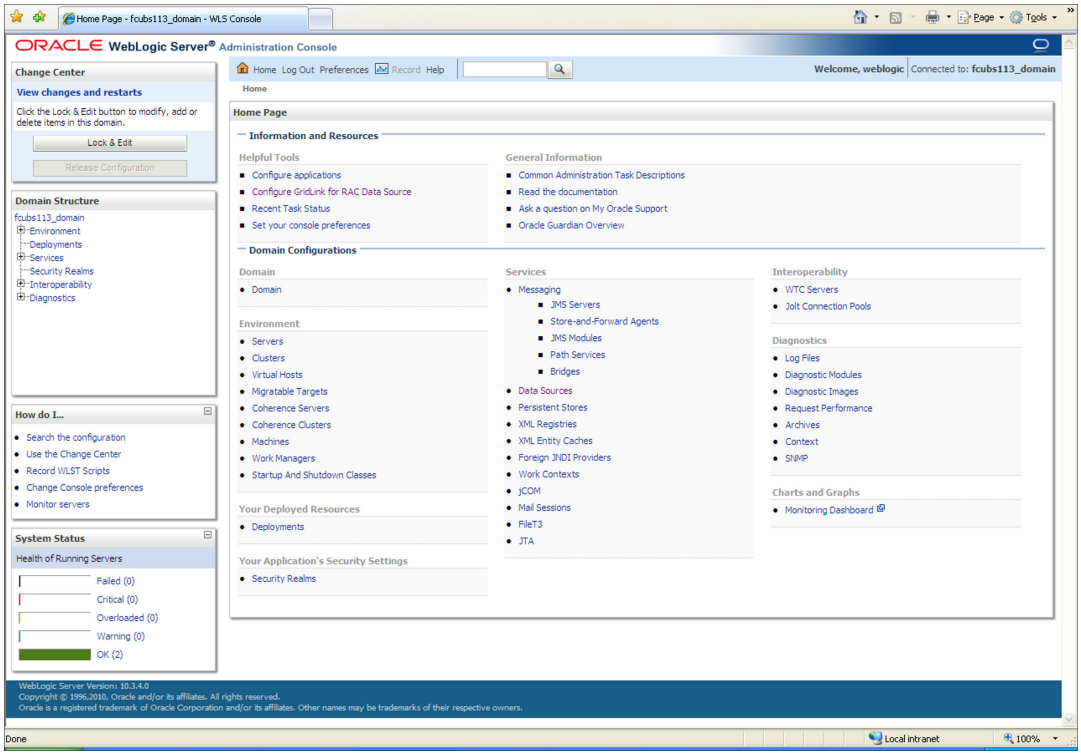
Figure 7-40 Weblogic Application Server Login



2. Specify the **Username** and **Password** in the WebLogic Server domain and click **Login**.

   The **Oracle Weblogic Server Home Page** screen is displayed.

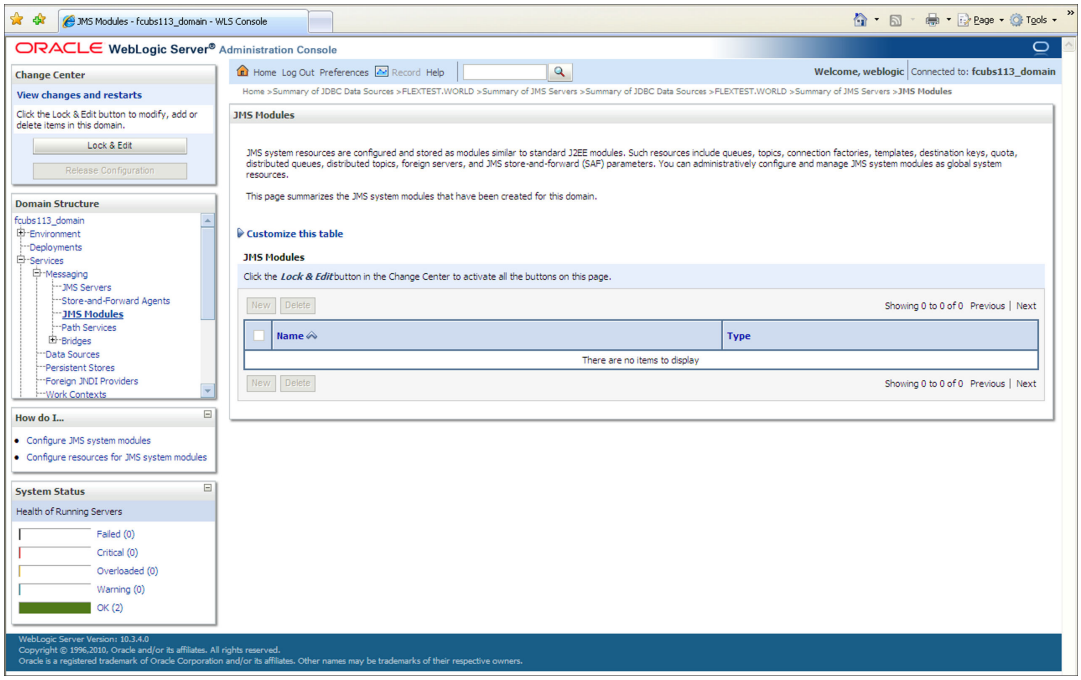
Figure 7-41 Oracle Weblogic Server Home Page



3. In the **Domain Structure**, expand the node **Services** and **Messaging**, and click **JMS Modules** from the list.

   The **JMS Modules** screen is displayed.

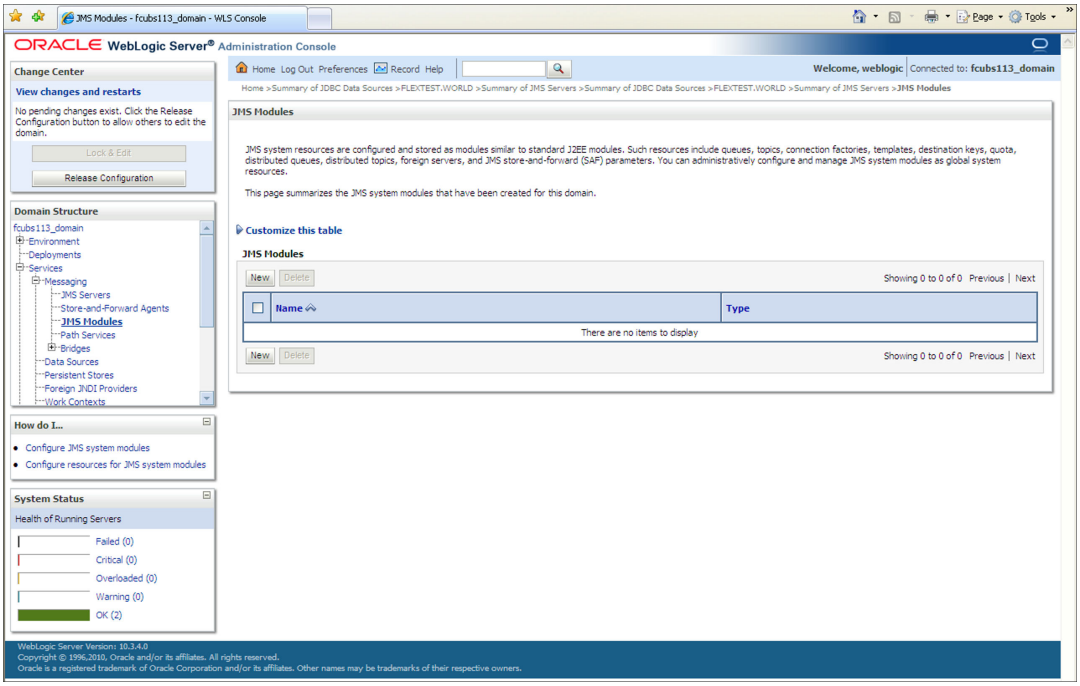
Figure 7-42 JMS Modules



4. Click the **Lock & Edit** button in the **Change Center** to add, modify or delete items by acivating all the buttons on this screen.

   The **JMS Modules** screen is displayed with all the buttons enabled to edit.

Figure 7-43 Click Lock and Edit_JMS Modules_New



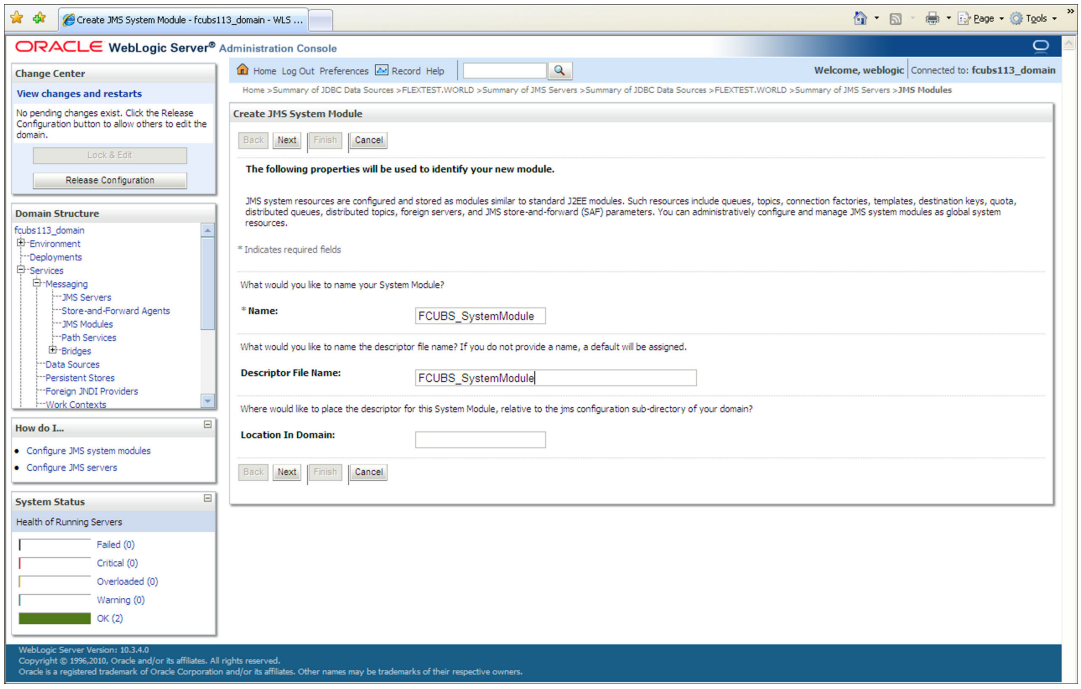
5. Navigate to **JMS Modules** section.

   The **JMS Modules_JMS Modules section** screen is displayed.

6. Click **New**.

   Create JMS System Module screen is displayed.

Figure 7-44 Create JMS System Module



7. On **Create JMS System Module** screen, specify the fields.

   For more information on fields, refer to the field description table.

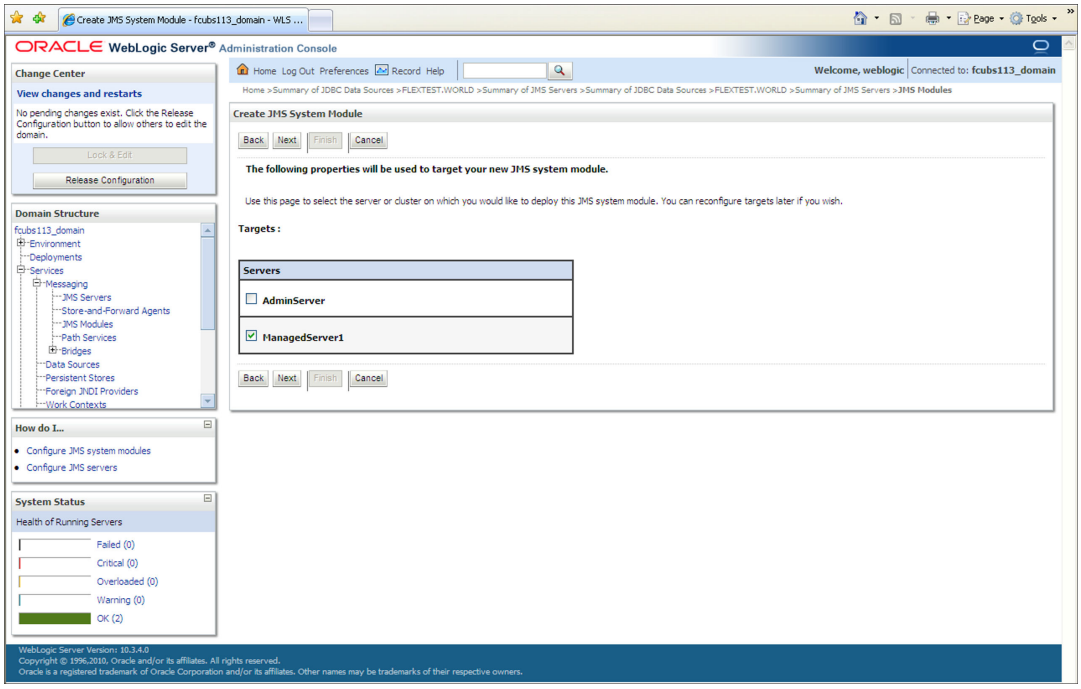
Table 7-7 Create JMS System Module

Field	Description
Name	Enter the System Module Name as **FCUBS_SystemModule**.
Description File Name	Enter the Description File Name as **FCUBS_SystemModule**.

8. Click **Next** to select the server where you want to deploy the JMS system module.

   Create JMS System Module_Select Targets screen is displayed.

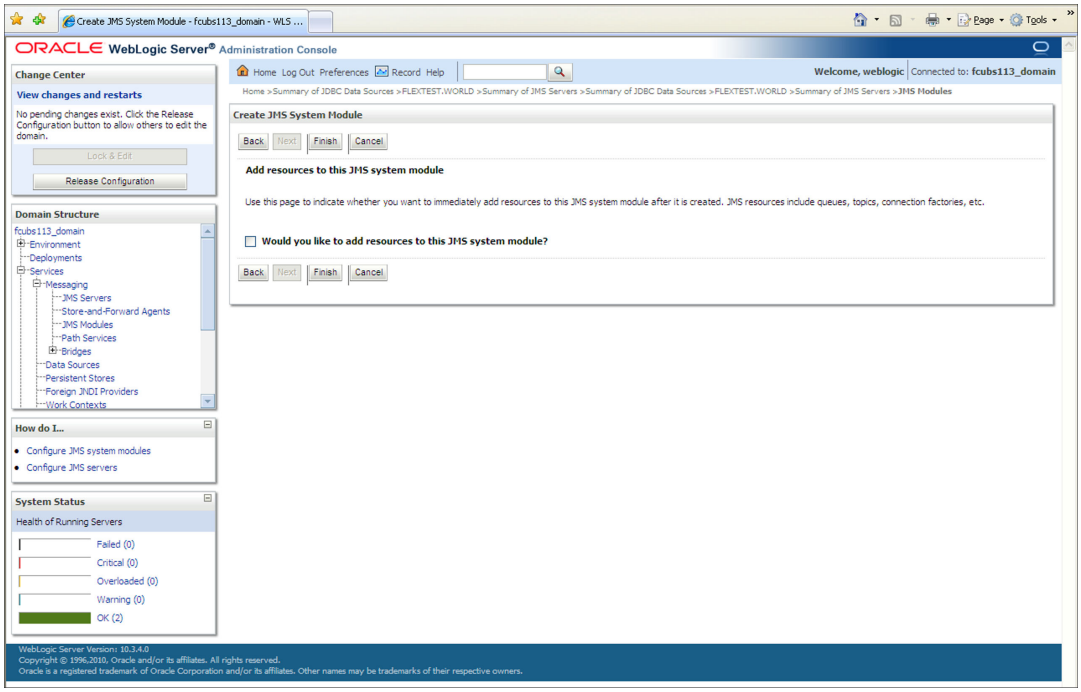
Figure 7-45 Create JMS System Module_Select Targets



9. Select the box against the server created and click **Next**.

Create JMS System Module_Add Resources screen is displayed.

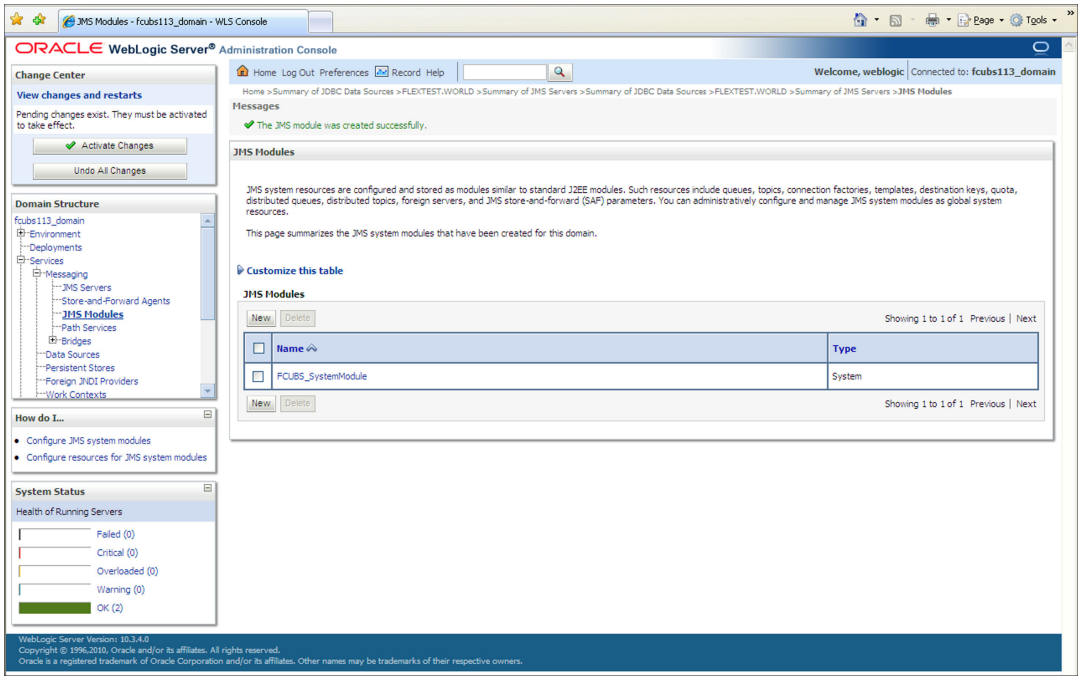
Figure 7-46 Create JMS System Module_Add Resources



10. Click **Finish**.

JMS Modules_New JMS Module created screen is displayed.

