Oracle FLEXCUBE Direct Banking

IBM WebSphere 8.5 Installation Manual

Release 12.0.3.0.0

Part No. E52543-01

April 2014

IBM WebSphere 8.5 Installation Manual
April 2014

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India
Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax:+91 22 6718 3001
www.oracle.com/financialservices/

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products,orservices.

Contents

1. Preface

Intended Audience
This document is primarily targeted at

- Oracle FLEXCUBE Direct Banking Development Teams
- Oracle FLEXCUBE Direct Banking Implementation Teams
- Oracle FLEXCUBE Direct Banking Implementation Partners

Documentation Accessibility
For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to OFSS Support
https://support.us.oracle.com

Structure

This document consists of the following chapter

Chapter 3, "Websphere v7 features”

This chapter discusses the new features of WAS.

Chapter 4, "Pre-requisites”
This chapter discusses the necessary installations/requirements prior to deployment on IBM WAS 8.5.

Chapter 5, "Application Server Configuration"
This chapter discusses configuration of application server, creation of instance of IBM WAS 8.5.

Chapter 6, "Deploying Application"
This chapter discusses the steps to be followed to deploy the application modules on IBM WAS 8.5.

Chapter 7, "Setup JDBC Data Source and Connection Pooling"

The steps for JDBC Data source creation and Connection Pool creation are discussed in this chapter.

Chapter 8, “Setup of WebSphere MQ entities”

The steps for WebSphere MQ queues and queue connections.

Chapter 9, "Appendix"

All the miscellaneous issues, for reference purpose can be found in this section.

Related Information Sources

For more information on Oracle FLEXCUBE Direct Banking Release 12.0.3.0.0, refer to the following documents:

- Oracle FLEXCUBE Direct Banking Licensing Guide

2. Abbreviations

FCDB	Oracle FLEXCUBE Direct Banking
HTTP	Hyper Text Transfer Protocol
J2EE	Java 2 Enterprise Edition
WAS	IBM WebSphere Application Server
FCDB BASE DIR	FCDB Base Working Directory

3. Websphere v7 features

- Extends the brand experience to social networking sites
- Improves customer experience with support for advanced payment options and digital coupons
- Delivers localized experience for Brazil and China
- Provides powerful business user tools to centrally manage targeted marketing campaigns and promotions across Web sites
- Helps reduce total cost of implementation by easing data loading and integration with sub-systems
- Enables 64-bit WebSphere Application Server support with select operating systems
- Provides WebSphere eXtreme Scale integration to support large scale high performance caching

Websphere v7 release notes –

1. **Simplified Development**

Increase developer productivity with a proven, standards-based platform. WebSphere Application Server V7 offers enhanced support for standards, emerging technology and a choice of development frameworks that simplifies programming models to increase developer productivity. Integral tooling makes the most of existing technology skills for rapid time-to-value.

- Java EE 5 certification, EJB 3.0 support and Java Persistence API (JPA) and Java Development Kit (JDK) 6.0, deliver simplified programming models for building reusable persistent object.
- Web services support, including JAX-WS, SOAP 1.2, MTOM, XOP, WS-ReliableMessaging, WS-Trust, WS-SecureConversation, WS-Policy, and Kerberos Token Profile, simplifies interoperability in mixed environments.
- Web 2.0 support (via Feature Pack for Web 2.0) extends Service Oriented Architecture (SOA) by connecting external Web services, internal SOA services, and Java Platform Enterprise Edition (JEE) objects into highly interactive Web application interfaces.
- Session Initiation Protocol (SIP) servlets simplify development with standardized support for interactive user sessions involving real-time multimedia elements such as voice, video, instant messaging and online games.
- Spring has certification for use with WebSphere Application Server to enable module-based programming.
- Single-step installation and configuration, wizards and default configurations, and easy- to-navigate documentation that includes extensive use of sample code help ensure fast and smooth start-up.
- WebSphere Application Server Feature Packs simplify the adoption of new standards such Services Component Architecture (SCA), and improve consumability by enabling users to selectively take advantage of new standards and features while maintaining a more stable internal release cycle.

2. **High Performance**

Keep your business agile in a constantly changing environment. WebSphere Application Server V7 provides a fast, reliable, available, secure and scalable environment that enables development and application efficiencies so that businesses can do more with fewer resources

Unparalleled Performance
Do more with less. WebSphere Application Server V7 delivers optimized runtime performance through provisioning, Web services, and EJB3 enhancements which can result in fewer energy consuming processors performing the same workloads of previous versions.

3. **Rock-solid Security**

Rest assured, applications and data are secure from attack. Out-of-the-box security configurations and user registry, compliance with government standards, and stringent Web services security. New security capabilities add deeper levels of management, user governance and auditing to decrease system vulnerabilities while maximizing developer productivity.

- WebSphere Security Domains deliver greater granularity, flexibility and control over users and infrastructure to increase administration efficiencies
- Security auditing capabilities ensure security compliance to ease development
- Broader implementation of Kerberos improves security interoperability with other applications and environments while increasing developer productivity
- Security Assertion Markup Language (SAML) token support for web services security provides for secure, interoperable Web services

4. **Intelligent Management**

Anticipate and adjust to mission critical issues. WebSphere Application Server V7 offers a simplified infrastructure, flexible and effective application control and runtime efficiency so you can respond to evolving business needs with infrastructure and application intelligence.

- Runtime provisioning and OSGi technology dynamically selects only the needed functions for memory and space reducing the application server footprint
- Flexible Management features in WebSphere Application Server Network Deployment improve efficiencies of administration and allow remote production resource management of WAS Base and WAS Express
- Administration tools enable attention to the deepest level of security providing fine grained security management levels
- New WebSphere Business Level Applications (WBLA) expands the notion of "application" beyond Java EE 5 to significantly improve the management of multi-component applications, simplifying administrative tasks
- Improved Console Command Assistant, easier security configuration and database connectivity, wizards and a stand-alone thin administration client enable efficient management of the deployment environment.

5. Investment Protection

Protect investments in applications through reuse and integration. From service-enabling legacy assets to inventing new ones, our technology makes your business accessible to new users in innovative ways, giving you immediate insight and interaction with partners, suppliers and customers and increasing your return on investment.

- New Web services support including WS-Business Activity, WS-Notification and WS-I Basic Security Profile, helps you more securely extend your reach and gives better application portability and control. On top of already extensive Web services support, Web 2.0 and a powerful Java Messaging Service (JMS) engine help you extend the reach of your existing applications and maximize asset utilization.
- Pre-integrated support for WebSphere MQ and tight integration with WebSphere ESB - the combination of these products form a powerful Enterprise Service Bus that can integrate the most diverse set of applications and environments
- New Deployment manager enables you to manage previous versions of WebSphere Application Server so that you can adopt newer infrastructure as your plans require
- New multi-cell support and compatibility features offer the ability to seamlessly adopt newer infrastructure and run applications developed in previous Java Enterprise Edition versions (backwards compatible to JEE v1.2) for WebSphere Application Server to eliminate intensive cost and resource requirements

4. Pre- requisites

1. All the relevant software requirements mentioned in the sheet “Oracle_FLEXCUBE_Direct_Banking_Software_Stack” must be installed.
2. IBM WebSphere v7 must be installed on the system.
3. Flexcube Direct Banking Application is installed using Oracle FCDB Installer.
4. The document expects the user to have fair knowledge of application deployment on IBM Websphere. The document only explains deployment and configuration of Oracle FLEXCUBE Direct Banking on IBM Websphere; IBM Websphere documentation should be referred for details on IBM Websphere.

5. Application Server Configuration

5.1 Creating and Managing a Server Instance

Perform following steps for creating the WAS instance.

1) Go to the Admin console
2) Enter User Id as “WEBSPHERE”.

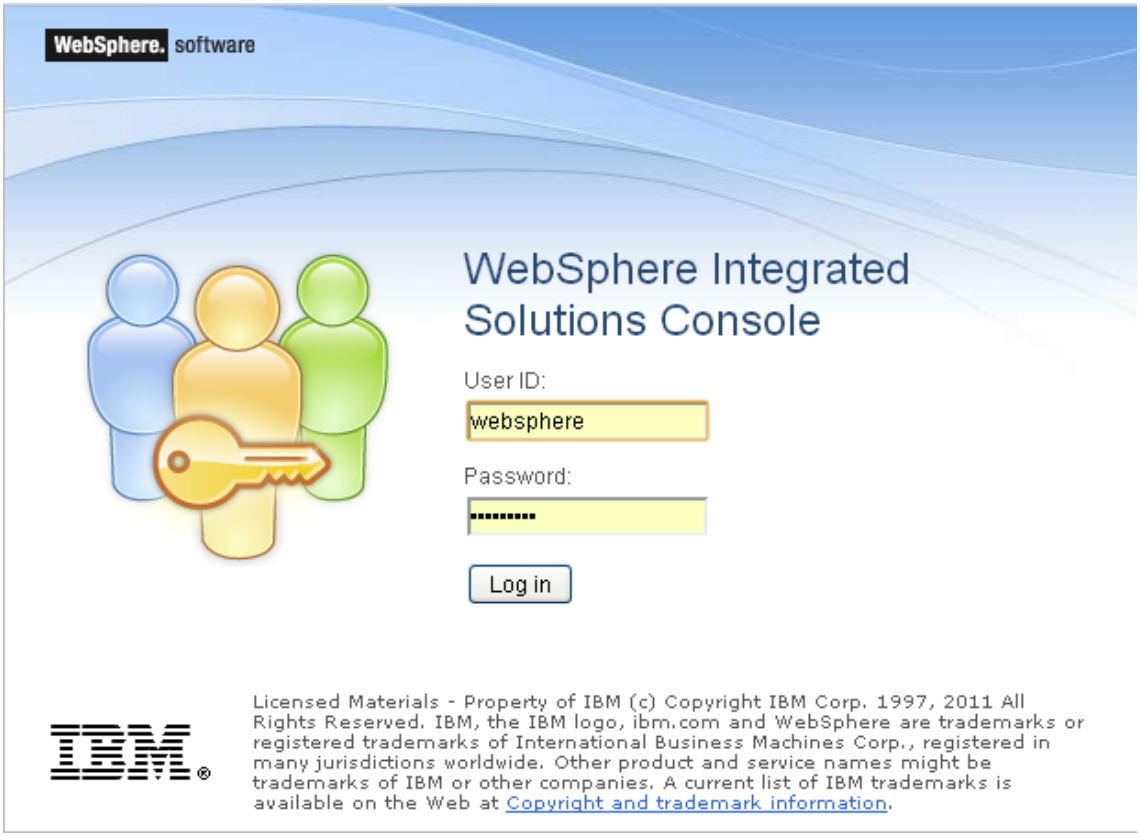


Figure 5.1.1

3) Following Menu will be displayed.

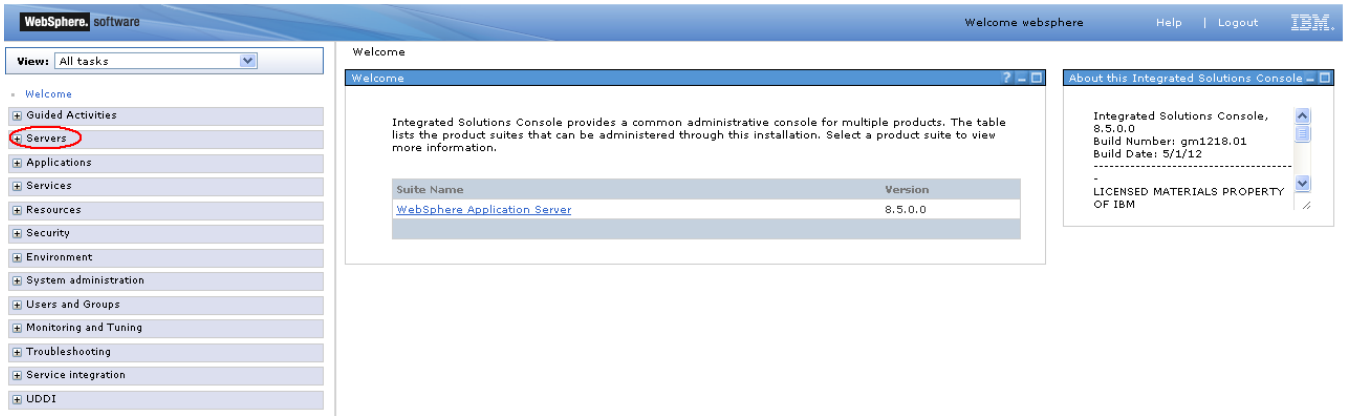


Figure 5.1.2

4) Go to Servers -> Server Types->Application servers.
Click on the server to be configured from the list. Screen as displayed in *Figure 4.1.3* will appear.

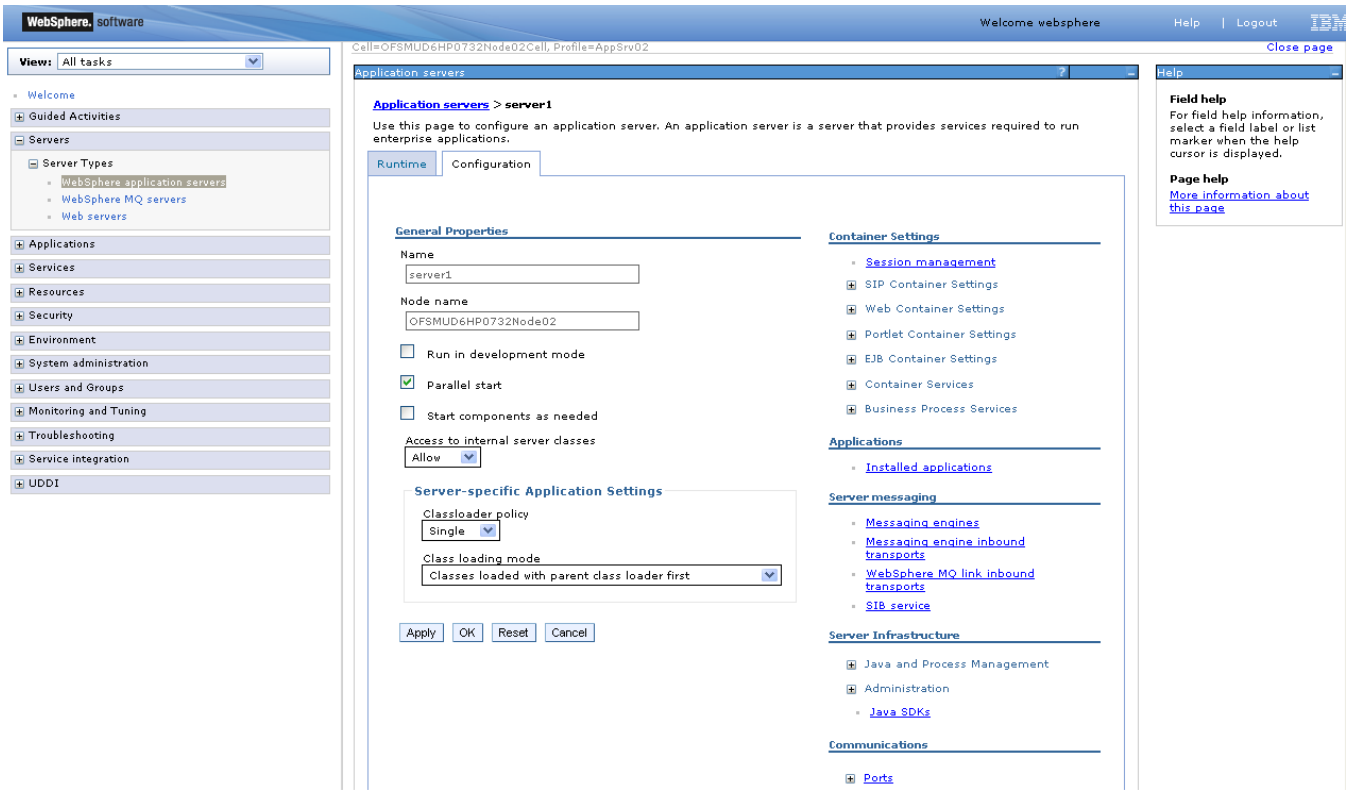


Figure 5.1.3

5) Select “single” as classloader policy. Click Apply.

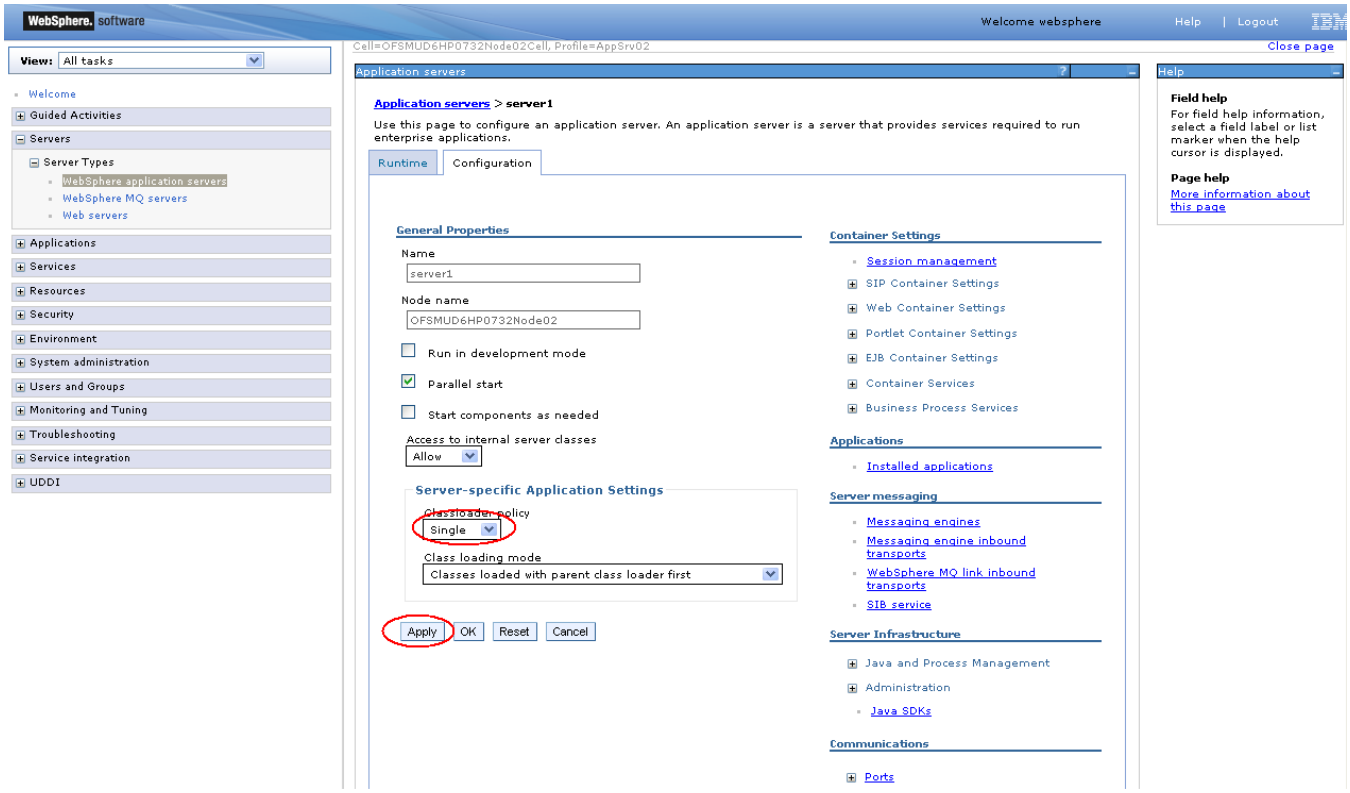
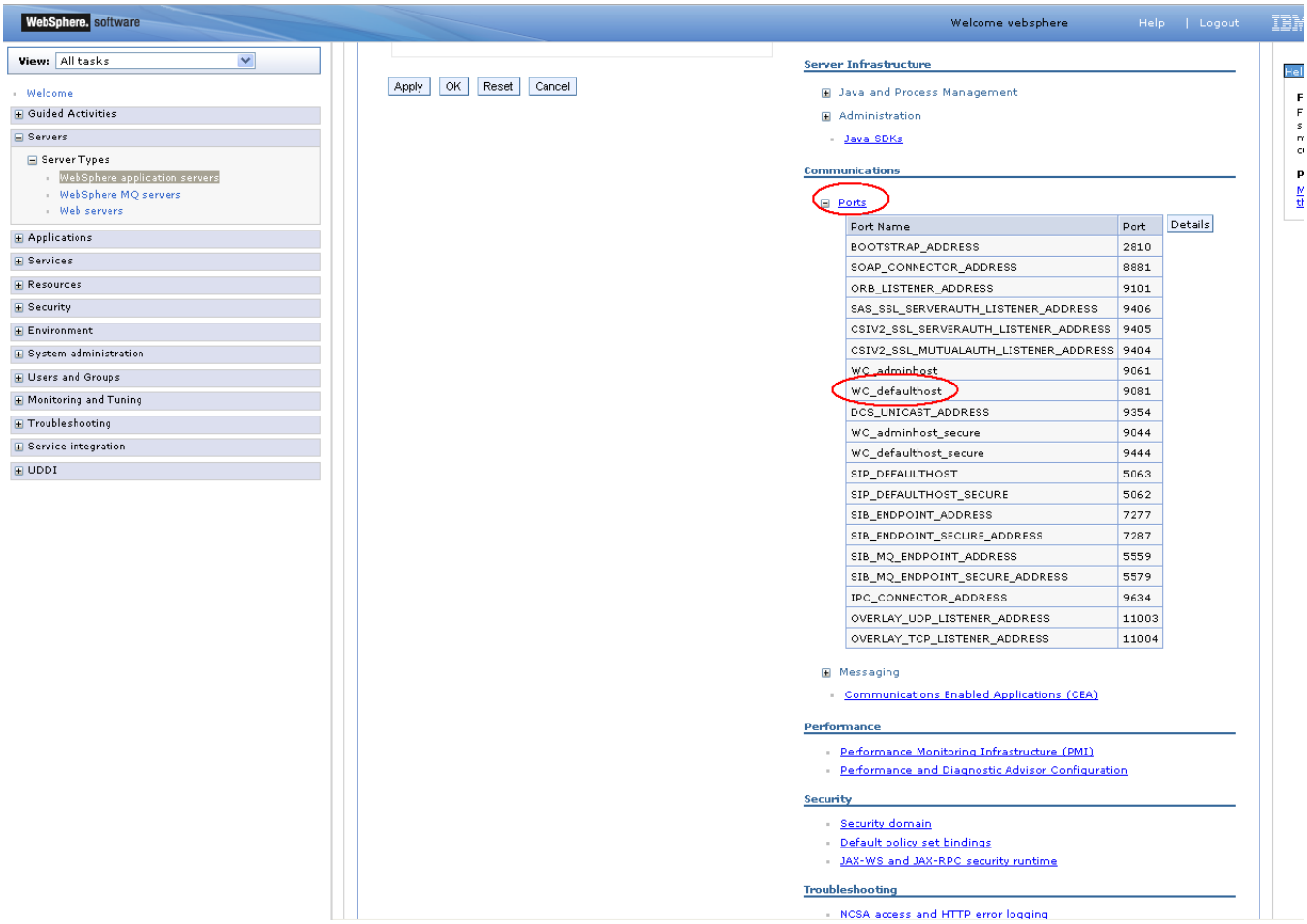
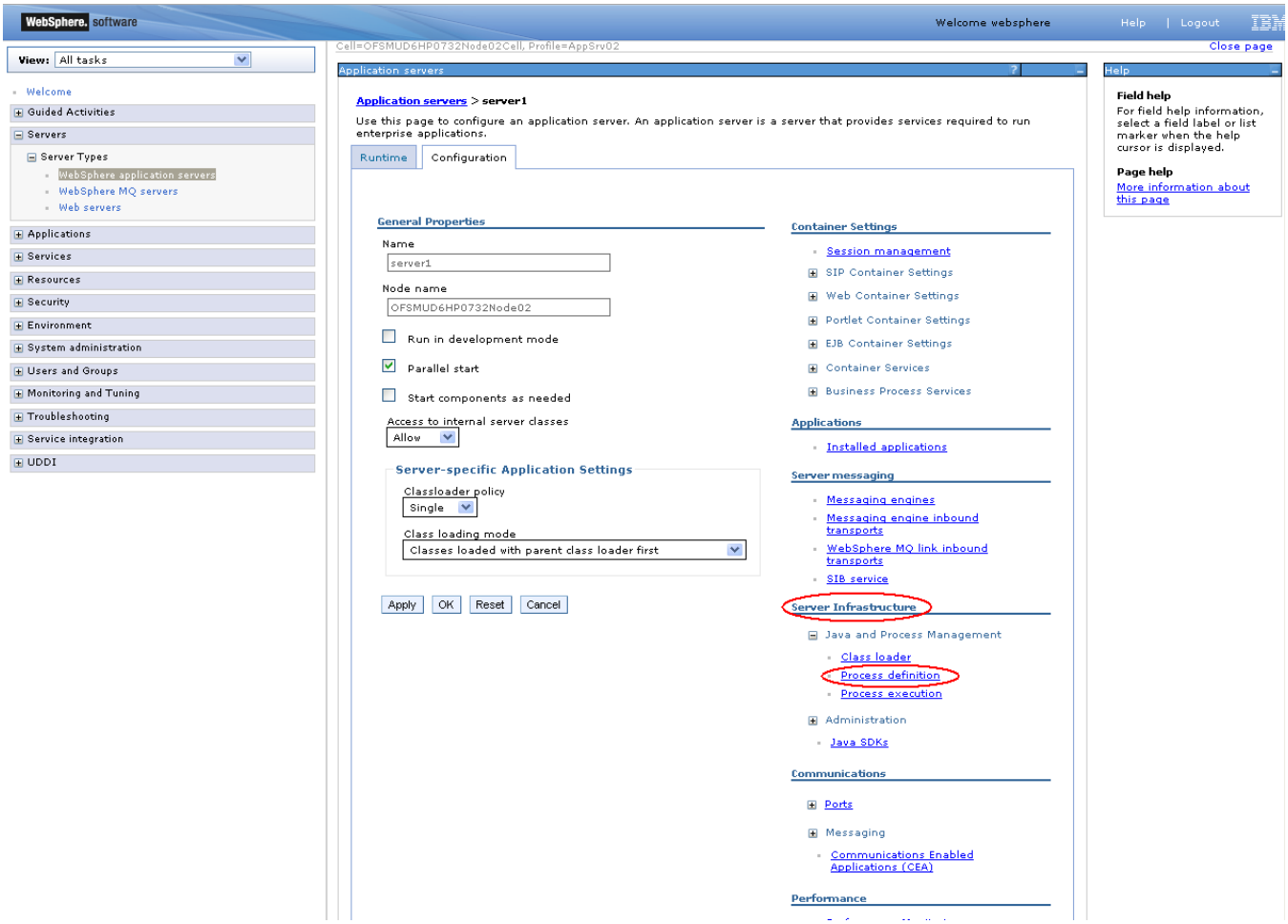


Figure 5.1.4

6) On the same screen, find “Port” section, expand the same to find WC_defaulthost port (termed as default host port in document). Note down the port for later use.


WebSphere. software
Welcome websphere
Help | Logout
View: All tasks
Welcome
Guided Activities
Servers
Server Types
WebSphere application servers
WebSphere MQ servers
Web servers
Applications
Services
Resources
Security
Environment
System administration
Users and Groups
Monitoring and Tuning
Troubleshooting
Service integration
UDDI
Apply
OK
Reset
Cancel
Server Infrastructure
Java and Process Management
Administration
Java SDKs
Communications
Ports
Port Name
Port
Details
BOOTSTRAP_ADDRESS 2810
SOAP_CONNECTOR_ADDRESS 8881
ORB_LISTENER_ADDRESS 9101
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS 9406
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS 9405
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS 9404
WC_adminhost 9061
WC_defaulthost 9081
DCS_UNICAST_ADDRESS 9354
WC_adminhost_secure 9044
WC_defaulthost_secure 9444
SIP_DEFAULTHOST 5063
SIP_DEFAULTHOST_SECURE 5062
SIB_ENDPOINT_ADDRESS 7277
SIB_ENDPOINT_SECURE_ADDRESS 7287
SIB_MQ_ENDPOINT_ADDRESS 5559
SIB_MQ_ENDPOINT_SECURE_ADDRESS 5579
IPC_CONNECTOR_ADDRESS 9634
OVERLAY_UDP_LISTENER_ADDRESS 11003
OVERLAY_TCP_LISTENER_ADDRESS 11004
Messaging
Communications Enabled Applications (CEA)
Performance
Performance Monitoring Infrastructure (PMI)
Performance and Diagnostic Advisor Configuration
Security
Security domain
Default policy set bindings
JAX-WS and JAX-RPC security runtime
Troubleshooting
NCSA access and HTTP error logging

7) On the same Screen, locate section “Server Infrastructure”, expand the option “Java and Process Management“ and click on the link “Process Definition”.



8) On the “Process Definition” screen, click on the link “Java Virtual Machine”

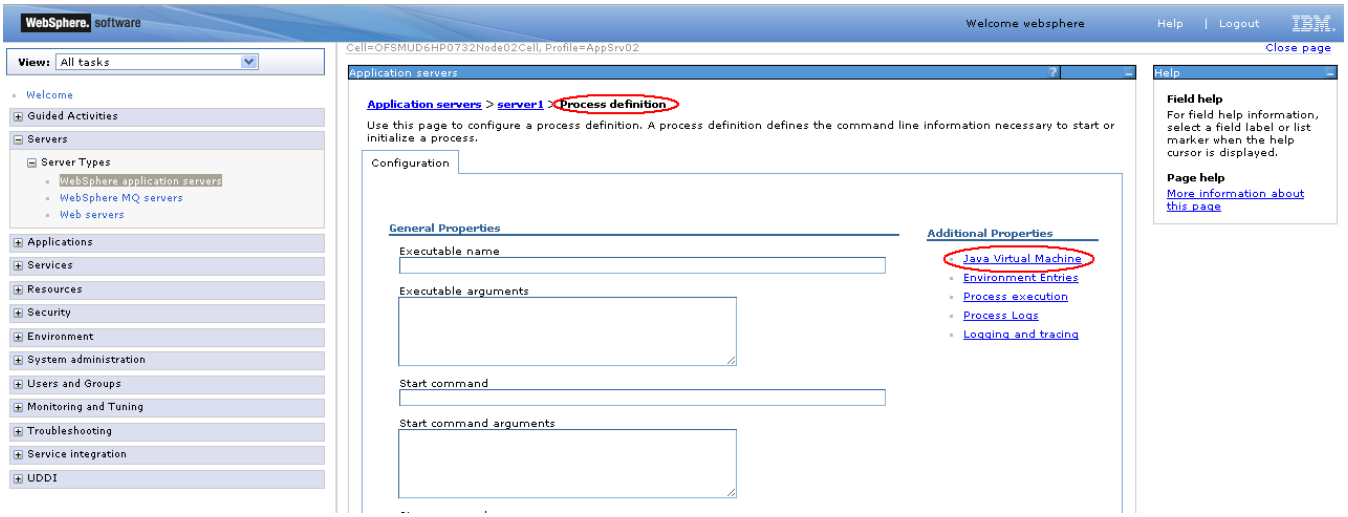


Figure 4.1.5

9) On the “Java Virtual Machine” screen, click on “Configuration” tab and then click on the link “Custom properties”.

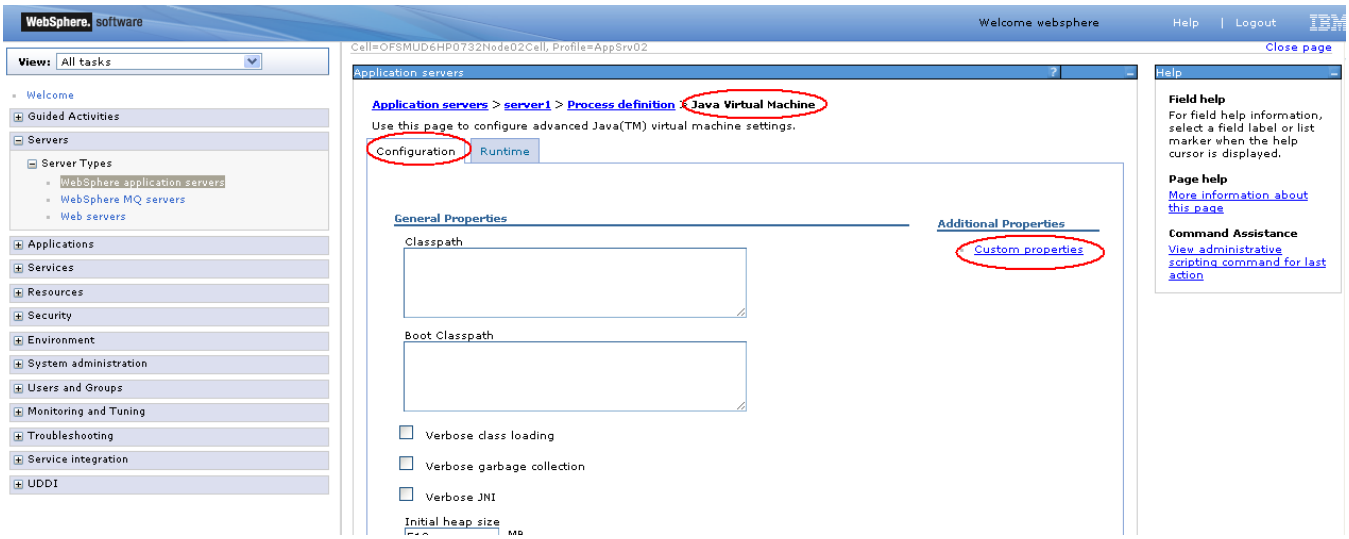


Figure 5.1.6

10) On the “Custom Properties” screen, click on “New” to configure a new custom property.

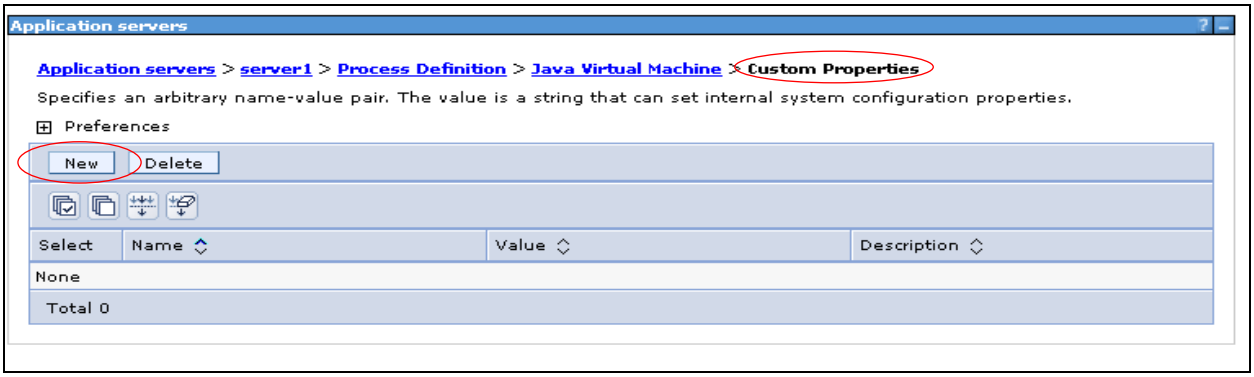


Figure 5.1.7

11) Add the property name, value and click “OK”.

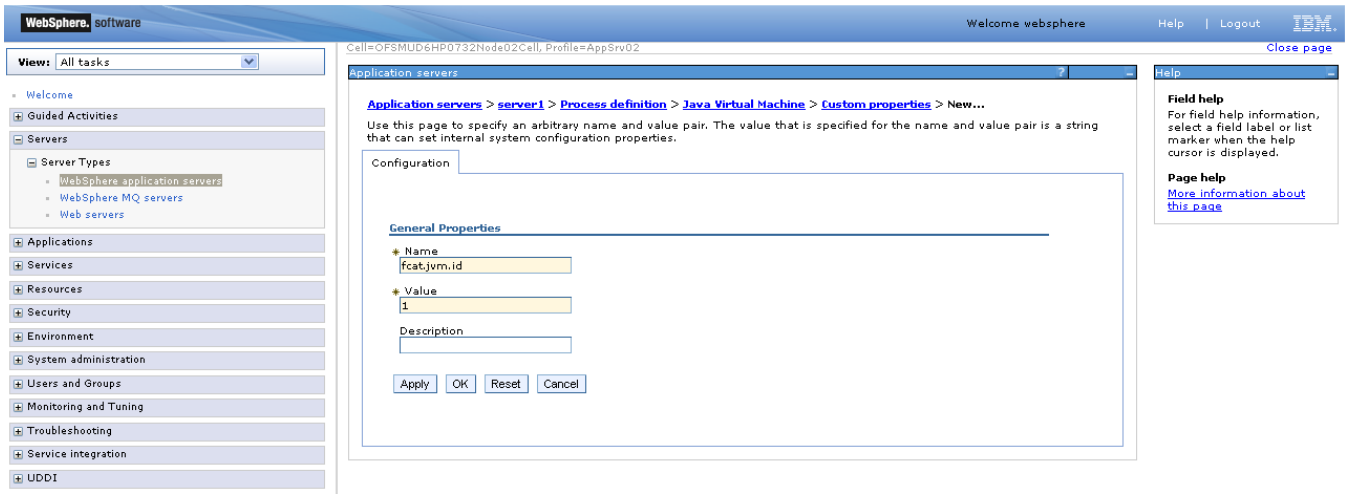


Figure 5.1.8

12) Add all below mentioned properties by following steps 11 - 12.

Property Name	Value	Explanation
fcat.jvm.id	A unique number.	This unique number identifies the server amongst all the cluster members.
fcat.basedir	<FCDB BASE DIR>	Path to the base folder containing FCDB application on file system. E.g.: D:\FCDB
fcat.propfile.isencr	true/false	This property indicates the properties file's encryption status. Set as 'true' if files are encrypted, else set as 'false'. The property files are not encrypted when installed but needs to be explicitly encrypted using FCDB tool provided. Please refer Encryption and Decryption of Property Files
ws.ext.dirs	E.g.: d:\fcdb\system\build\kernel;d:\fcdb\system\build\extclasses\jars\axis;d:\fcdb\system\build\extclasses\jars\cocoon;d:\fcdb\system\build\extclasses\jars\dom;d:\fcdb\system\build\extclasses\jars\fcr;d:\fcdb\system\build\extclasses\jars\flexcube;d:\fcdb\system\build\extclasses\jars\fop;d:\fcdb\system\build\extclasses\jars\ibm;d:\fcdb\system\build\extclasses\jars\oracle;d:\fcdb\system\build\extclasses\jars\others;d:\fcdb\system\build\extclasses\jars\poi;d:\fcdb\system\build\extclasses\jars\sunjai;d:\fcdb\system\build\extclasses\jars\sunjndi;d:\fcdb\system\build\extclasses\jars\toolkit;d:\fcdb\system\build\extclasses\jars\txt.txt;d:\fcdb\system\build\extclasses\jars\xalan;	Path to all folders holding application relevant jar files. Folders are ";" colon separated for Windows OS. Folders are ":" colon separated for Linux/Unix OS. Following folder should be added: <FCDB BASE DIR>/system/build/kernel And all subdirectories under folder <FCDB BASE DIR>/system/build/extclasses/jars (**Note: 1.**Remove commons-beanutils-1.7.jar from <FCDB BASE DIR>/ system\build\extclasses\jars\json **2**.In case of **Patch Installer** delete the *FCDB_12.0.0.jar from the* < FCDB BASE DIR >*/system/build/kernel)*

13) After above steps are completed, save the configuration changes by clicking on the "Save" link on the top of the screen.
14) Since all the configurations of IBM Websphere application server are XML files, the server uses `xalan` parser to parse the configuration files. However Oracle FLEXCUBE Direct Banking application uses oracle parser for reporting module and hence is added to the classpath. This results in the server failing to parse its configuration files and hence failing to start. Please update following property as remedy:

- Locate the jre folder present in the application server installation folder. Usually `<IBM Websphere Installation folder>/java/jre/lib`
- In the above folder locate a file `jaxp.properties` or `jaxp.properties.template`. If template file is present, rename it to `jaxp.properties`.
- Edit the file `jaxp.properties` and uncomment the `javax.xml.*` properties. Hence following properties should now be present in the properties file.

```
javax.xml.transform.TransformerFactory=org.apache.xalan.processor.TransformerFactoryImpl
javax.xml.xpath.XPathFactory=org.apache.xpath.jaxp.XPathFactoryImpl
```

```
javax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl
javax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBuilderFactory
Impl
```

- Restart the server.

5.2 Creating virtual host for Web Application

This section is not mandatory. In case FCDB is to be setup on the "`default_host`", this section can be skipped.

ⓘ Please refer to IBM Websphere documents to find details on virtual host and default host.

1) Click Administrative Console → Environment → Virtual Hosts

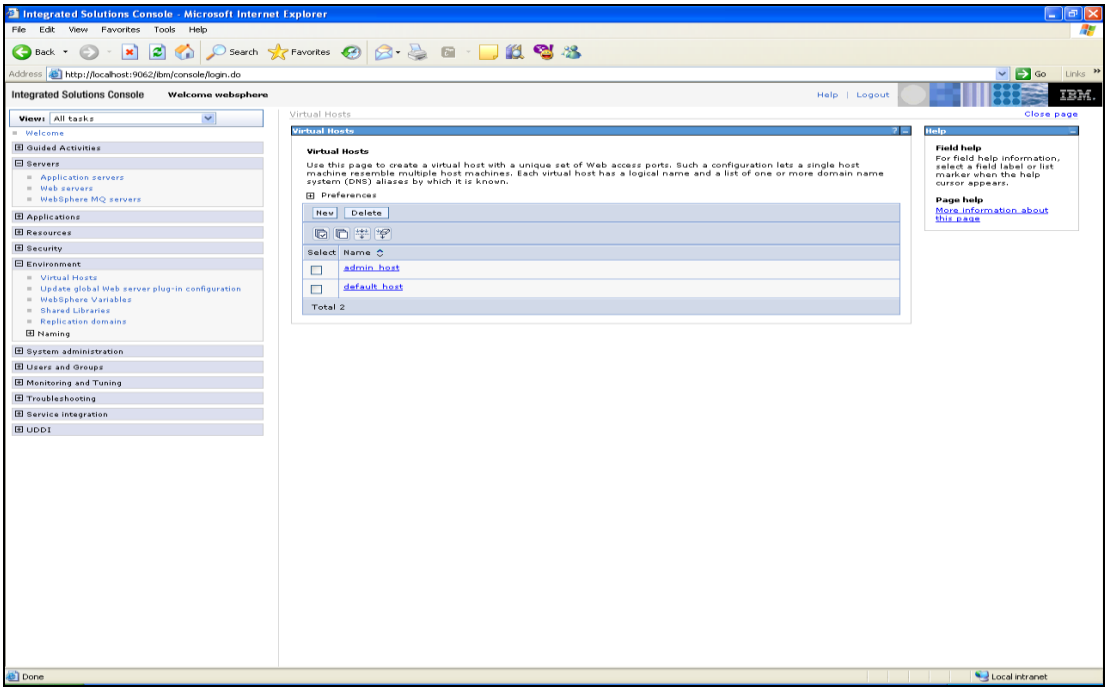


Figure 5.2.1

2) Click on New button. Enter the Virtual Host Name (e.g. FCDB)

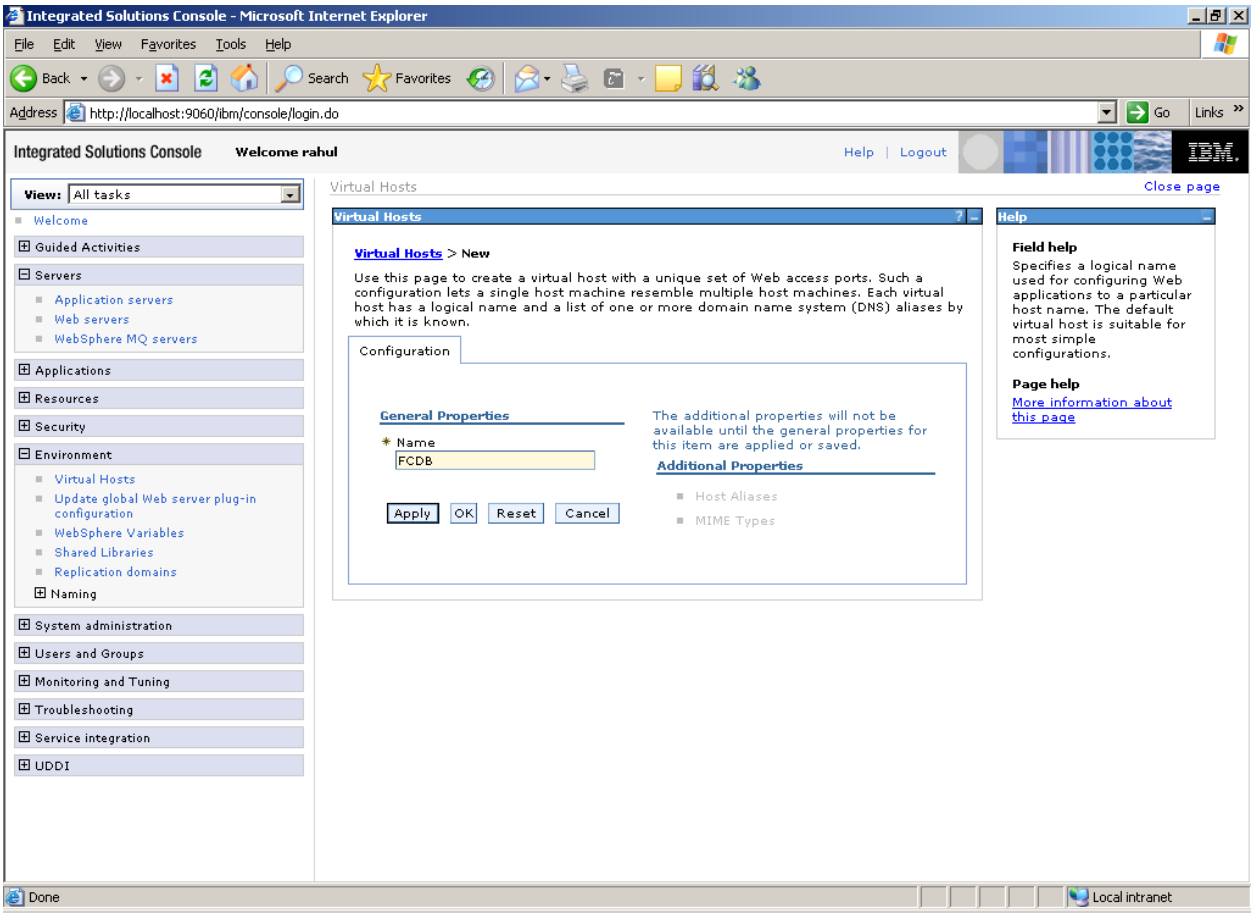


Figure 5.2.2

3) Click Apply

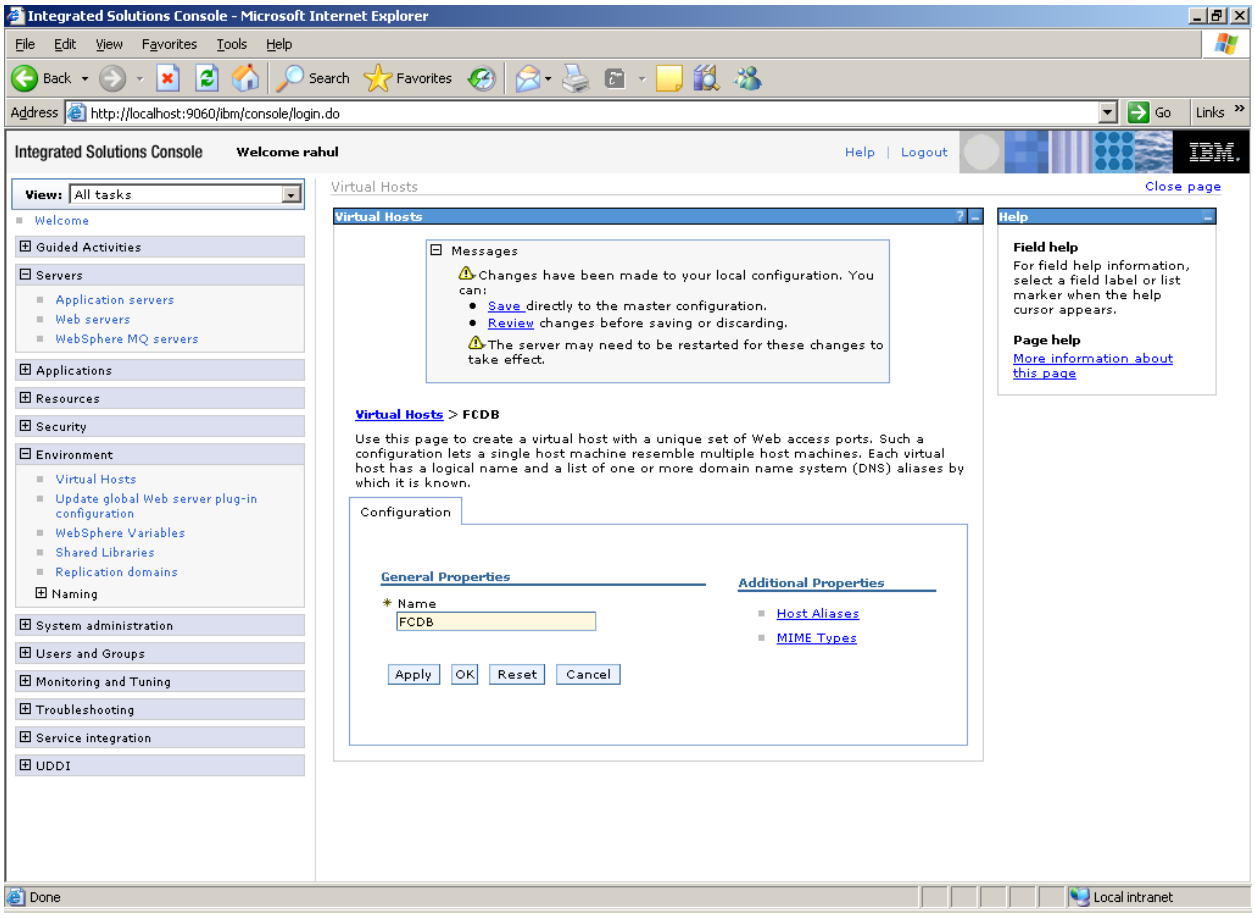


Figure 5.2.3

4) Click “Host Aliases”

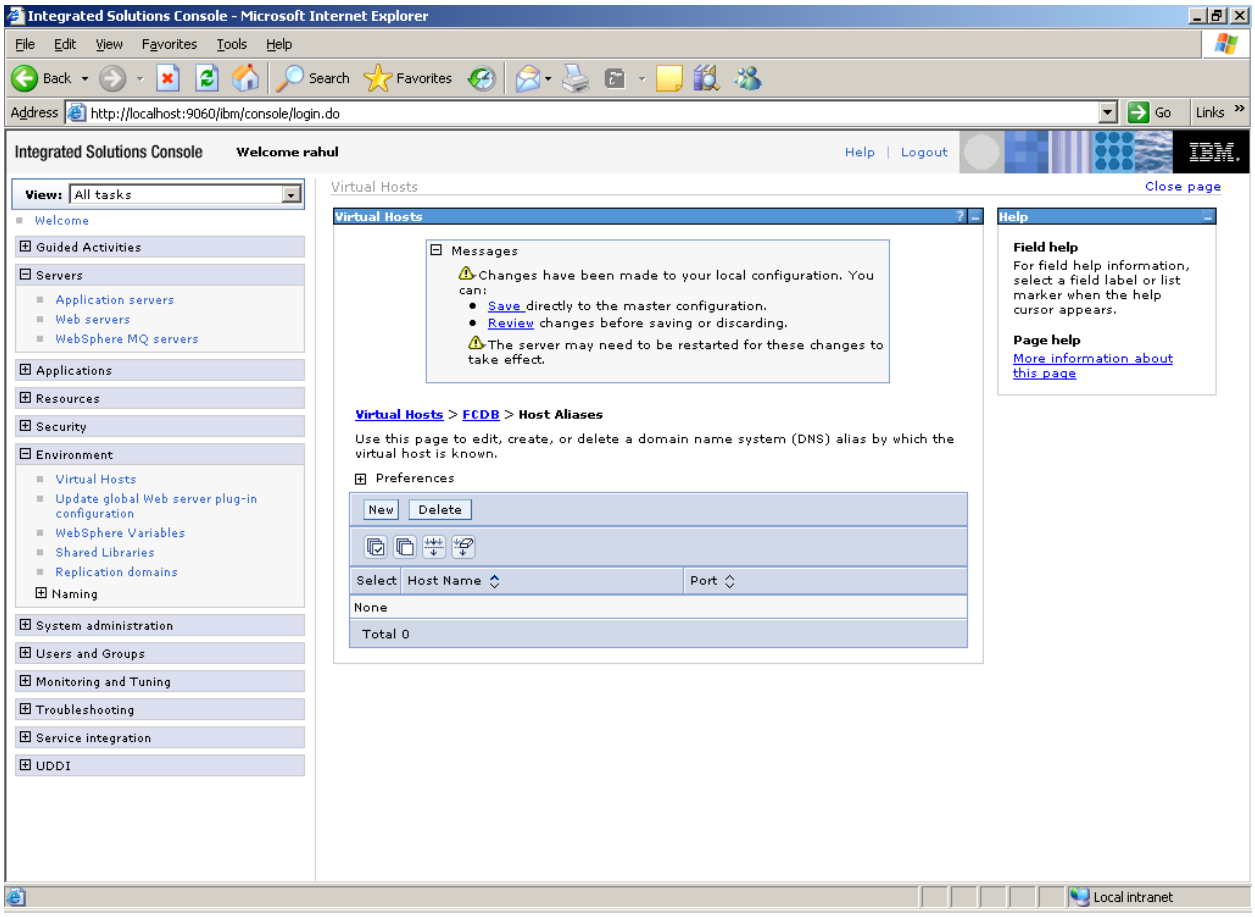


Figure 5.2.5

5) Click “New” button

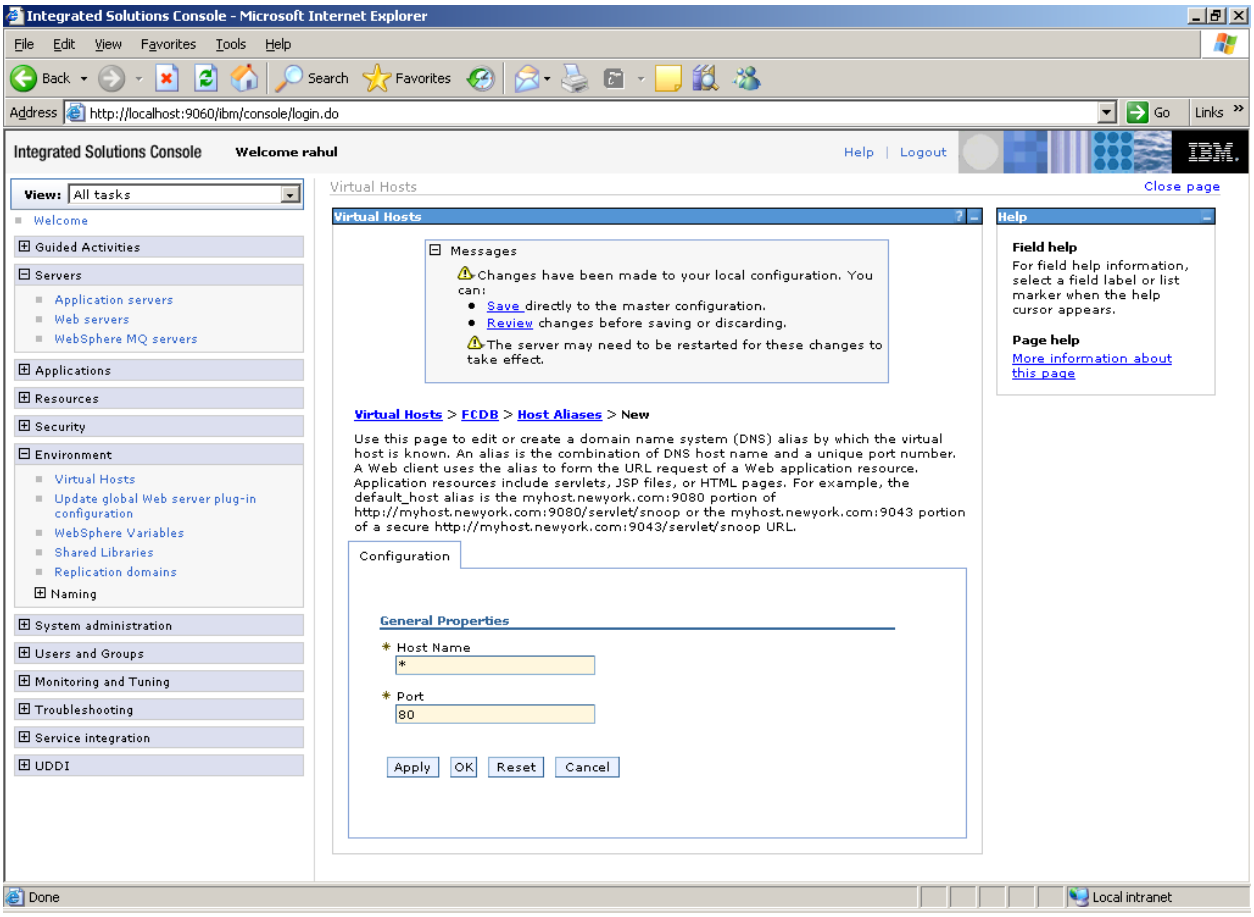


Figure 5.2.6

6) Add following host aliases (noted as default host port earlier)

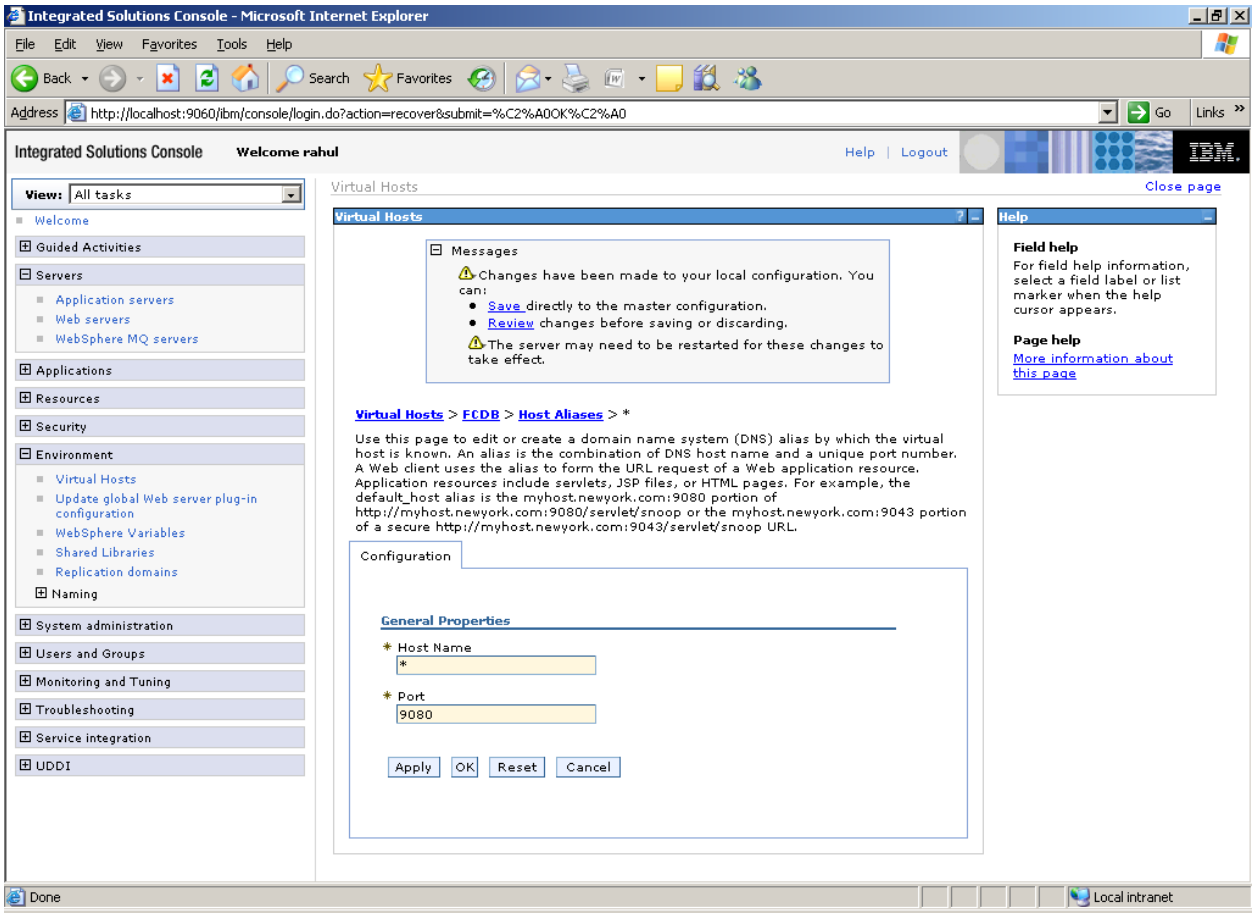


Figure 5.2.7

7) Click Apply

8) Click Save/OK

6. Deploying Applications

6.1 Web Application Deployment

Following section explains deploying Web application on IBM Websphere. For details and different options of deployment refer to documentations provided by IBM WebSphere.

These steps should be carried out for each FCDB web application mentioned below.

Deployable	Suggested Context	Path Location	Description
F001.war	F001	<FCDB BASE DIR>\deploy	Application for Super entity F001
B001.war	B001	< FCDB BASE DIR>\deploy	Application for Bank entity B001. This is available only for Base release.
T001.war	T001	< FCDB BASE DIR>\deploy	Application for Model Third Party Bank entity T001. This is applicable only for Third party Accelerator Pack release.
SMS.war	SMS	< FCDB BASE DIR>\deploy	Required for SMS Banking.

Update following property for websphere in MSTPROPERTIES table of FCDB:

ORACLE.UTILS.IMPL.CLASS=com.iflex.fcat.xml_sql.JFWSOracleUtils.

1) Go to Admin Console. On the left Menu expand option “Applications” and click on the link “New Applications”

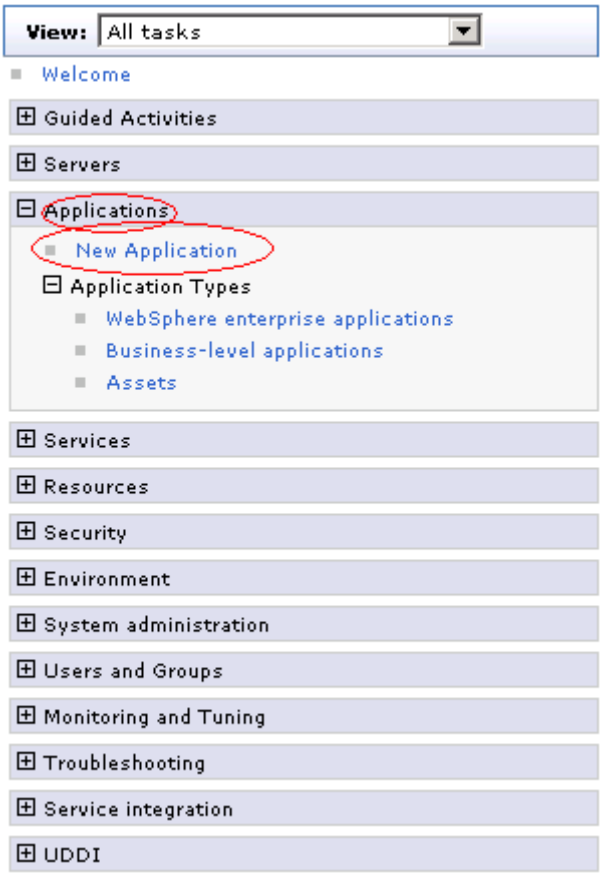
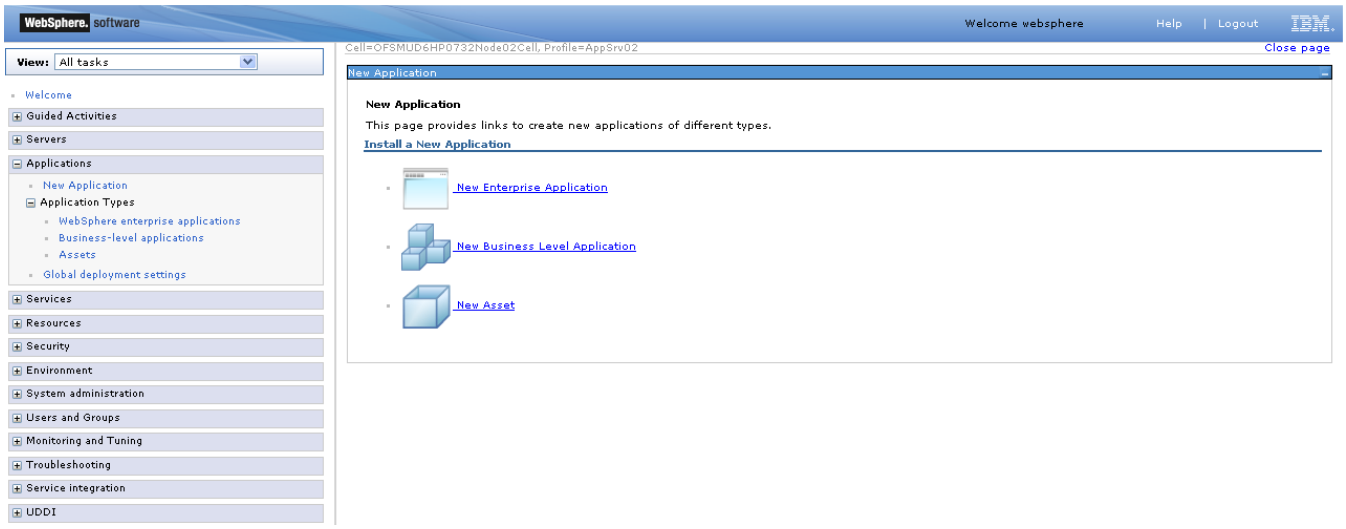


Figure 6.1

2) Screen as shown in below figure will appear. On the screen, click “New Enterprise Application”.



3) Screen as shown in below figure will appear. On the screen, click “Browse” button and browse to the web application deployable.. Click “Next” button.

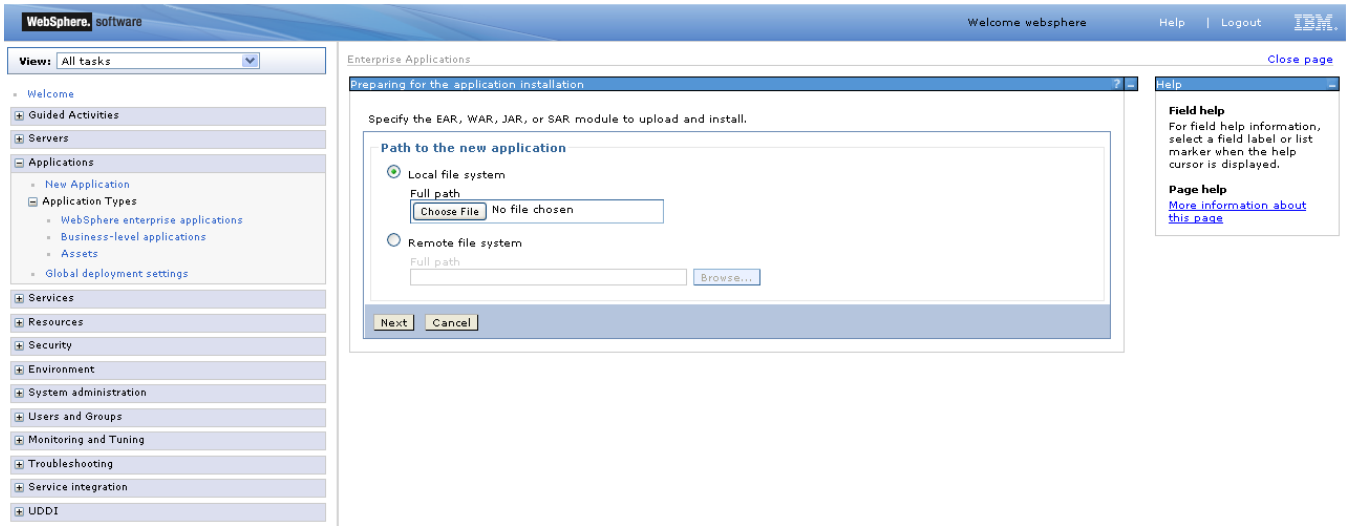
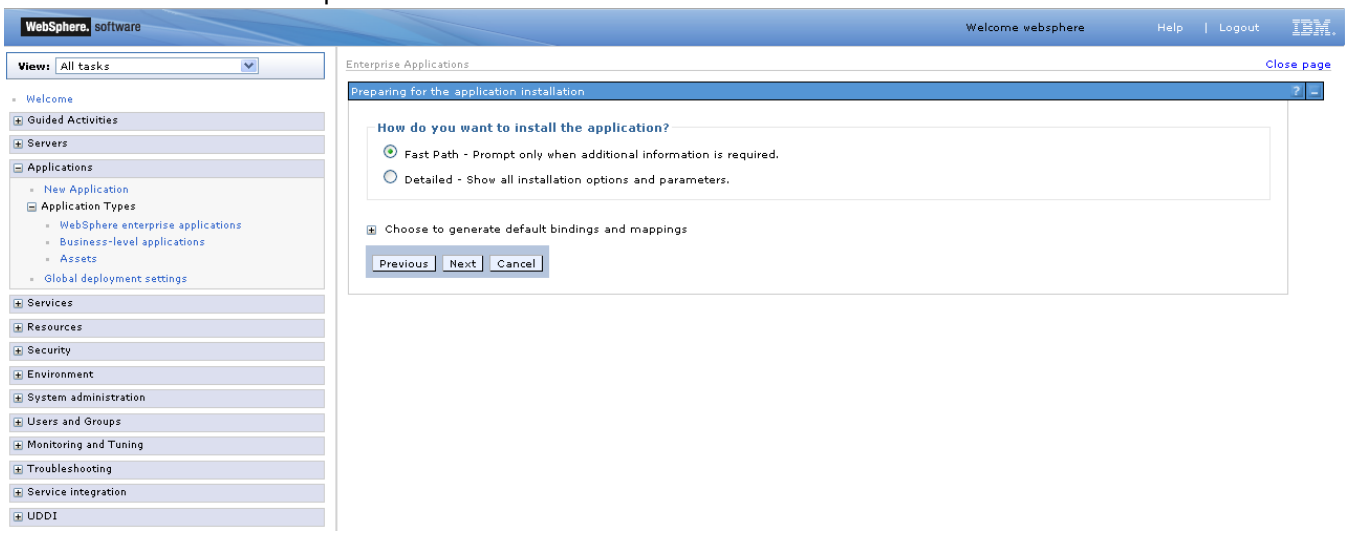


Figure 6.2

4) Screen as shown in below figure will appear. On the screen, click on “Fast Path - Prompt only when additional information is required” radio button and click “Next”.



5) On the Screen as shown in figure below, change the application name if required and click “Next”.