Figure 7-47 JMS Modules_JMS Module Created Message



11. Click the **Activate Changes** button in the **Change Center** section of the screen.

    The message `All the changes have been activated. No restarts are necessary.` is displayed.

The **JMS Module** is created.

7.4 Subdeployment Creation

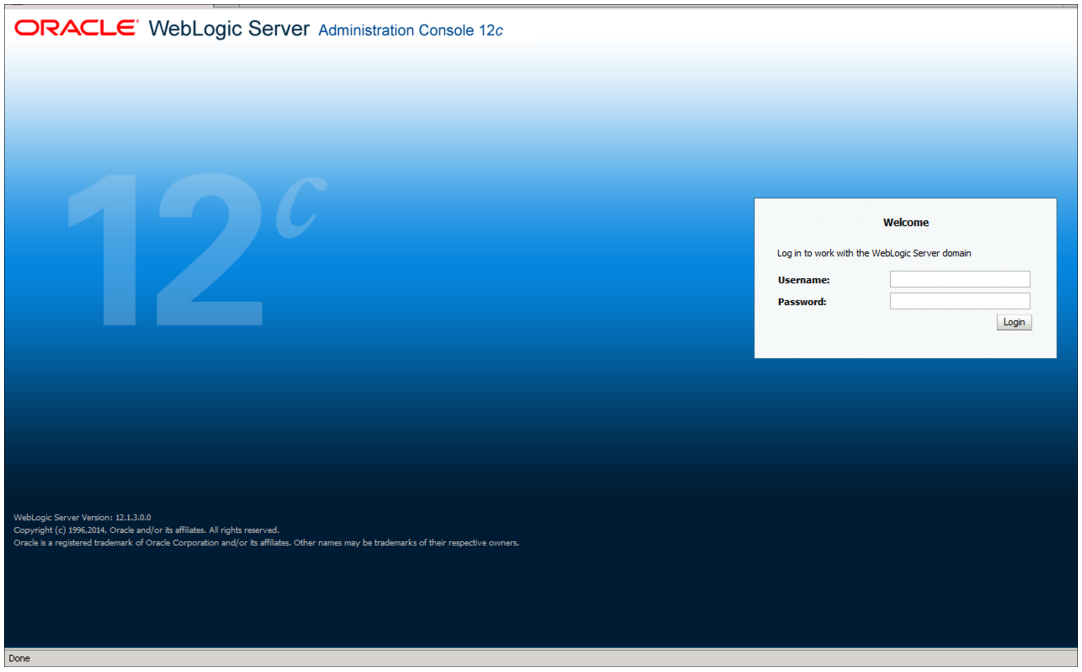
This topic explains the systematic instructions to create the subdeployment in the Weblogic application server.

Follow the steps given below to create the subdeployments:

1. Start the **Administration Console** of the WebLogic Application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   The **Oracle WebLogic Server Login** screen is displayed.

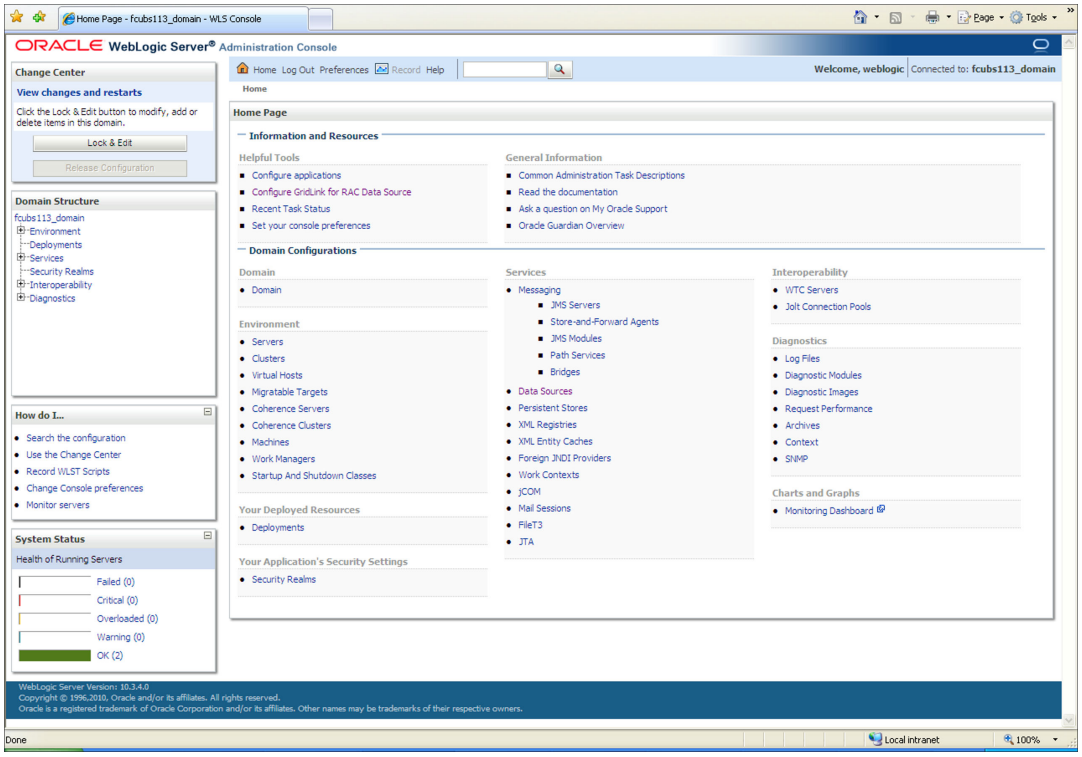
Figure 7-48 Weblogic Application Server Login



2. Specify the **Username** and **Password** in the WebLogic Server domain and click **Login**.

   The **Oracle Weblogic Server Home Page** screen is displayed.

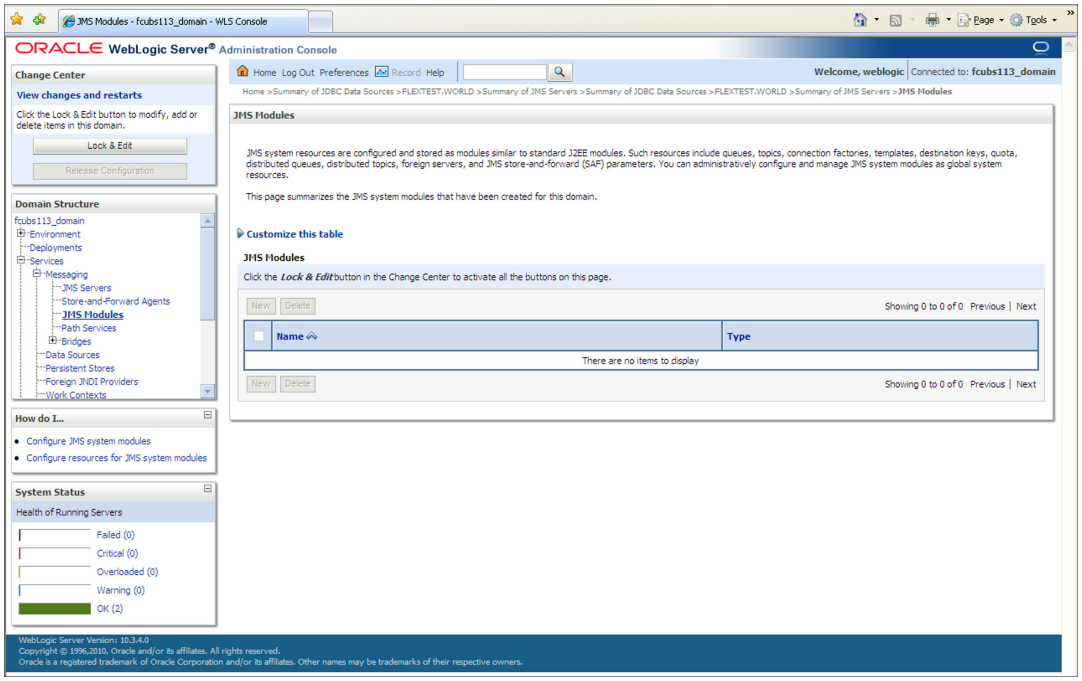
Figure 7-49 Oracle Weblogic Server Home Page



3. In the **Domain Structure**, expand the node **Services** and **Messaging**, and click **JMS Modules** from the list.

The **JMS Modules** screen is displayed.

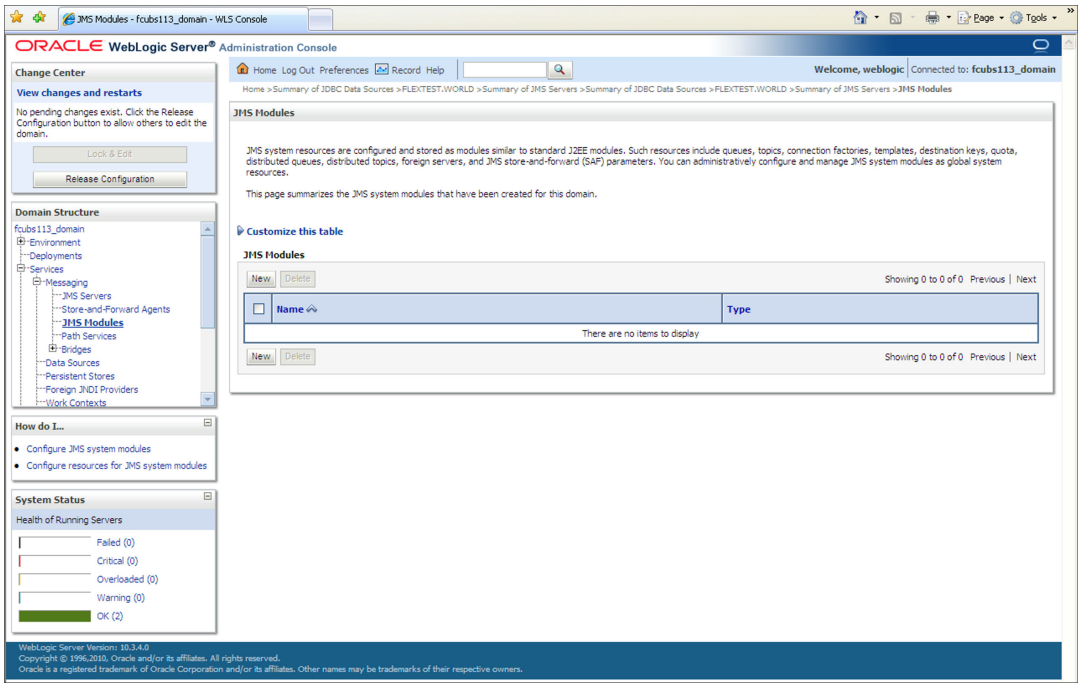
Figure 7-50 JMS Modules



4. Click the **Lock & Edit** button in the **Change Center** to add, modify or delete items by acivating all the buttons on this screen.

   The **JMS Modules** screen is displayed with all the buttons enabled to edit.

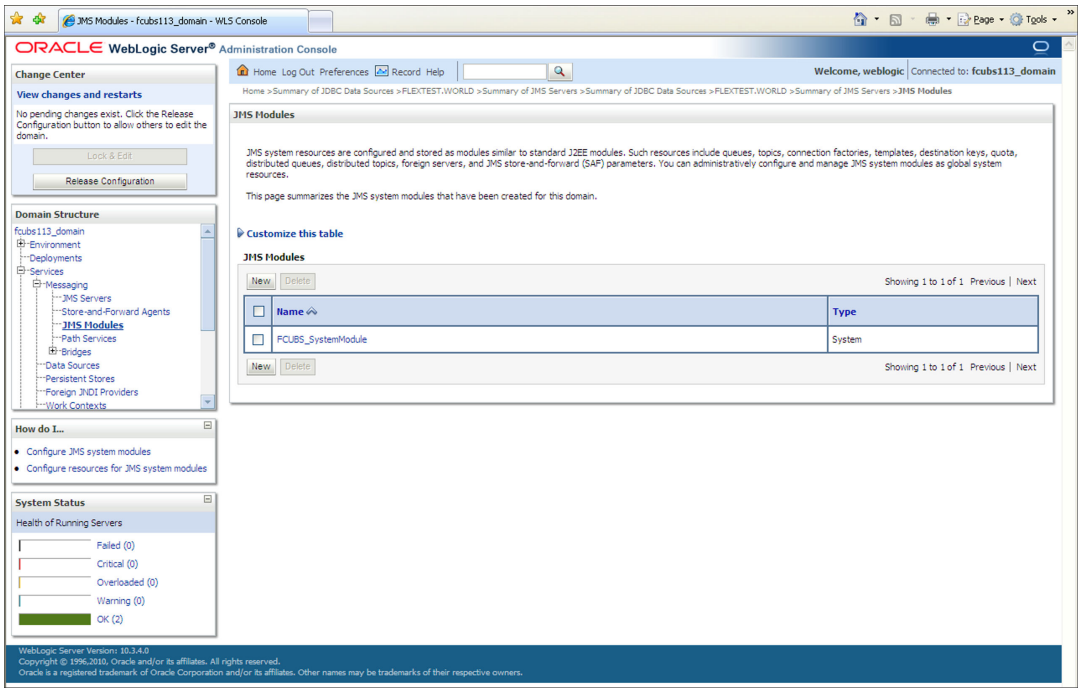
Figure 7-51 Click Lock and Edit_JMS Modules_New



5. Select the JMS module created earlier.

   Settings for the SystemModule screen is displayed.

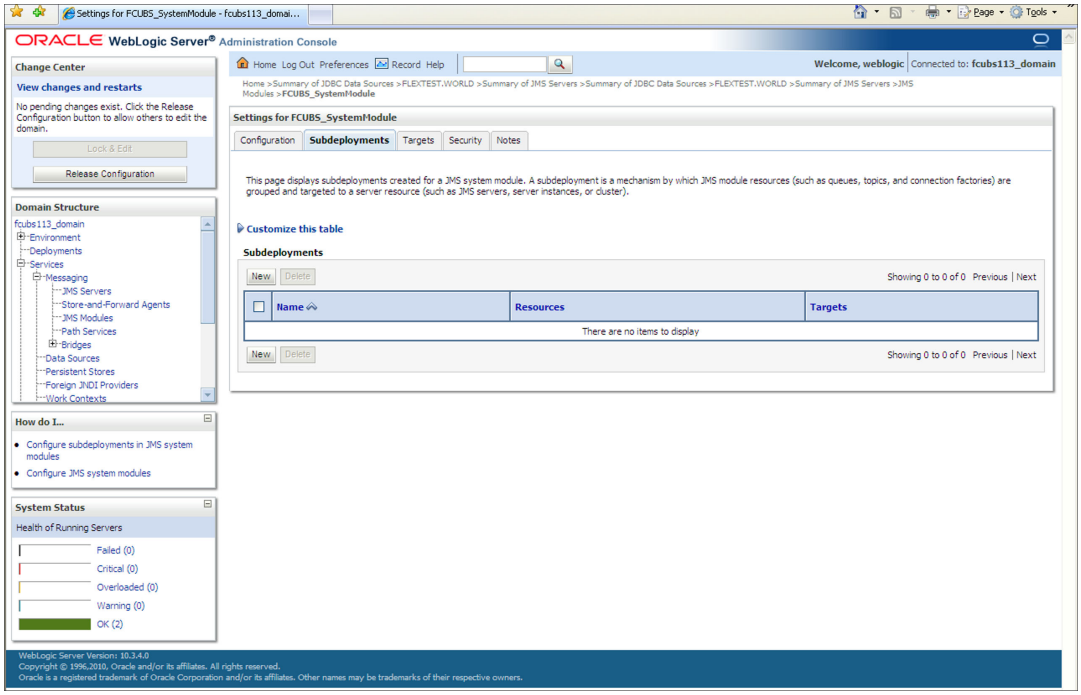
Figure 7-52 JMS Modules_Select JMS Moduled created



6. Click **Subdeployments** tab.

   The **Subdeployments** screen is displayed.

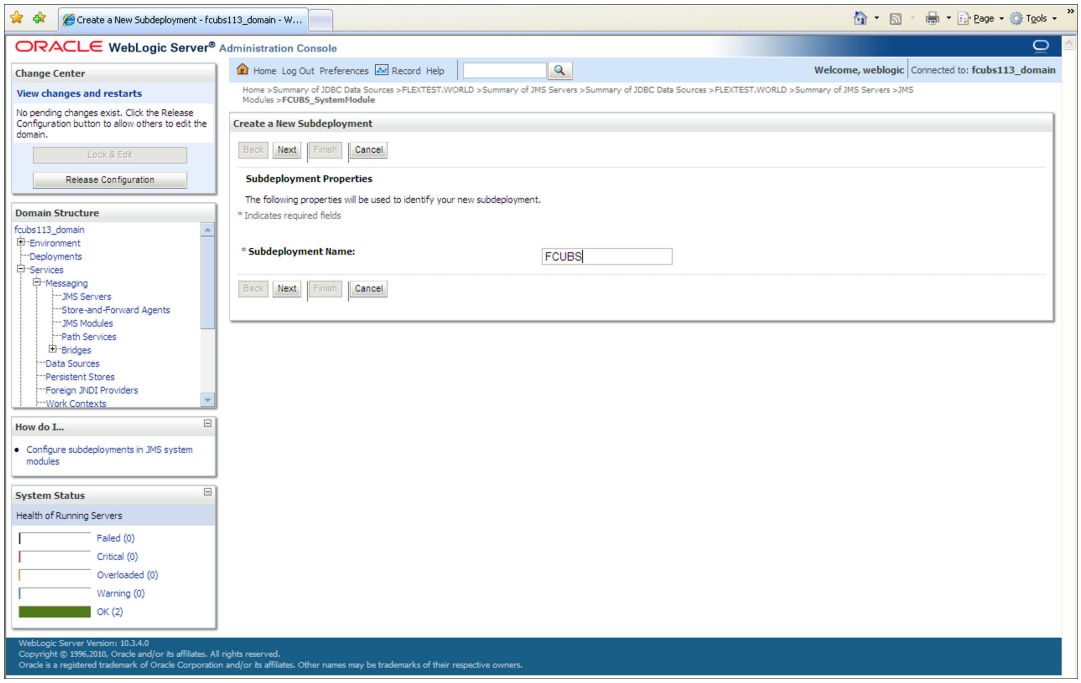
Figure 7-53 Subdeployments



7. On the **Subdeployments** section, click **New**.

   Create a New Subdeployment screen is displayed.

Figure 7-54 Create a New Subdeployment



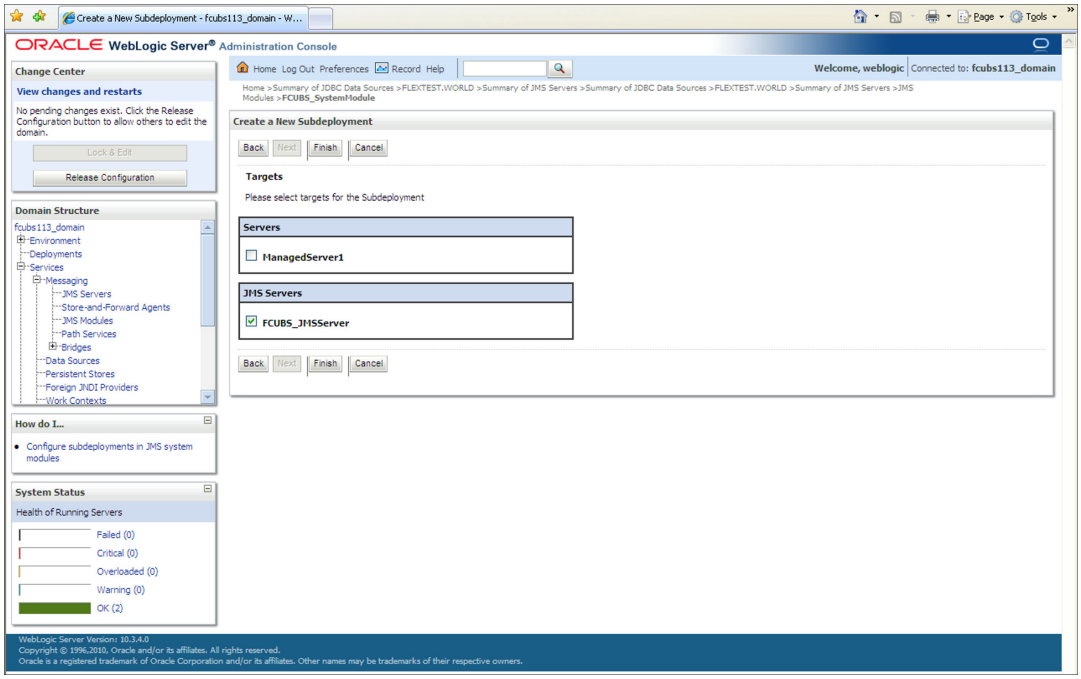
8. On **Create a New Subdeployment** screen, specify the **Subdeployment Name**.

   Create a New Subdeployment_Subdeployment Properties screen is displayed.

9. Click **Next** to select targets for the subdeployment.

   Create a New Subdeployment_Select Targets screen is displayed.

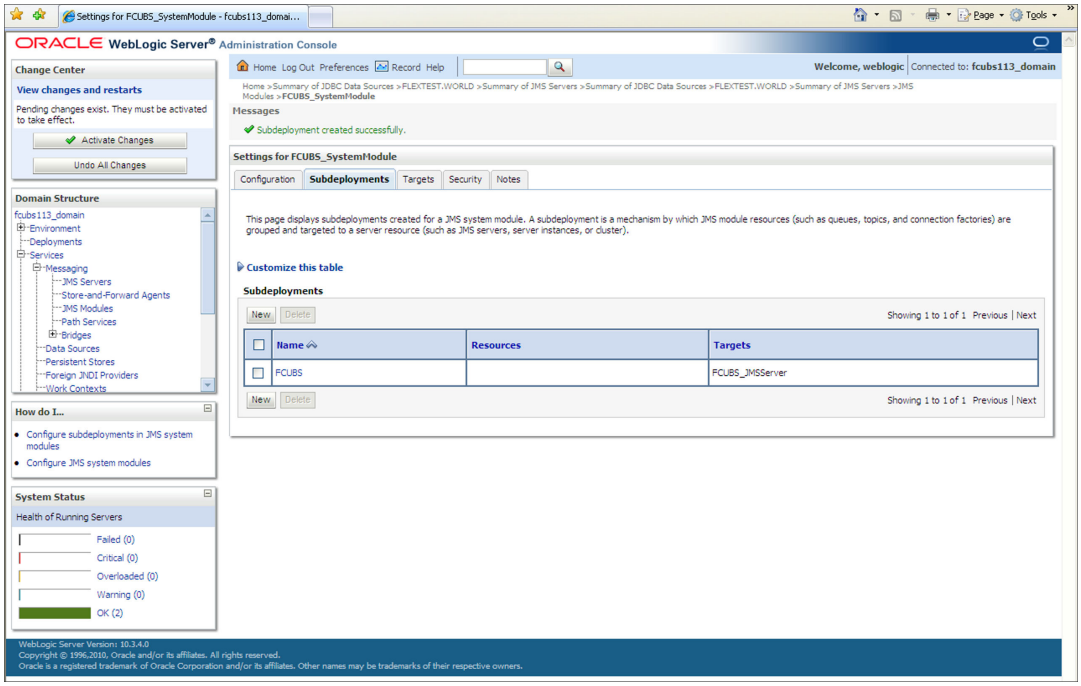
Figure 7-55 Create a New Subdeployment_Select Targets



10. Select the **JMS Server** (as created by the user).
11. Click **Finish**.

    The new subdeployment created is displayed in the **Subdeployments** section.

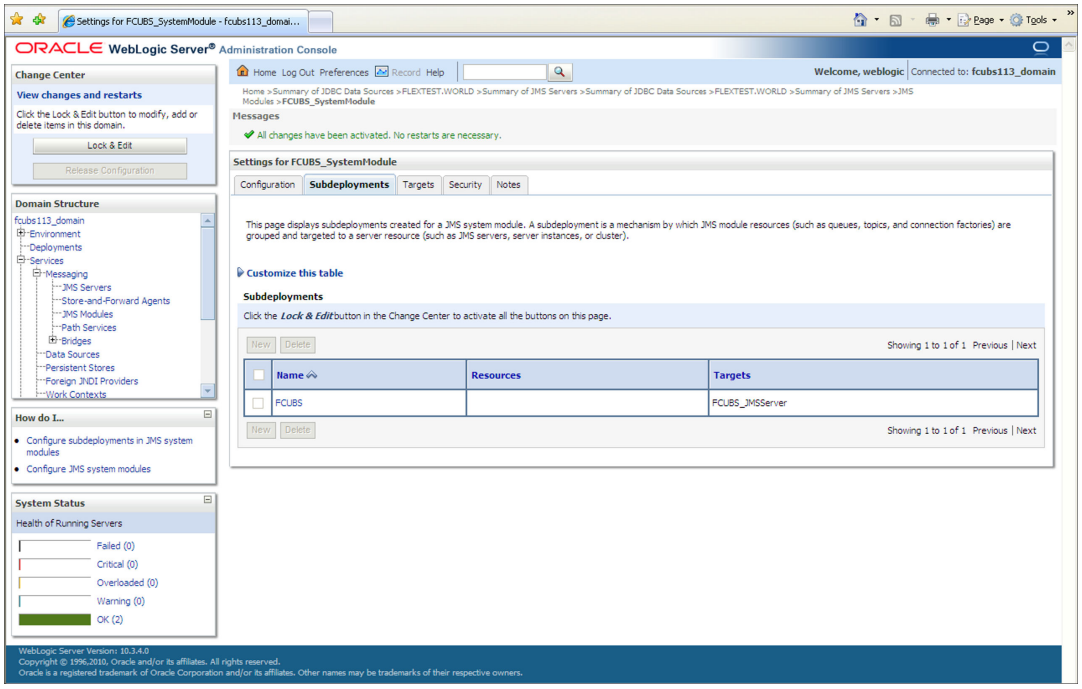
Figure 7-56 Subdeployments Created



12. Click the **Activate Changes** button in the **Change Center** section of the screen to accept the changes made.

The message `All the changes have been activated. No restarts are necessary.` is displayed.

Figure 7-57 Subdeployments_All Changes Activated



The new **Subdeployment** is created.

7.5 JMS Queue Creation

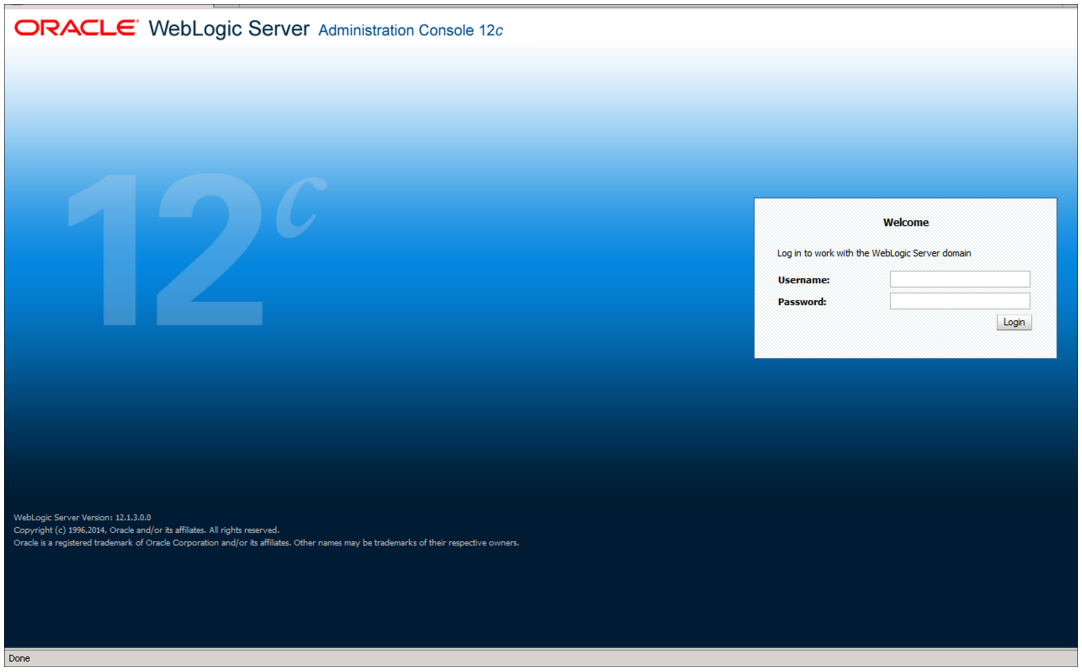
This topic explains the systematic instructions to create the JMS Queue in the Weblogic application server.

Follow the steps given below to create the JMS Queue:

1. Start the **Administration Console** of the WebLogic Application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   The **Oracle WebLogic Server Login** screen is displayed.

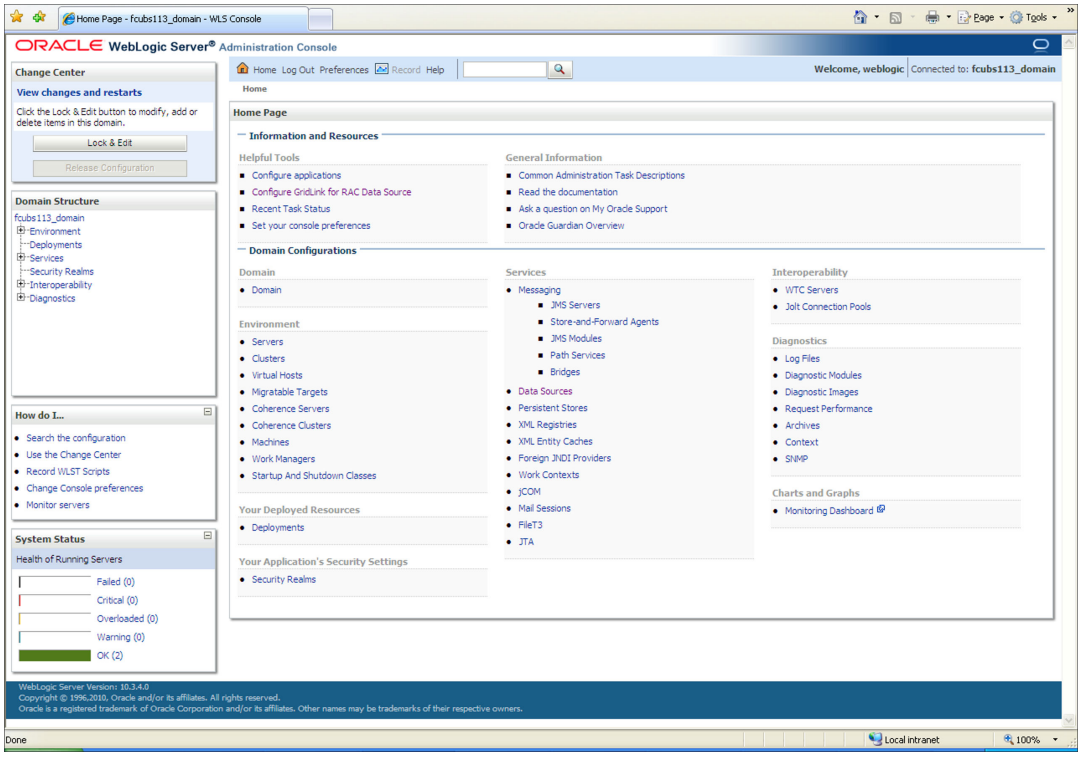
Figure 7-58 Weblogic Application Server Login



2. Specify the **Username** and **Password** in the WebLogic Server domain and click **Login**.

   The **Oracle Weblogic Server Home Page** screen is displayed.

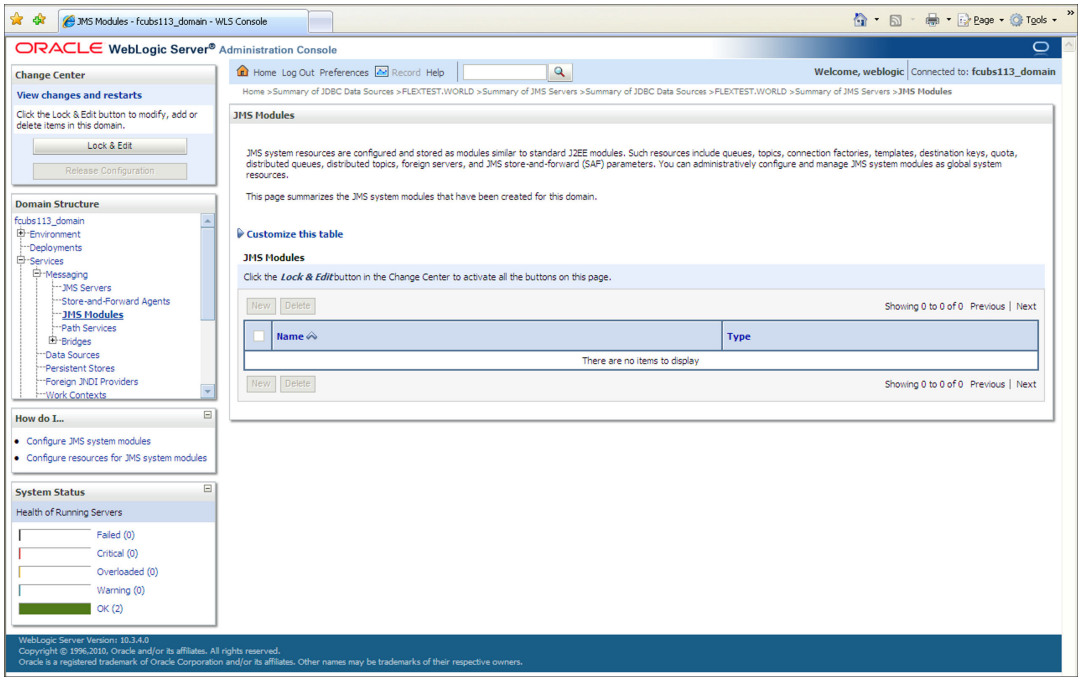
Figure 7-59 Oracle Weblogic Server Home Page



3. In the **Domain Structure**, expand the node **Services** and **Messaging**, and click **JMS Modules** from the list.

The **JMS Modules** screen is displayed.