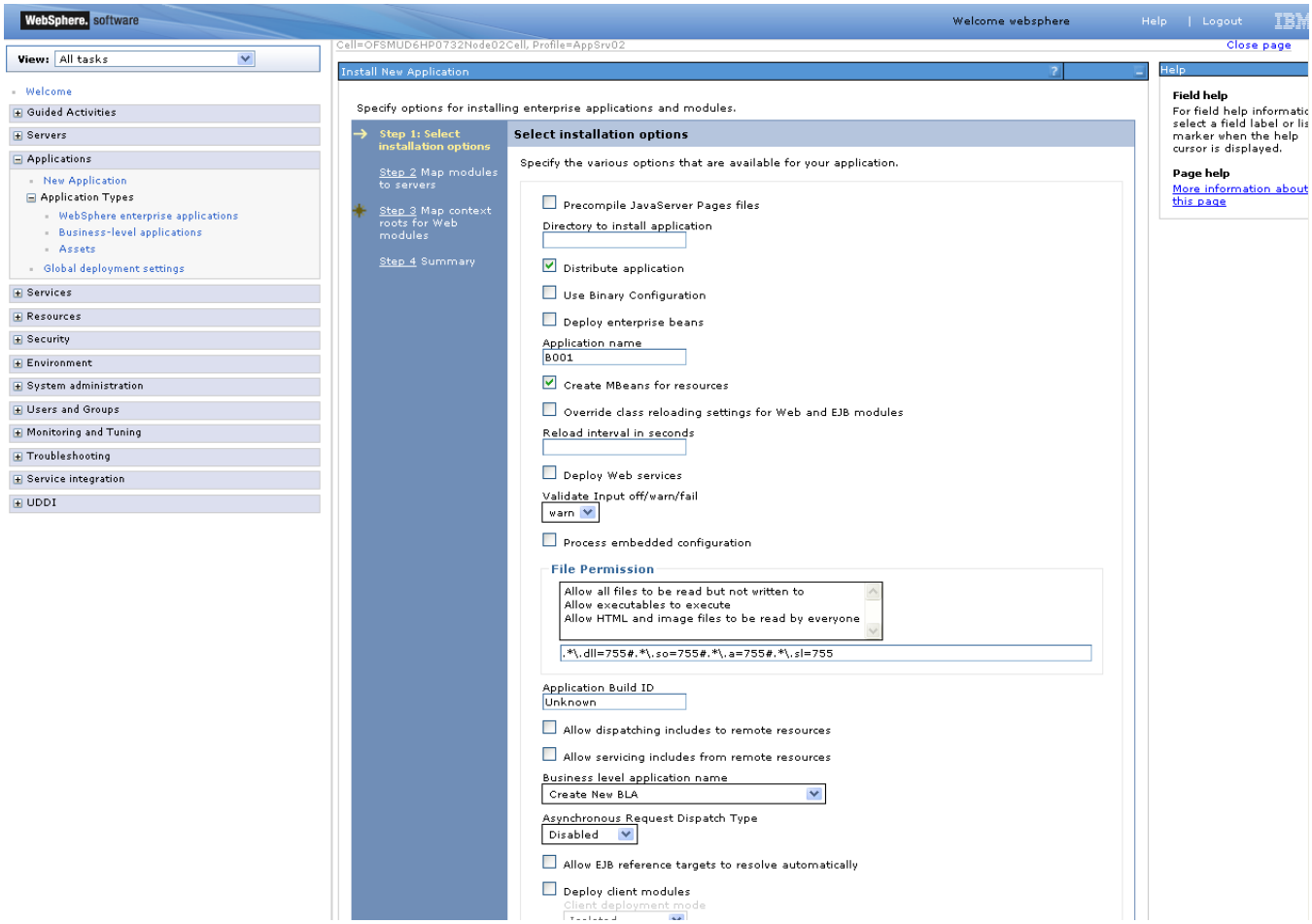
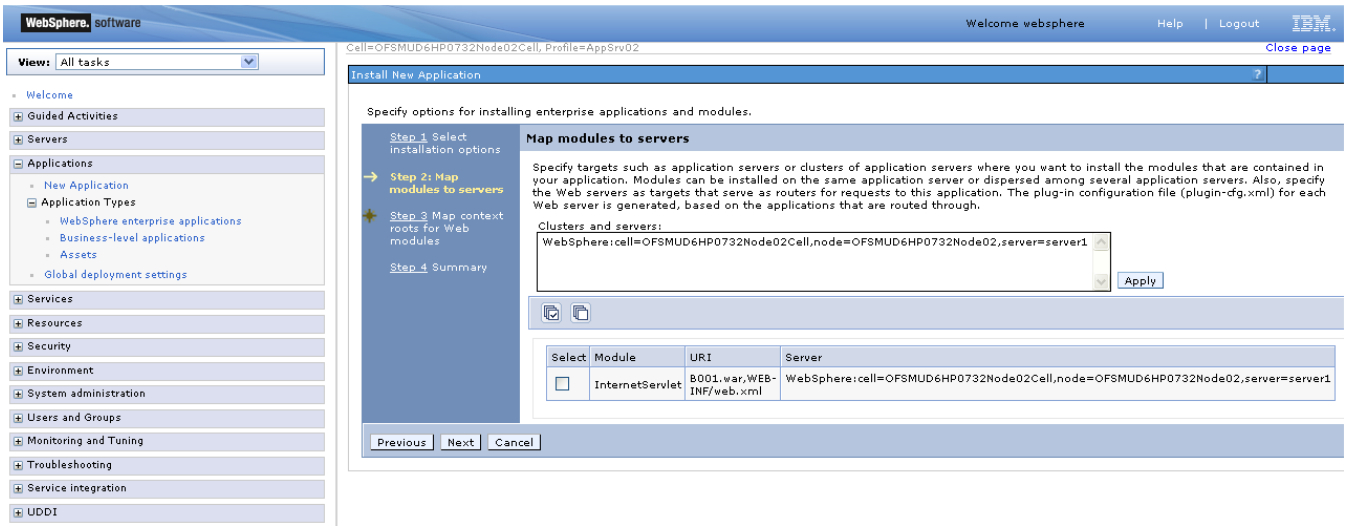
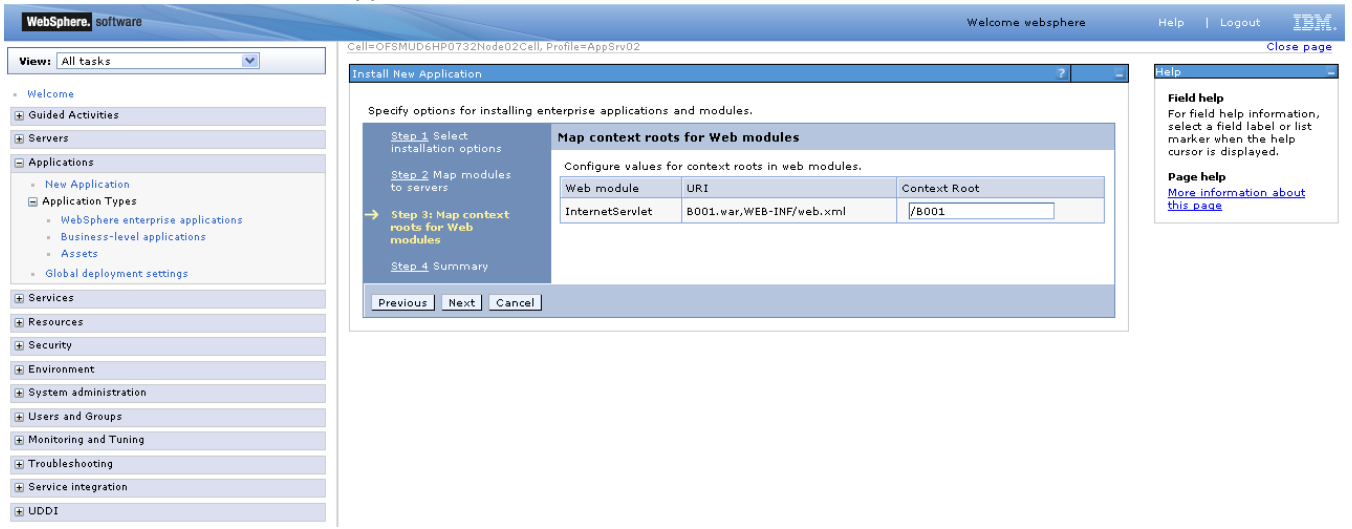


Figure 6.4

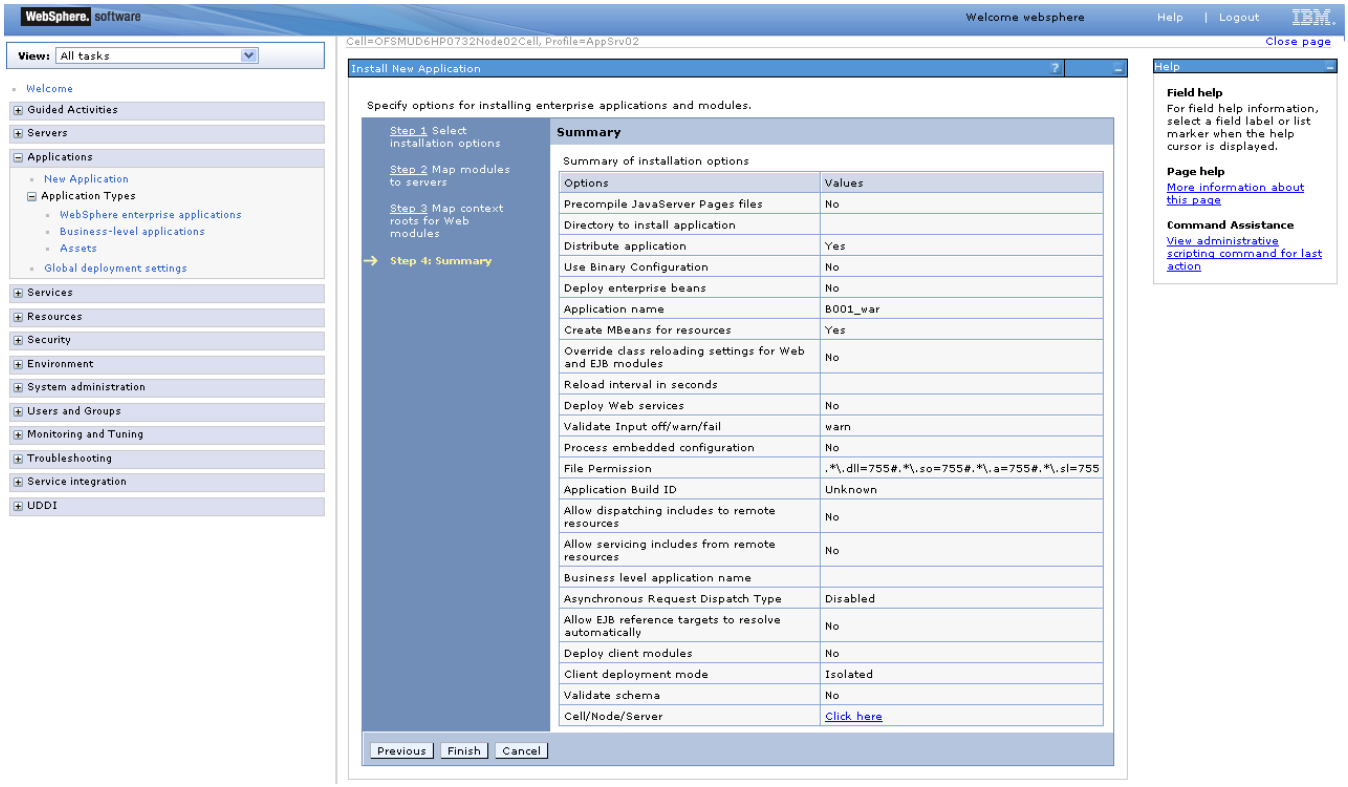
6) On the Screen as shown in figure below, Map the web module to expected server. Click “Next”.



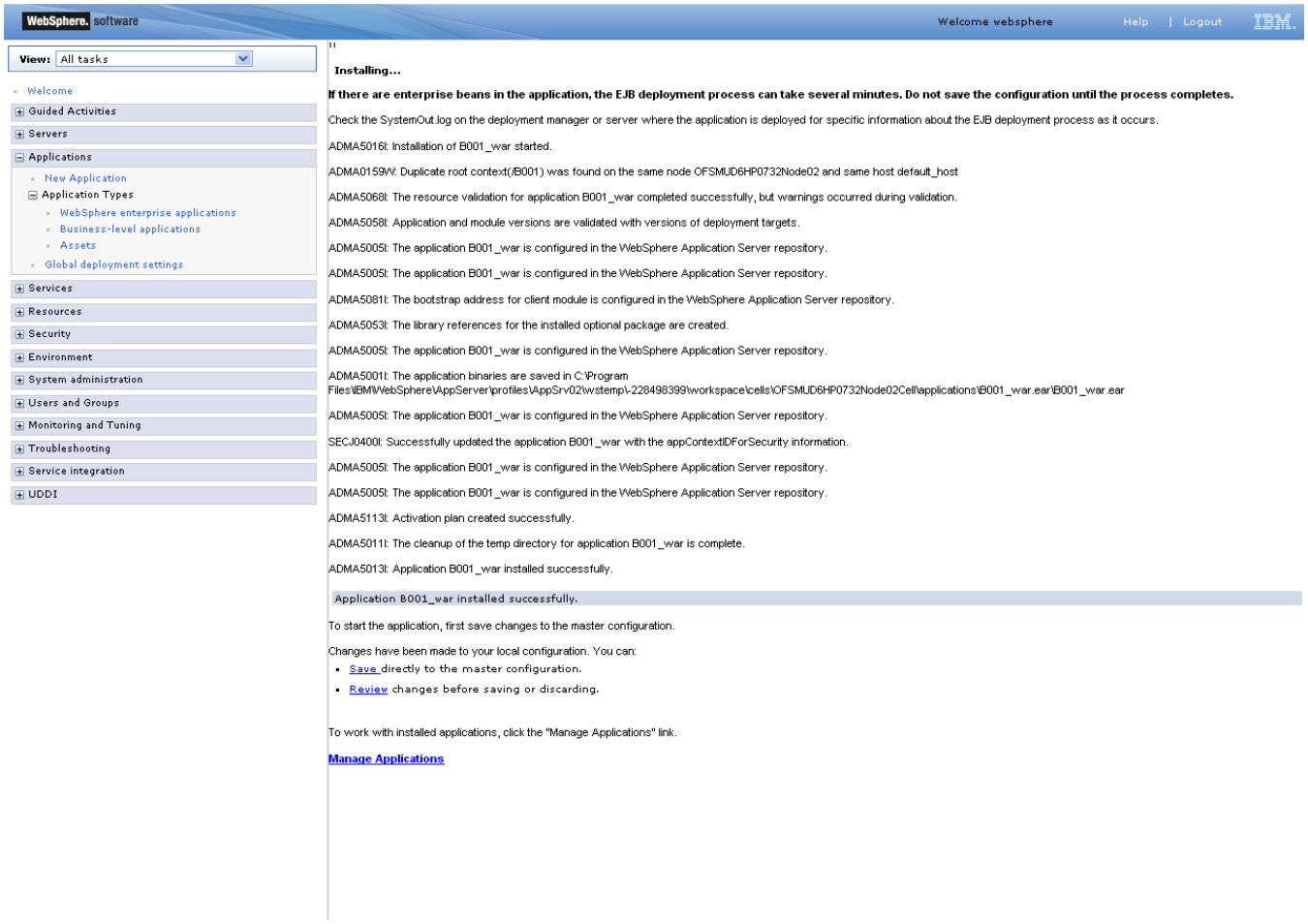
7) Enter the context root of the application and click next.



8) Summary page will appear with selected installation options, review the same and click “Finish” to begin with installation of the web application.


WebSphere software
Welcome websphere
Help | Logout
IBM
View: All tasks
Welcome
Guided Activities
Servers
Applications
New Application
Application Types
WebSphere enterprise applications
Business-level applications
Assets
Global deployment settings
Services
Resources
Security
Environment
System administration
Users and Groups
Monitoring and Tuning
Troubleshooting
Service integration
UDDI
Cell=OFSMUD6HP0732Node02Cell, Profile=AppSrv02
Close page
Install New Application
Specify options for installing enterprise applications and modules.
Step 1 Select installation options
Step 2 Map modules to servers
Step 3 Map context roots for Web modules
Step 4: Summary
Summary
Summary of installation options
Options
Values
Precompile JavaServer Pages files
No
Directory to install application
Distribute application
Yes
Use Binary Configuration
No
Deploy enterprise beans
No
Application name
B001_war
Create MBeans for resources
Yes
Override class reloading settings for Web and EJB modules
No
Reload interval in seconds
Deploy Web services
No
Validate Input off/warn/fail
warn
Process embedded configuration
No
File Permission
.*\.dll=755#.*\.so=755#.*\.a=755#.*\.sl=755
Application Build ID
Unknown
Allow dispatching includes to remote resources
No
Allow servicing includes from remote resources
No
Business level application name
Asynchronous Request Dispatch Type
Disabled
Allow EJB reference targets to resolve automatically
No
Deploy client modules
No
Client deployment mode
Isolated
Validate schema
No
Cell/Node/Server
Click here
Previous
Finish
Cancel
Help
Field help
For field help information, select a field label or list marker when the help cursor is displayed.
Page help
More information about this page
Command Assistance
View administrative scripting command for last action

9) Screen as shown in below figure will appear, detailing the status of the installation. On successful installation, click “Save” to complete the deploy application process.



6.2 JAR Application Deployment

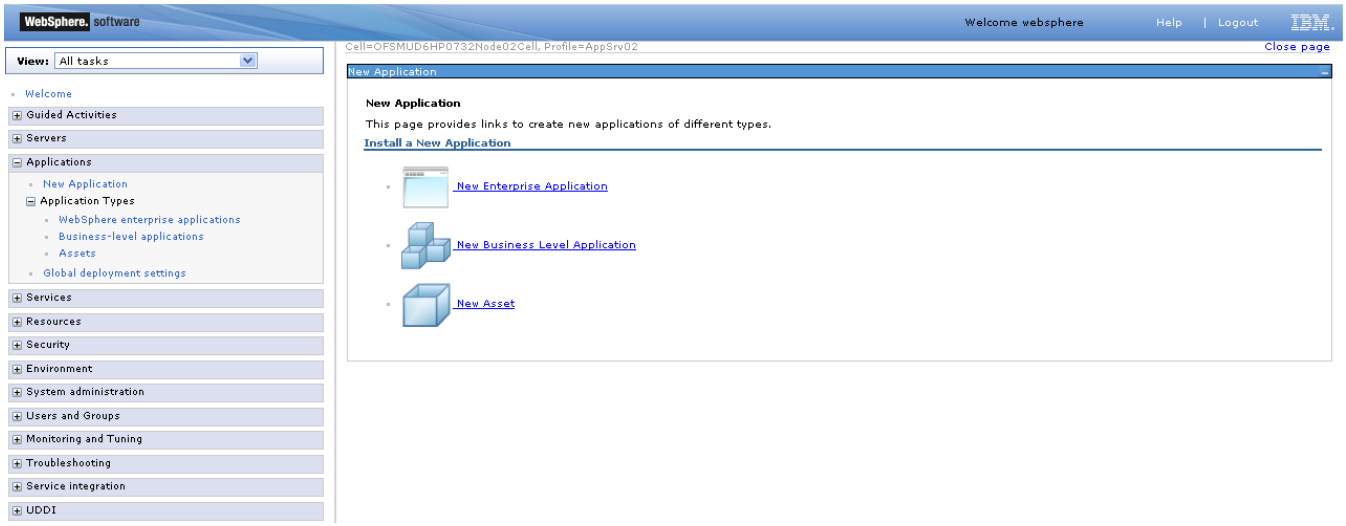
Following section explains deploying JAR application on IBM Websphere. For details and different options of deployment refer to documentations provided by IBM Websphere.

These steps should be carried out for each FCDB jar application mentioned below.

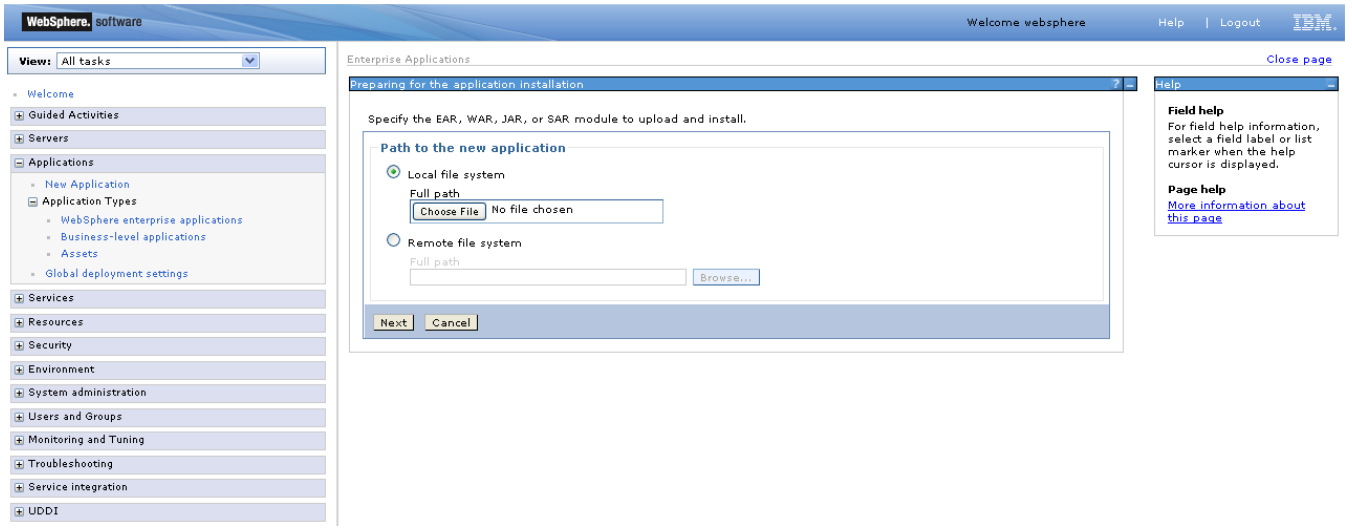
Deployable	JNDI	Path Location	Description
EJBChannleController.jar	EJBChannelController	<FCDB BASE DIR>\deploy	Deployment required. Used as interface to Channel tier.
ServiceEndPointRI.jar	ServiceEndPointEJB	<FCDB BASE DIR>\deploy	This application should be deployed if the mode of service tier invocation is EJB. If webservice mode is required, service.ear needs to be deployed. Please check "Enterprise Application Deployment" section
Service.ear	JNDI: service Context Root: service	<FCDB BASE DIR>\deploy	This application should be deployed if the mode of service tier invocation is webservice.
TransactionBean.jar	TransactionBean	<FCDB BASE DIR>\deploy	Deployment required.
TimerBean.jar	TimerBean	<FCDB BASE DIR>\deploy	Application should be deployed for timer services.

1) On Admin console, in the left menu, click on “New Application”.

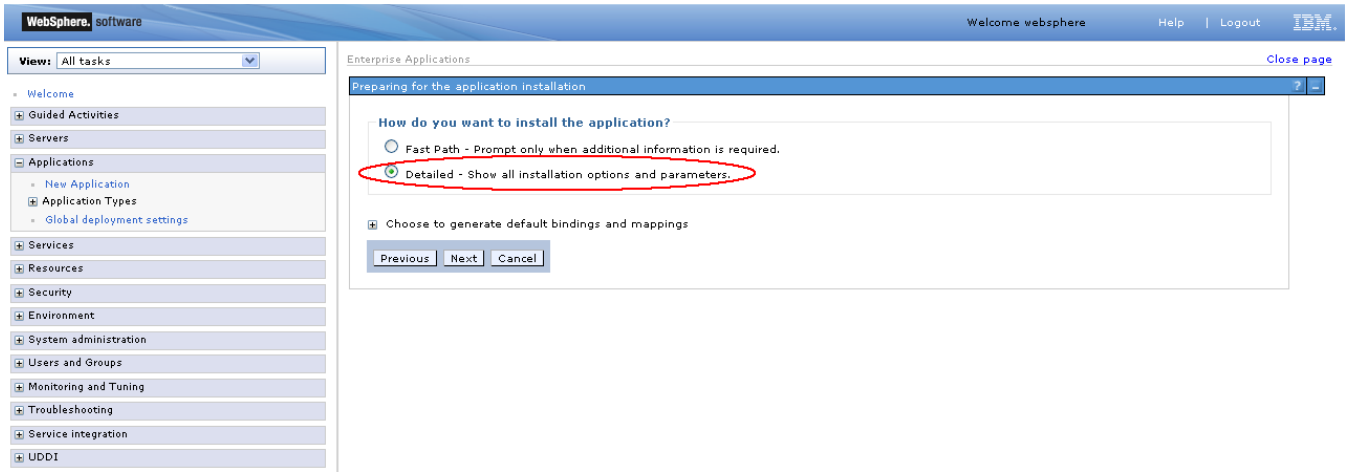
2) Screen as shown in below figure will appear. On the screen, click “New Enterprise Application”.



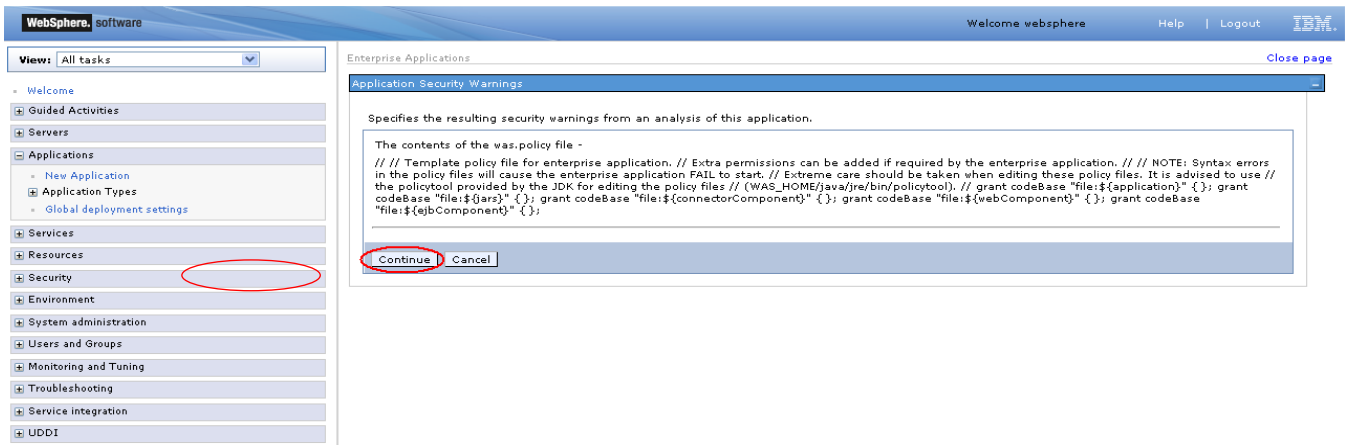
3) On the screen as show in figure below, use browse button to select the JAR file. Click on “Next” button.



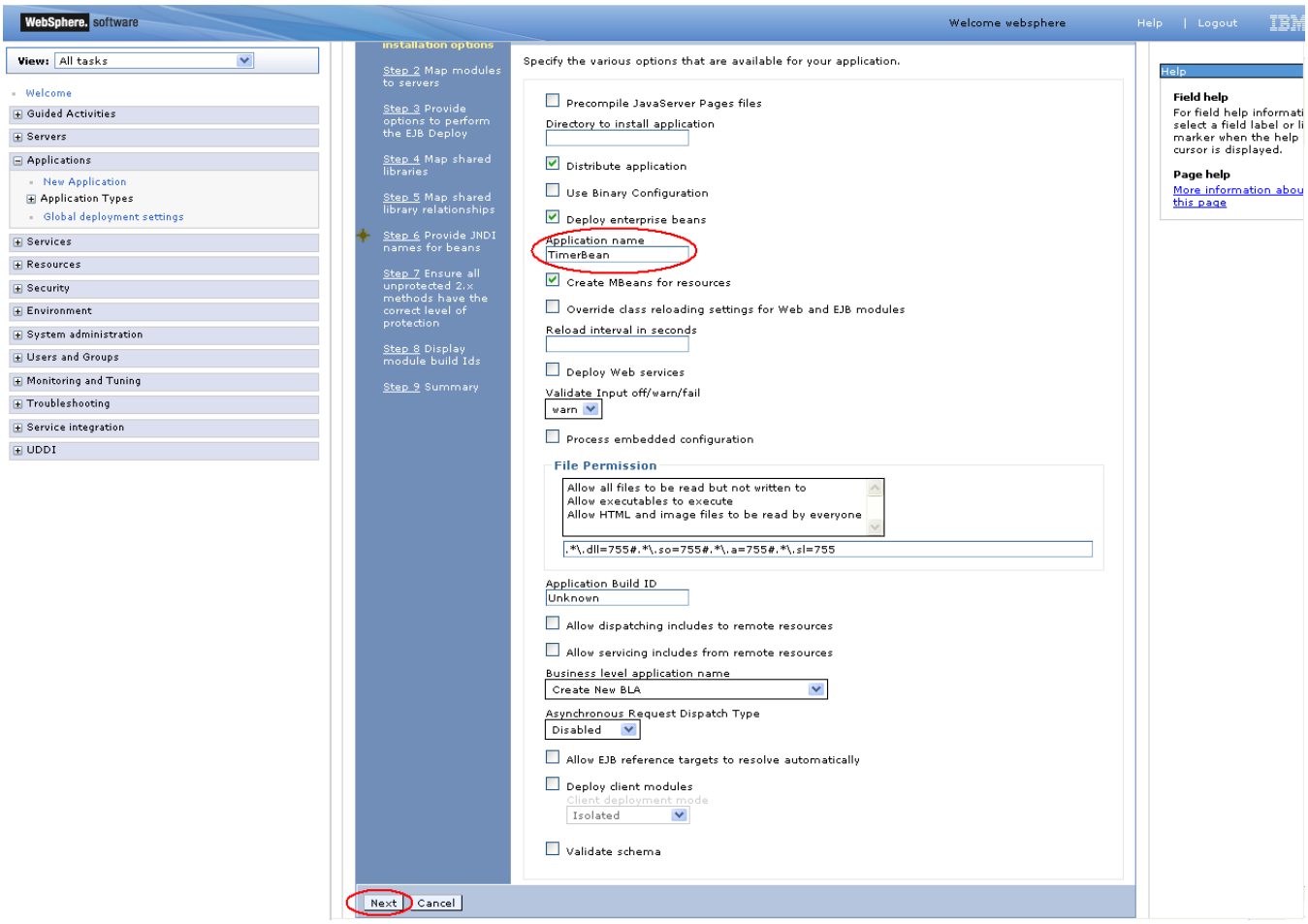
4) On the screen as show in figure below,Select the radio option “Show all installation options and parameters”. Click on “Next” button.



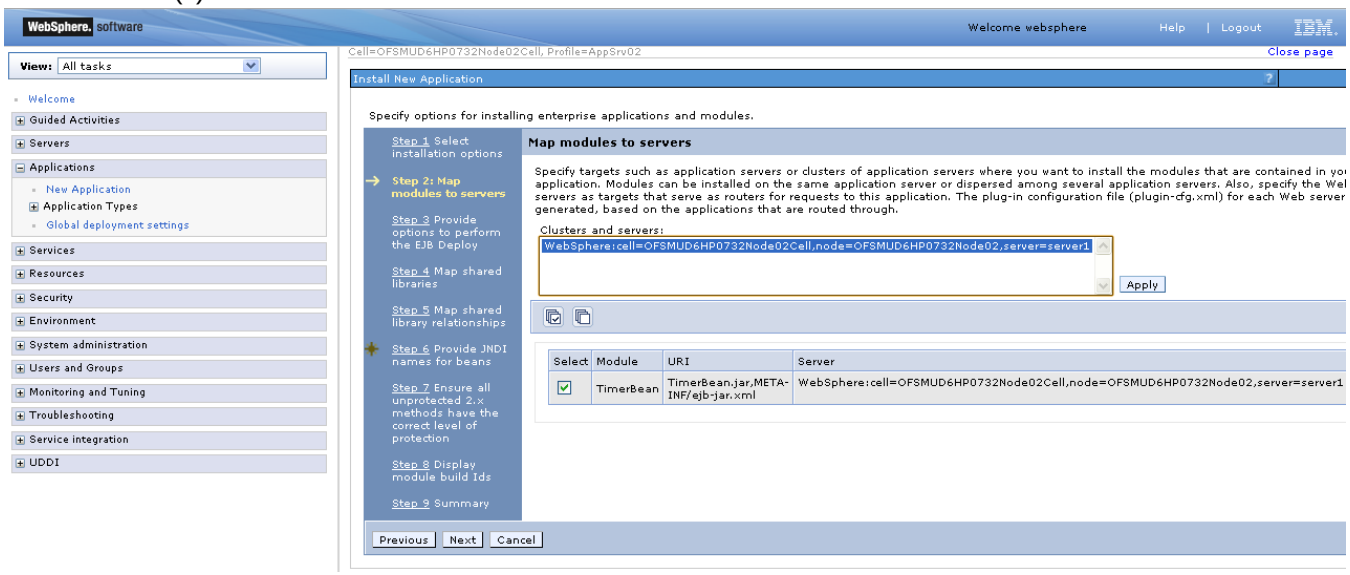
5) On the screen as shown below, click “Continue” button.