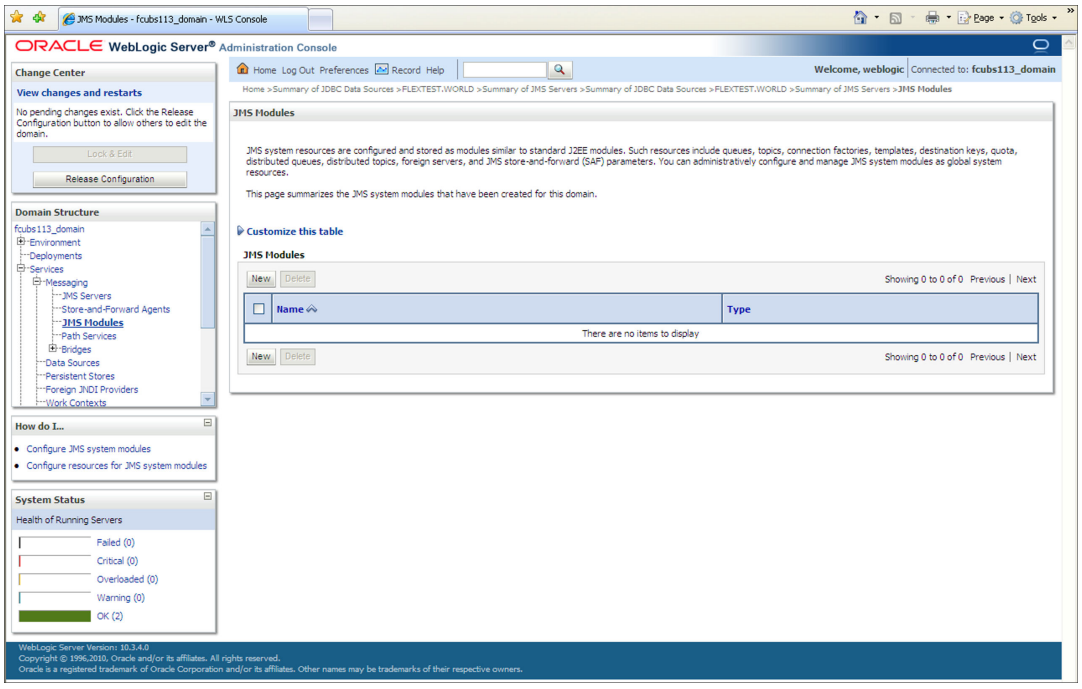
Figure 7-60 JMS Modules



4. Click the **Lock & Edit** button in the **Change Center** to add, modify or delete items by acivating all the buttons on this screen.

   The **JMS Modules** screen is displayed with all the buttons enabled to edit.

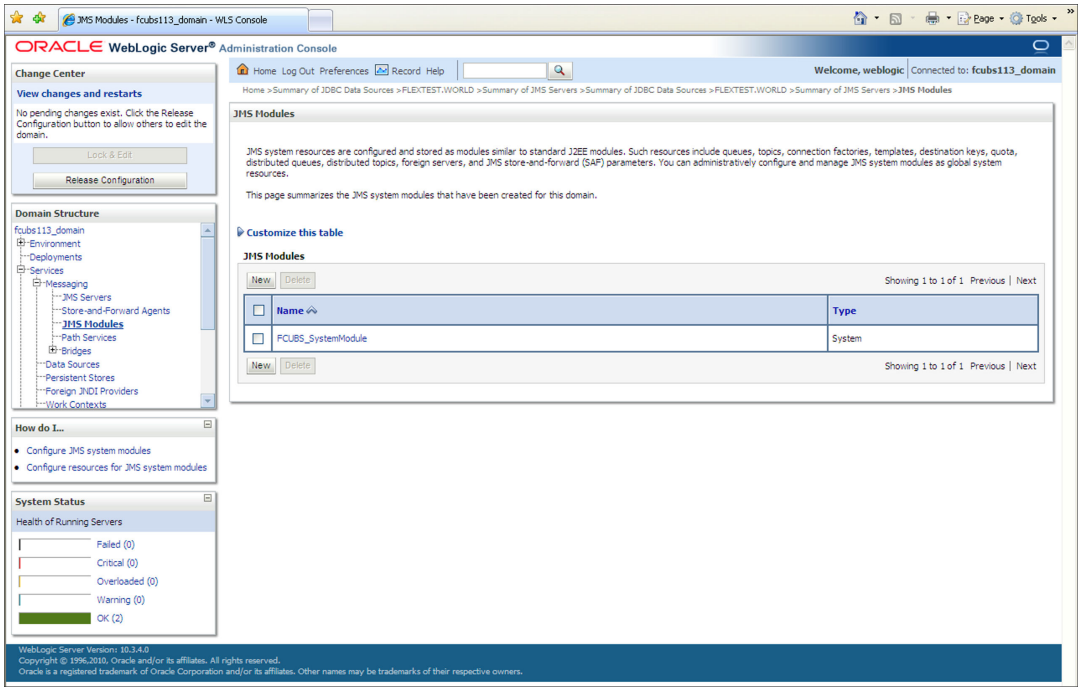
Figure 7-61 Click Lock and Edit_JMS Modules_New



5. Select the JMS module created earlier in the **JMS Modules** section.

   The screen displays the list of JMS modules created.

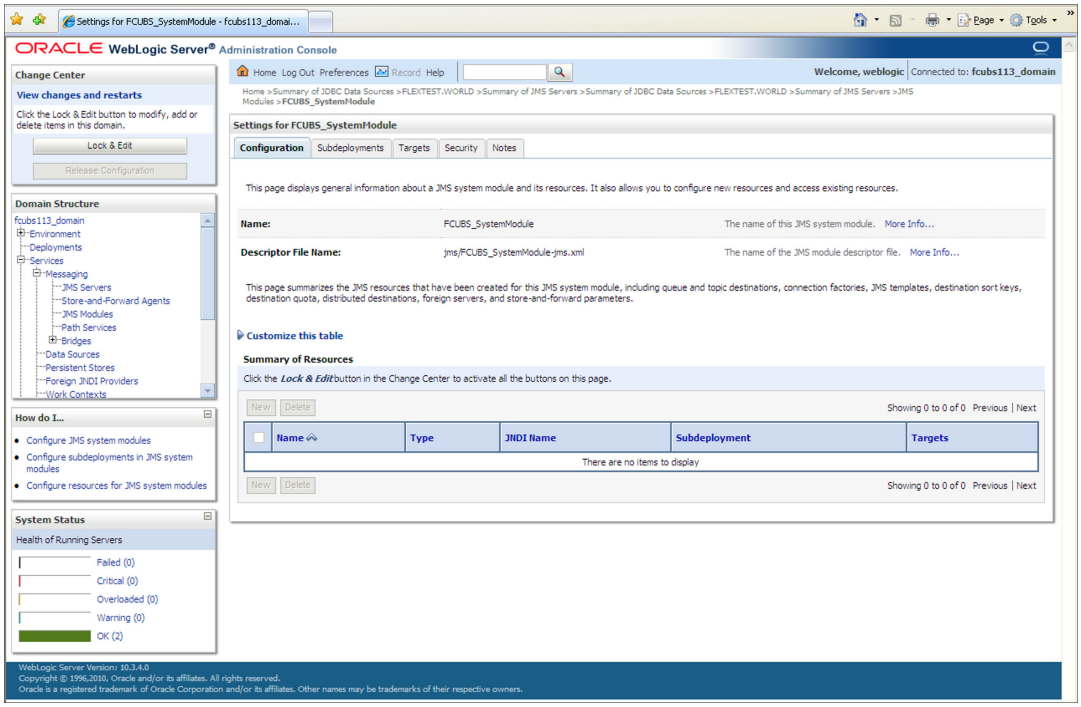
Figure 7-62 JMS Modules_Select JMS Moduled created



6. Click the **Configuration** tab to set the configuration and then click **Lock & Edit** button in the **Change Center**.

   Settings for the SystemModule_Configuration tab is displayed.

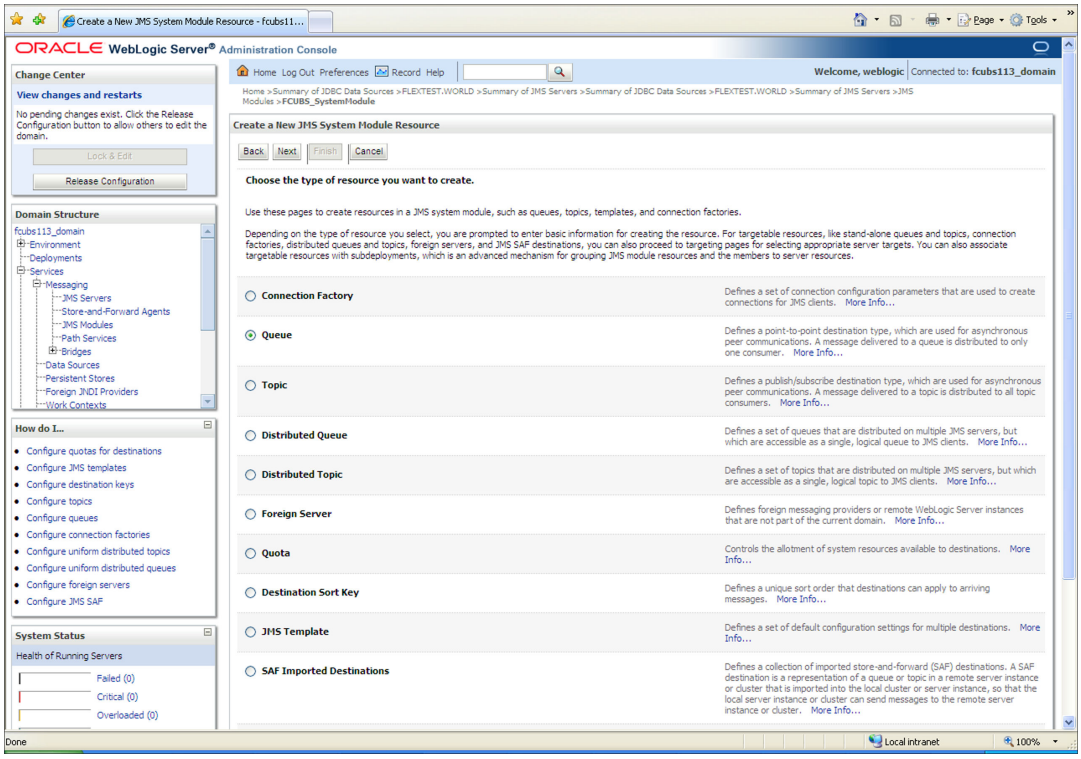
Figure 7-63 Settings for the SystemModule_Configuration



7. On the **Settings for the SystemModule_Configuration** tab, click **New** in the **Summary of Resources** section.

   Create a New JMS System Module Resource screen is displayed.

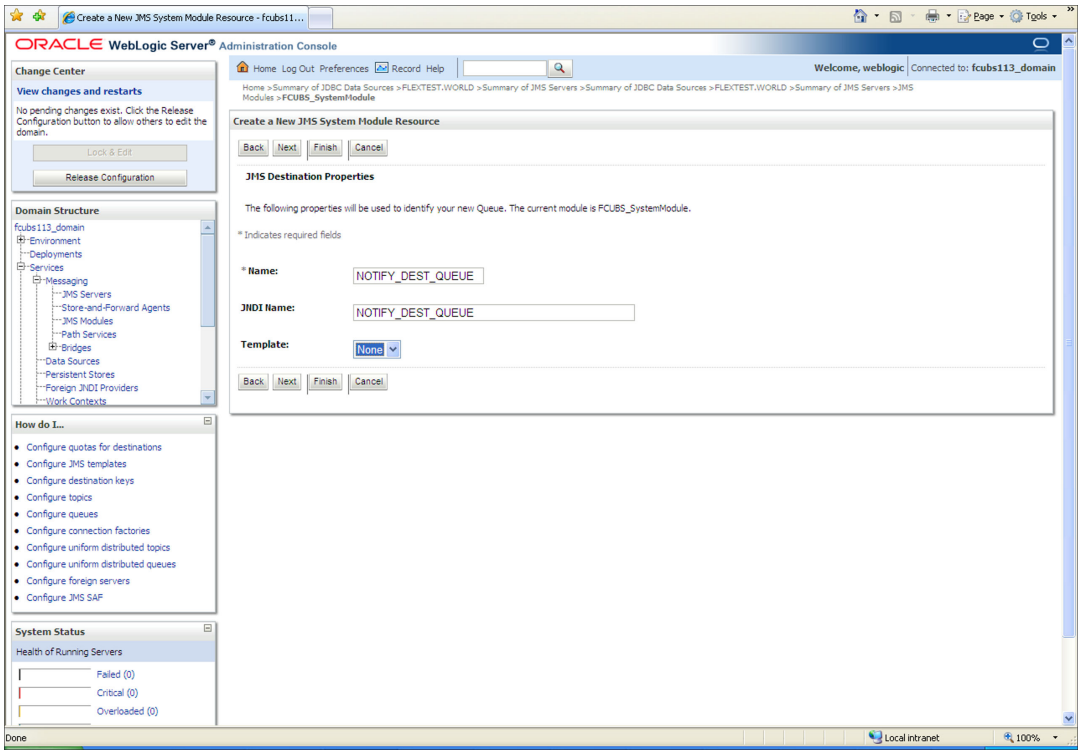
Figure 7-64 Create a New JMS System Module Resource



8. Select the **Queue** option and click **Next**.

   Create a New JMS System Module Resource_JMS Destination Properties screen is displayed.

Figure 7-65 Create a New JMS System Module Resource_JMS Destination Properties



9. To **Create a New JMS System Module Resource**, specify the fields.

   For more information on fields, refer to the field description table.

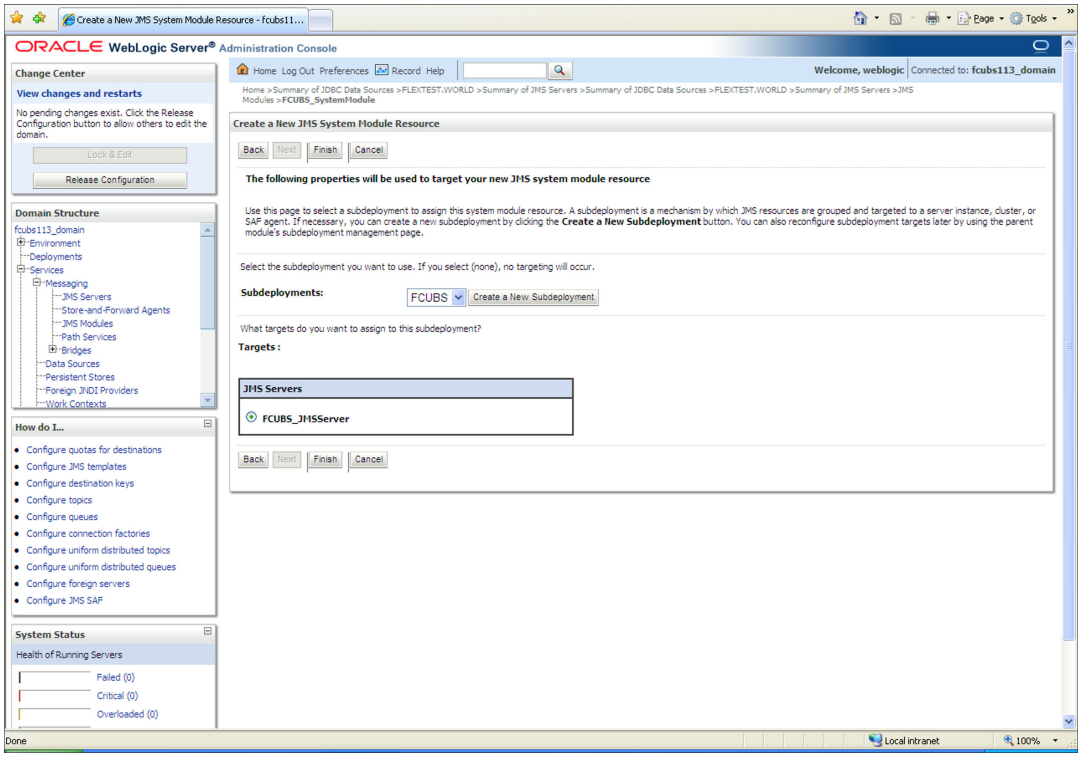
Table 7-8 JMS Destination Properties

Filed	Description
Name	Specify the Name of the Queue as **NOTIFY_DEST_QUEUE**.
JNDI Name	Specify the JNDI Name as **NOTIFY_DEST_QUEUE**
Template	Select the Template as **None** from the drop-down.

10. Click **Next** to select the subdeployment.

    Create a New JMS System Module Resource_Select Subdeployments screen is displayed.