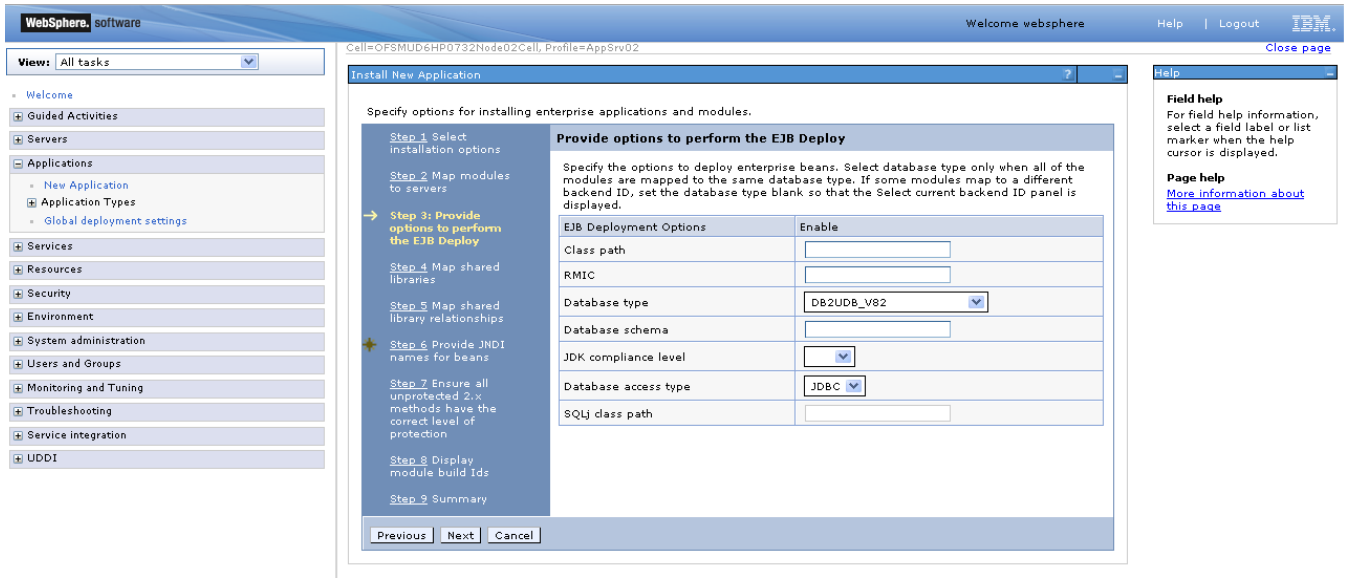
6) On the Screen as shown in below figure, Update the application name if required and click “Next”



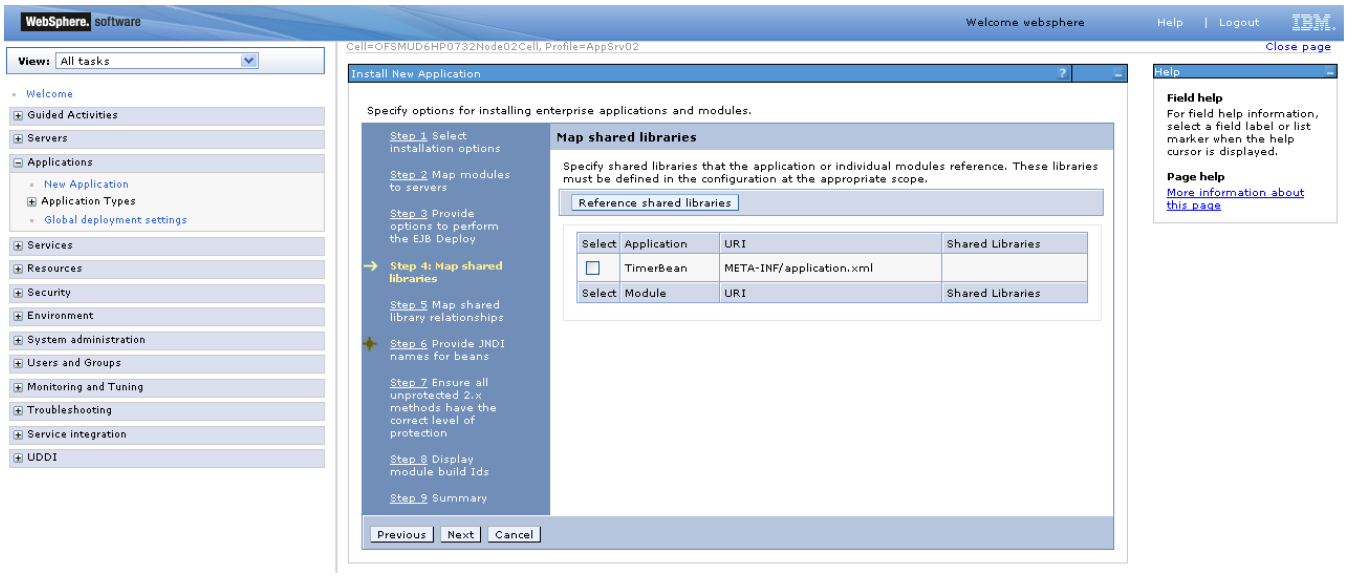
7) In screen “Map modules to server” as shown in figure below, map the JAR module to the applicable server(s) and click “Next”.



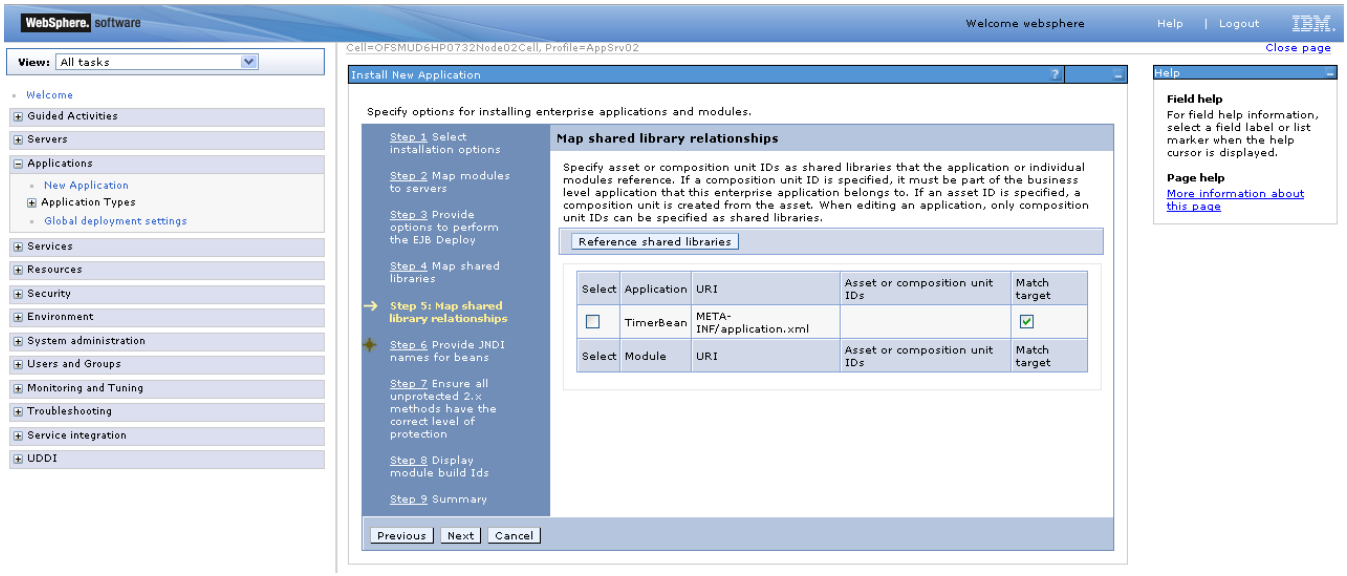
8) On Screen “Provide options to perform the EJB Deploy” Click next.



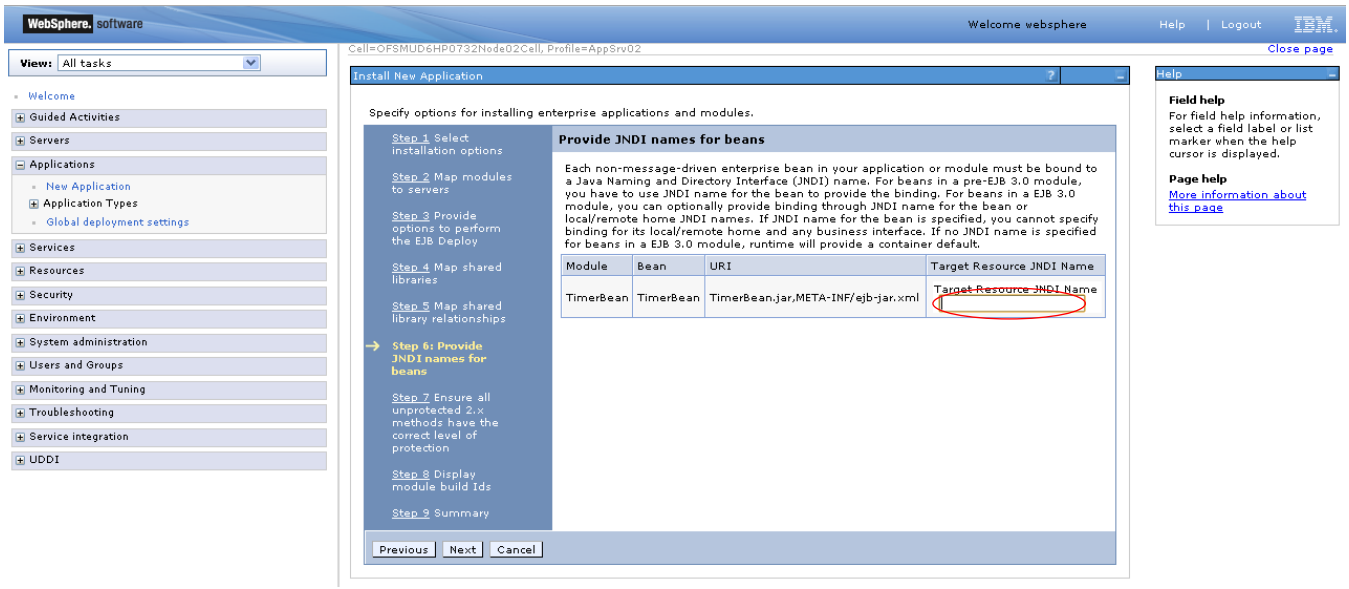
9) On the screen “Map Shared libraries” as shown below, click “Next”



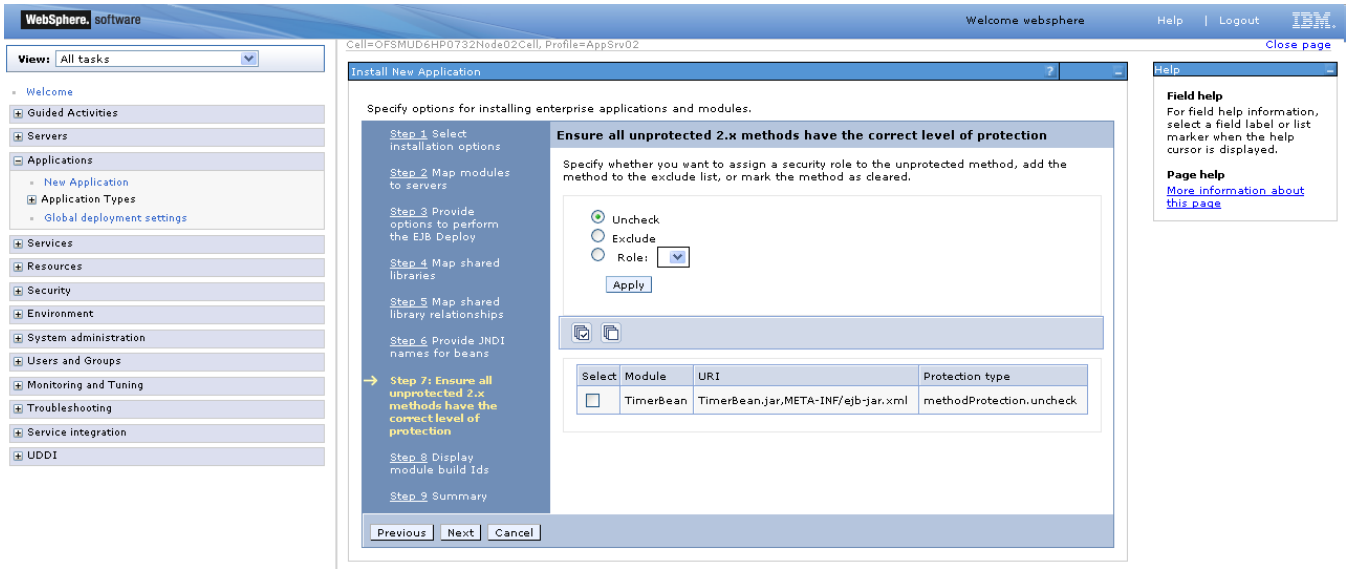
10) On the Screen “Map shared libraries relationships” as shown below, click “Next”.



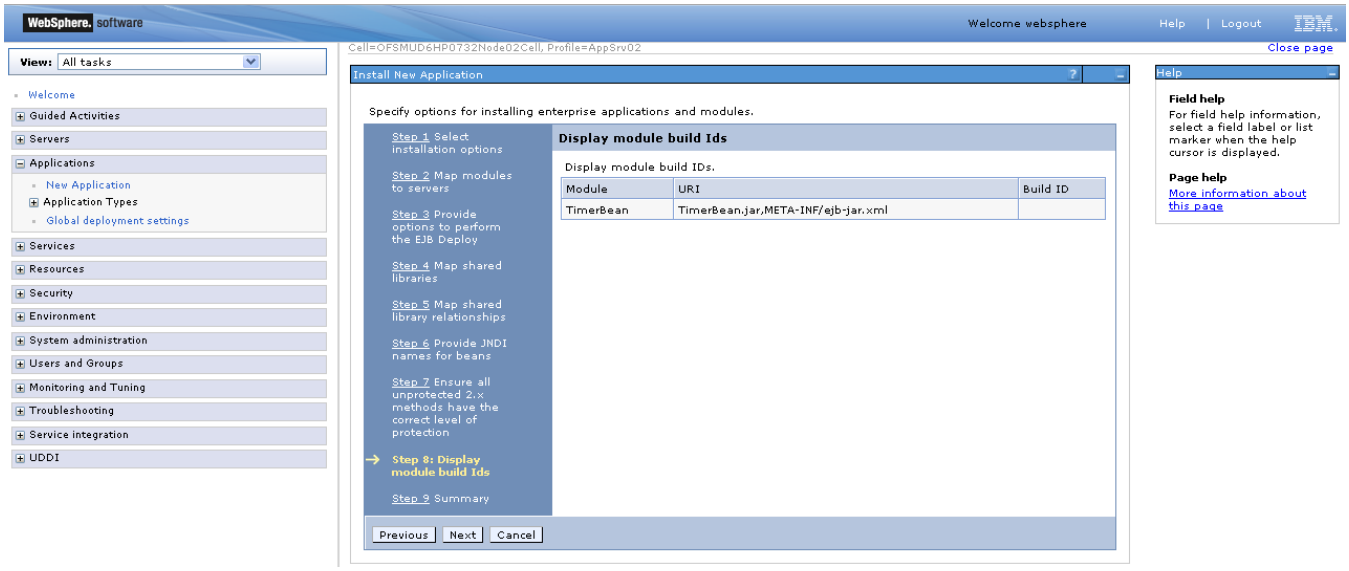
11) On the Screen “Provide JNDI names” as show below, provide the JNDI name for the application in field “Target Resource JNDI Name”. The JNDI name for each FCDB application is provided.


WebSphere. software
Welcome websphere
Help | Logout
IBM.
View: All tasks
Welcome
Guided Activities
Servers
Applications
New Application
Application Types
Global deployment settings
Services
Resources
Security
Environment
System administration
Users and Groups
Monitoring and Tuning
Troubleshooting
Service integration
UDDI
Cell=OFSMUD6HP0732Node02Cell, Profile=AppSrv02
Close page
Install New Application
Specify options for installing enterprise applications and modules.
Step 1 Select installation options
Step 2 Map modules to servers
Step 3 Provide options to perform the EJB Deploy
Step 4 Map shared libraries
Step 5 Map shared library relationships
Step 6: Provide JNDI names for beans
Step 7 Ensure all unprotected 2.x methods have the correct level of protection
Step 8 Display module build Ids
Step 9 Summary
Provide JNDI names for beans
Each non-message-driven enterprise bean in your application or module must be bound to a Java Naming and Directory Interface (JNDI) name. For beans in a pre-EJB 3.0 module, you have to use JNDI name for the bean to provide the binding. For beans in a EJB 3.0 module, you can optionally provide binding through JNDI name for the bean or local/remote home JNDI names. If JNDI name for the bean is specified, you cannot specify binding for its local/remote home and any business interface. If no JNDI name is specified for beans in a EJB 3.0 module, runtime will provide a container default.
Module
Bean
URI
Target Resource JNDI Name
TimerBean
TimerBean
TimerBean.jar,META-INF/ejb-jar.xml
Target Resource JNDI Name
Previous
Next
Cancel
Help
Field help
For field help information, select a field label or list marker when the help cursor is displayed.
Page help
More information about this page

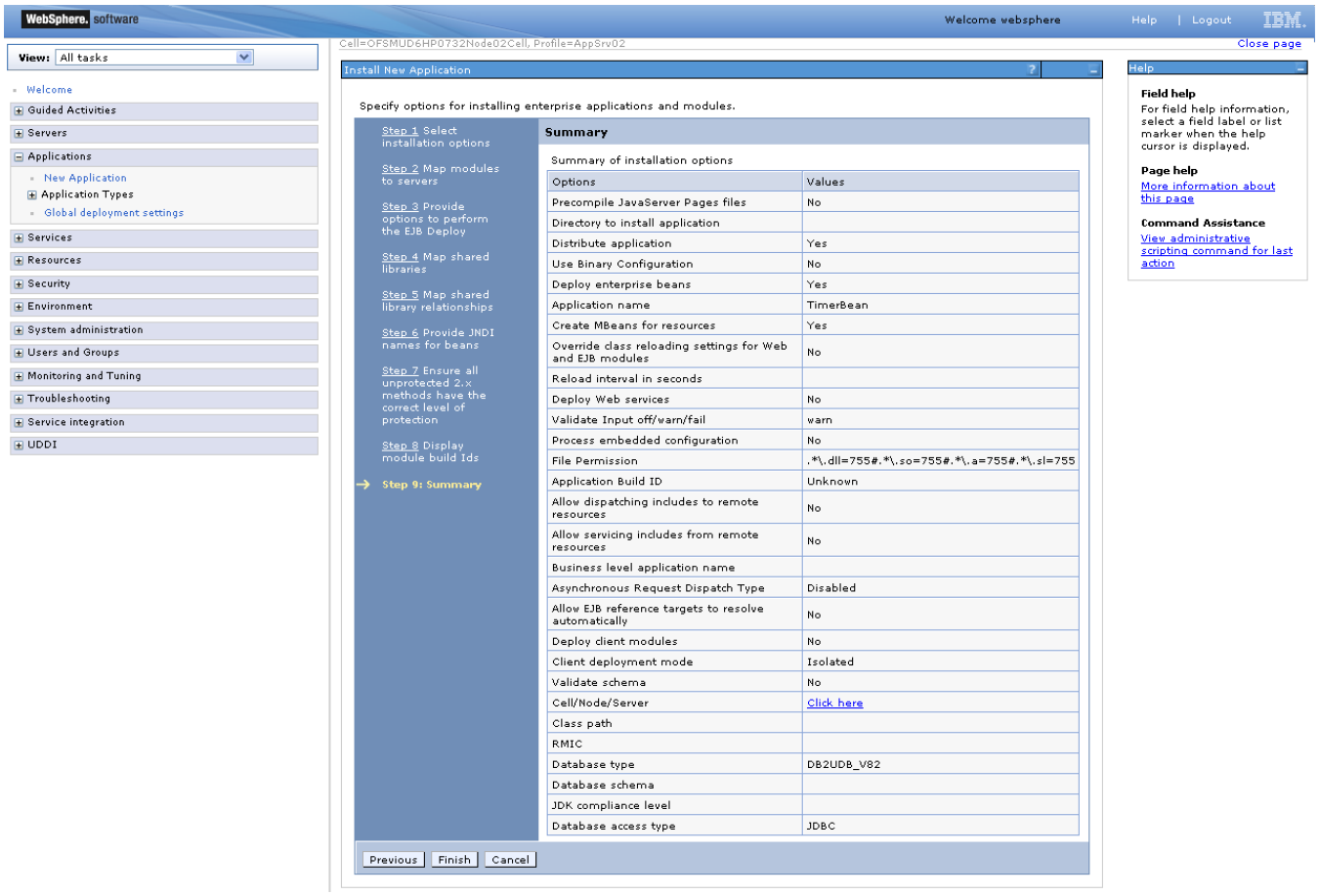
12) On the screen as show below, click “Next”.



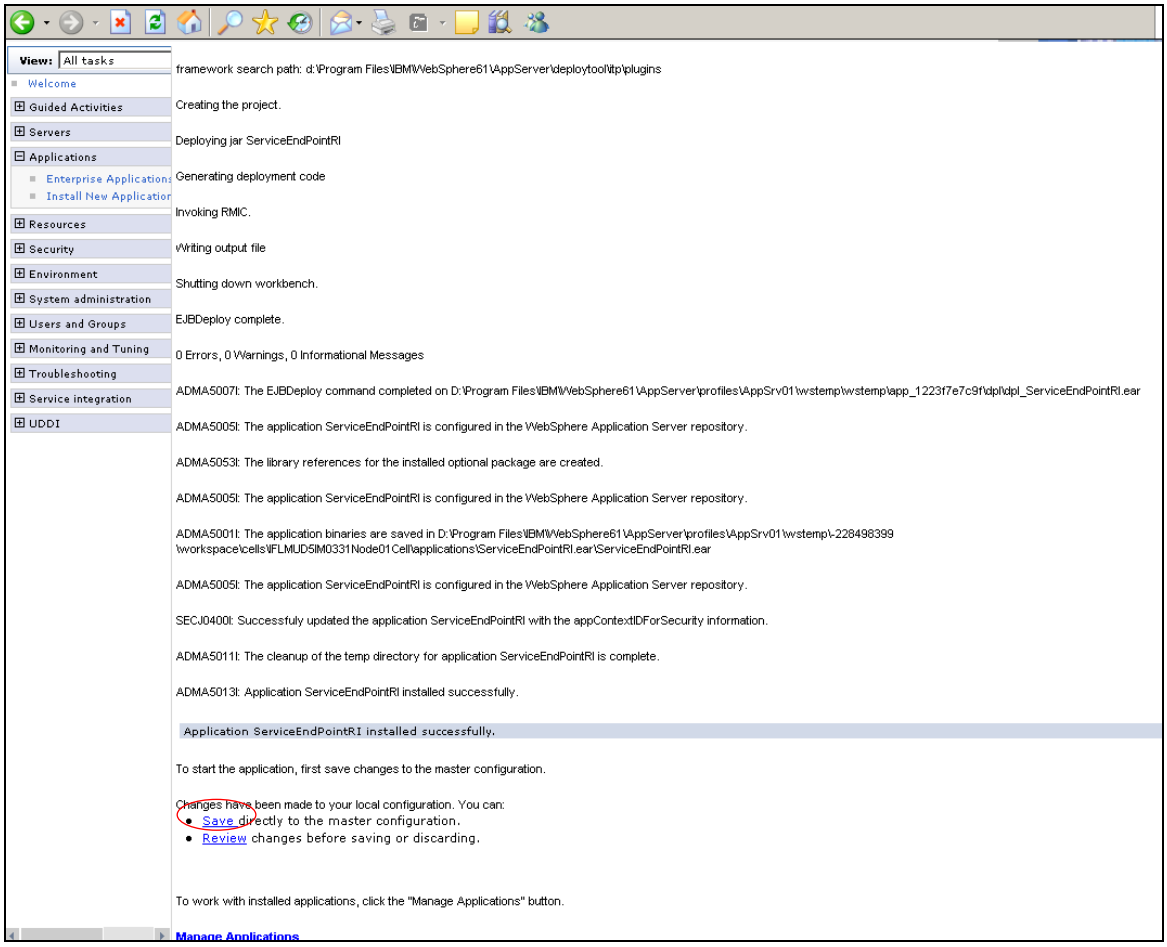
13) On the screen as show below, click “Next”.



14) On the “Summary” page, summary of selected installation options will be displayed. Verify the same and click “Next” to begin with application installation.



15) The next screen displays application installation status. On successful installation, click the link “Save” to save the installation.



6.3 Enterprise Application Deployment

Following section explains deploying EAR application on IBM Websphere. For details and different options of deployment refer to documentations provided by IBM Websphere.

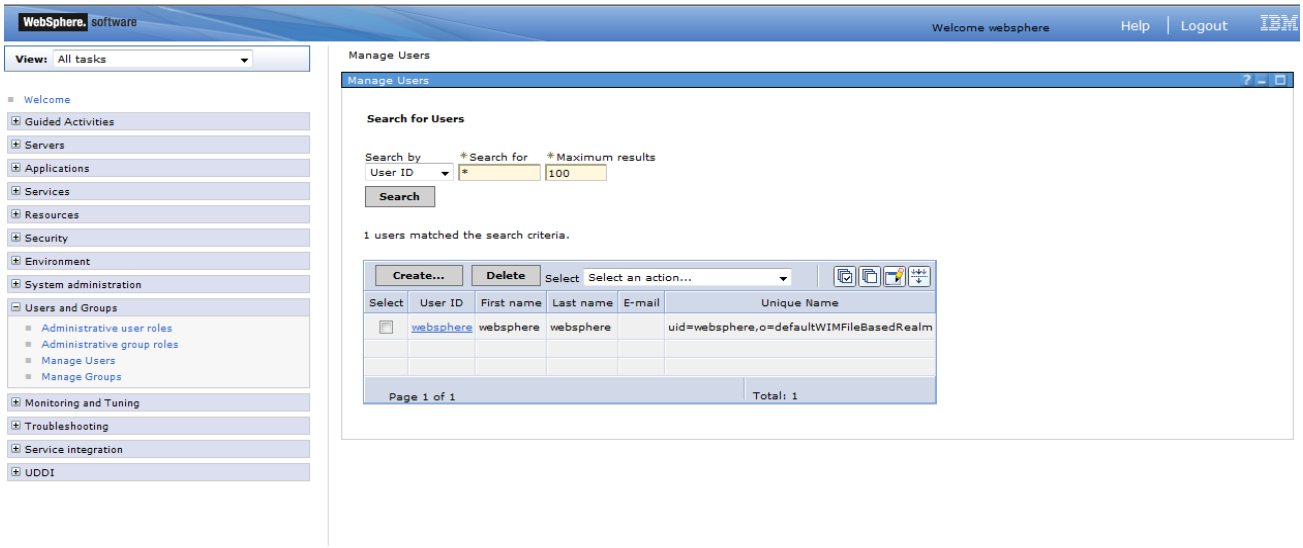
These steps should be carried out for each FCDB jar application mentioned below.

Deployable	JNDI Name/Root	Path Location	Description
Service.ear	JNDI: service Context Root: service	<FCDB BASE DIR>\deploy\ websphere	This application should be deployed if the mode of service tier invocation is webservice. If EJB mode is required, ServiceEndPointRI.jar needs to be deployed.

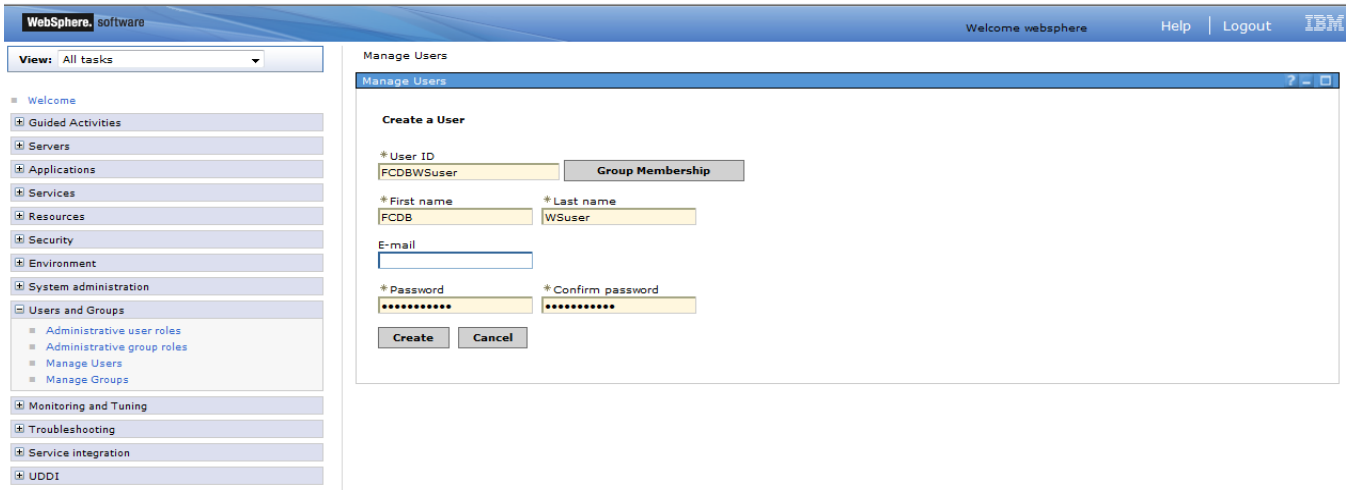
Before deploying services.ear, the below steps needs to be followed to create an user for enabling webservice authentication –

1. On Admin console, in the left menu, click on “Manage Users”.

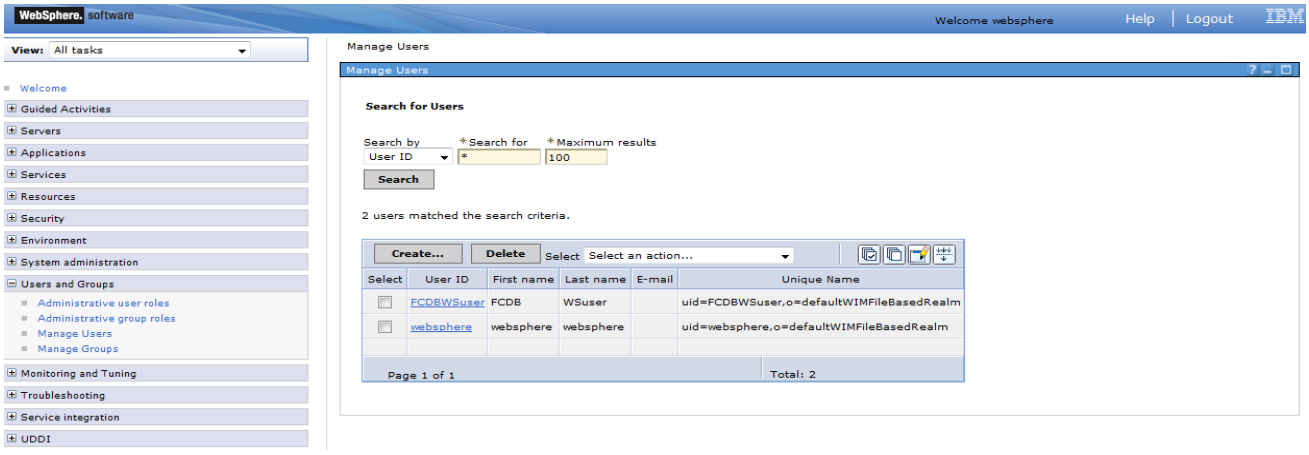
2. Screen as shown in below figure will appear. On the screen, click “Create”.



3. On the screen as show in figure below, enter the user details (e.g. FCDBWSuser). Click on “Create” button

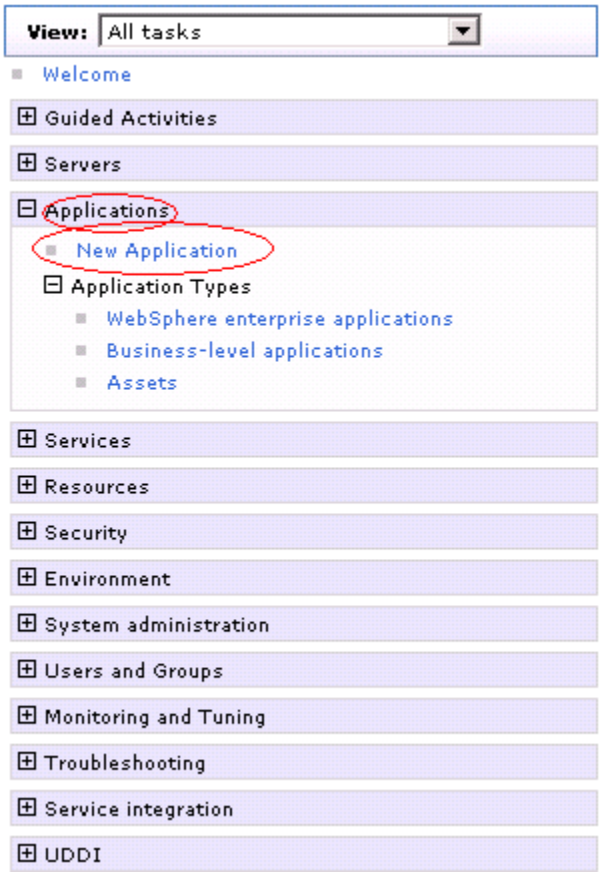


4. User is created successfully in the current security realm.

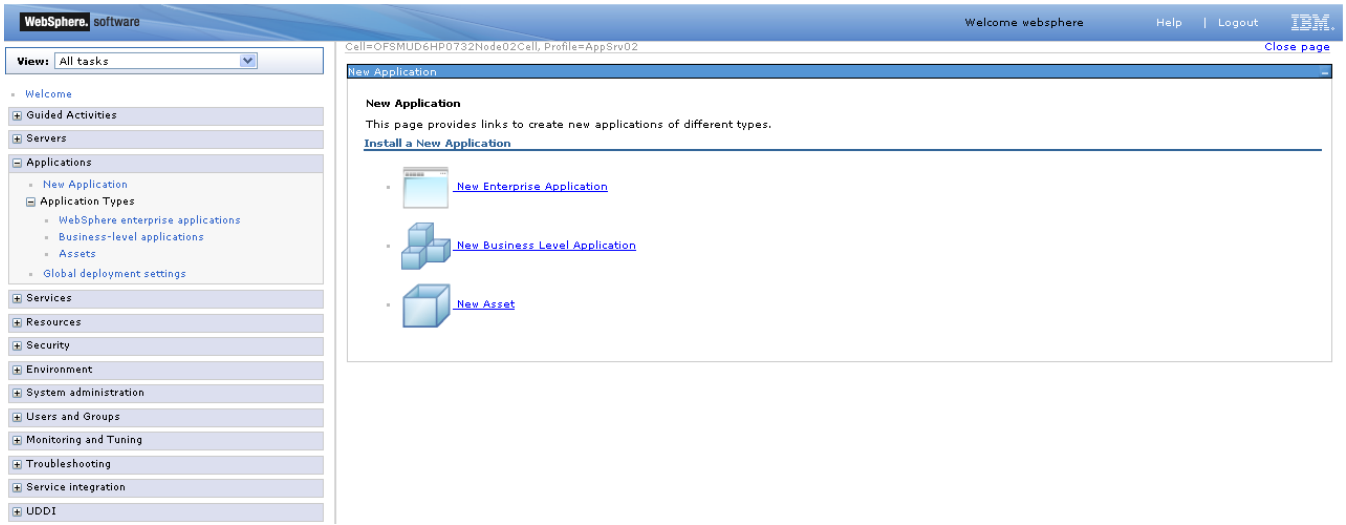


These steps needs to be followed for deploying service.ear-

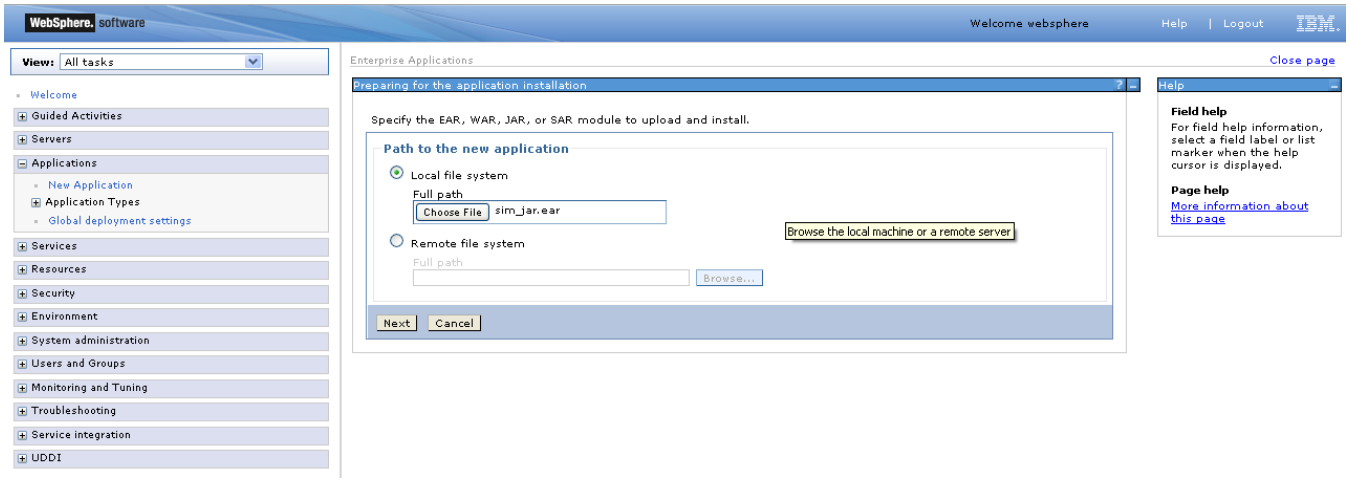
On Admin console, in the left menu, click on “New Application”.



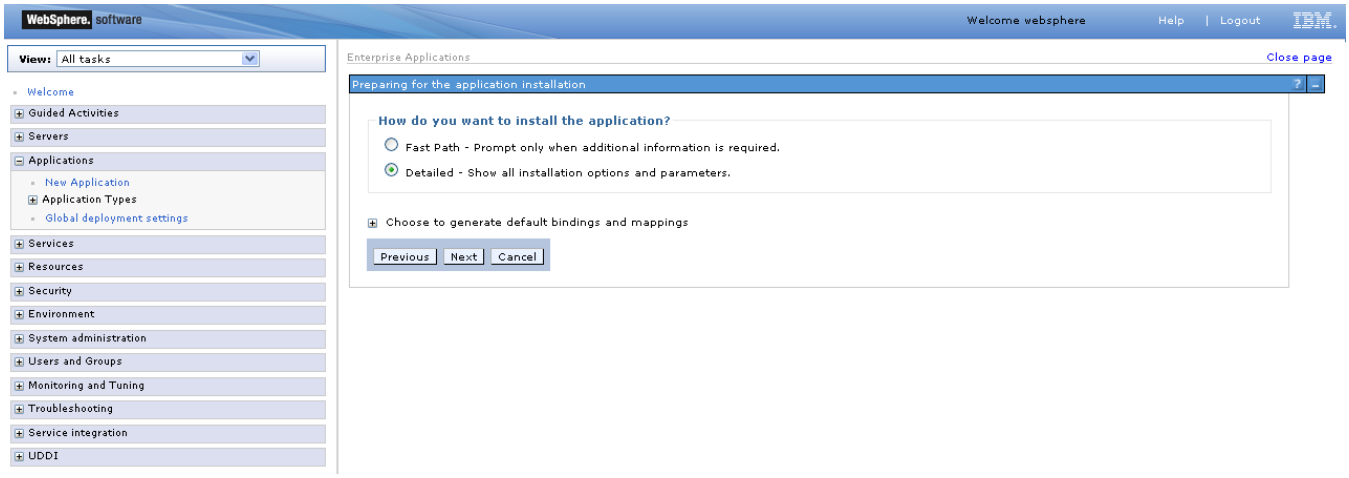
1) Screen as shown in below figure will appear. On the screen, click “New Enterprise Application”.



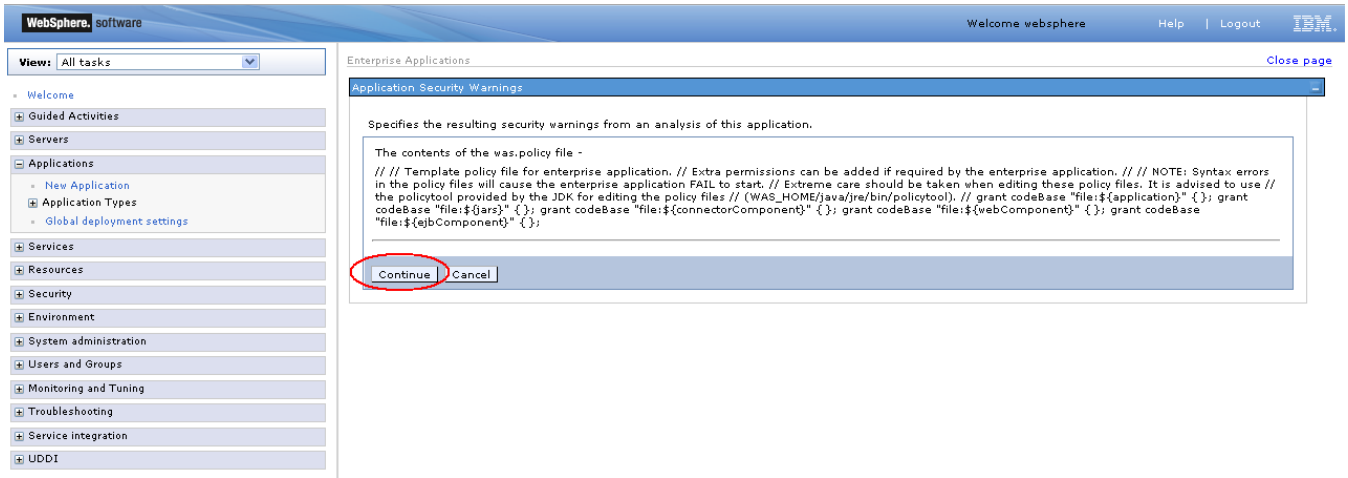
2) On the screen as show in figure below, use browse button to select the EAR file. Click on “Next” button.



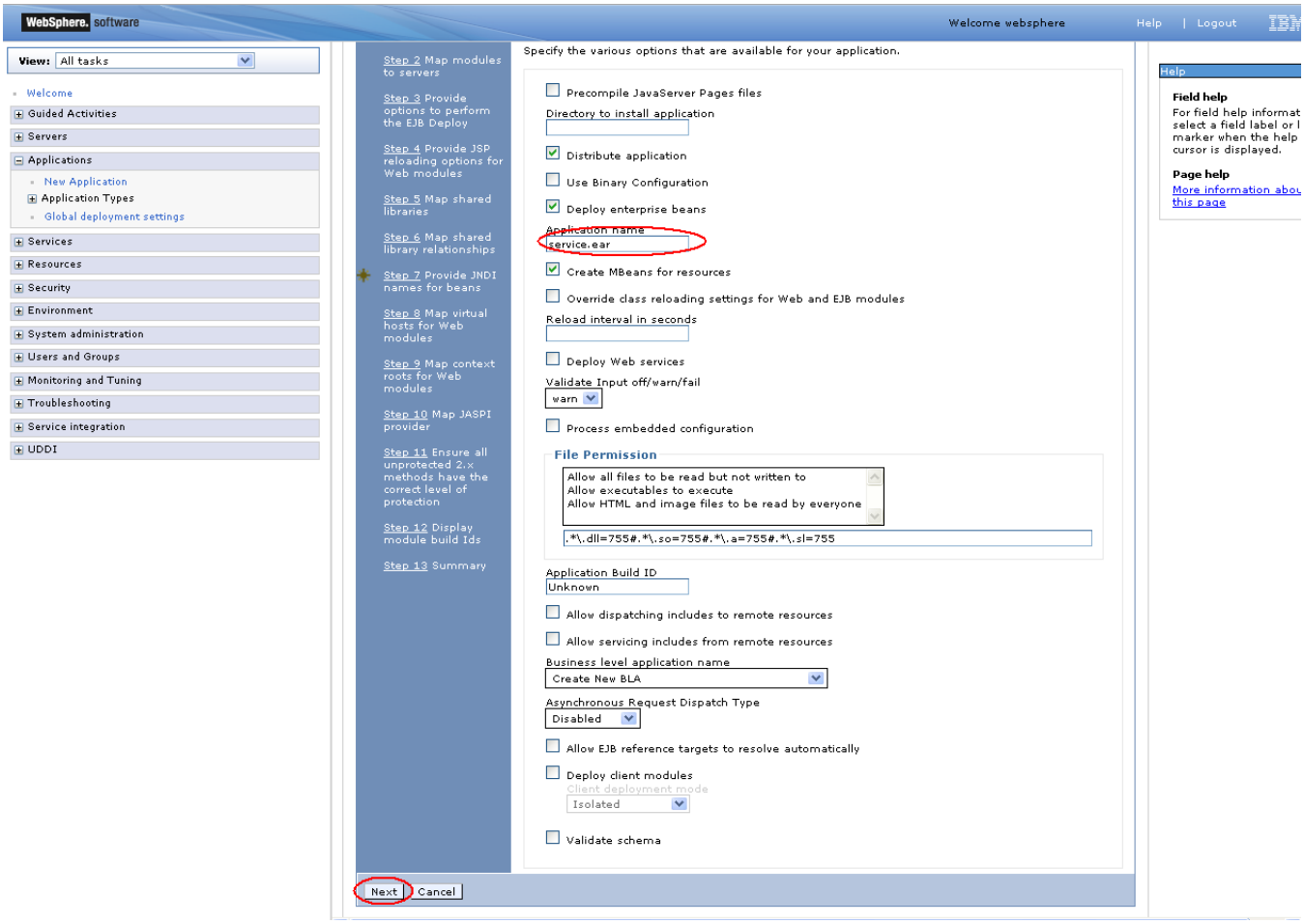
3) On the screen as show in figure below, select the radio option “Show all installation options and parameters”. Click on “Next” button.



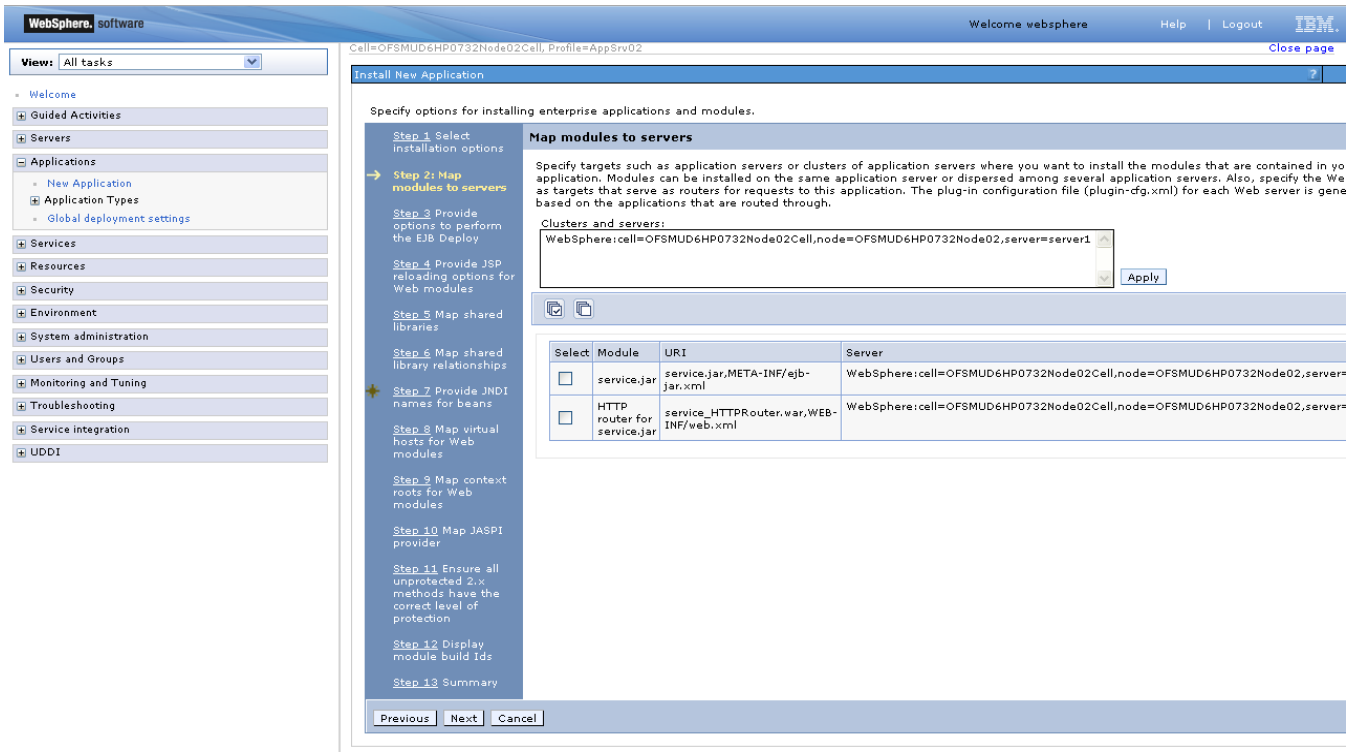
4) On the screen as shown below, click “Continue” button.



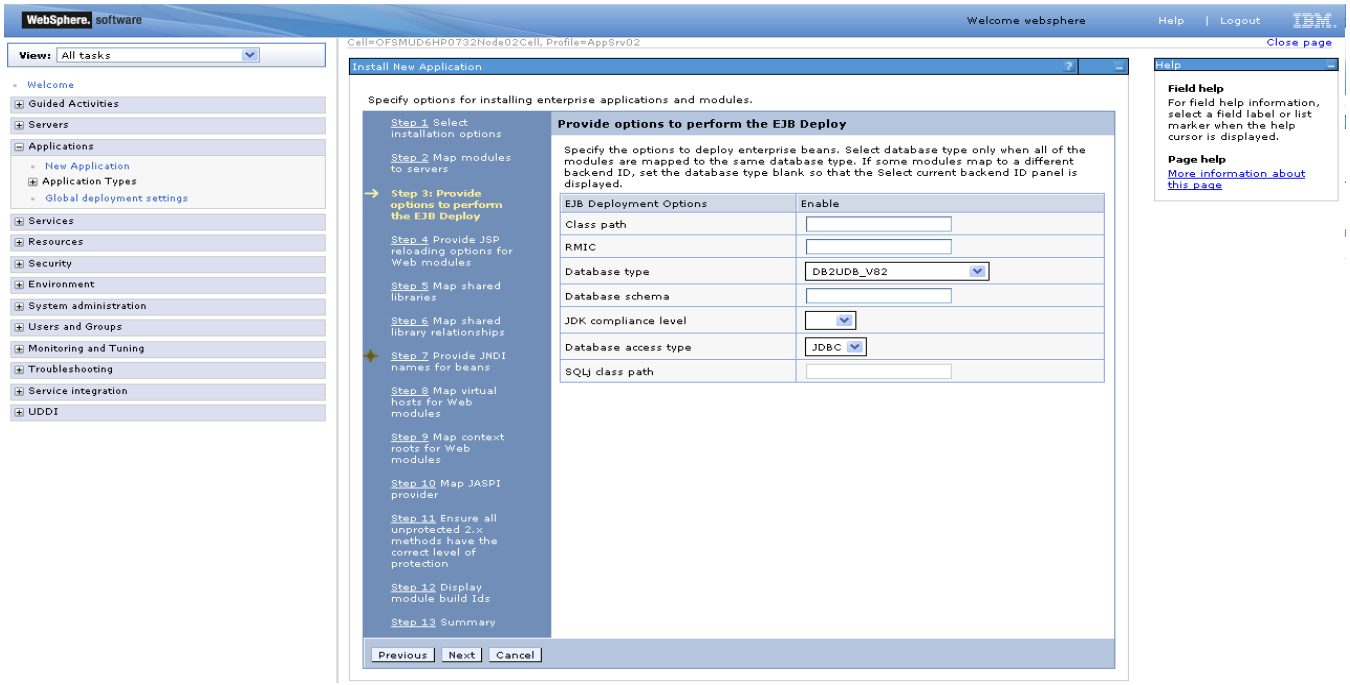
5) On the Screen as shown in below figure, Update the application name if required and click “Next”



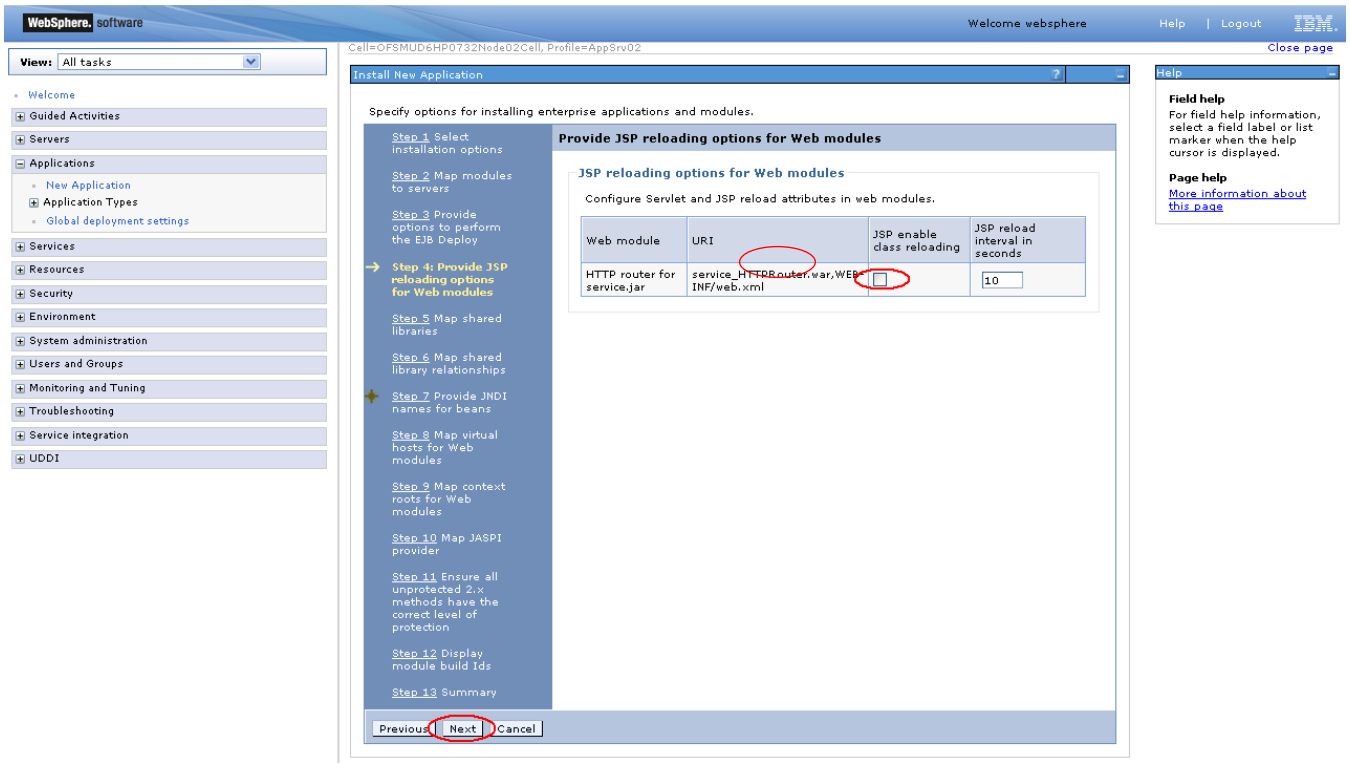
6) In screen “Map modules to server” as shown in figure below, map the modules to the applicable server(s) and click “Next”.



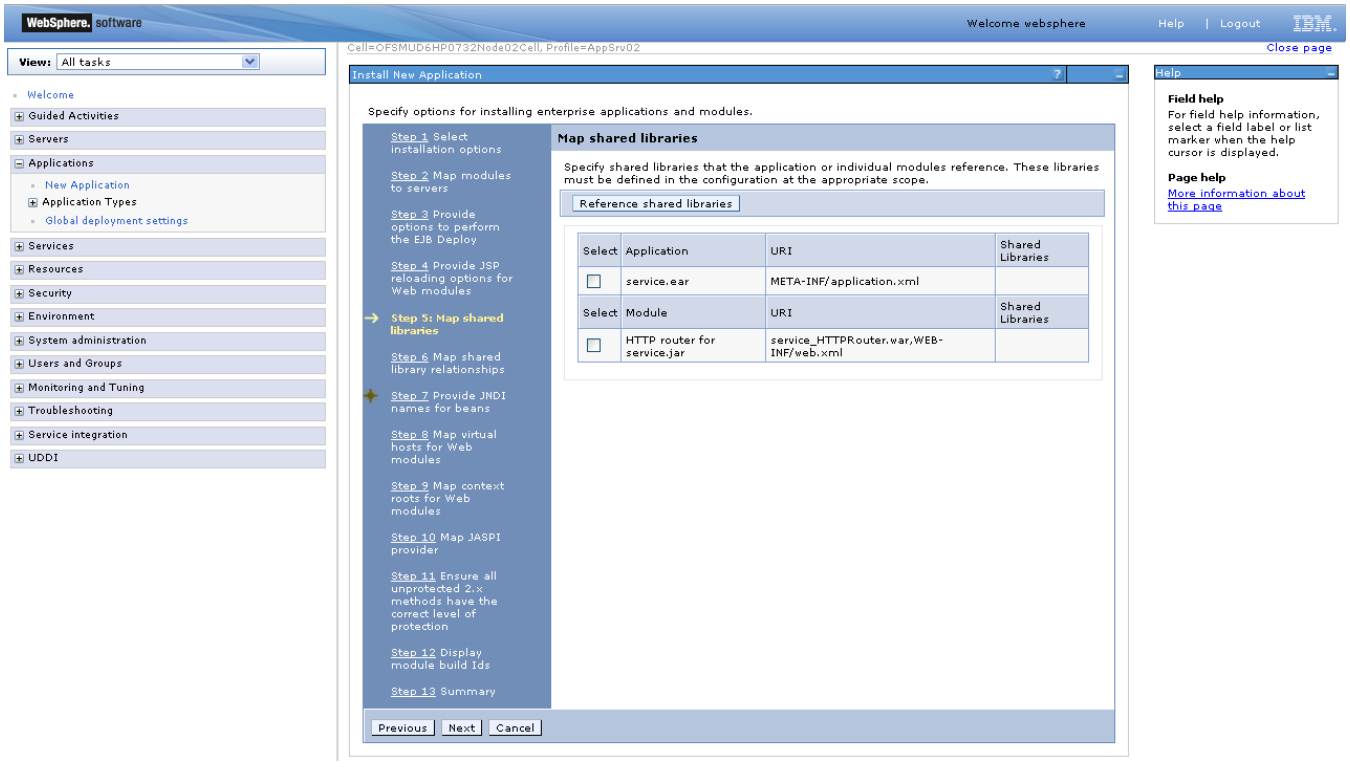
7) On screen “Provide options” as shown in figure below, add the FCDB Kernel jar to the classpath. Add <FCDB BASE DIR>/system/build/kernel/FCDB_<version>.jar to the class path field and click “Next”.



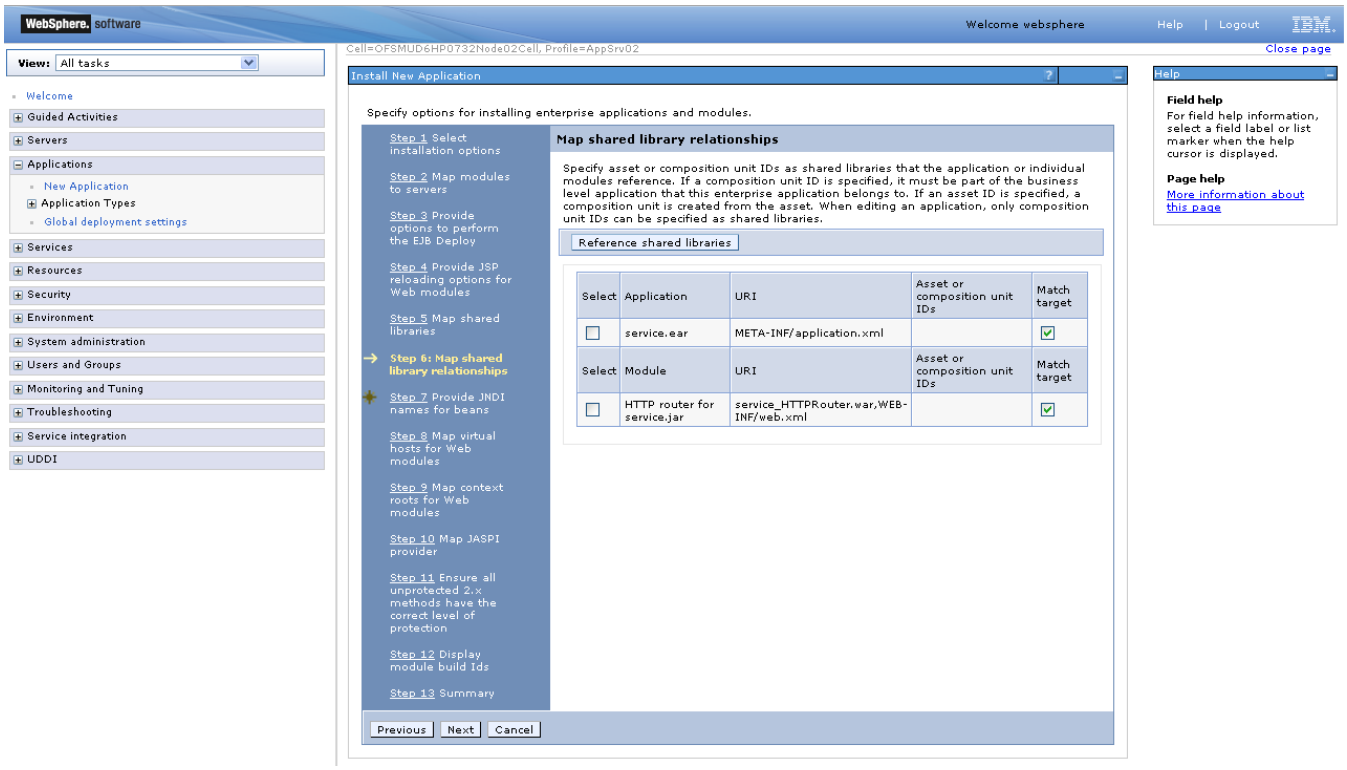
8) Disable JSP Class reloading for all web modules and click “Next”



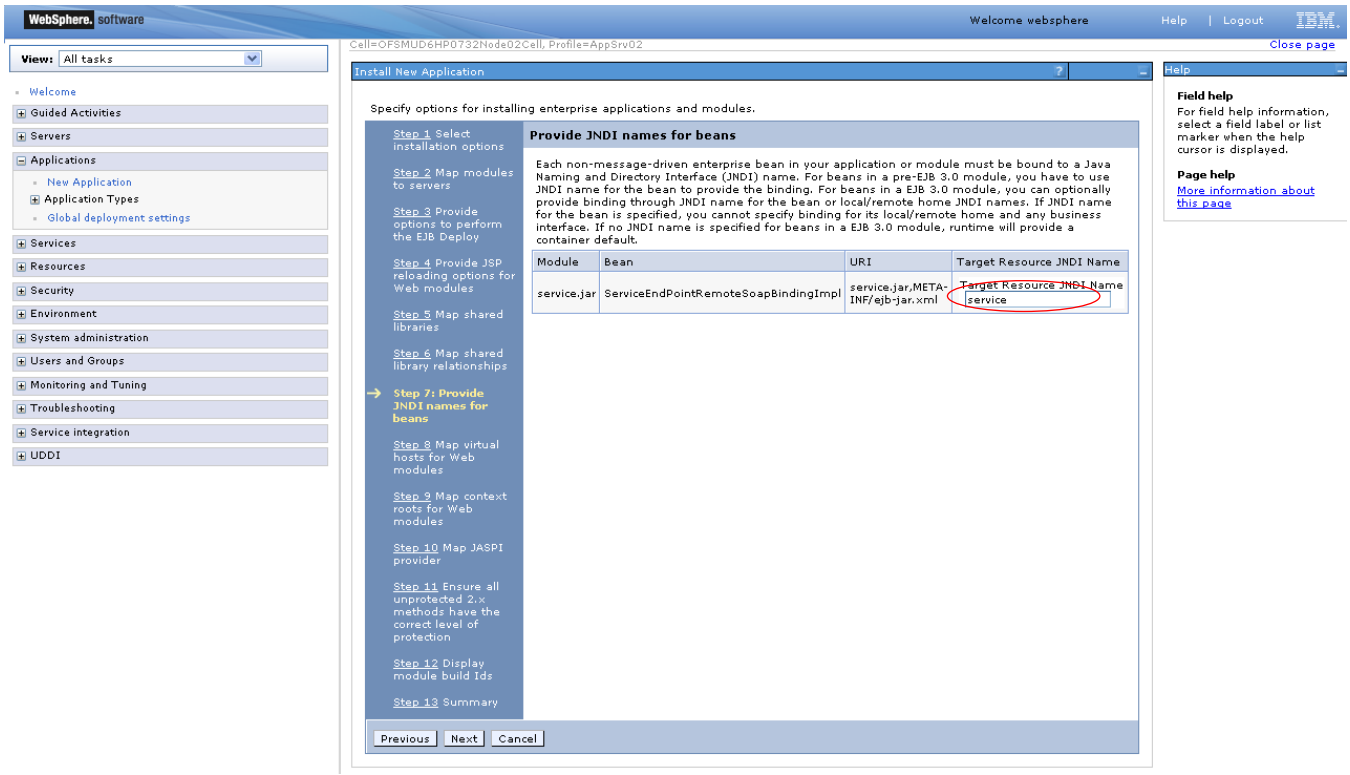
9) On the screen “Map Shared libraries” as shown below, click “Next”



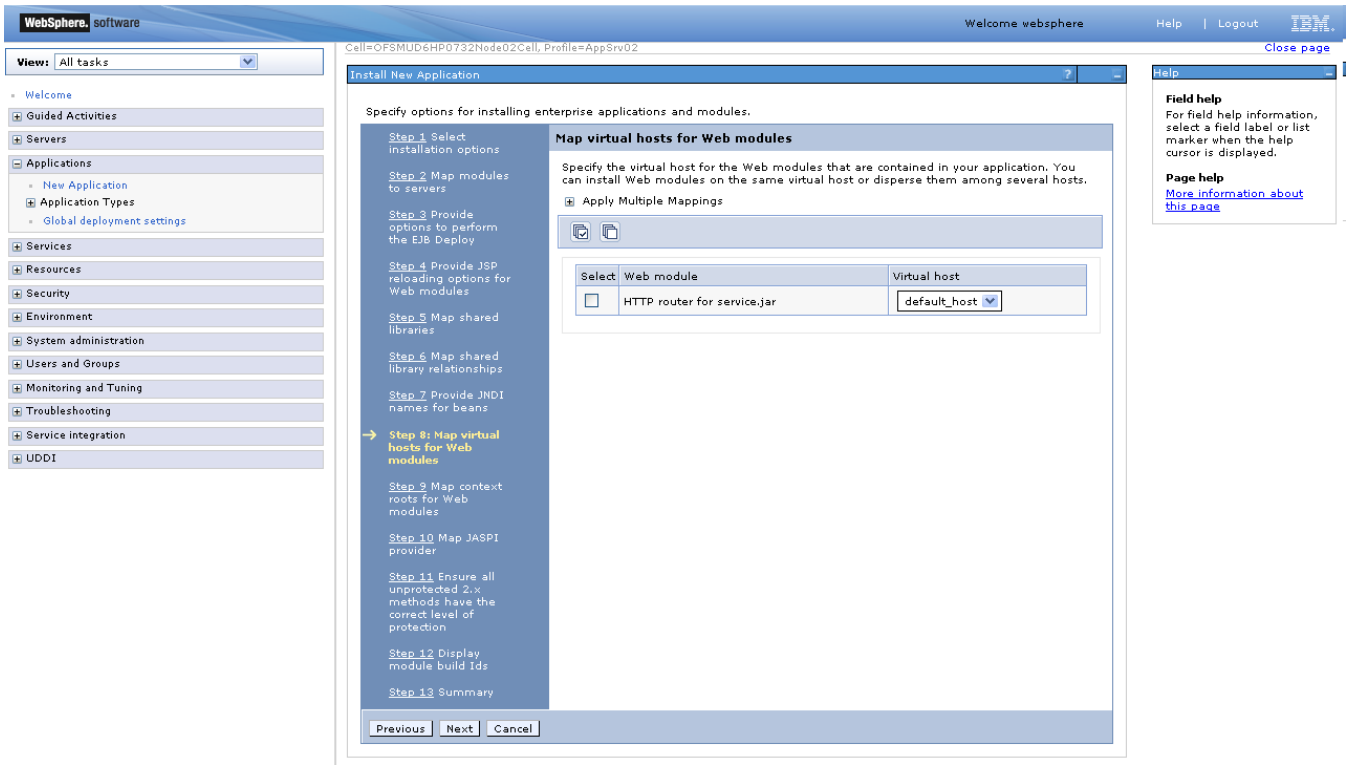
10) On “Map shared library relationships” click next.



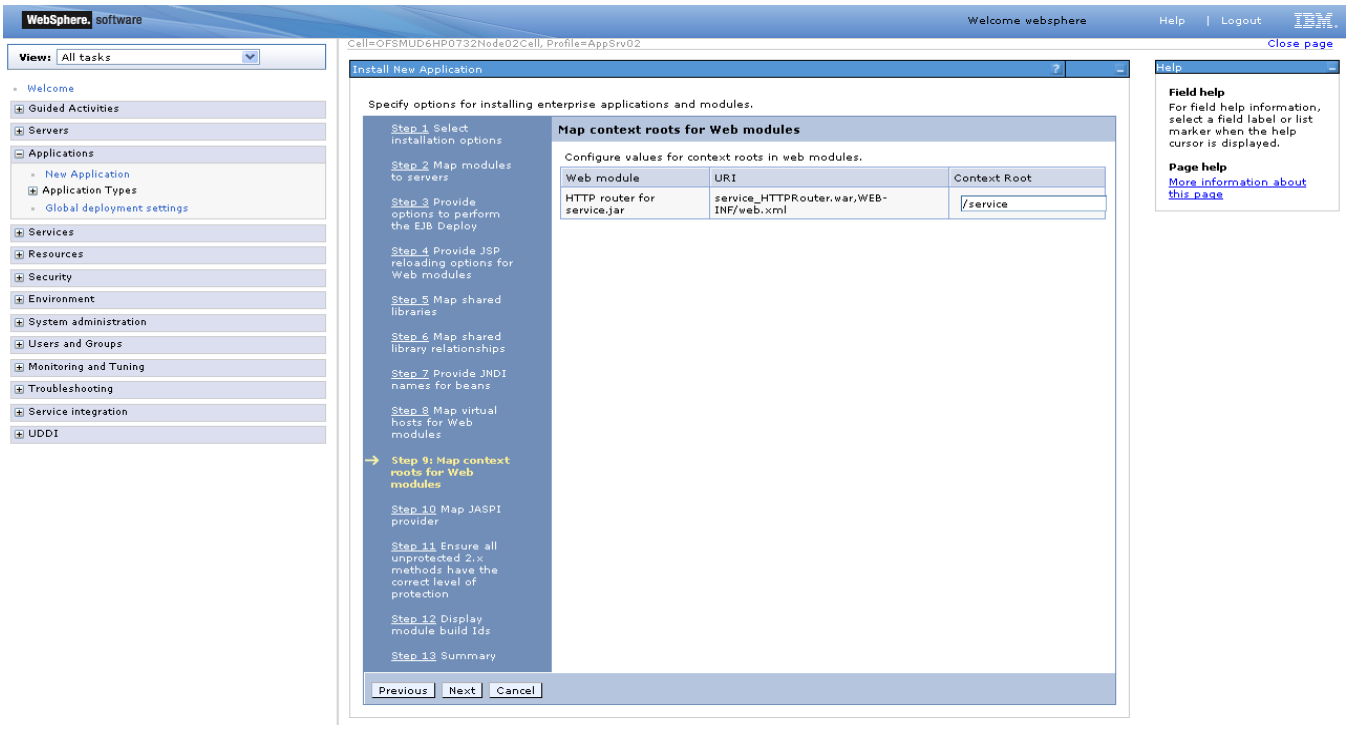
11) On the Screen “Provide JNDI names” as show below, provide the JNDI name for the application in field “Target Resource JNDI Name”. The JNDI name for each FCDB application has been provided at the start of this section.