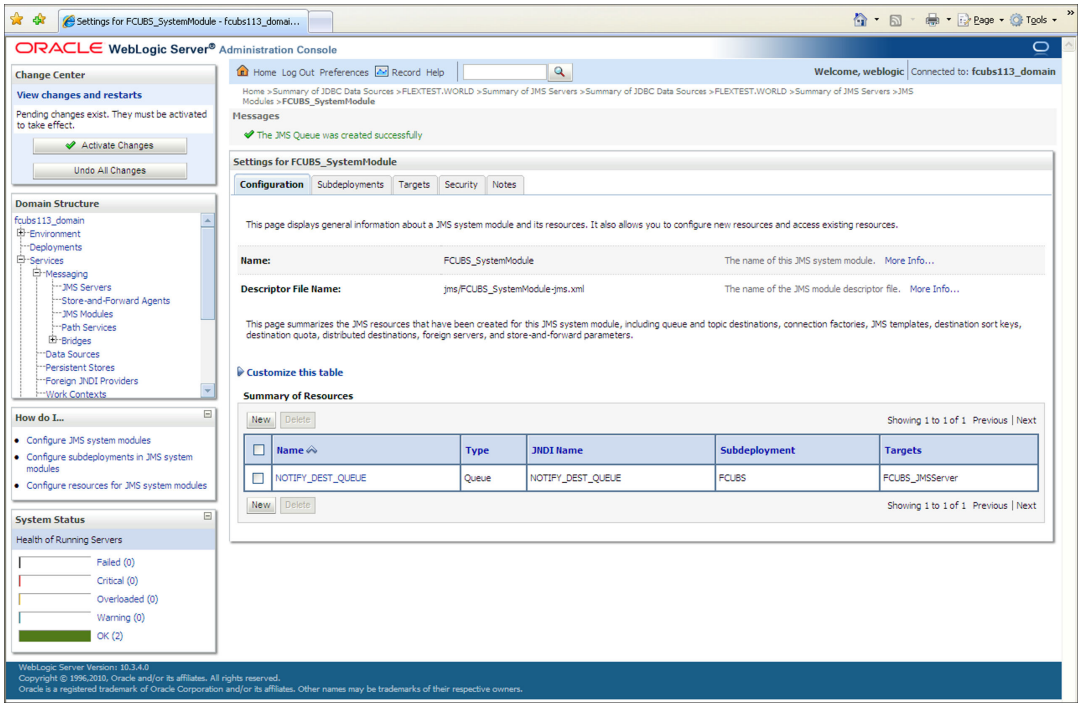
Figure 7-66 Create a New JMS System Module Resource_Select Subdeployments



11. Select the server created earlier as the target to assign to this subdeployment and click **Finish** button.

    The new **JMS Queue** is created.

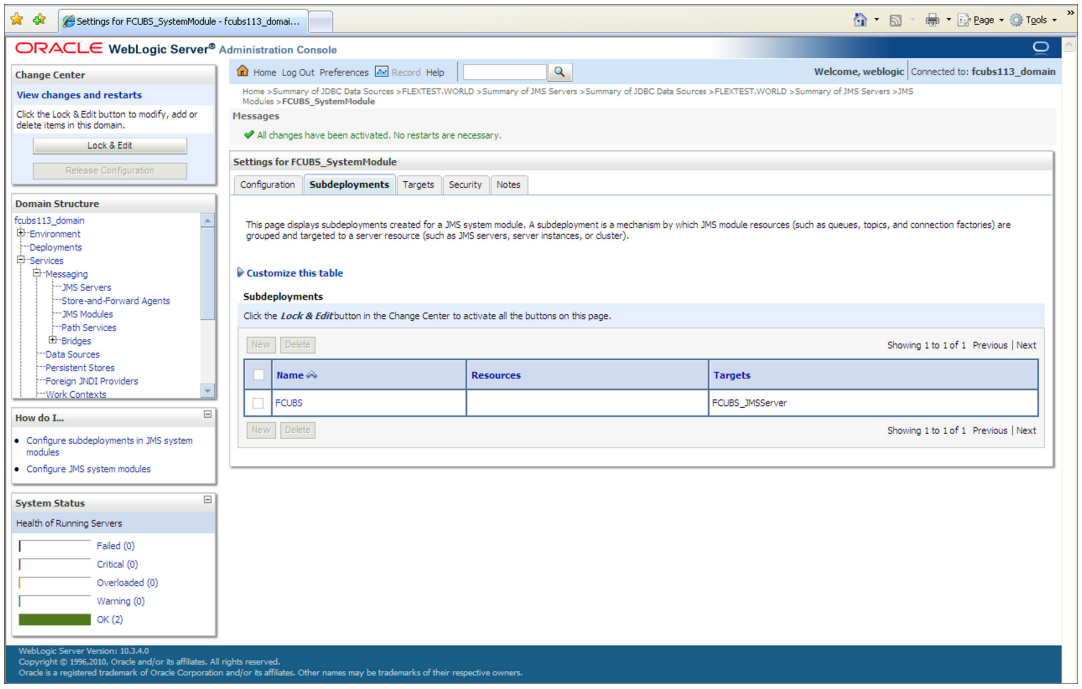
Figure 7-67 New JMS Queue Created



12. Click the **Activate Changes** button in the **Change Center** section of the screen to accept the changes made.

    The message `All the changes have been activated. No restarts are necessary.` is displayed.

Figure 7-68 Subdeployments_All Changes Activated



The **JMS Queue** has been created successfully

7.6 JMS Connection Factory Creation

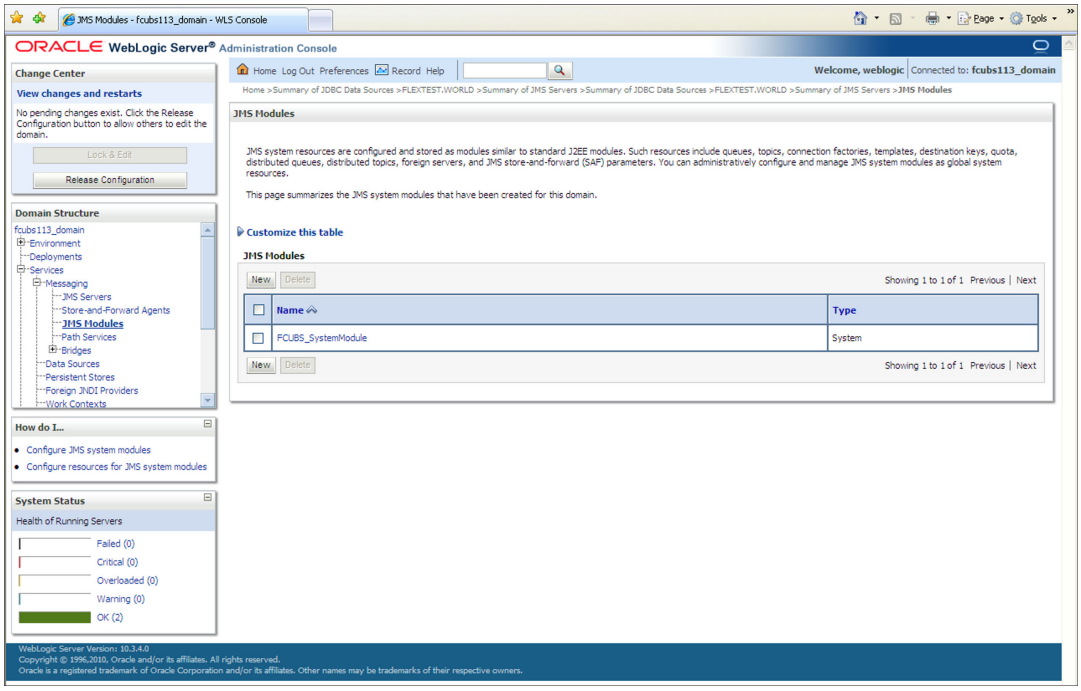
This topic explains the systematic instructions to create the JMS Connection Factory in the Weblogic application server.

You need to create the connection factory after creating the queues. To create the JMS Connection Factory, follow the steps given below:

1. Select the JMS Module created earlier in the **JMS Modules** section.

   The screen displays the list of JMS modules created.

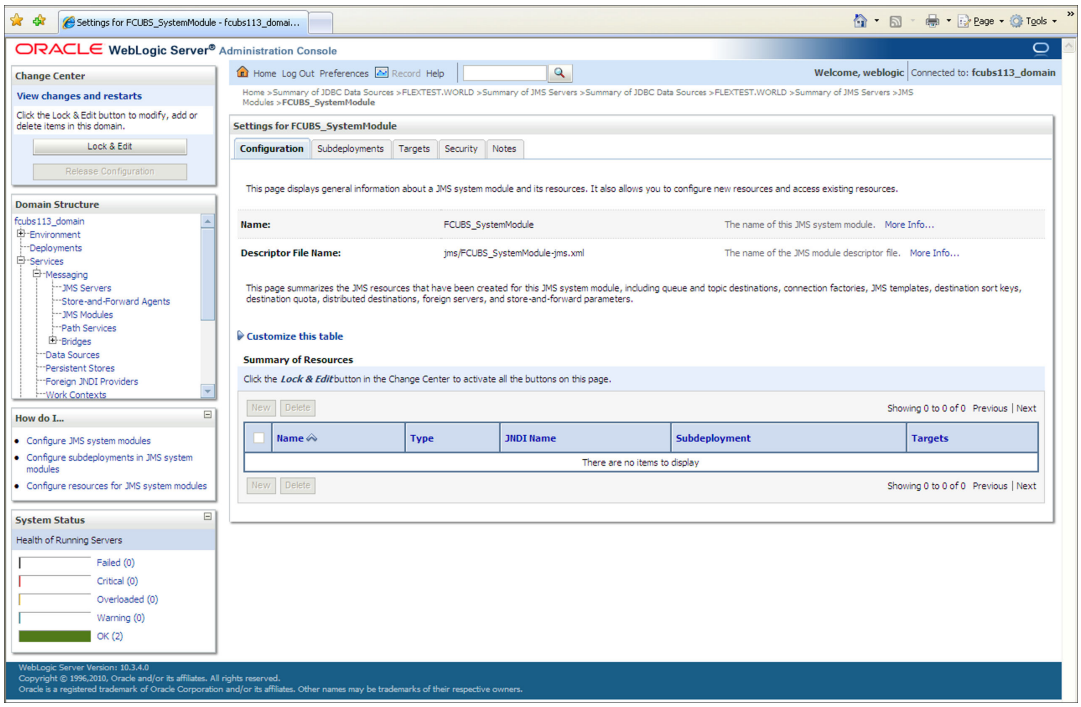
Figure 7-69 JMS Modules_Select JMS Moduled created



2. On the **Settings for the SystemModule** screen, click the **Configuration** tab to configure new resources or to access the existing resources.

   Settings for the SystemModule_Configuration tab is displayed.

Figure 7-70 Settings for the SystemModule_Configuration



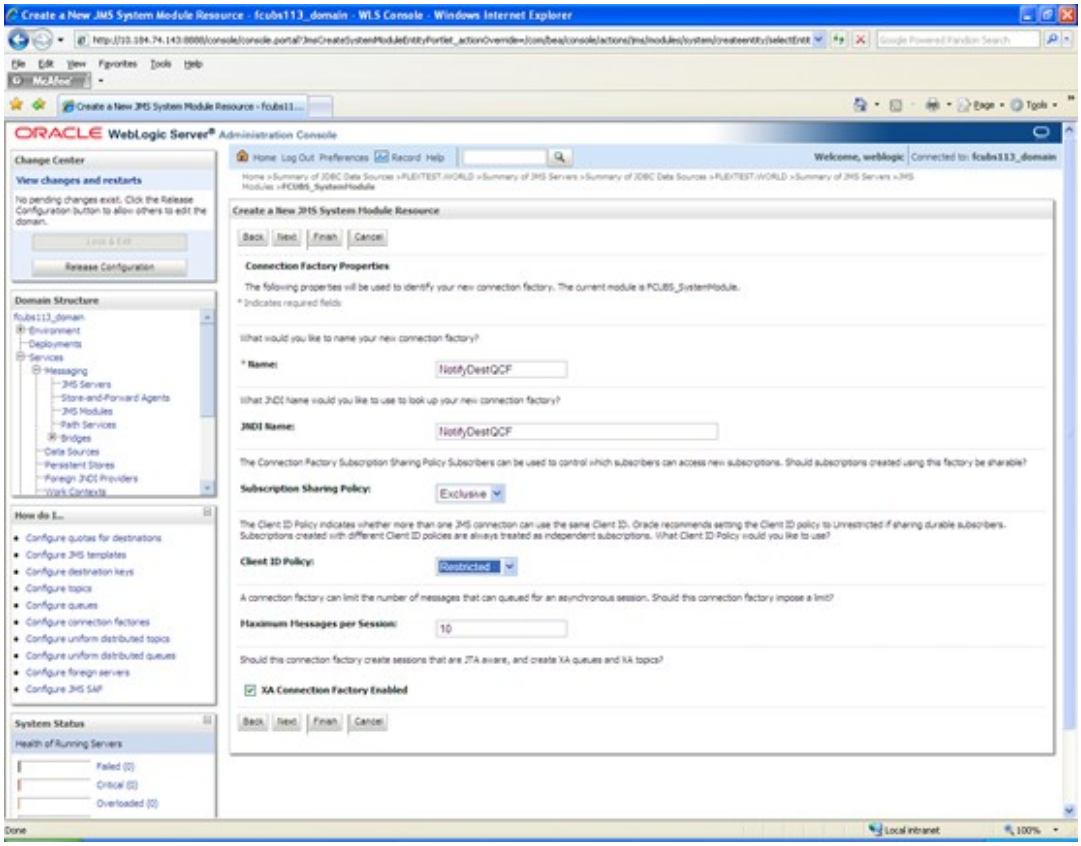
3. Click the **Lock & Edit** button in the **Change Center**. Click **New** in the **Summary of Resources** section to create new resources.

Create a New JMS System Module Resource screen is displayed.

4. Select the **Connection Factory** option and click **Next**.

   Create a New JMS System Module Resource_Connection Factory Properties screen is displayed.

Figure 7-71 Create a New JMS System Module Resource - Connection Factory Properties



5. On the **Connection Factory Properties** screen, specify the fields.

   For more information on fields, refer to the field description table.

Table 7-9 Connection Factory Properties

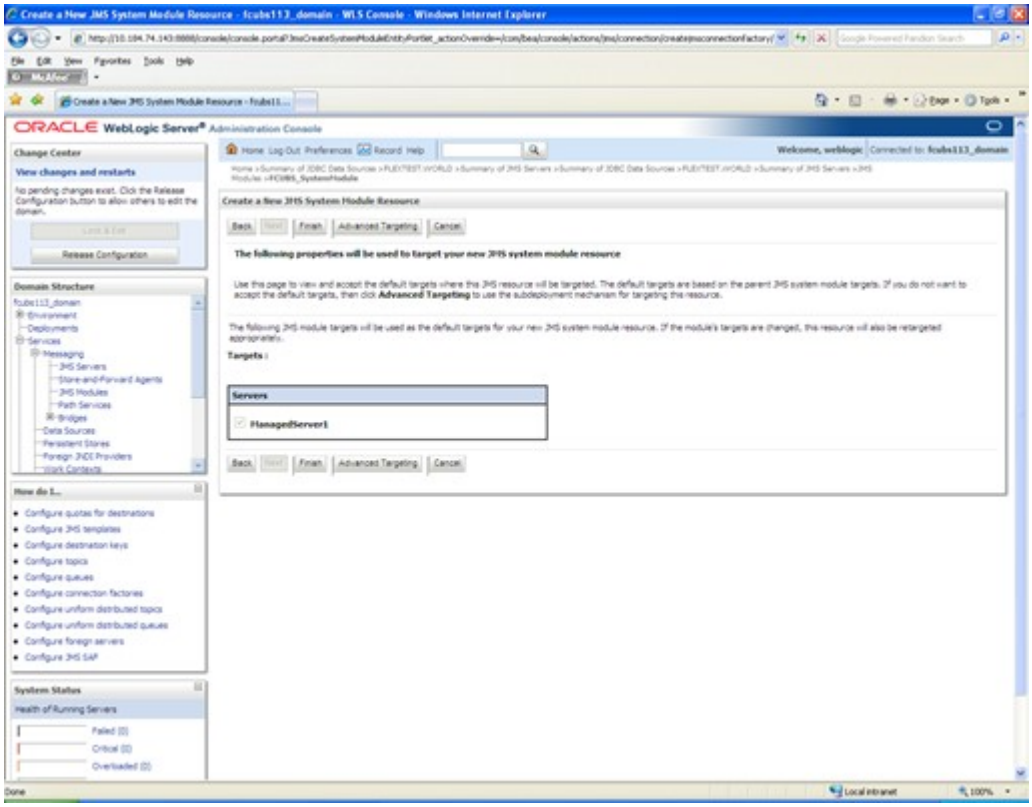
Filed	Description
Name	Specify the Name of the Connection factory as**NotifyDestQCF**.
JNDI Name	Specify the JNDI Name as **NotifyDestQCF**.
Client ID Policy	Select the Client ID policy as **Restricted** from the drop-down.

6. Select the box **XA Connection Factory Enabled**.
7. Click **Next** to use the default target for new JMS system module resource.

   The default targets are based on the parent JMS system module targets.

   Create a New JMS System Module Resource_Select Targets screen is displayed.