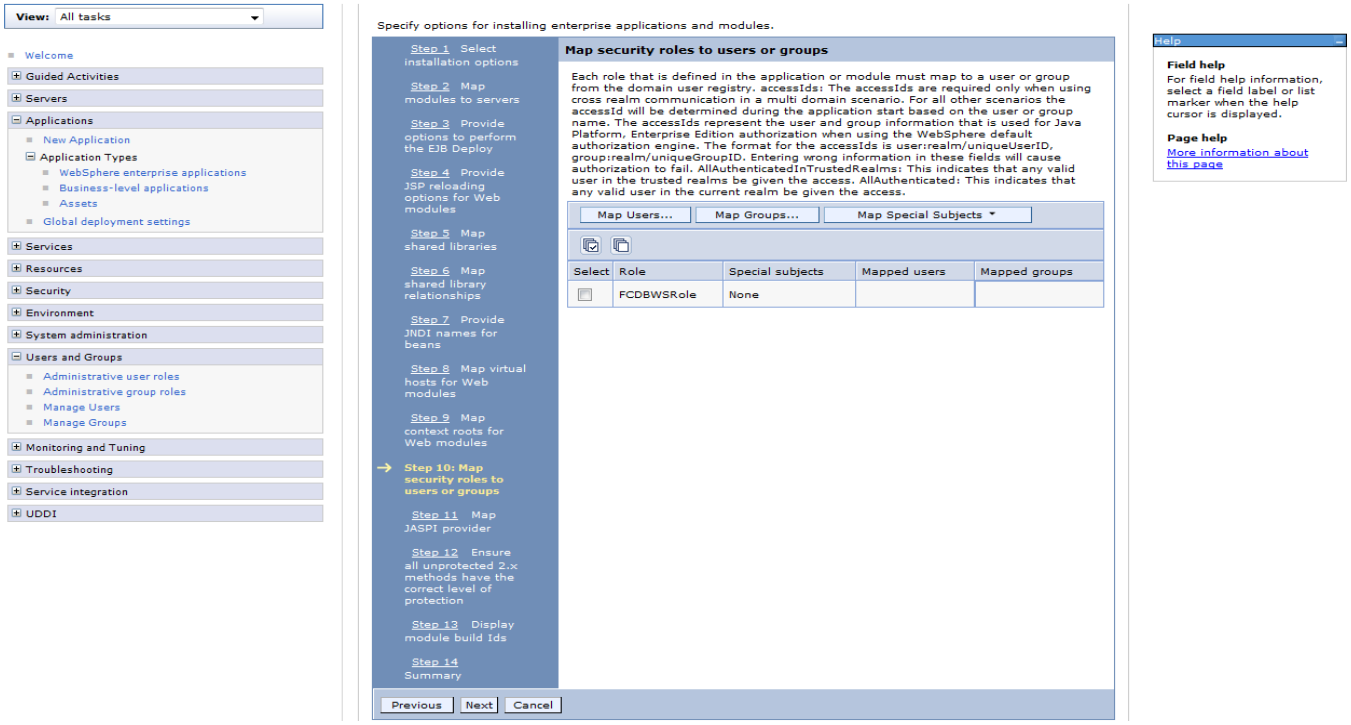
12) On the screen “Map Virtual host”, map web modules to the relevant virtual host & click Next.



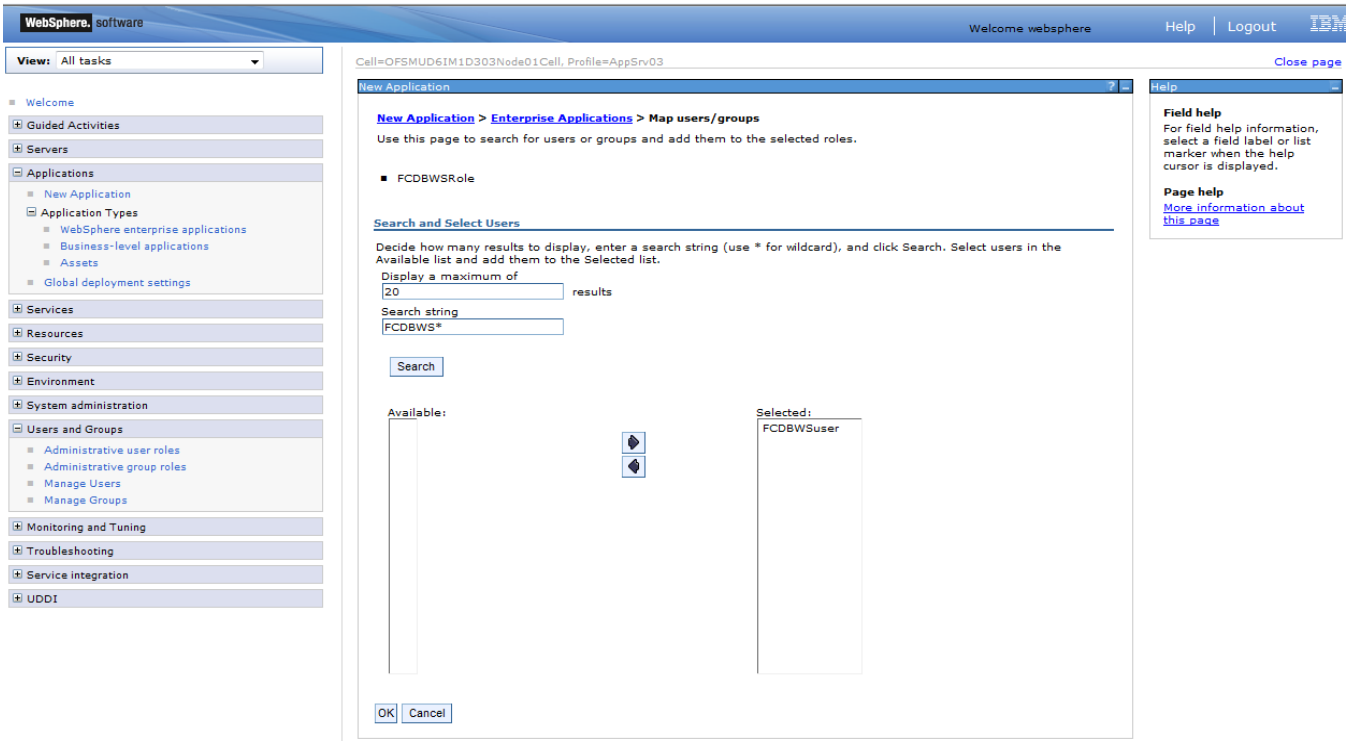
13) On screen “Map Context roots”, map the web modules with respective context root. The context root has been configured in the deployable and the default context root being displayed can be used. If no context root is displayed, use the context root as mentioned for the application at the start of this section.



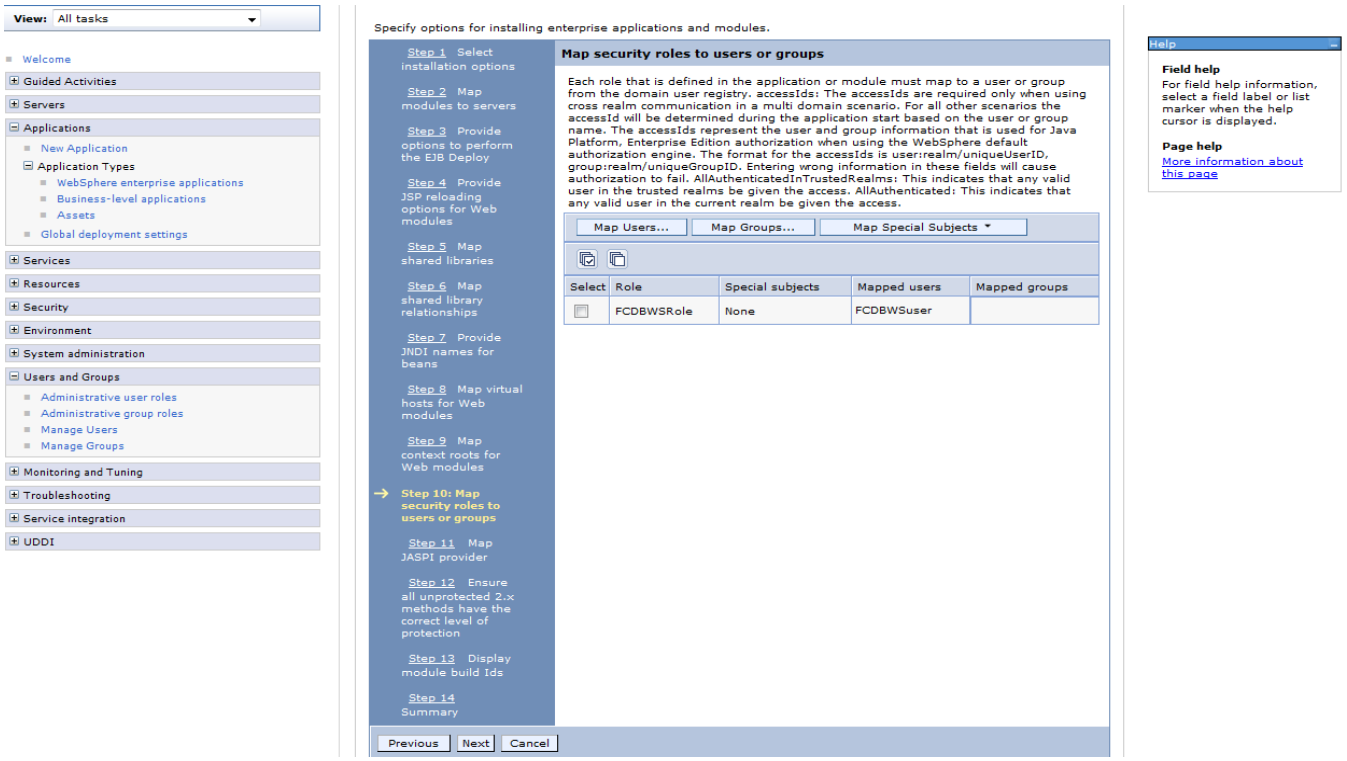
14) On screen “Map security roles to users or groups”, map the role “FCDBWSRole” and map it with the user created before deploying the ear.



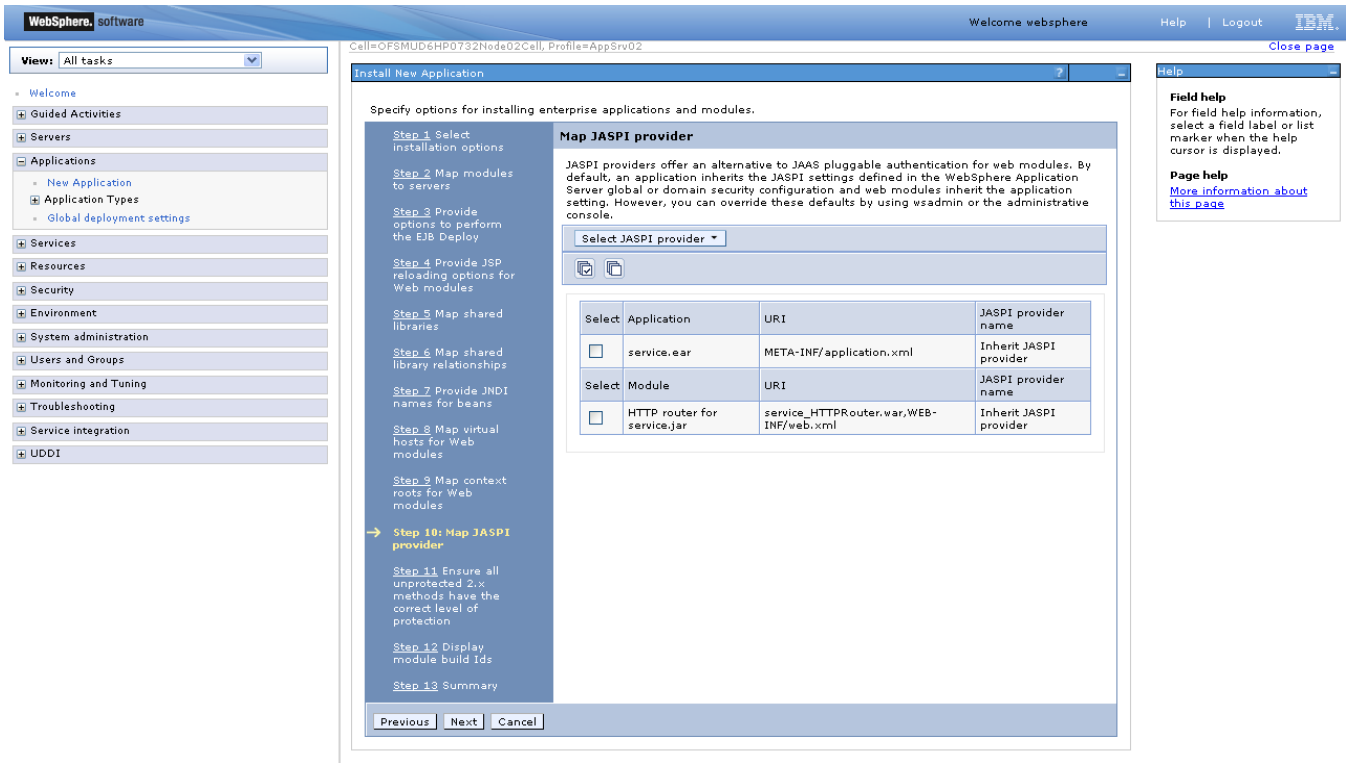
15) On the screen as show below, click “Ok”.



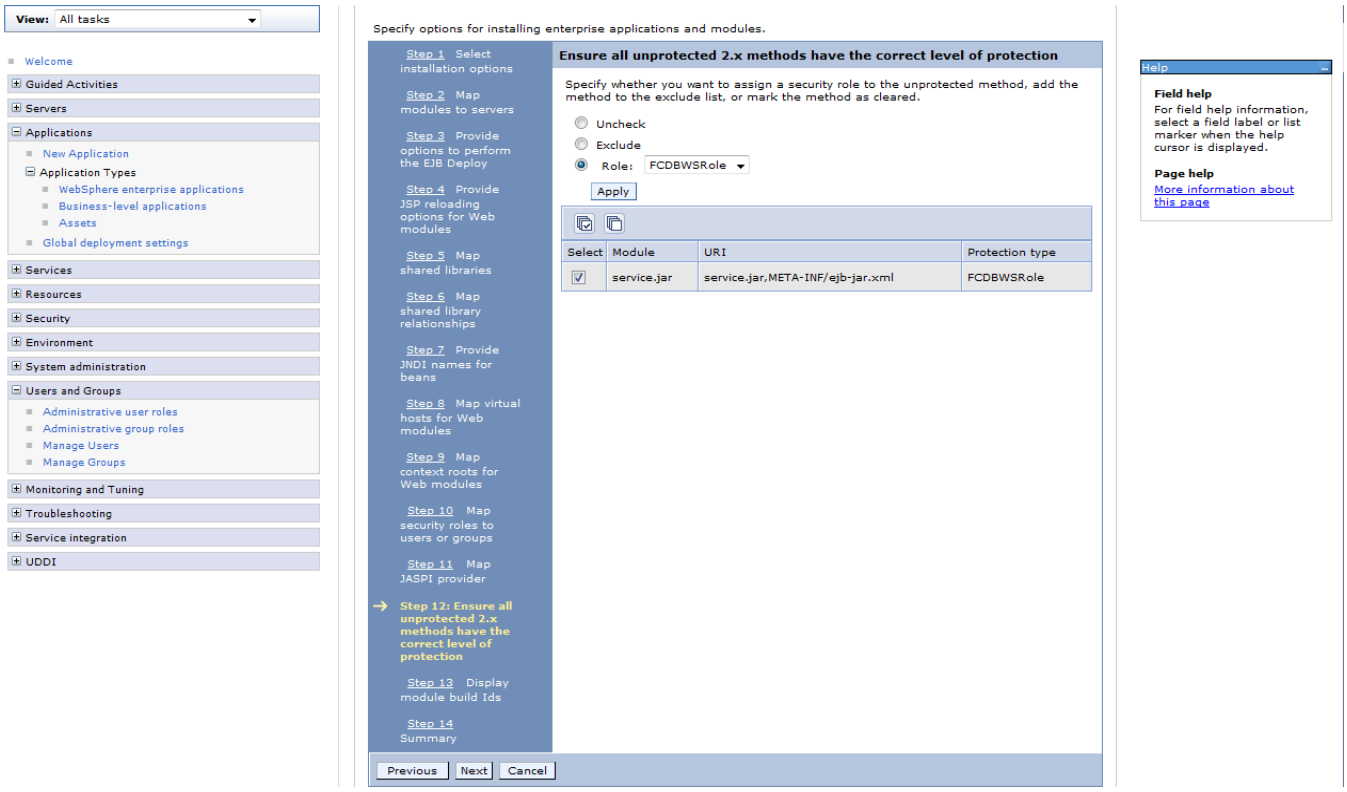
16) On the screen as show below, click “Next”.



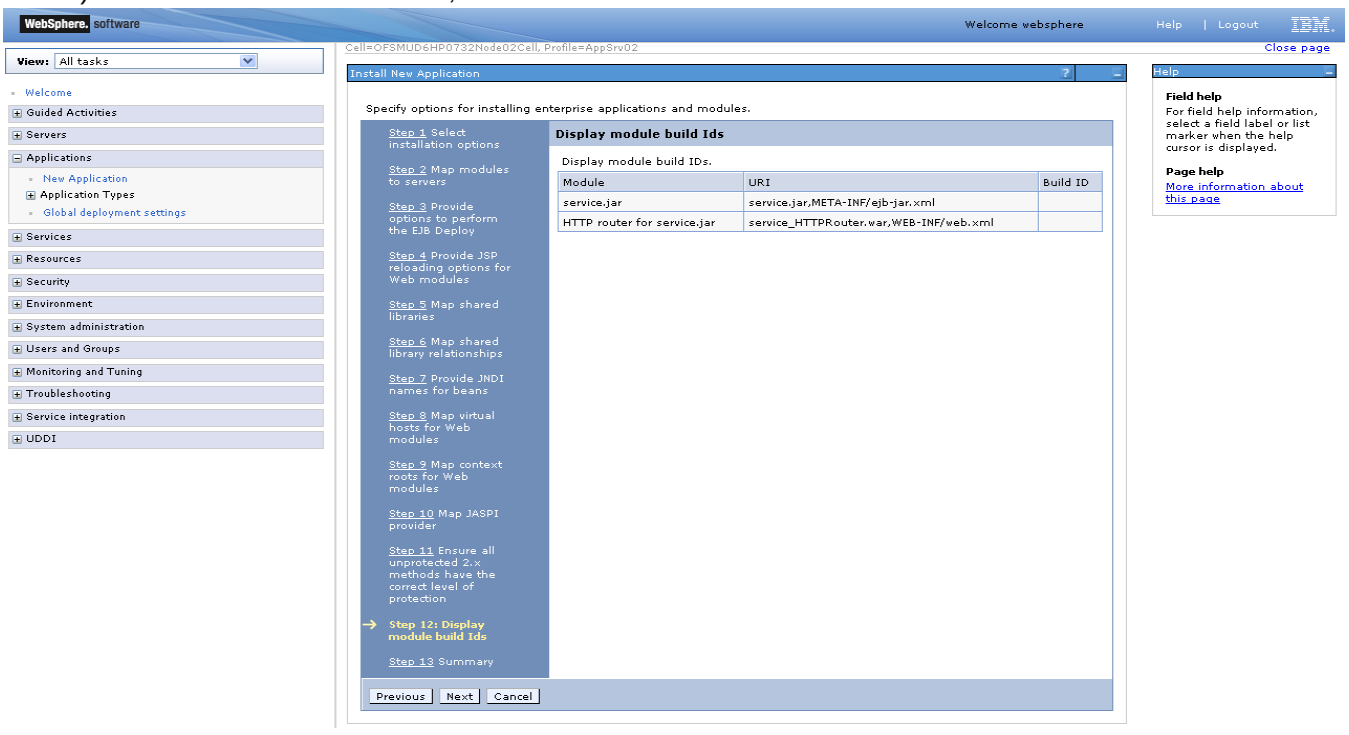
17) On the screen as show below, click “Next”.



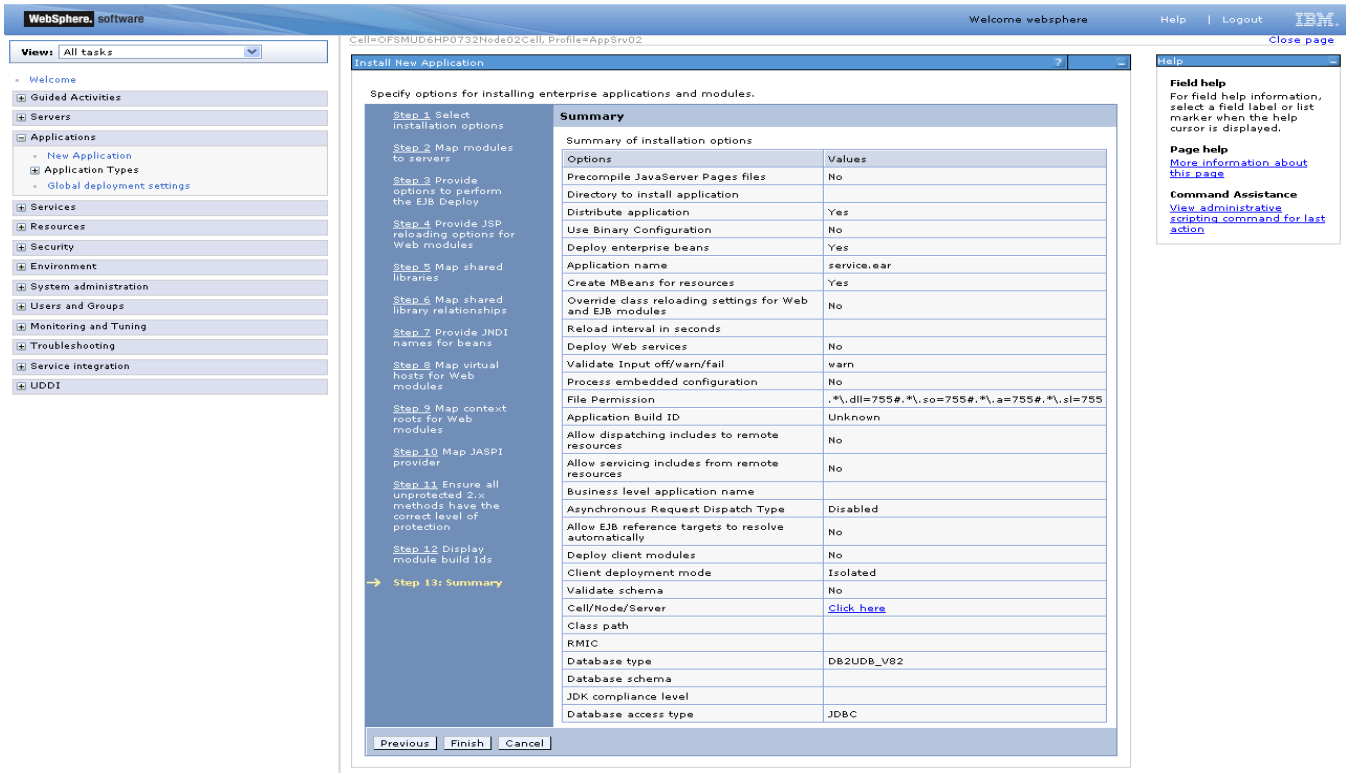
18) On the screen as show below, select the role and associate it with the module by clicking on “Apply”. Now click on “Next”



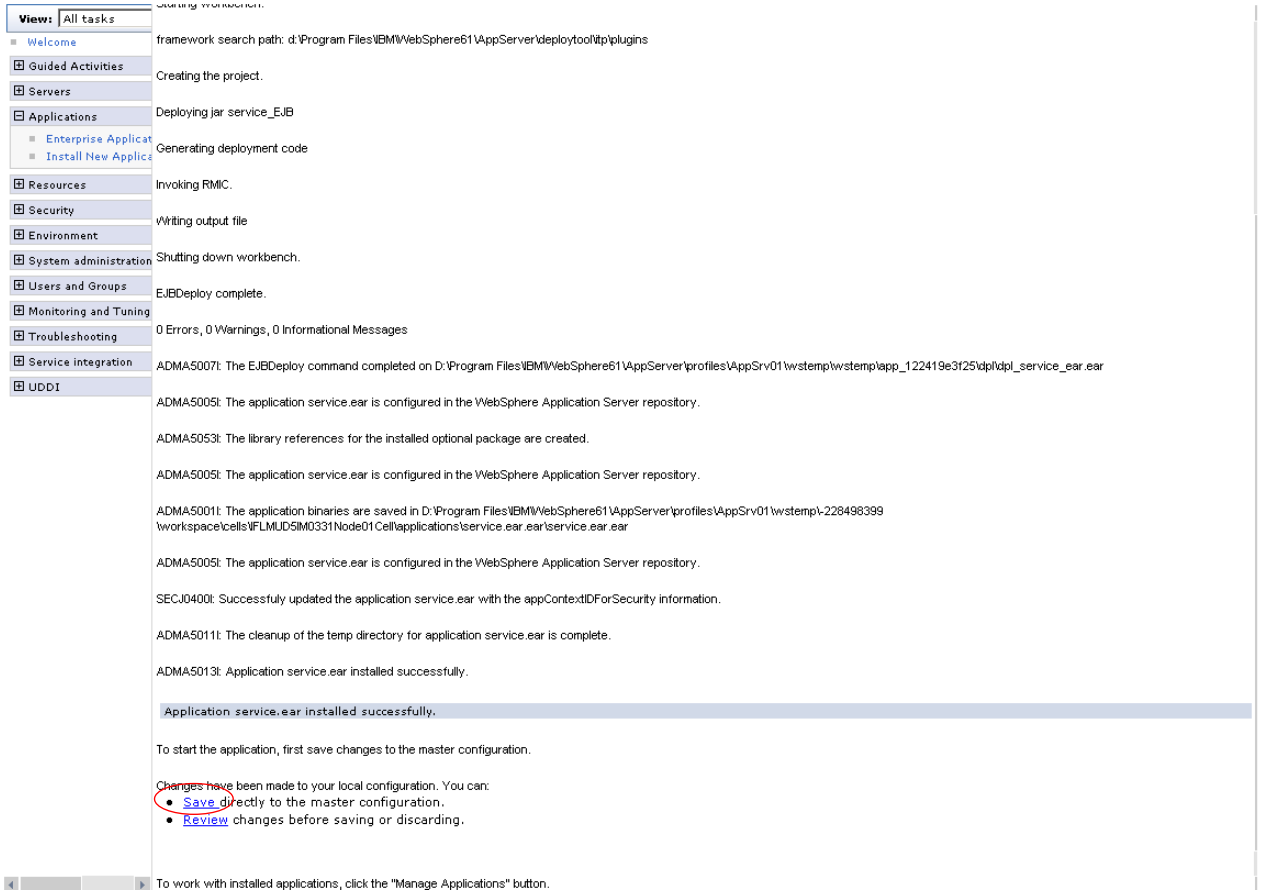
19) On the screen as show below, click “Next”.



20) On the “Summary” page, summary of selected installation options will be displayed. Verify the same and click “Next” to begin with application installation.


WebSphere. software
Welcome websphere
Help | Logout
IBM.
Cell=OFSMUD6HP0732Node02Cell, Profile=AppSrv02
Close page
View: All tasks
Welcome
Guided Activities
Servers
Applications
New Application
Application Types
Global deployment settings
Services
Resources
Security
Environment
System administration
Users and Groups
Monitoring and Tuning
Troubleshooting
Service integration
UDDI
Install New Application
Specify options for installing enterprise applications and modules.
Step 1 Select installation options
Step 2 Map modules to servers
Step 3 Provide options to perform the EJB Deploy
Step 4 Provide JSP reloading options for Web modules
Step 5 Map shared libraries
Step 6 Map shared library relationships
Step 7 Provide JNDI names for beans
Step 8 Map virtual hosts for Web modules
Step 9 Map context roots for Web modules
Step 10 Map JASPI provider
Step 11 Ensure all unprotected 2.x methods have the correct level of protection
Step 12 Display module build Ids
Step 13: Summary
Summary
Summary of installation options
Options
Values
Precompile JavaServer Pages files
No
Directory to install application
Distribute application
Yes
Use Binary Configuration
No
Deploy enterprise beans
Yes
Application name
service.ear
Create MBeans for resources
Yes
Override class reloading settings for Web and EJB modules
No
Reload interval in seconds
Deploy Web services
No
Validate Input off/warn/fail
warn
Process embedded configuration
No
File Permission
.*\.dll=755#.*\.so=755#.*\.a=755#.*\.sl=755
Application Build ID
Unknown
Allow dispatching includes to remote resources
No
Allow servicing includes from remote resources
No
Business level application name
Asynchronous Request Dispatch Type
Disabled
Allow EJB reference targets to resolve automatically
No
Deploy client modules
No
Client deployment mode
Isolated
Validate schema
No
Cell/Node/Server
Click here
Class path
RMIC
Database type
DB2UDB_V82
Database schema
JDK compliance level
Database access type
JDBC
Previous
Finish
Cancel
Help
Field help
For field help information, select a field label or list marker when the help cursor is displayed.
Page help
More information about this page
Command Assistance
View administrative scripting command for last action

21) The next screen displays application installation status. On successful installation, click the link “Save” to save the installation.



22) In fcat-config.xml set the following properties with username and password of the user created in above steps and restart the server.

Field	Value
FCAT.CONNECT.SVC.USERNAME	<Enter username>
FCAT.CONNECT.SVC.PASSWORD	<Enter password>

The following section explains deploying JAR application on IBM Websphere. For details and different options of deployment refer to documentations provided by IBM Websphere.

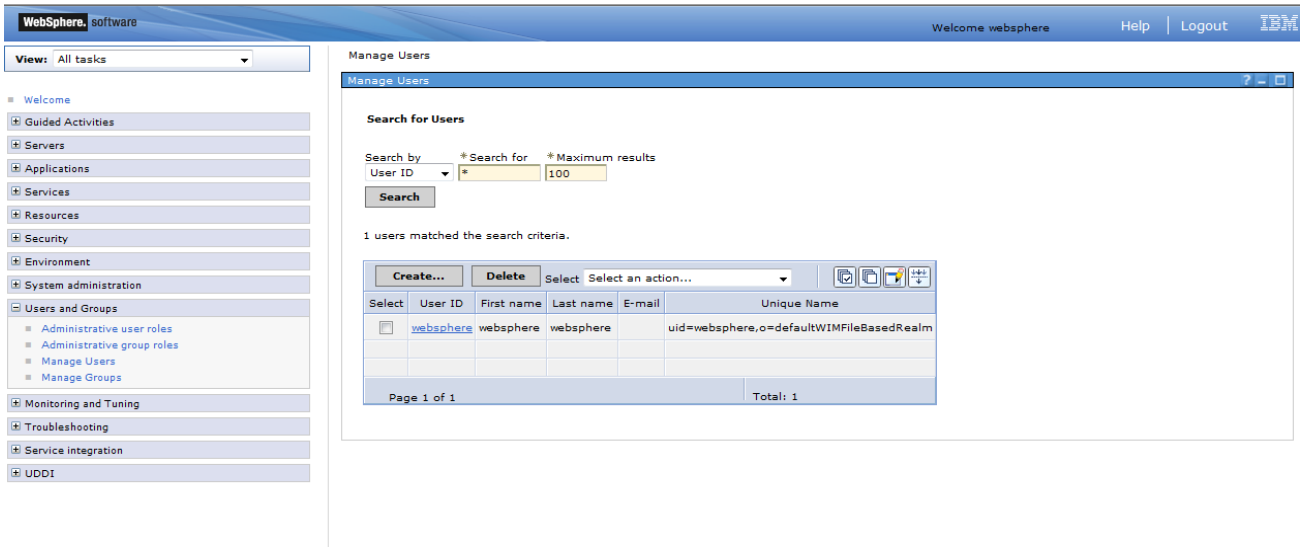
These steps should be carried out for each FCDB jar application mentioned below.

Deployable	JNDI Name/Root	Path Location	Description
ServiceEndPointRI.jar	ServiceEndPoint	<FCDB BASE DIR>\deploy\ websphere	This application should be deployed if the mode of service tier invocation is EJB mode.

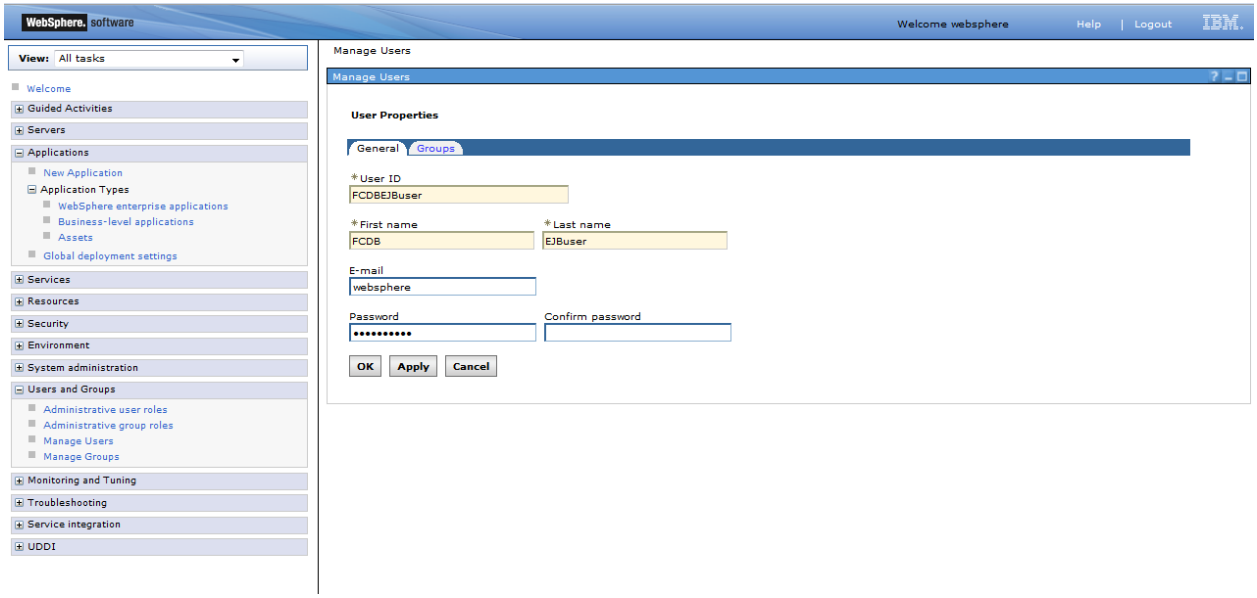
Before deploying ServiceEndPointRI.jar, the below steps needs to be followed to create an user for enabling EJB mode authentication –

5. On Admin console, in the left menu, click on “Manage Users”.

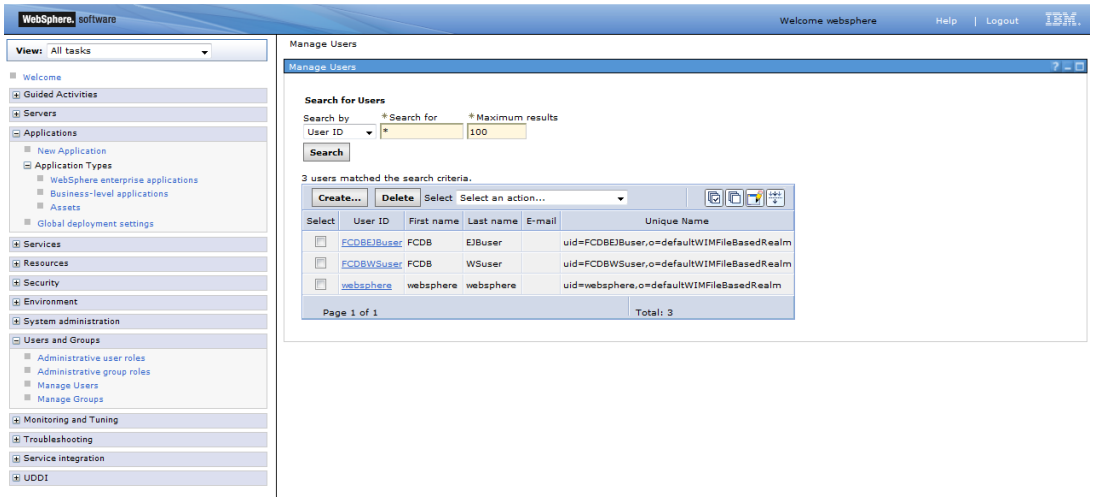
6. Screen as shown in below figure will appear. On the screen, click “Create”.