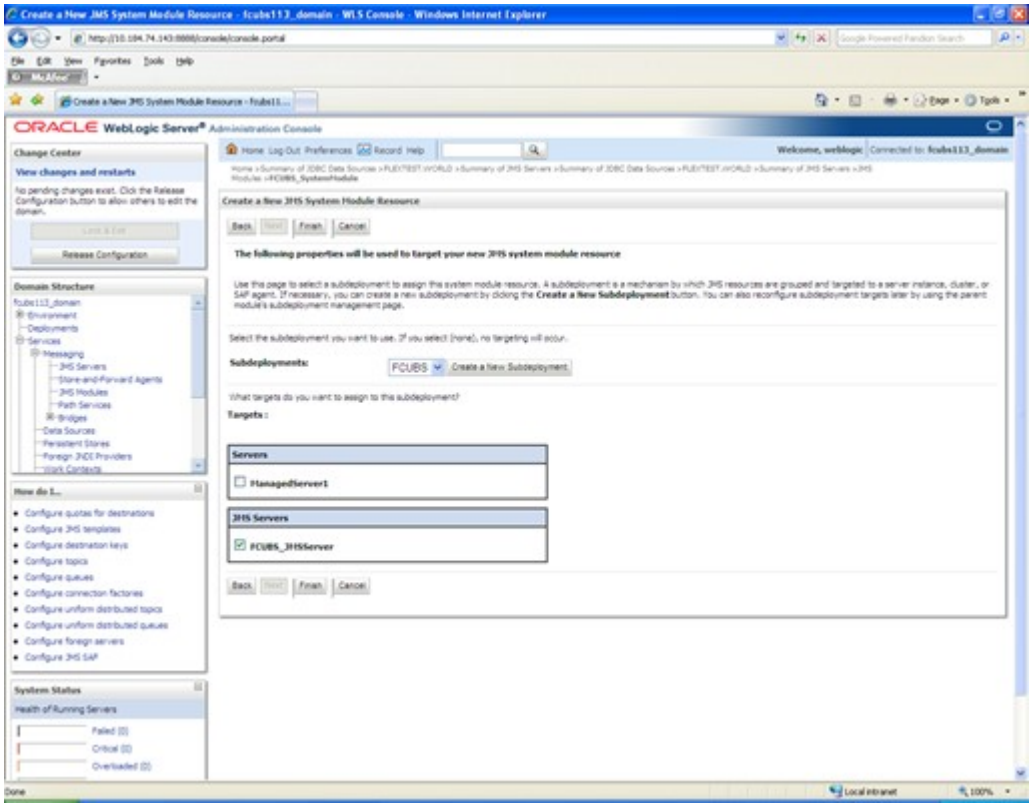
Figure 7-72 Create a New JMS System Module Resource - Targets



8. Click **Advanced Targeting** to use the subdeployment mechanism for targeting this source.

   The screen displays the targets you want to assign to this subdeployment.

Figure 7-73 Create a New JMS System Module Resource - Advance Targeting

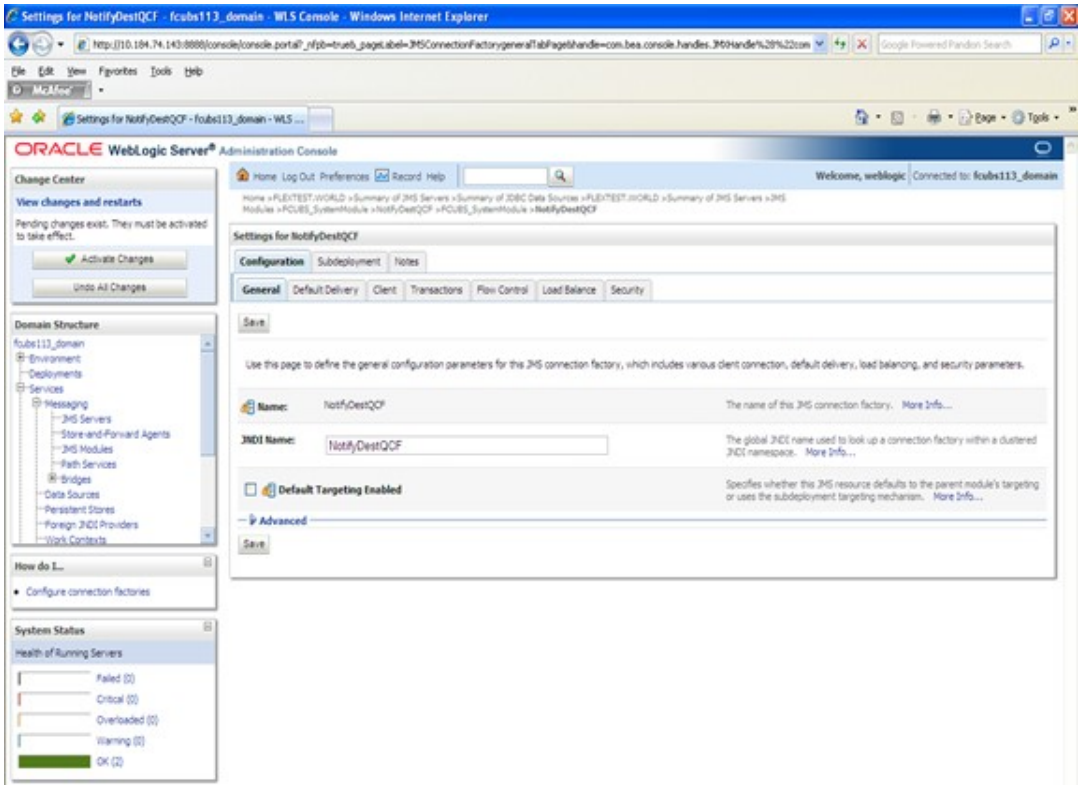


9. Select the **Subdeployments** as **FCIS** from the drop-down list.
10. Under the JMS Servers, check the box against Managed Server.
11. Click **Finish**.

    The message `Connection Factory created successfully` is displayed.

    Settings for the SystemModule_Messages screen is displayed.

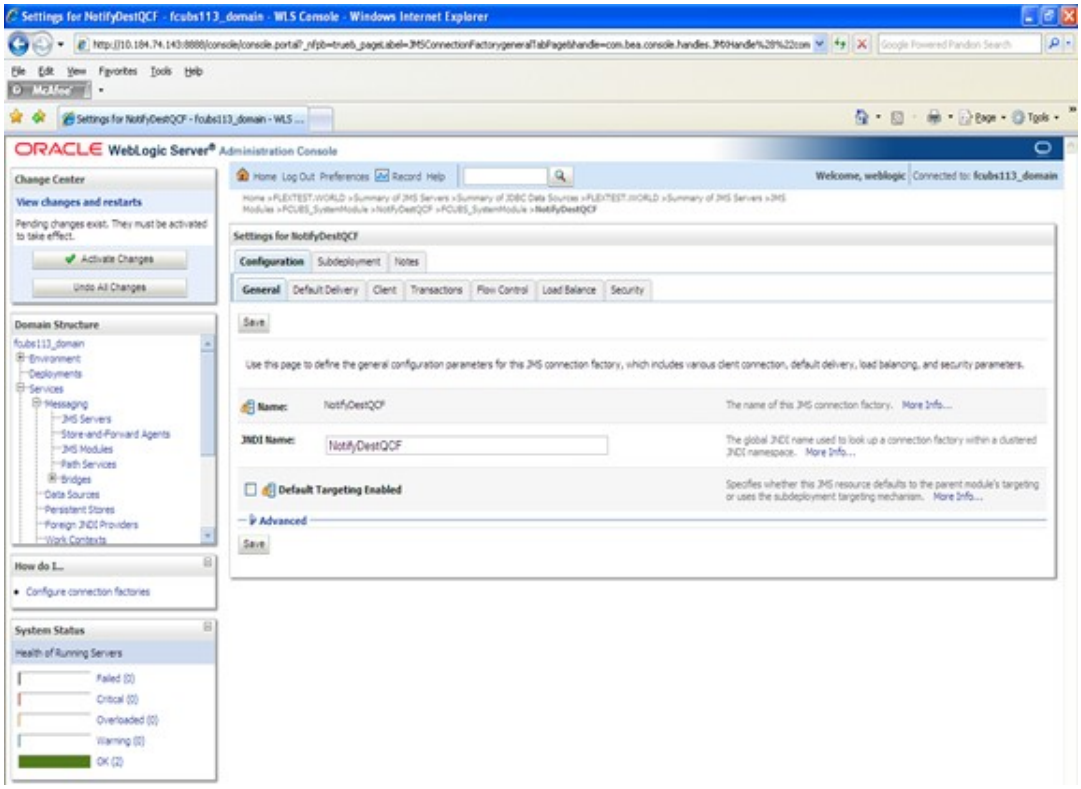
Figure 7-74 Settings for FCUBS_SystemModule - Messages



12. Click the Connection Factory **NotifyDestQCF** to have the XA Connection Factory enabled.

    Settings for NotifyDestQCF screen is displayed.

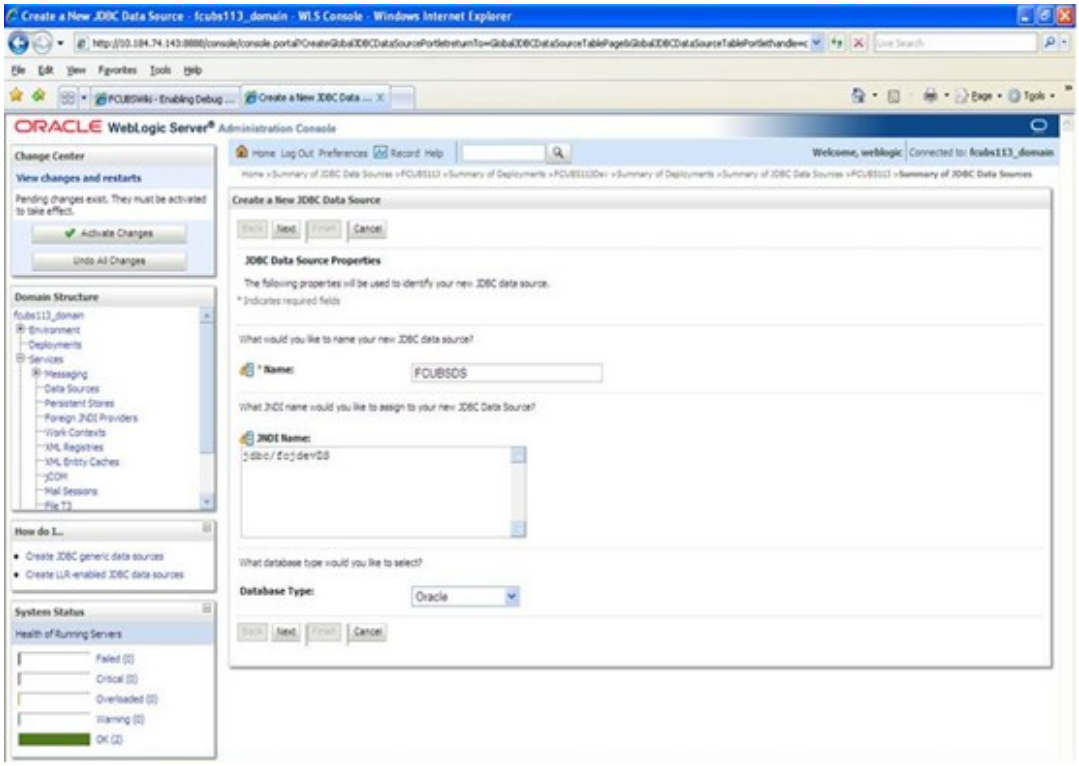
Figure 7-75 Settings for NotifyDestQCF



13. Click the **Transactions** tab.

    Settings for NotifyDestQCF_Transactions screen is displayed.

Figure 7-76 Settings for NotifyDestQCF - Transactions

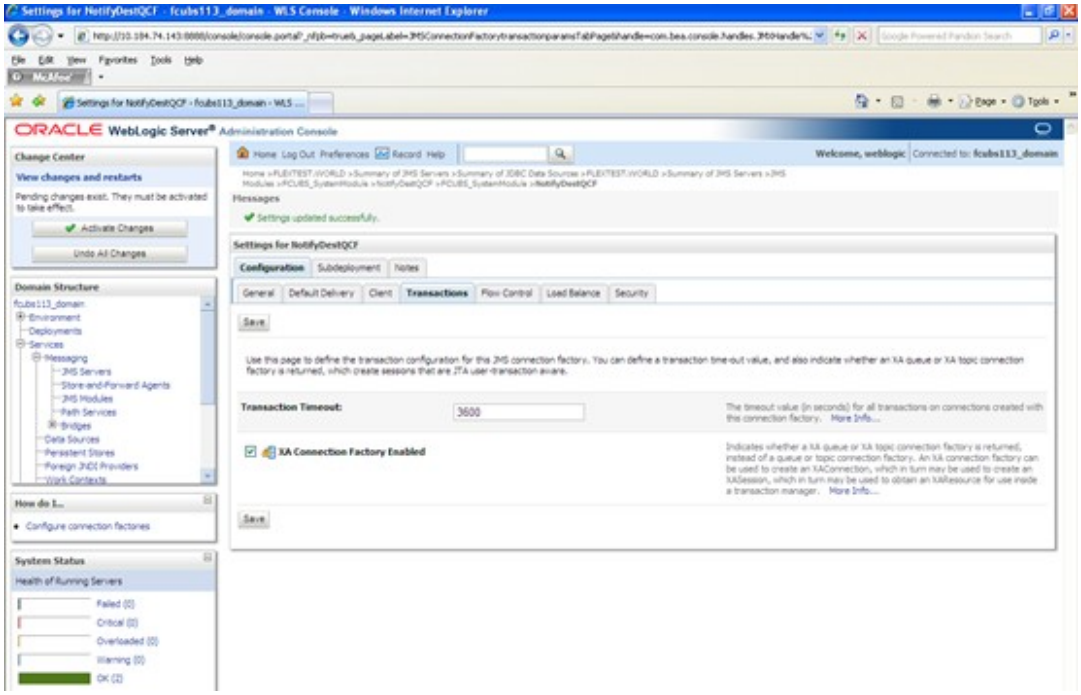


14. On the **Settings for NotifyDestQCF_Transactions** screen, you can define the transaction timeout value.
15. Check the box **XA Connection Factory Enabled**.
16. Click **Save**.

    The message `Settings updated successfully` is displayed.

    Settings for NotifyDestQCF_Messages screen is displayed.

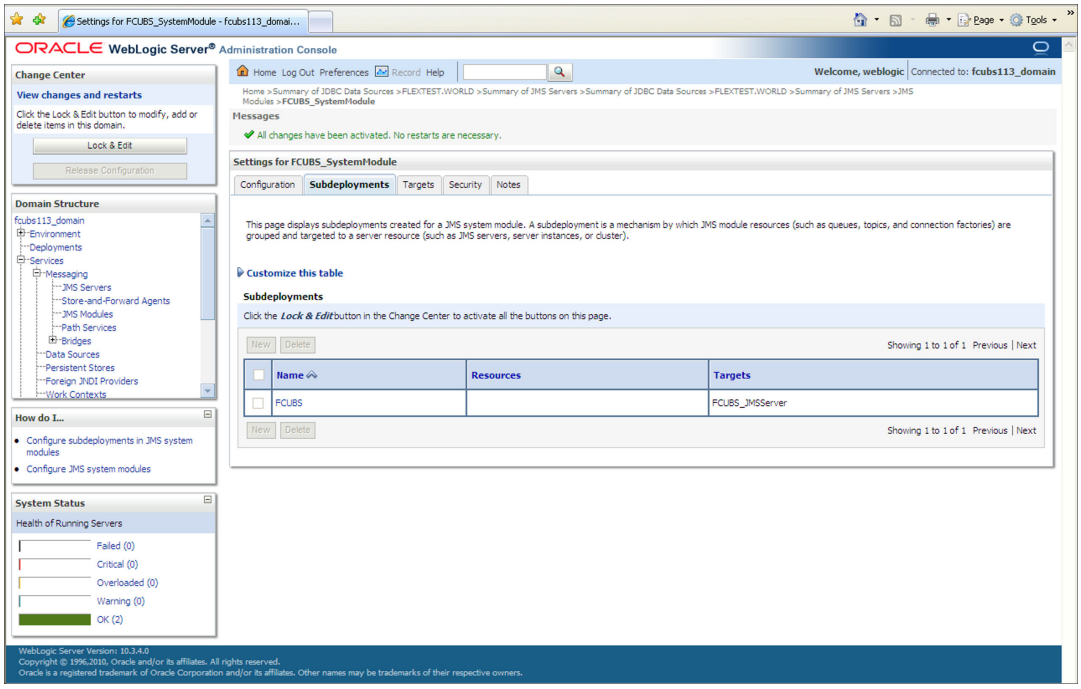
Figure 7-77 Settings for NotifyDestQCF - Messages



17. Click the **Activate Changes** button in the **Change Center** section of the screen to accept the changes made.

    The message `All the changes have been activated. No restarts are necessary.` is displayed.

Figure 7-78 Subdeployments_All Changes Activated



The **JMS Connection Factory** is created.

8

Configure Weblogic Server

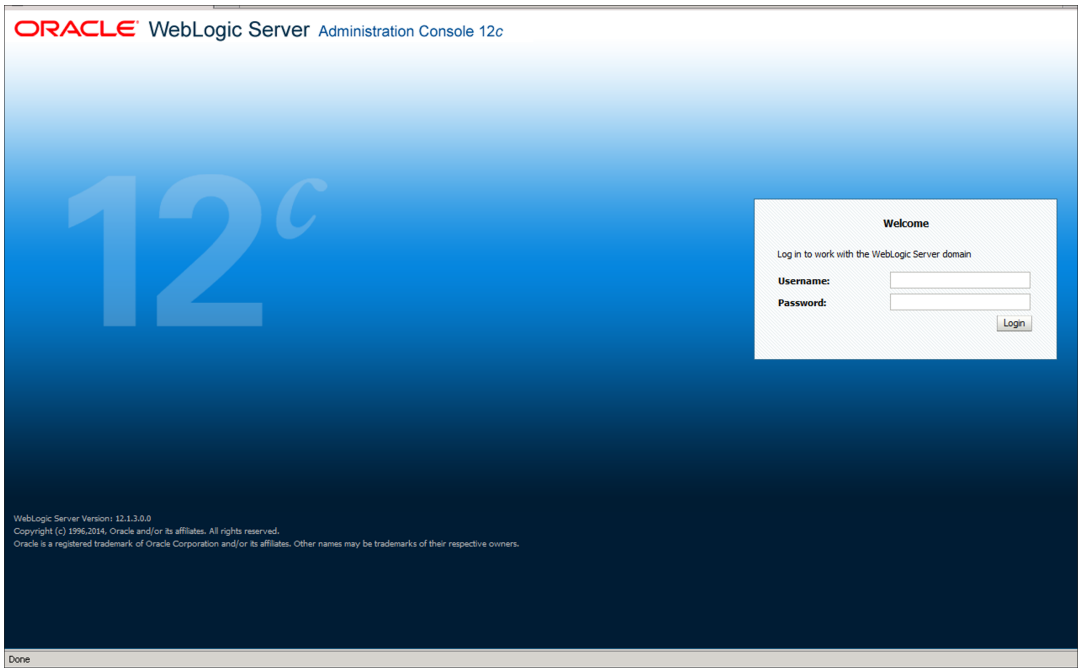
This section explains the systematic instructions to configure the Oracle WebLogic application server

To configure the Oracle WebLogic application server, follow the steps given below:

1. Start the Administrative Console of the WebLogic application server. Enter the Oracle WebLogic Admin Console URL in the address bar in an internet browser. For example:http://10.10.10.10:1001/console

   Oracle Weblogic Server - Welcome screen is displayed.

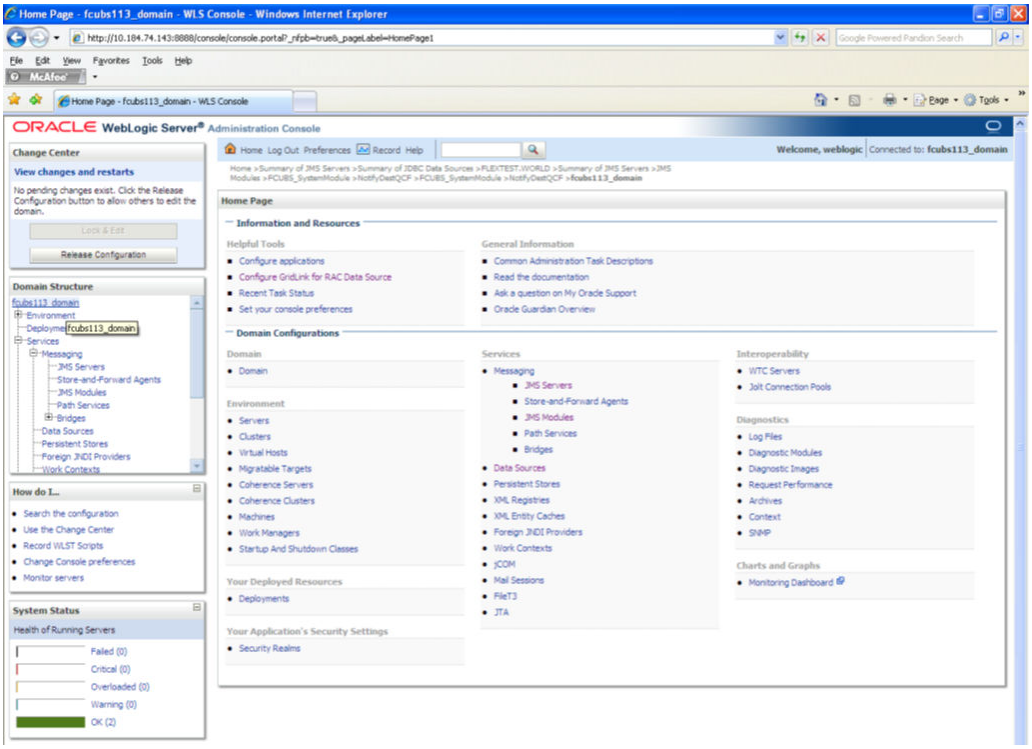
Figure 8-1 Oracle Weblogic Server - Welcome



   a. Specify the WebLogic administrator **Username**, **Password**

   b. Click **Log In**. The **Oracle Weblogic Server - Home Page** screen is displayed.

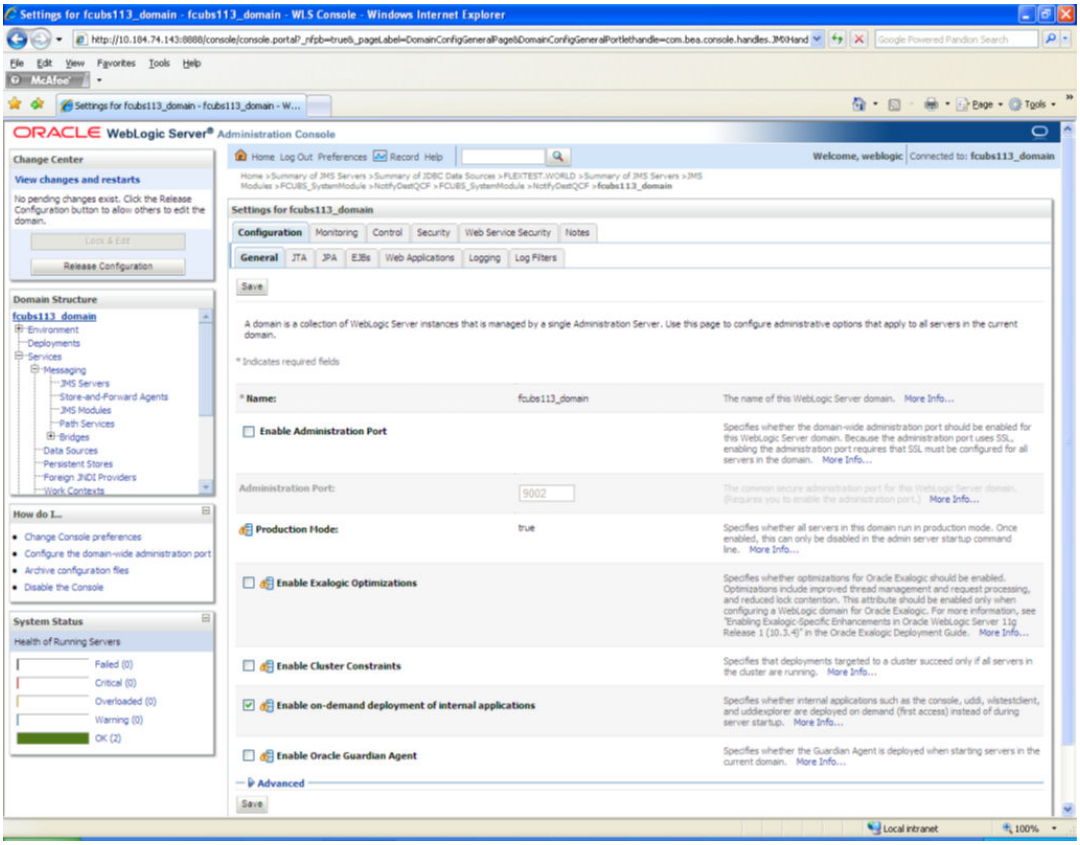
Figure 8-2 Oracle Weblogic Server - Home Page



2. Select the domain from the domain structure as shown below. (Eg: fcubs113_domain).

   Settings for fcubs113_domain screen is displayed.

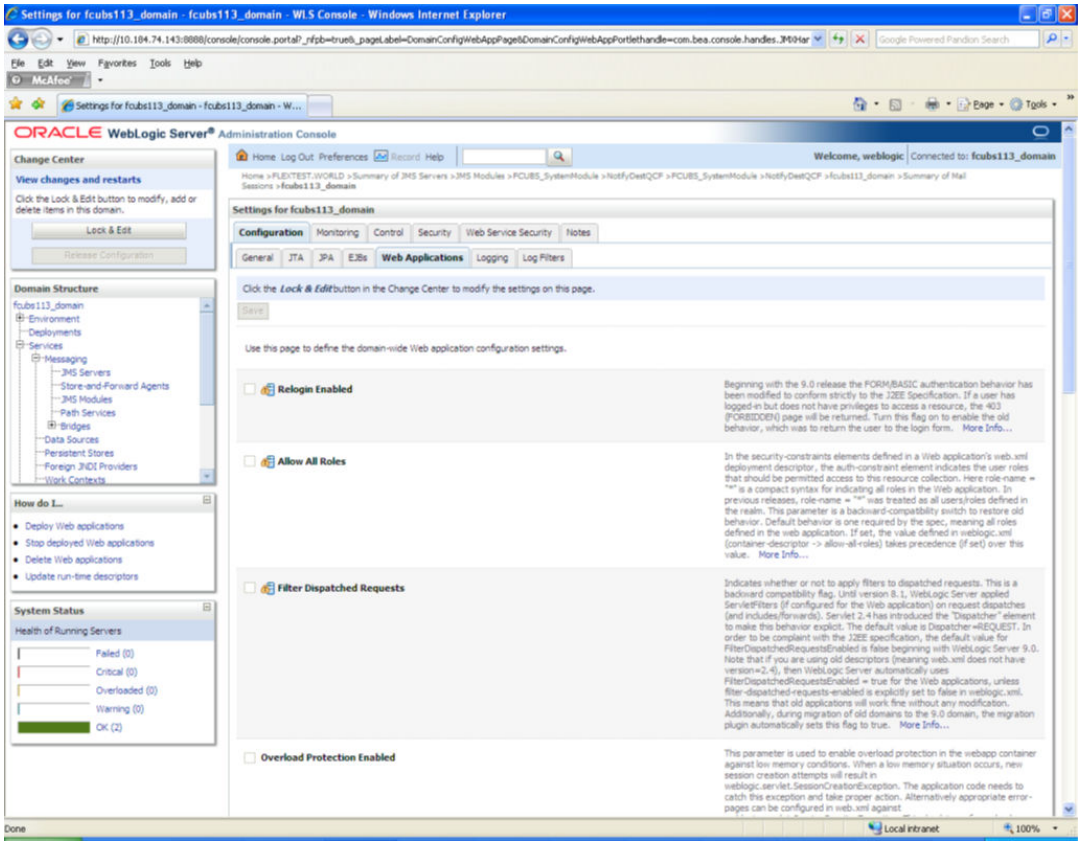
Figure 8-3 Settings for fcubs113_domain



3. Select **Web Applications** from the **Confugurations**.

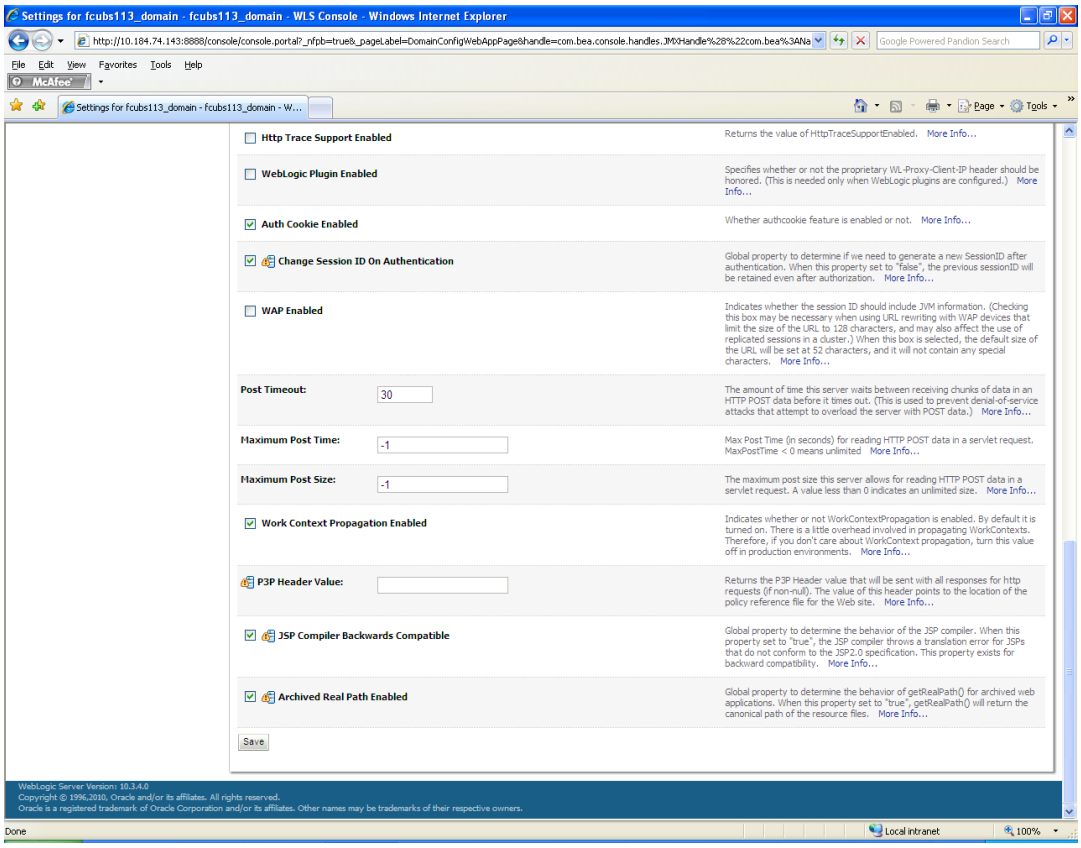
   Settings for fcubs113_domain - Web Applications is displayed.

Figure 8-4 Settings for fcubs113_domain - Web Applications



4. Ensure to select the **JSP Compiler Backwards Compatible** and **Archived Real Path Enabled** options in this screen.

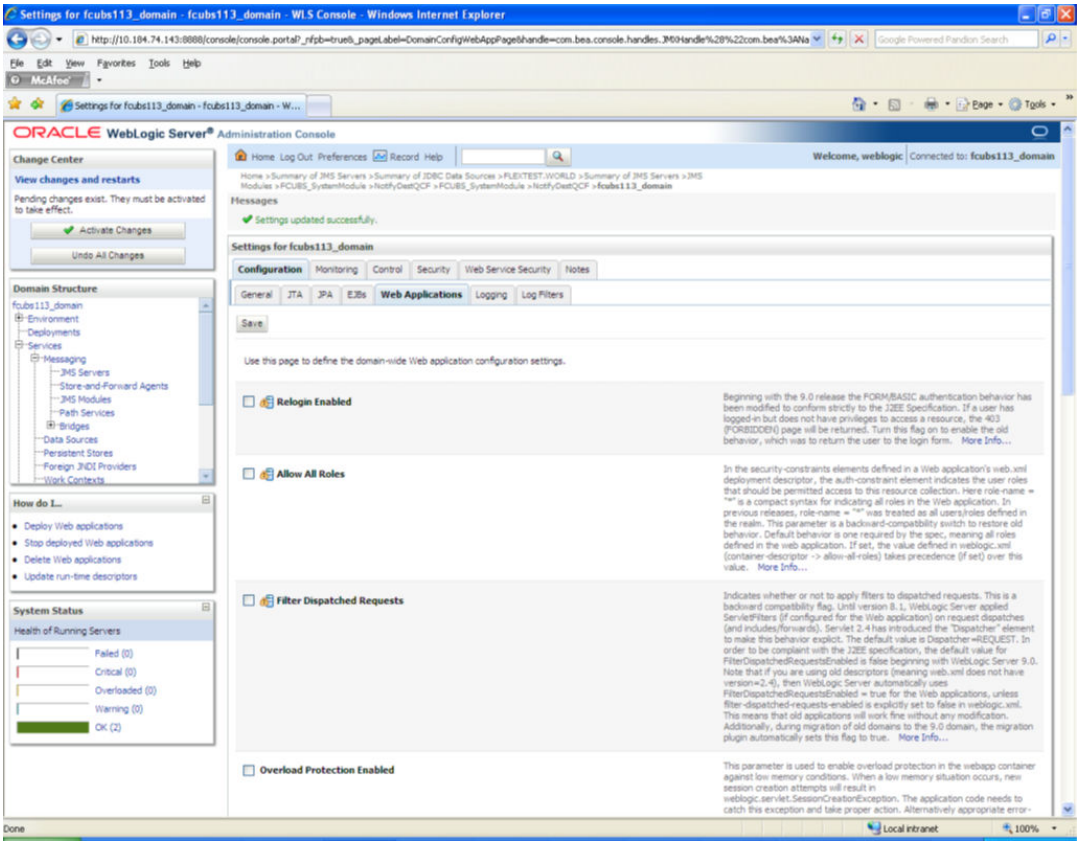
Figure 8-5 Settings for fcubs113_domain - Web Applications Configuration Settings



5. Click on **Save** button and the message `Settings are updated successfully` is displayed.

   Thw **Settings for fcubs113_domain - Messages** screen is displayed.

Figure 8-6 Settings for fcubs113_domain - Messages



6. Click the **Activate Changes** button under the **Change Center**. The message `All the changes have been activated. No restarts are necessary` is displayed.

9

Setup/Configure Mail Session in Weblogic

This topic explains to setup/configure mail sessions in Weblogic.