7. On the screen as show in figure below, enter the user details (e.g. FCDBEJBuser). Click on “Create” button

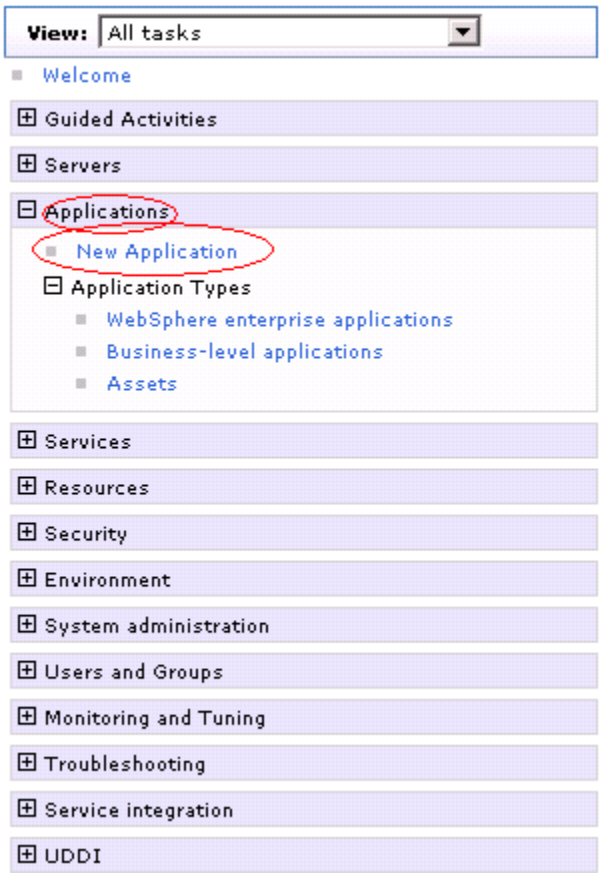


8. User is created successfully in the current security realm.

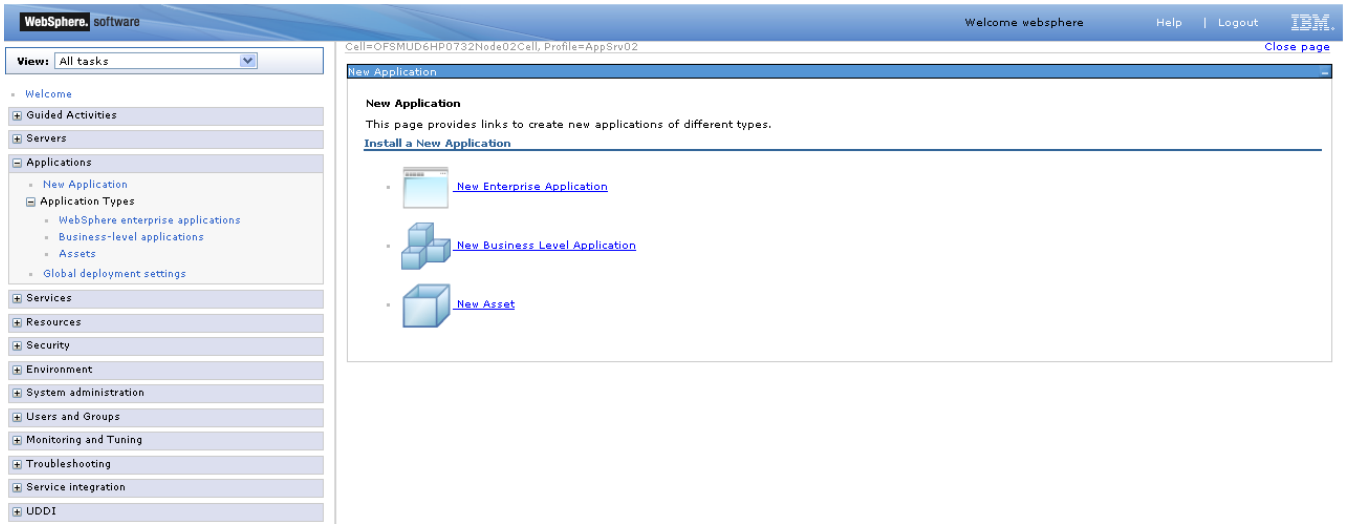


These steps needs to be followed for deploying ServiceEndPointRI.jar-

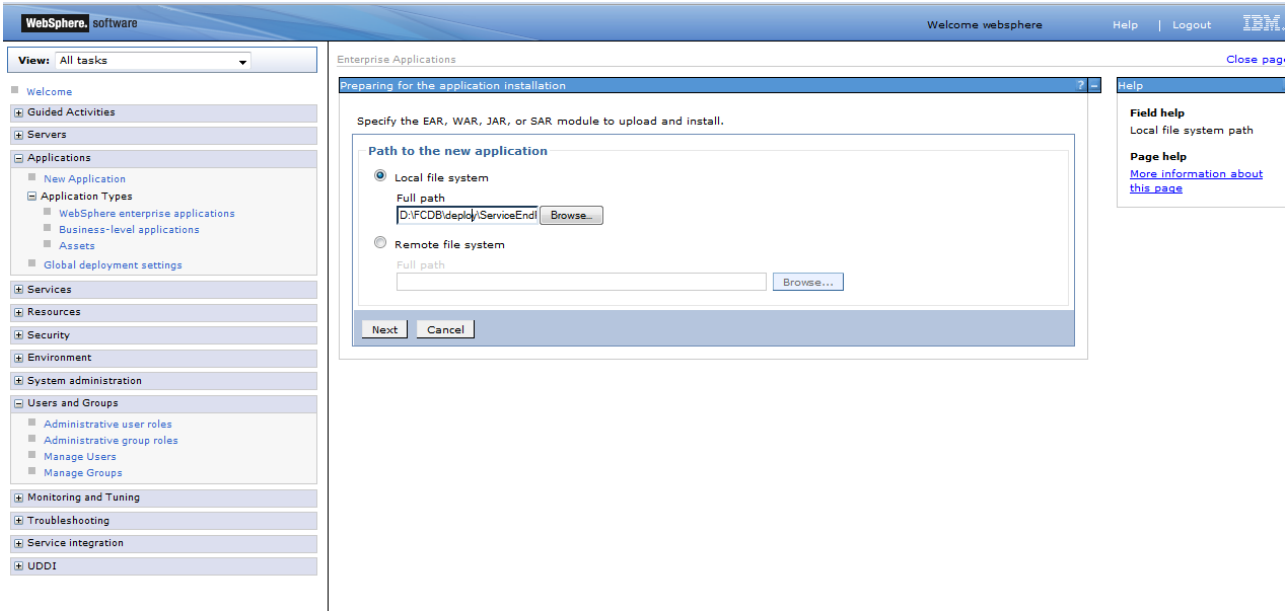
On Admin console, in the left menu, click on “New Application”.



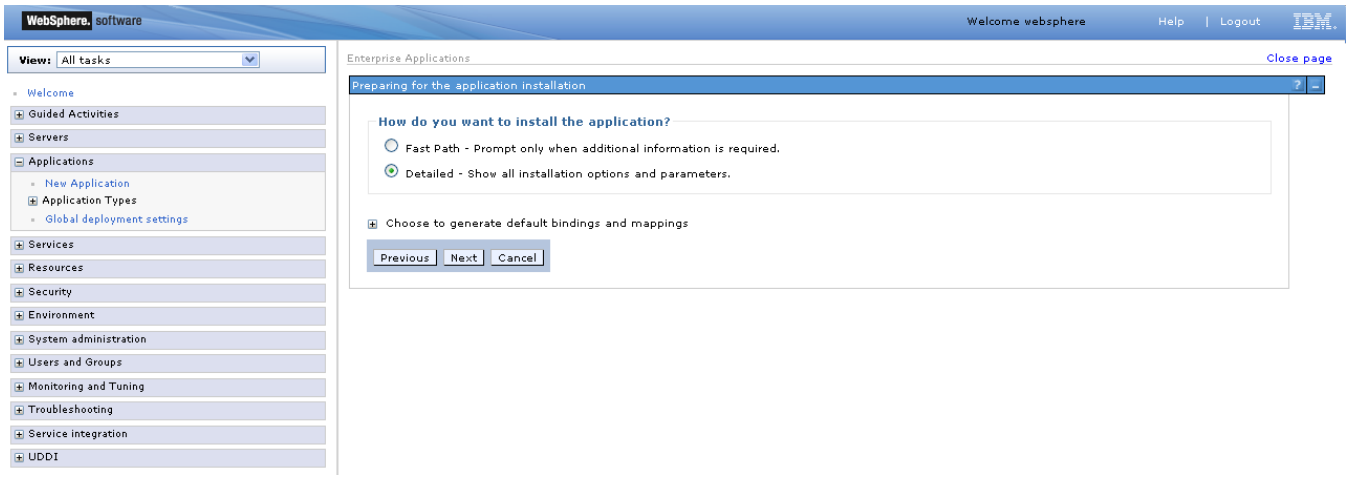
23) Screen as shown in below figure will appear. On the screen, click “New Enterprise Application”.



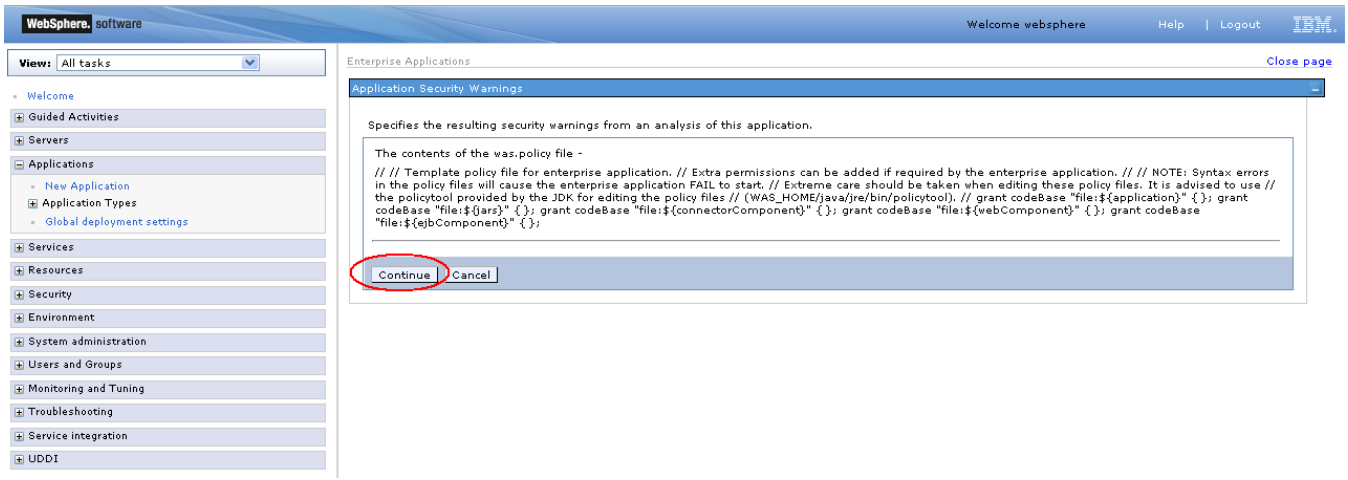
24) On the screen as show in figure below, use browse button to select the JAR file. Click on “Next” button.



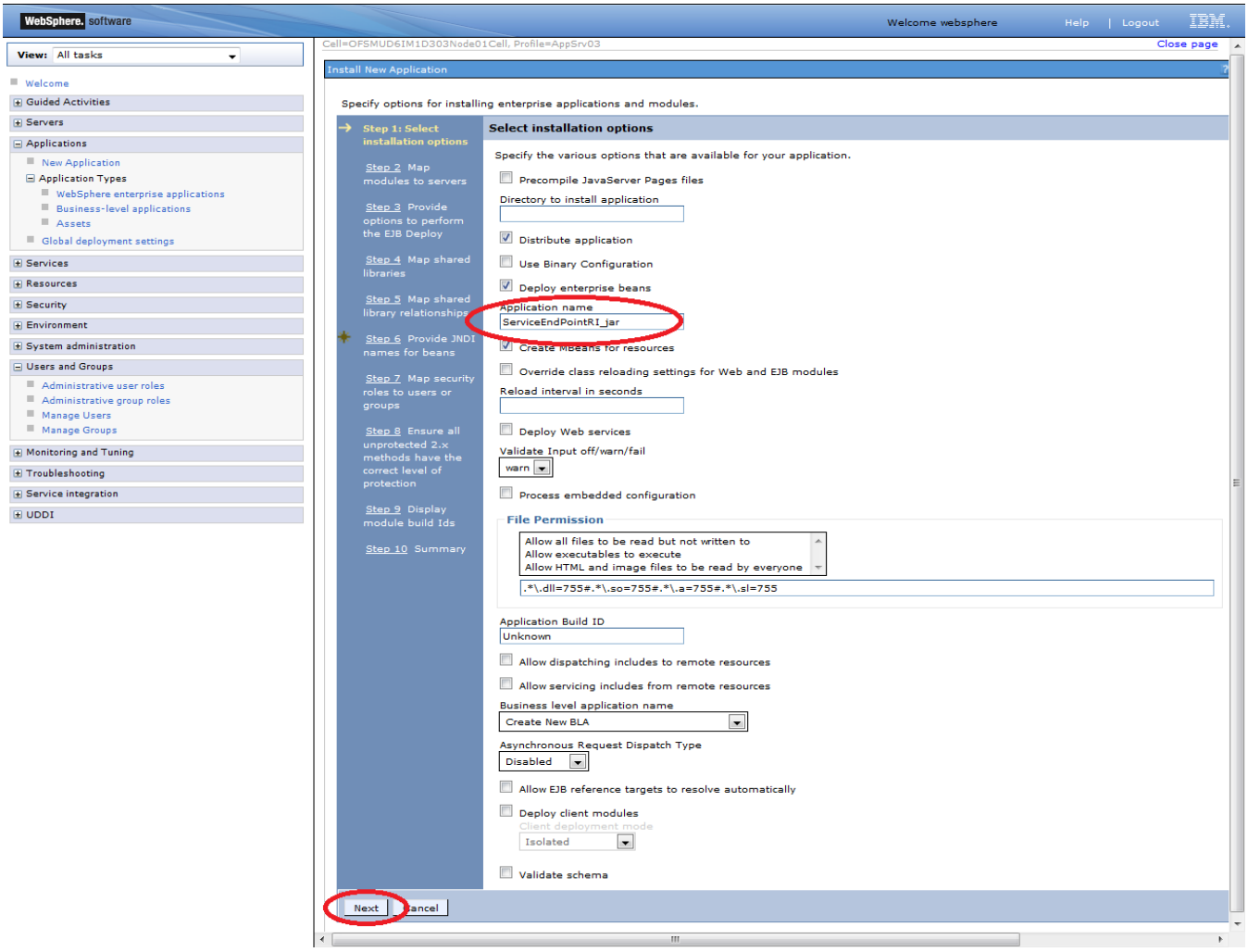
25) On the screen as show in figure below, select the radio option “Show all installation options and parameters”. Click on “Next” button.



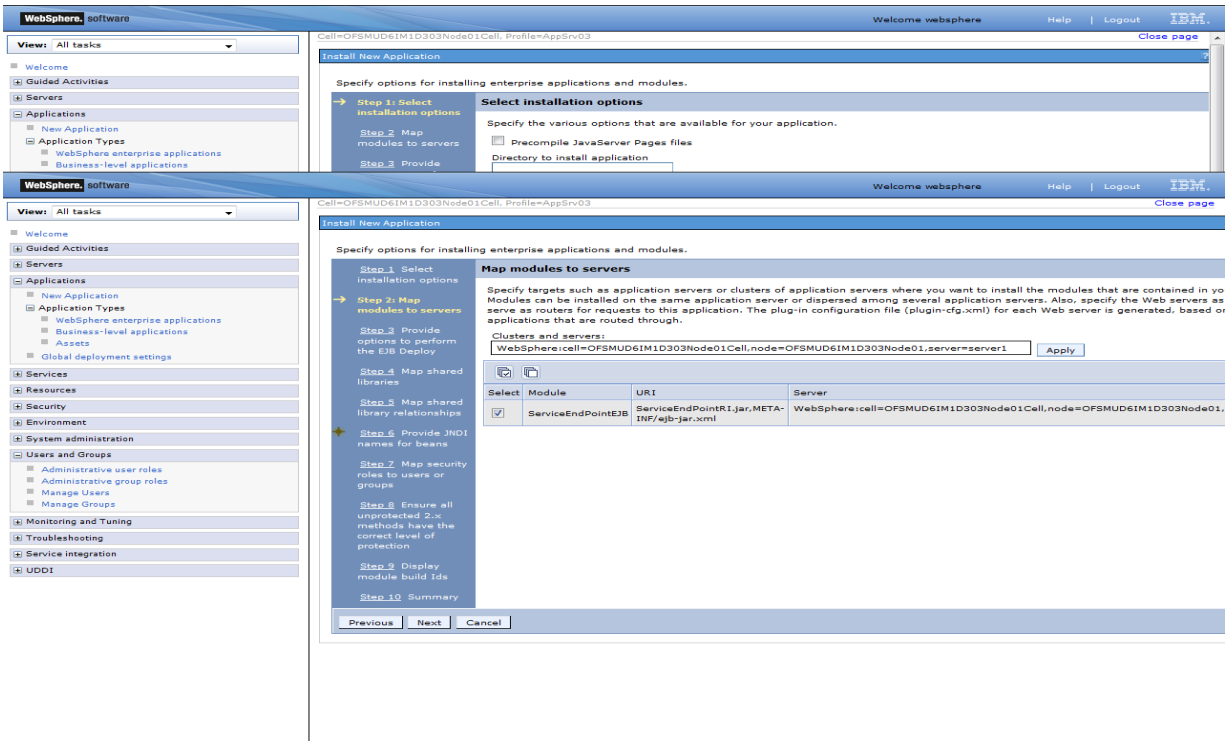
26) On the screen as shown below, click “Continue” button.



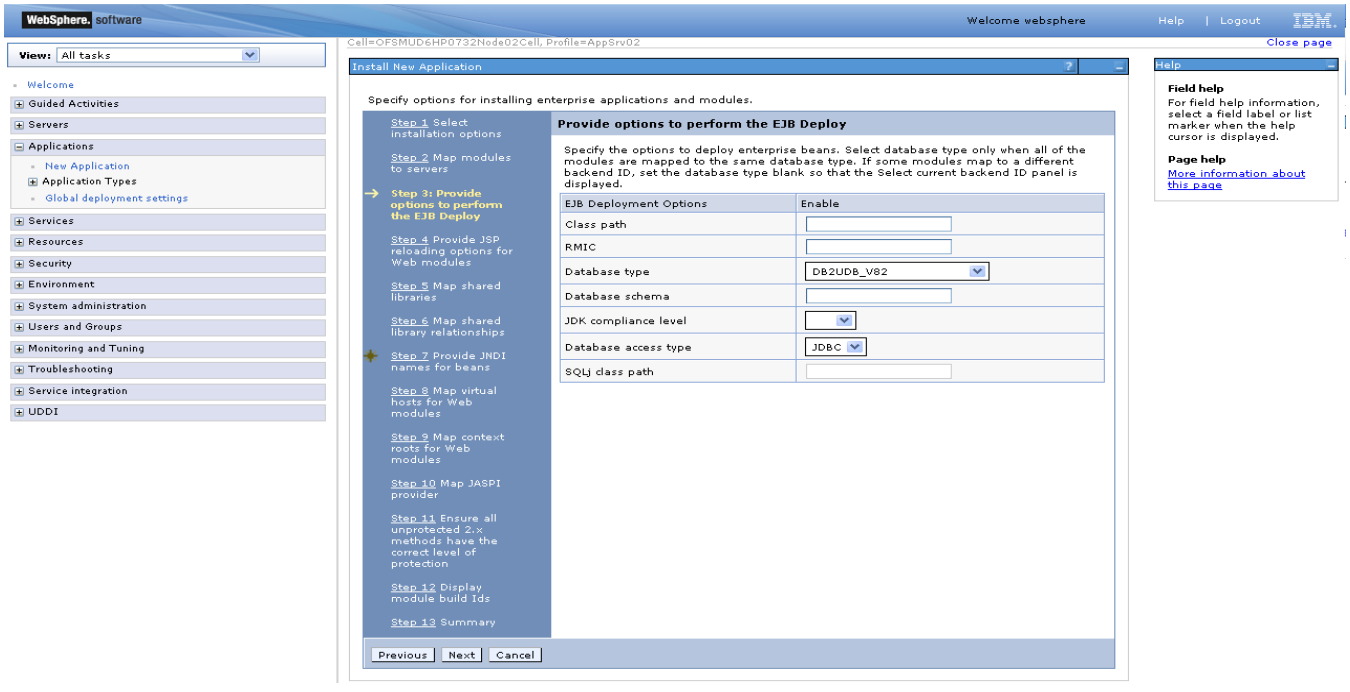
27) On the Screen as shown in below figure, Update the application name if required and click “Next”


WebSphere. software
Welcome websphere
Help | Logout
IBM.
View: All tasks
Welcome
Guided Activities
Servers
Applications
New Application
Application Types
WebSphere enterprise applications
Business-level applications
Assets
Global deployment settings
Services
Resources
Security
Environment
System administration
Users and Groups
Administrative user roles
Administrative group roles
Manage Users
Manage Groups
Monitoring and Tuning
Troubleshooting
Service integration
UDDI
Cell=OFSMUD6IM1D303Node01Cell, Profile=AppSrv03
Close page
Install New Application
Specify options for installing enterprise applications and modules.
Step 1: Select installation options
Step 2 Map modules to servers
Step 3 Provide options to perform the EJB Deploy
Step 4 Map shared libraries
Step 5 Map shared library relationships
Step 6 Provide JNDI names for beans
Step 7 Map security roles to users or groups
Step 8 Ensure all unprotected 2.x methods have the correct level of protection
Step 9 Display module build Ids
Step 10 Summary
Select installation options
Specify the various options that are available for your application.
Precompile JavaServer Pages files
Directory to install application
Distribute application
Use Binary Configuration
Deploy enterprise beans
Application name
ServiceEndPointRI_jar
Create MBeans for resources
Override class reloading settings for Web and EJB modules
Reload interval in seconds
Deploy Web services
Validate Input off/warn/fail
warn
Process embedded configuration
File Permission
Allow all files to be read but not written to
Allow executables to execute
Allow HTML and image files to be read by everyone
.*\.dll=755#.*\.so=755#.*\.a=755#.*\.sl=755
Application Build ID
Unknown
Allow dispatching includes to remote resources
Allow servicing includes from remote resources
Business level application name
Create New BLA
Asynchronous Request Dispatch Type
Disabled
Allow EJB reference targets to resolve automatically
Deploy client modules
Client deployment mode
Isolated
Validate schema
Next
Cancel

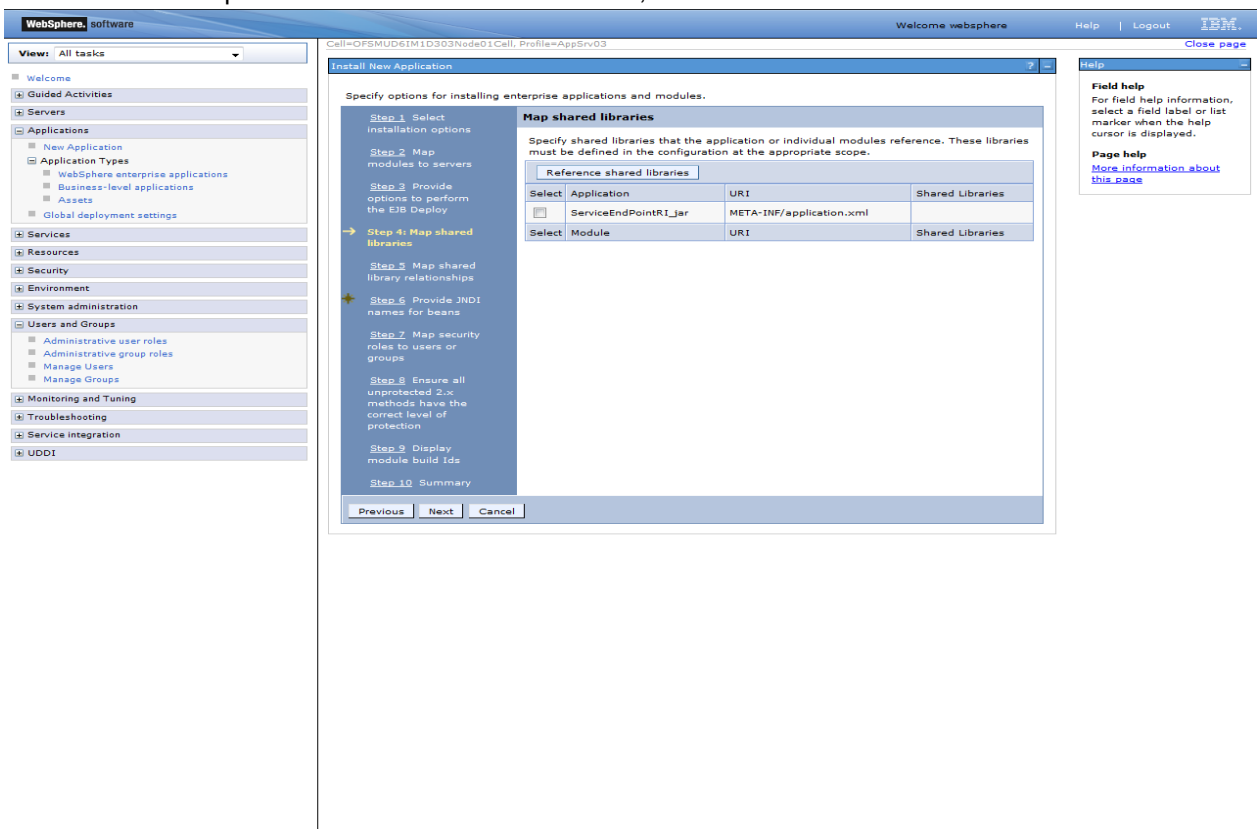
28) In screen “Map modules to server” as shown in figure below, map the modules to the applicable server(s) and click “Next”.



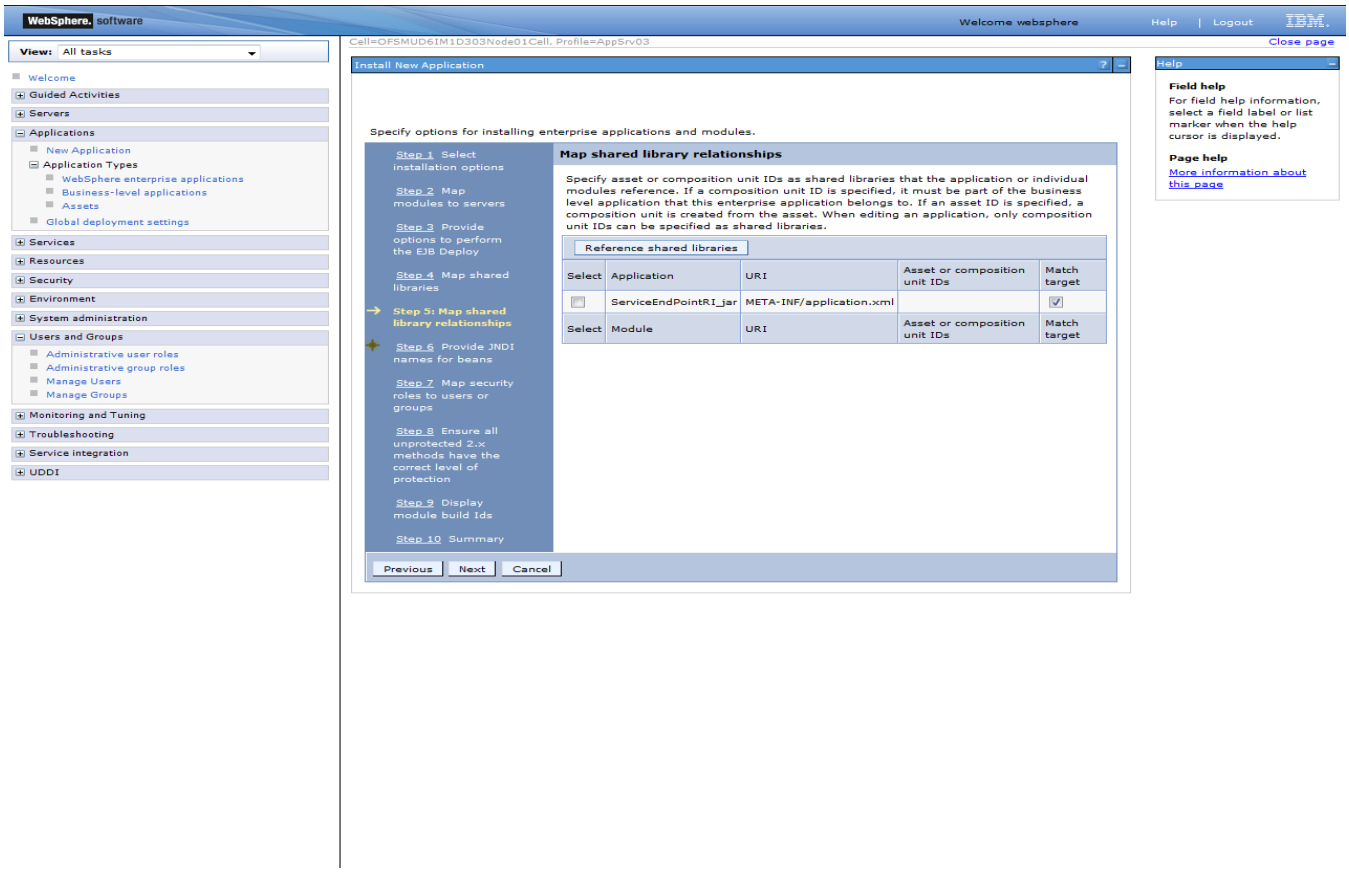
29) On screen “Provide options” as shown in figure below, add the FCDB Kernel jar to the classpath. Add <FCDB BASE DIR>/system/build/kernel/FCDB_<version>.jar to the class path field and click “Next”.



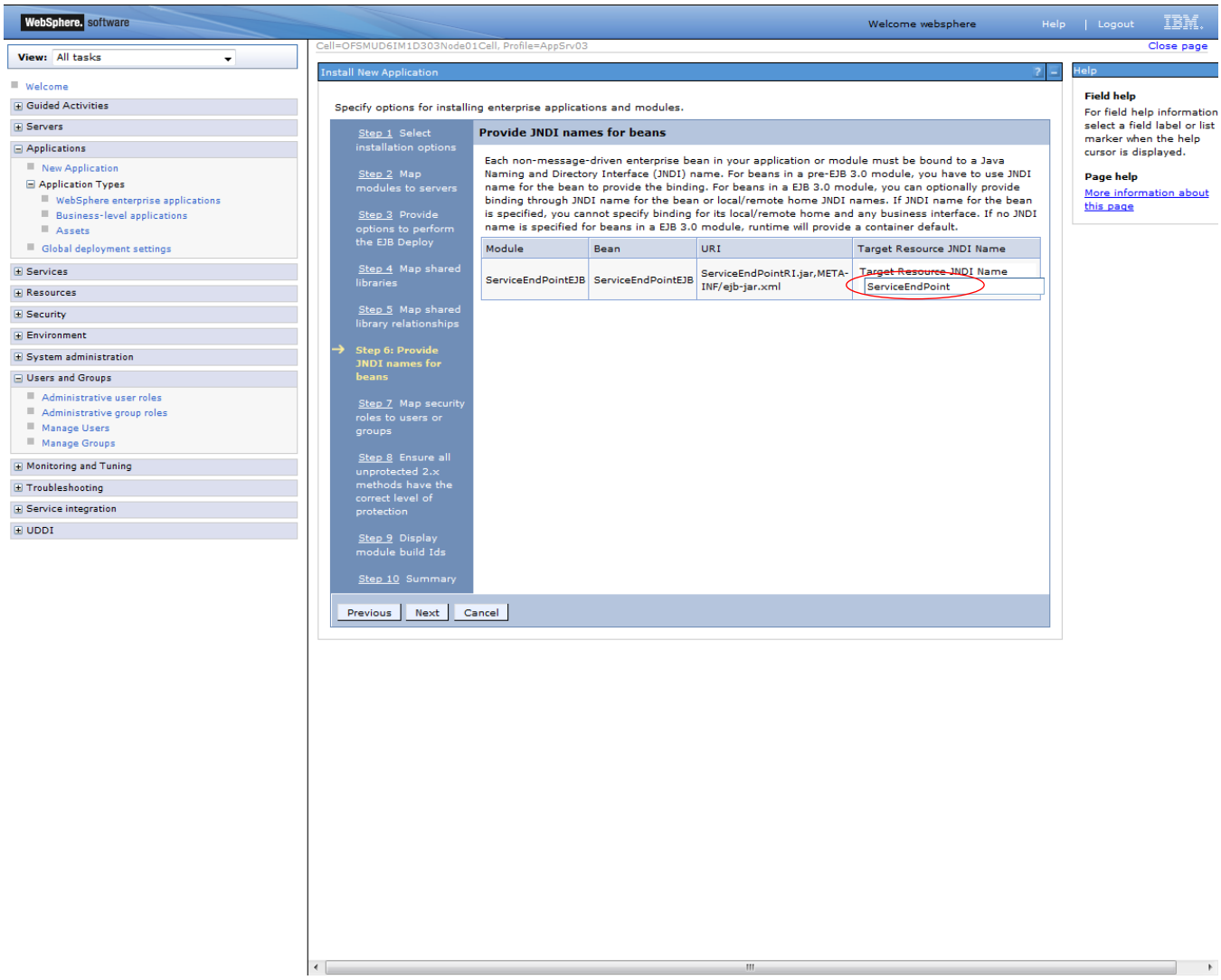
30) On the screen “Map Shared libraries” as shown below, click “Next”



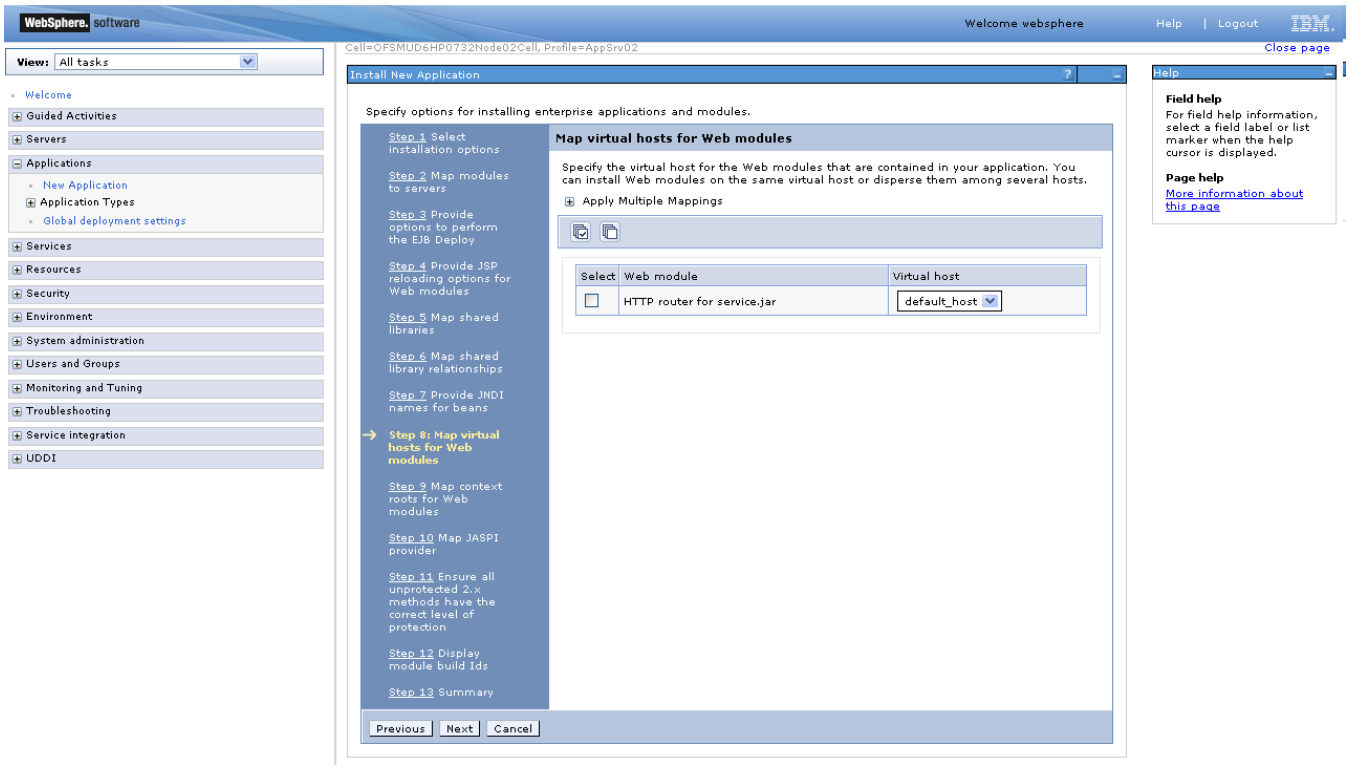
31) On “Map shared library relationships” click next.