This section describes the set of configurations changes required in the Oracle Weblogic Server when Oracle Banking Treasury Management is configured to generate and send passwords to users via e-mail.

- Create JavaMail Session
  This topic explains creating the JavaMail session in the Oracle weblogic server.
- Configuration of the TLS/SSL Trust Store for Weblogic Server

9.1 Create JavaMail Session

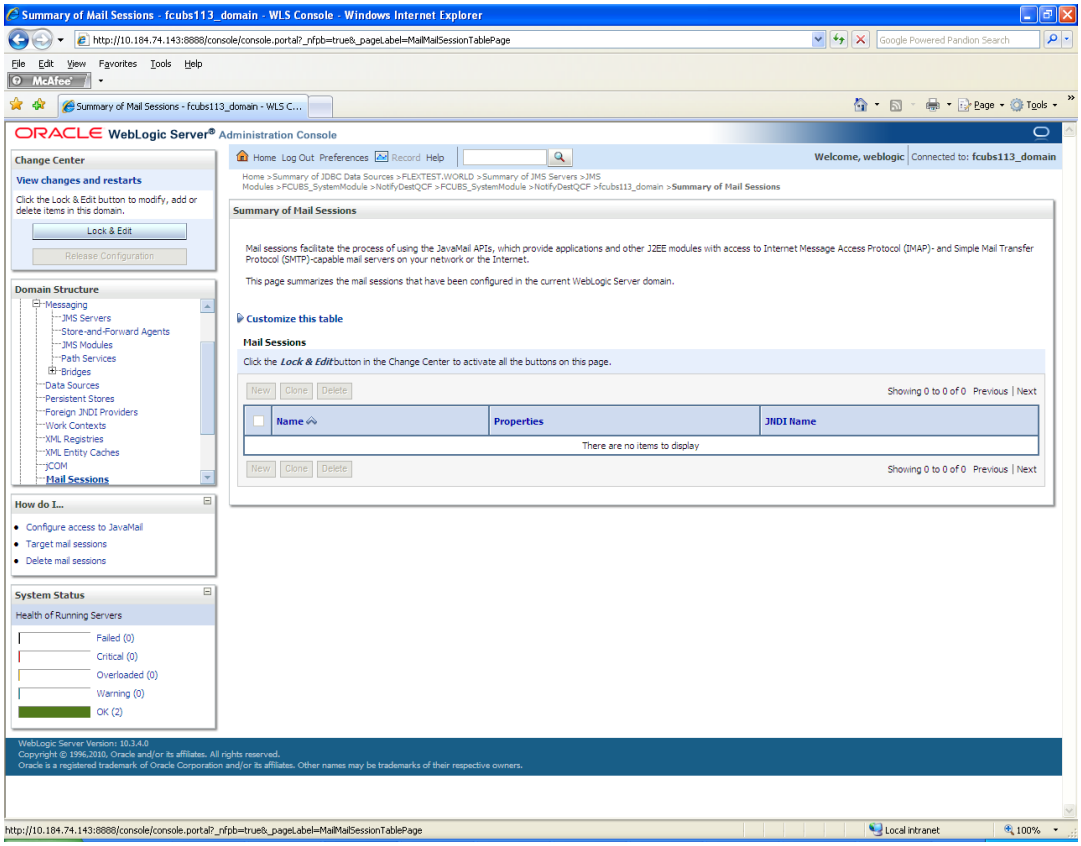
This topic explains creating the JavaMail session in the Oracle weblogic server.

To configure the mail session, perform the follwing:

1. Click **Mail Session** from the Domain Structure in the left pane.

   The following screen is displayed.

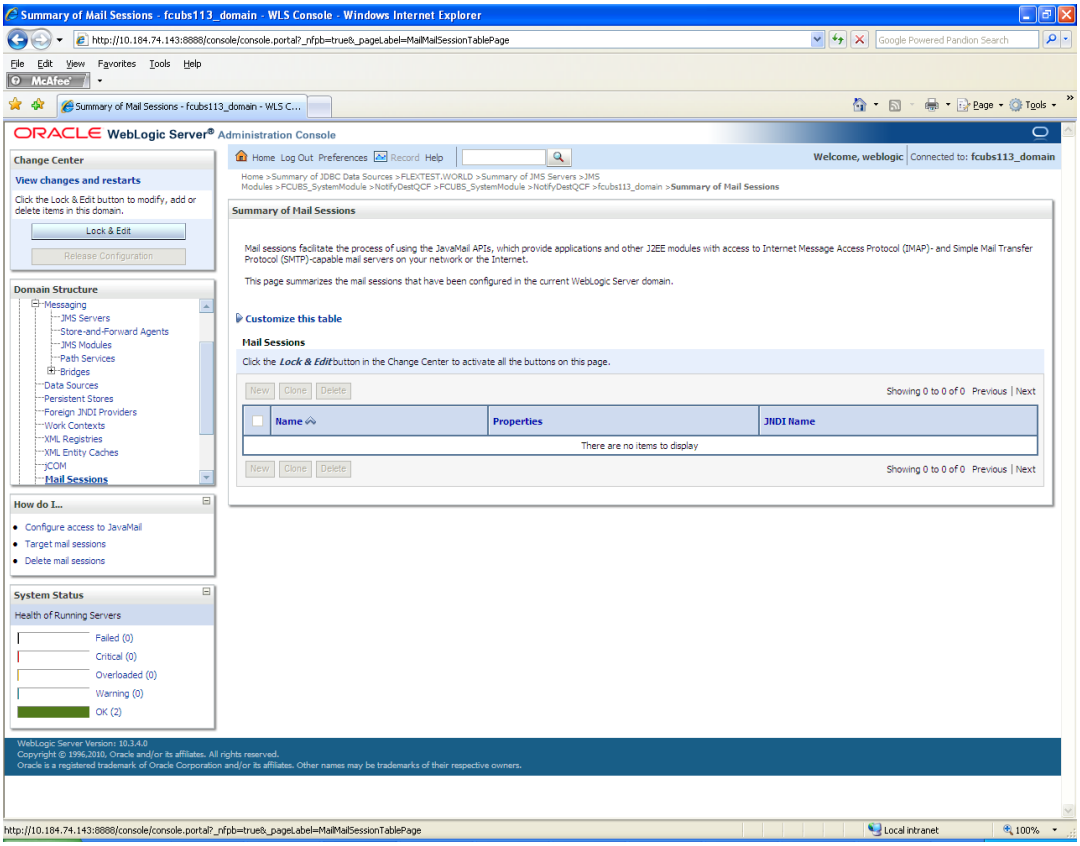
Figure 9-1 Summary of Mail Sessions



2. Click **Lock & Edit** in the Change Center to enable all the buttons in this page.

   The following screen is displayed.

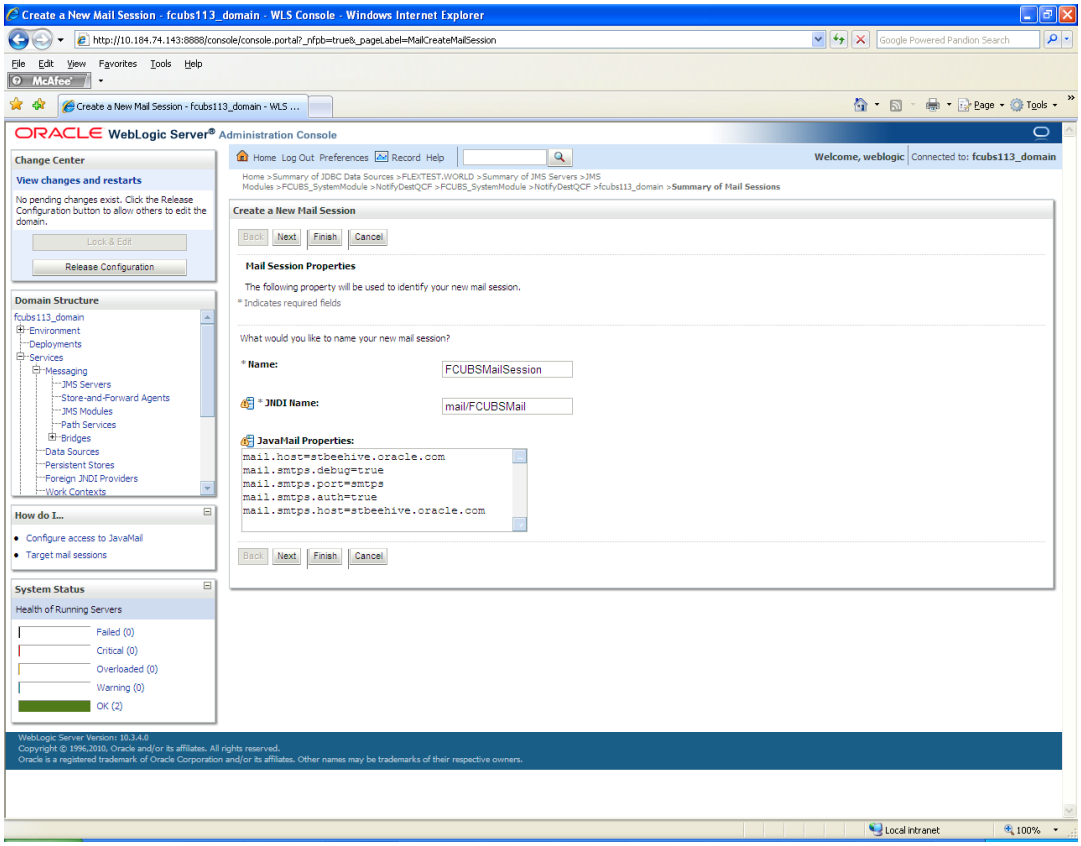
Figure 9-2 Summary of Mail Sessions_Lock & Edit



3. Click **New** to create a new session.

   The following screen is displayed.

Figure 9-3 Create a New Mail Session



4. Specify the Name of the Mail Session.

   Sample details are given below:

   - **Name**: FCUBSMailSession
   - **JNDI Name**: mail/FCUBSMail

   STOP This JNDI name needs to be maintained in fcubs.properties file with encrypted format.

   Java Mail Properties

   mail.host=<HOST_MAIL_SERVER>

   Eg: samplename.mail.com

   mail.smtps.port=<SMTPS_SERVER_PORT>

   Eg: 1010

   mail.transport.protocol=<MAIL_TRANSFER_PROTOCOL>

   Eg: smtps

   mail.smtps.auth=true

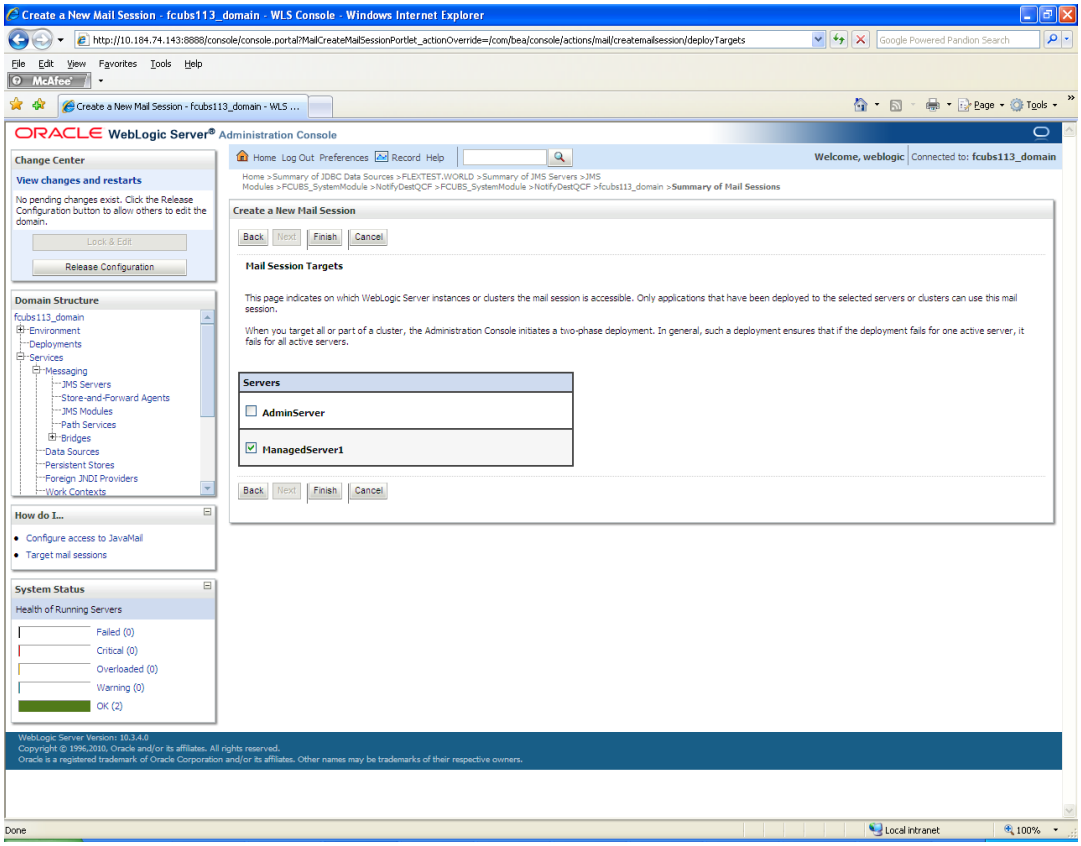
   mail.smtps.host==<HOST_SMTPS_MAIL_SERVER>

   Eg: samplename.mail.com

5. Click **Next**.

   The Following screen is displayed.

Figure 9-4 Create a Mail Session - Mail Session Targets



6. Select the **Required Servers** and click **Finish** to complete the configuration.

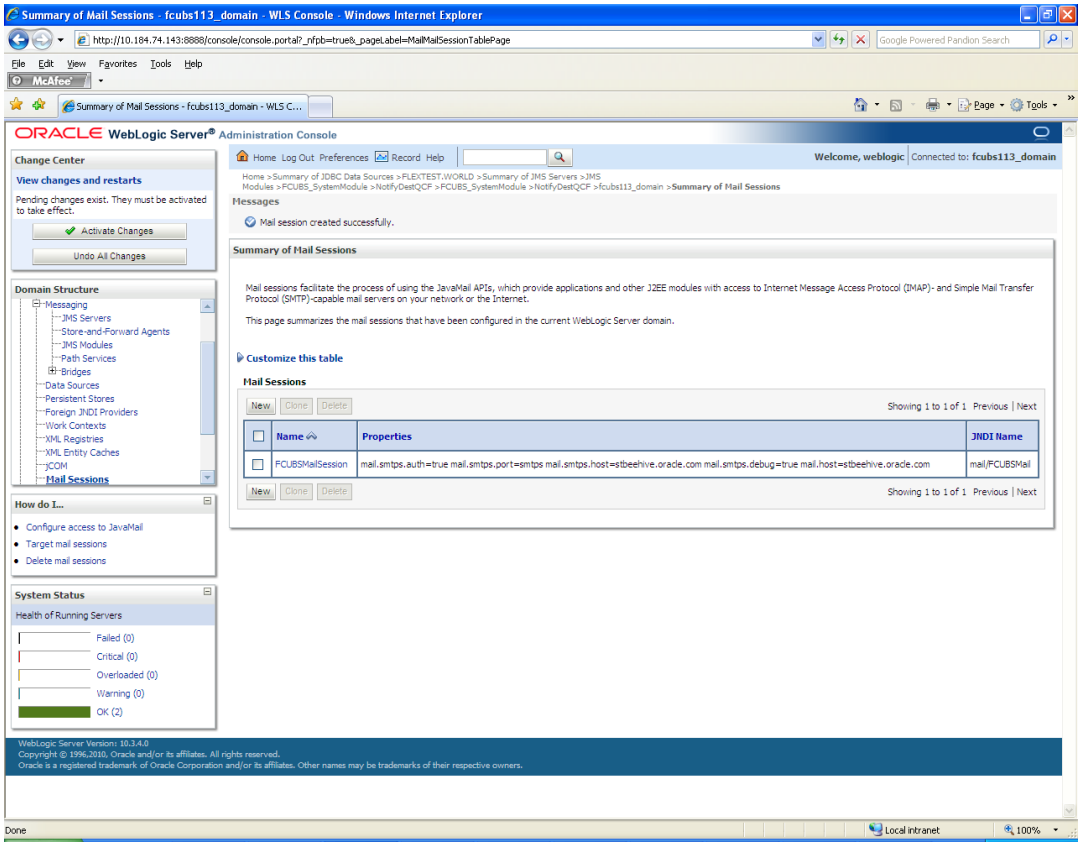
   fcubs.properties file needs to be updated with the encrypted values of

   - SMTP_HOST
   - SMTP_USER
   - SMTP_PASSWORD
   - SMTP_JNDI

7. Click **Active Changes** to activate the current mail session settings.

   The following screen is displayed.

Figure 9-5 Mail Sessions_Activate Changes



9.2 Configuration of the TLS/SSL Trust Store for Weblogic Server

As described in the previous section, Oracle Banking Trade Finance uses SMTPS to send outgoing mails. SMTPS uses SSL to ensure transport-level security of the mail messages and hence, the certificate of the mail server needs to be imported into the trust store(s) of the Managed Servers where Oracle Banking Trade Finance is deployed.

The certificate of the mail server needs to be specifically imported into the trust store configured for the Managed Server(s), as configured in the Oracle Banking Treasury Management Installation guide. For more information, refer to *SSL Configuration On Weblogic*.

For further details on importing the certificate of the mail server into the trust store, refer to the documentation for the Sun Java keytool utility (Key and Certificate Management tool).