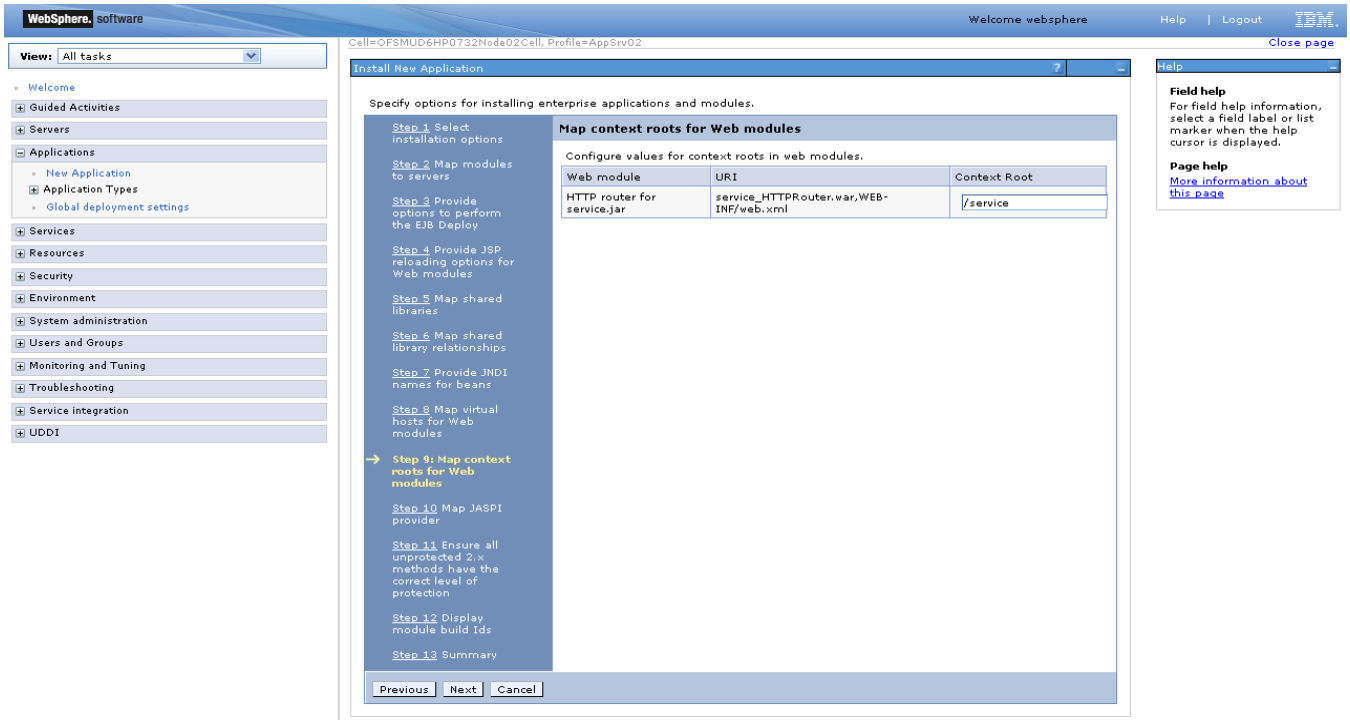
32) On the Screen “Provide JNDI names” as show below, provide the JNDI name for the application in field “Target Resource JNDI Name”. The JNDI name for each FCDB application has been provided at the start of this section.



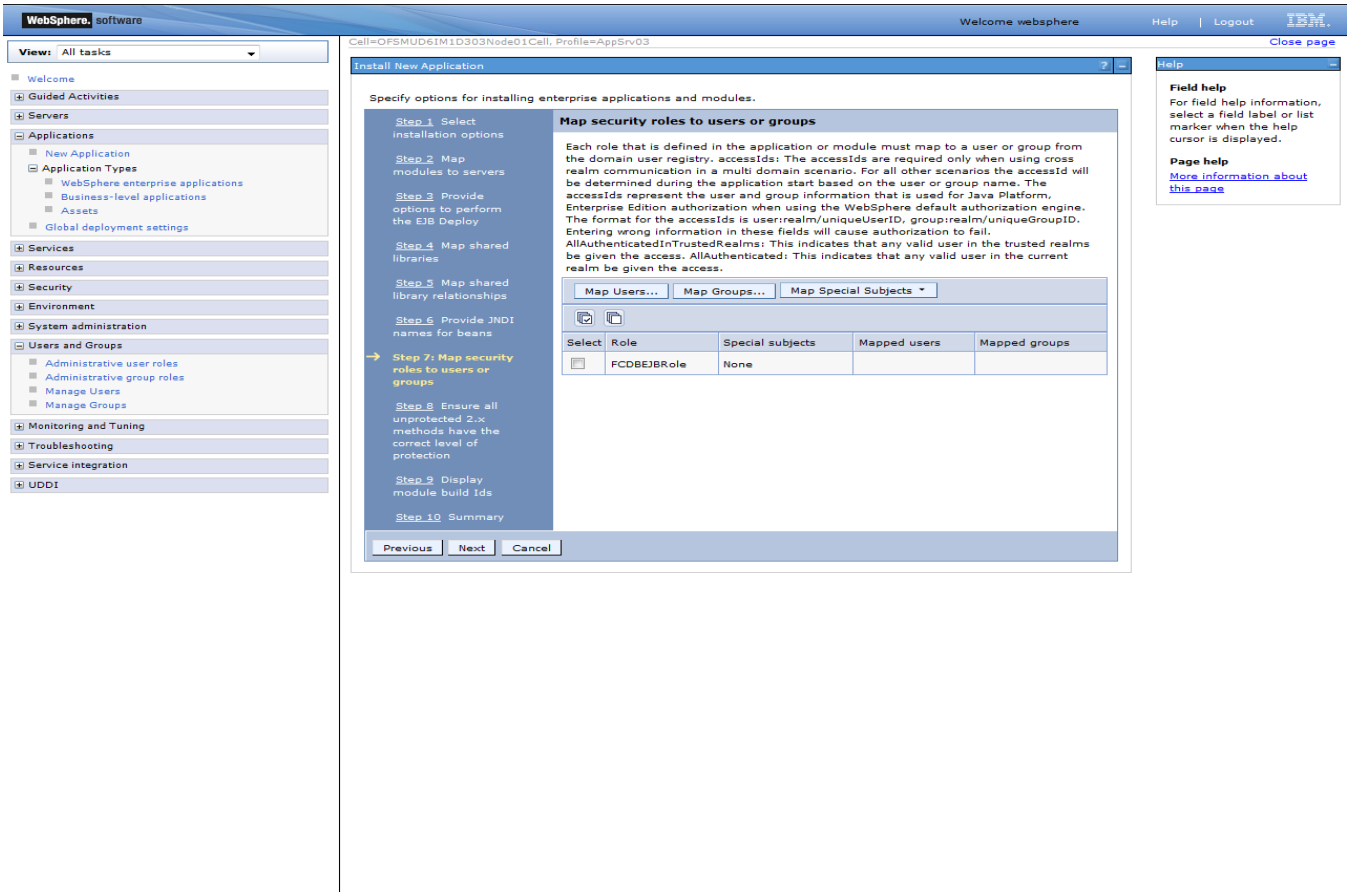
33) On the screen “Map Virtual host”, map web modules to the relevant virtual host & click Next.



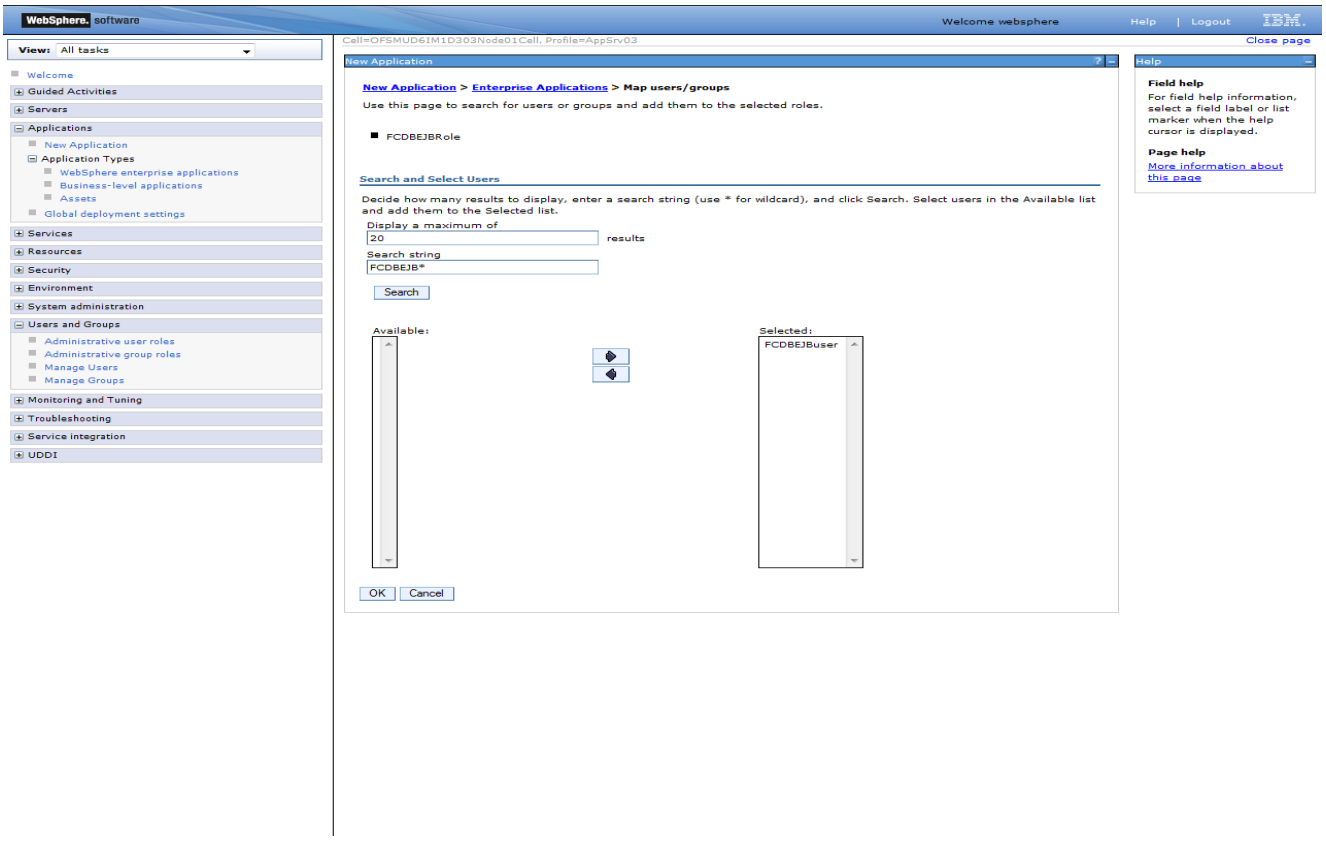
34) On screen “Map Context roots”, map the web modules with respective context root. The context root has been configured in the deployable and the default context root being displayed can be used. If no context root is displayed, use the context root as mentioned for the application at the start of this section.



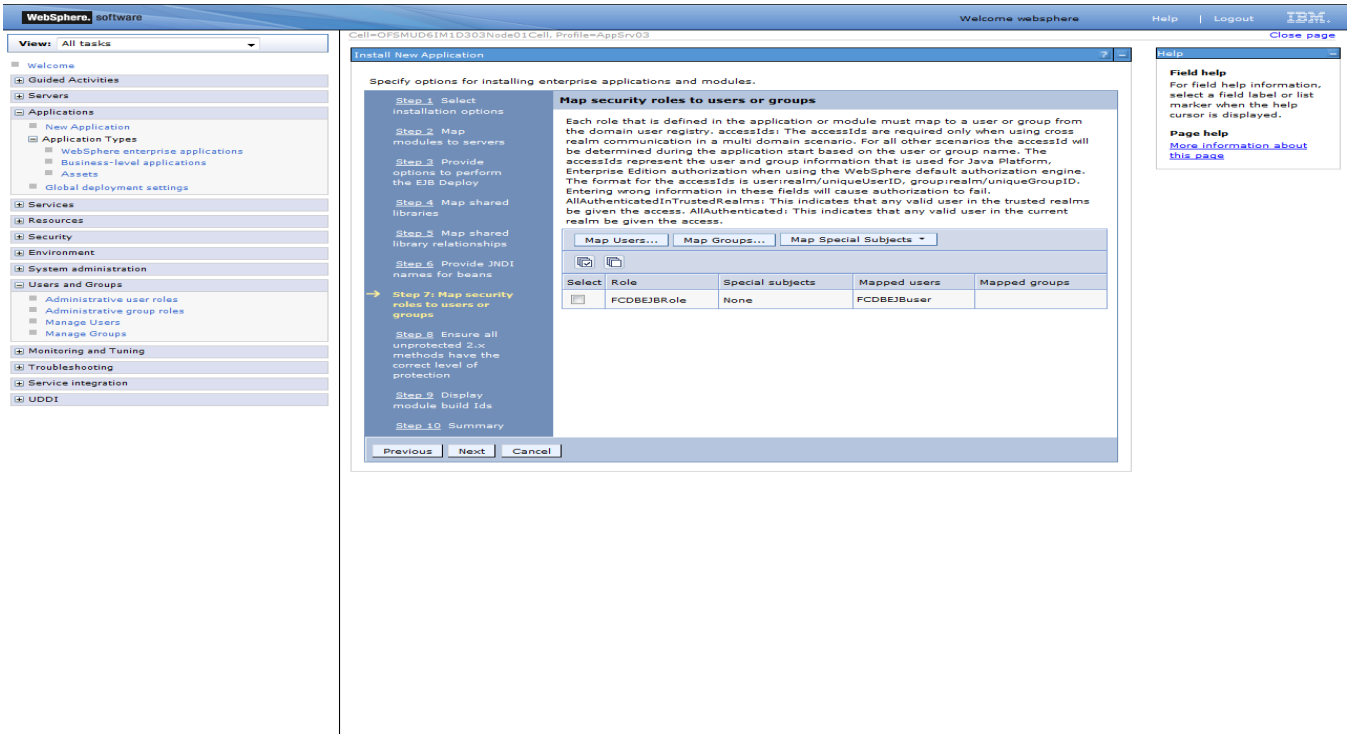
35) On screen “Map security roles to users or groups”, map the role “FCDBEJBRole” and map it with the user created before deploying the ear.



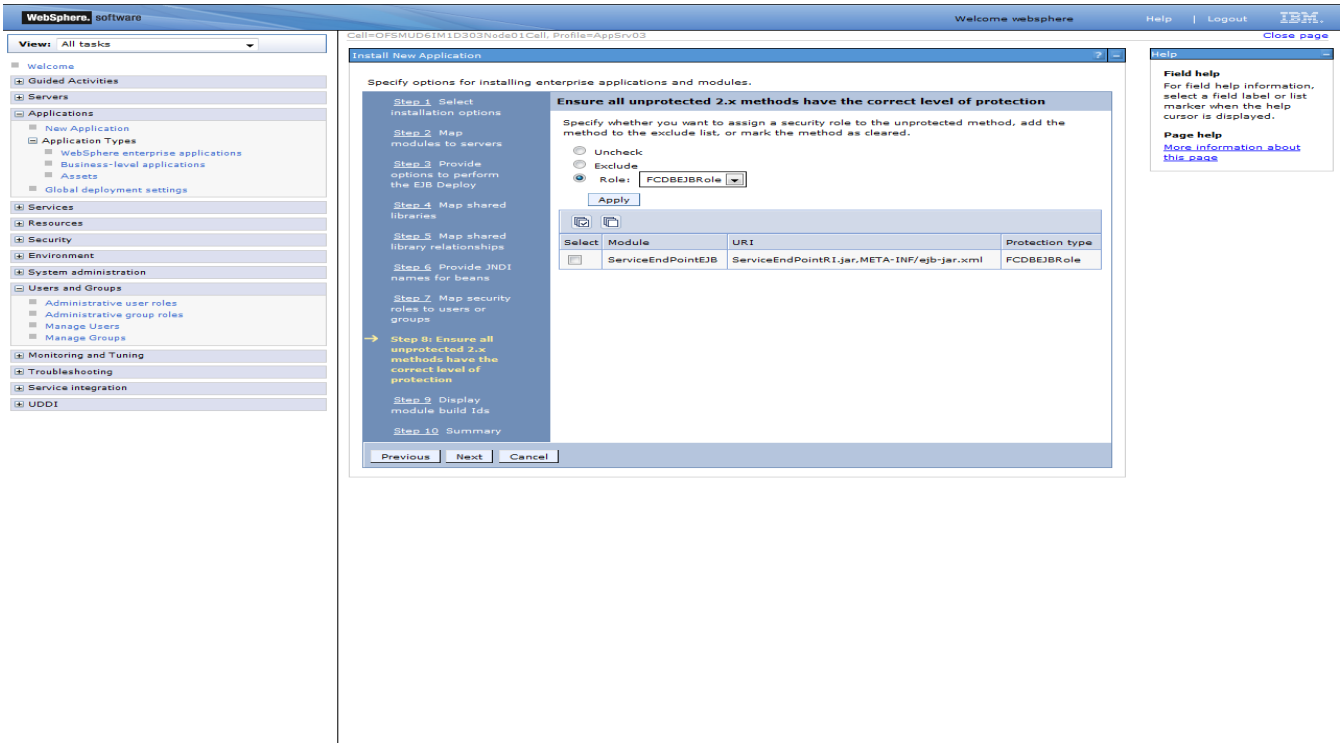
36) On the screen as show below, click “Ok”.



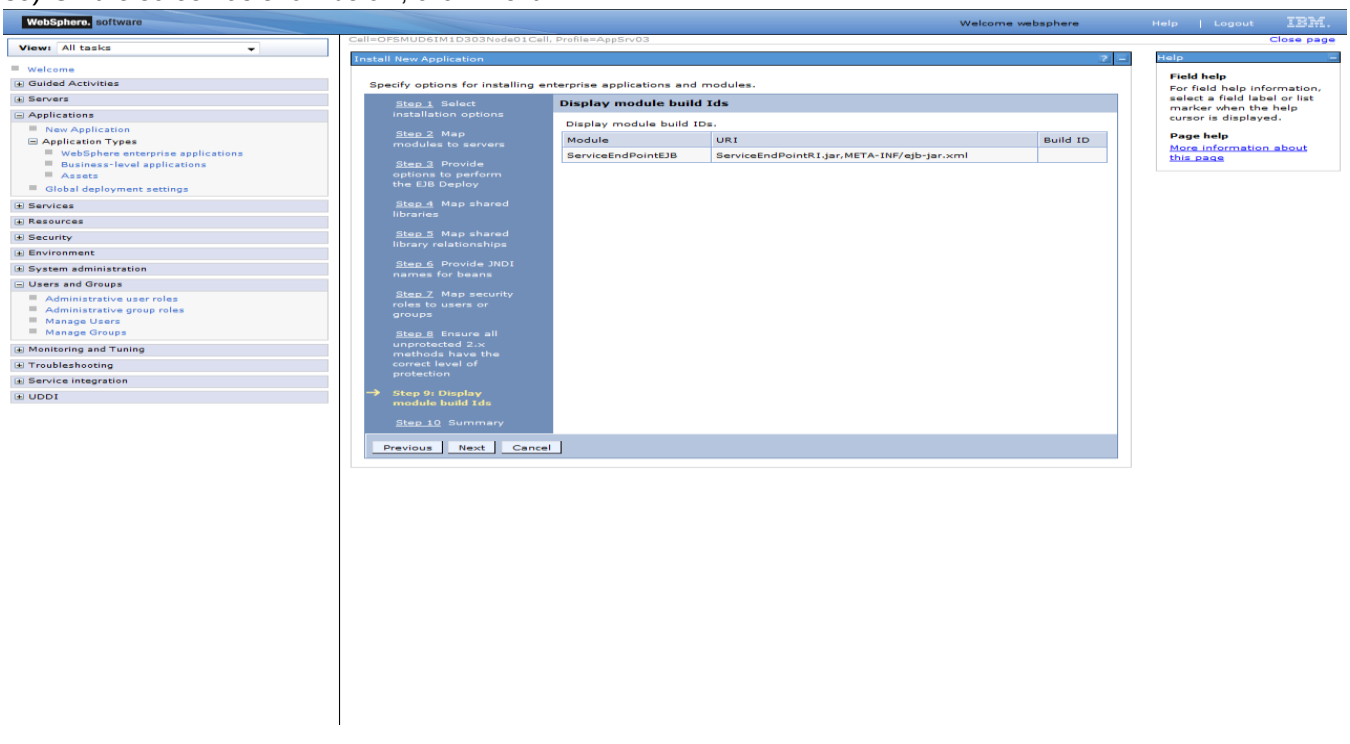
37) On the screen as show below, click “Next”.



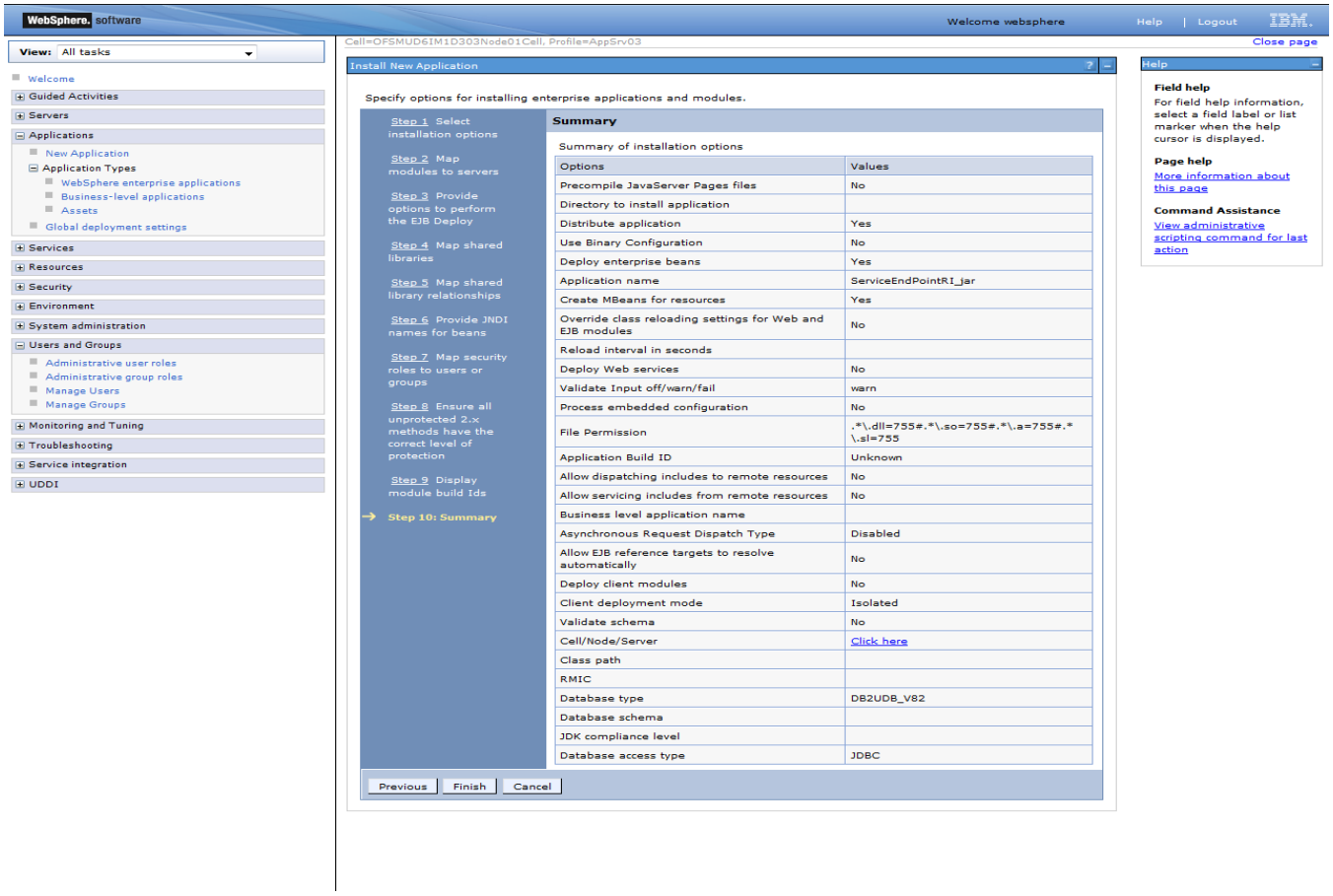
38) On the screen as show below, select the role and associate it with the module by clicking on “Apply”. Now click on “Next”



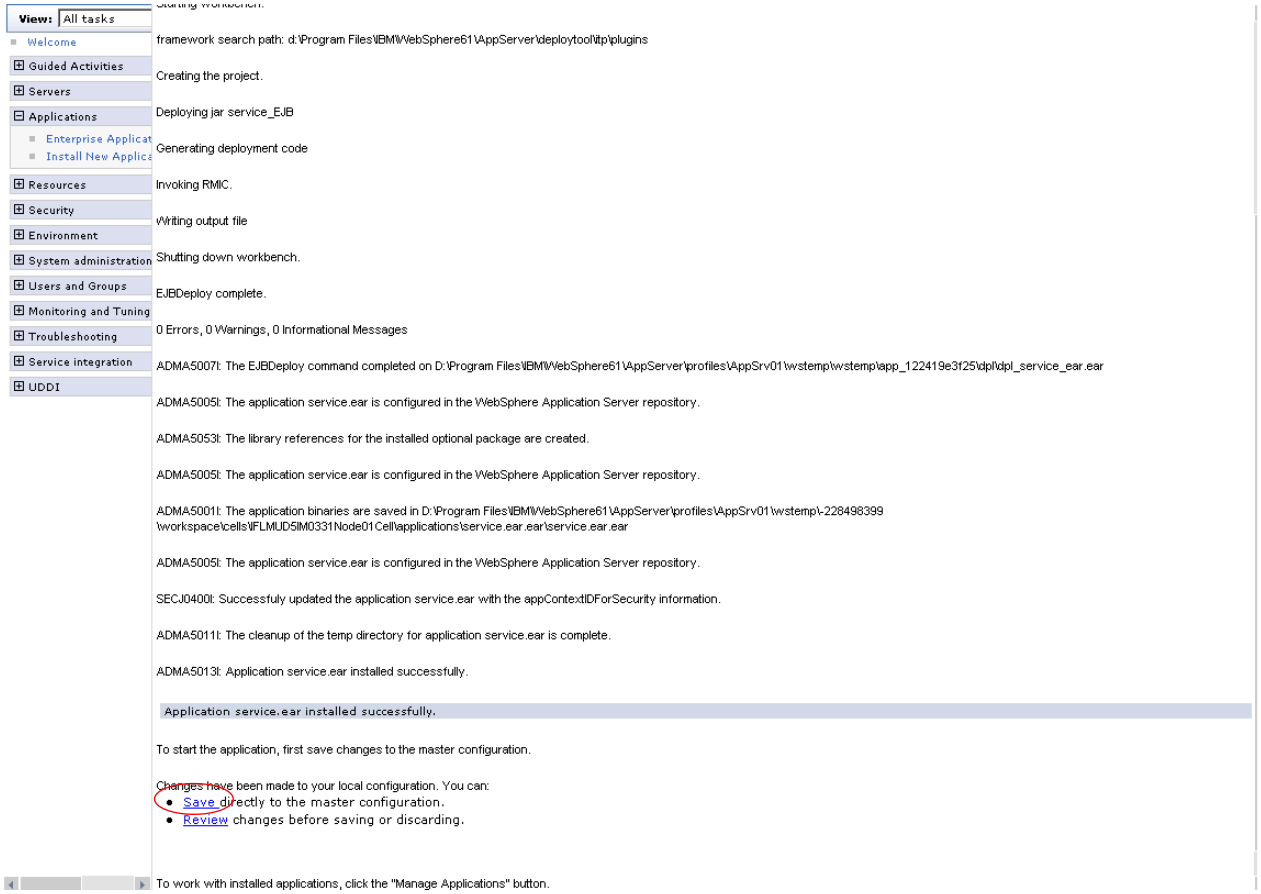
39) On the screen as show below, click “Next”.



40) On the “Summary” page, summary of selected installation options will be displayed. Verify the same and click “Next” to begin with application installation.



41) The next screen displays application installation status. On successful installation, click the link “Save” to save the installation.



1) In fcat-config.xml set the following properties with username and password of the user created in above steps and restart the server.

Field	Value
FCAT.CONNECT.SVC.USERNAME	<Enter username>
FCAT.CONNECT.SVC.PASSWORD	<Enter password>

7. Setup JDBC Datasource and Connection Pooling

7.1 Create Connection Pools

This activity will need to be repeated for all datasources (A1, B1A1 and B1AP).

The steps for both will be identical except where explicitly mentioned otherwise.

To create JDBC providers:

1) Admin Console: Click Resources → JDBC → JDBC Provider.
2) Select the scope for the new JDBC Provider to be created.

   Select “Node=<node name>, Server=<server name>”

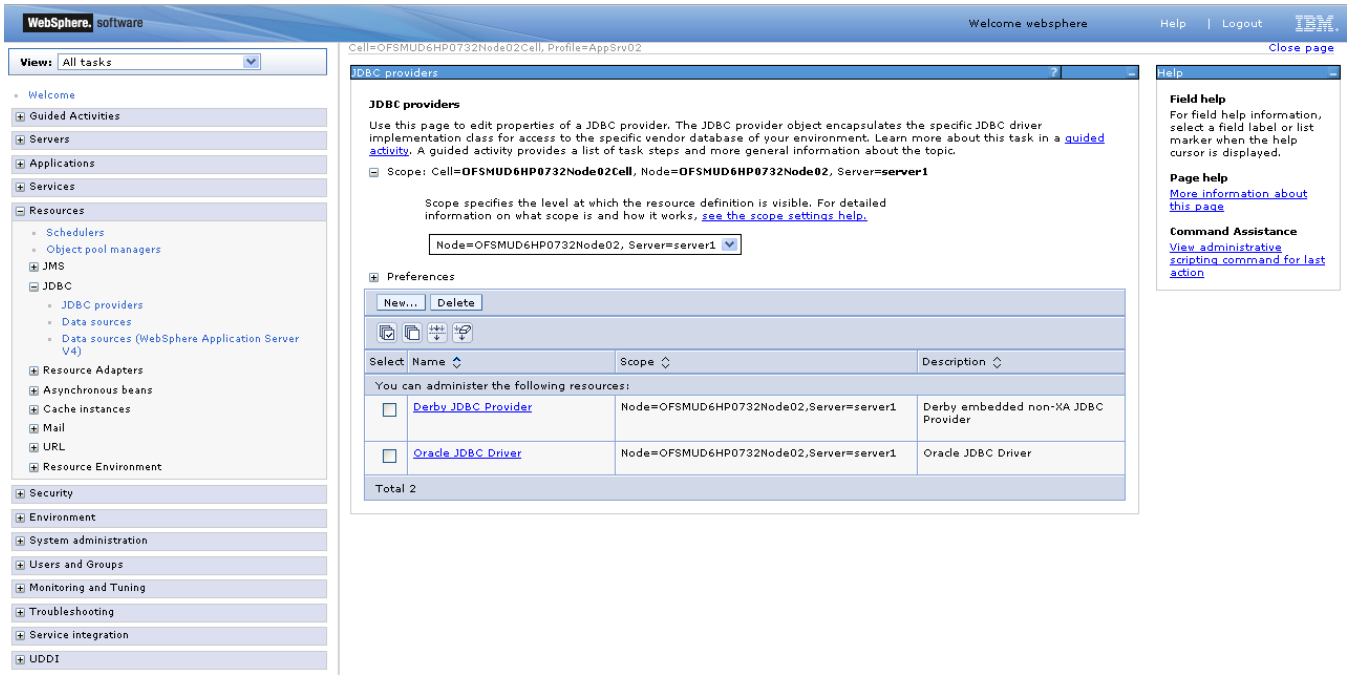


Figure 7.1

3) Click New.

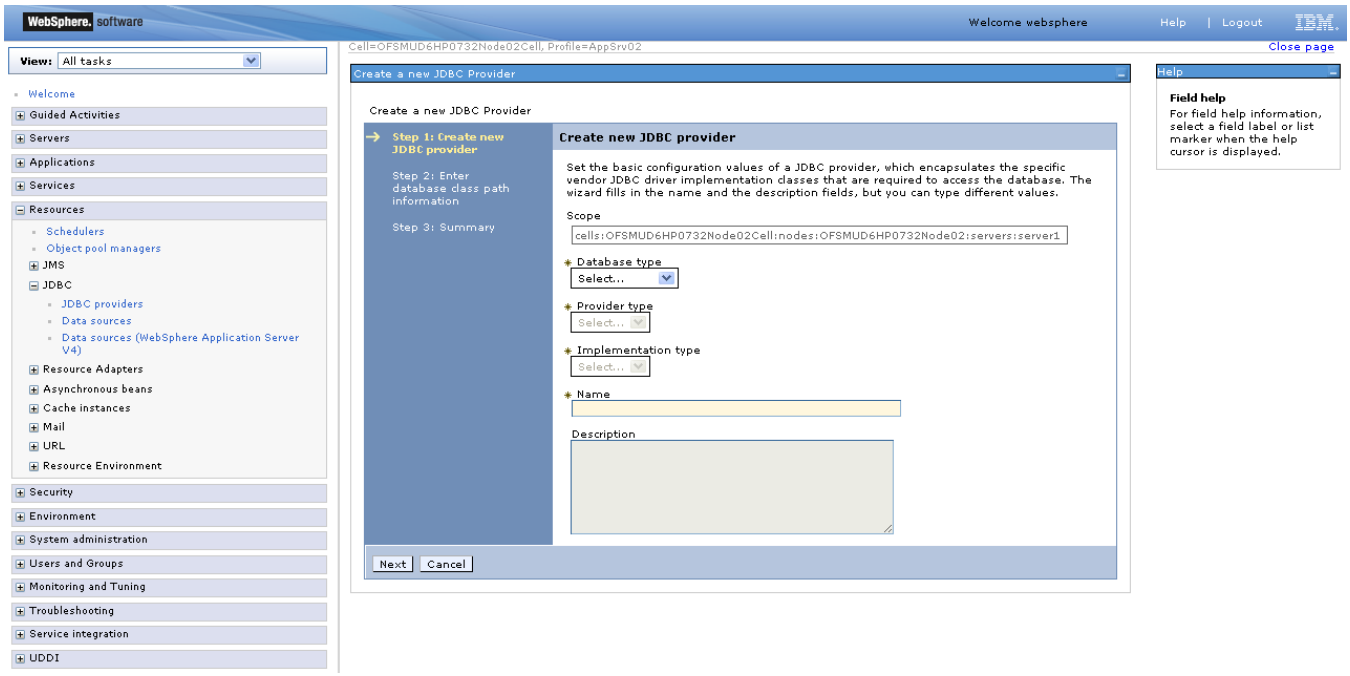


Figure 7.2

4) Select database type as Oracle
5) Select the provider type as Oracle JDBC Driver
6) Select the implementation type as Connection pool data source

   (During BT datasource creation, select XA datasource)

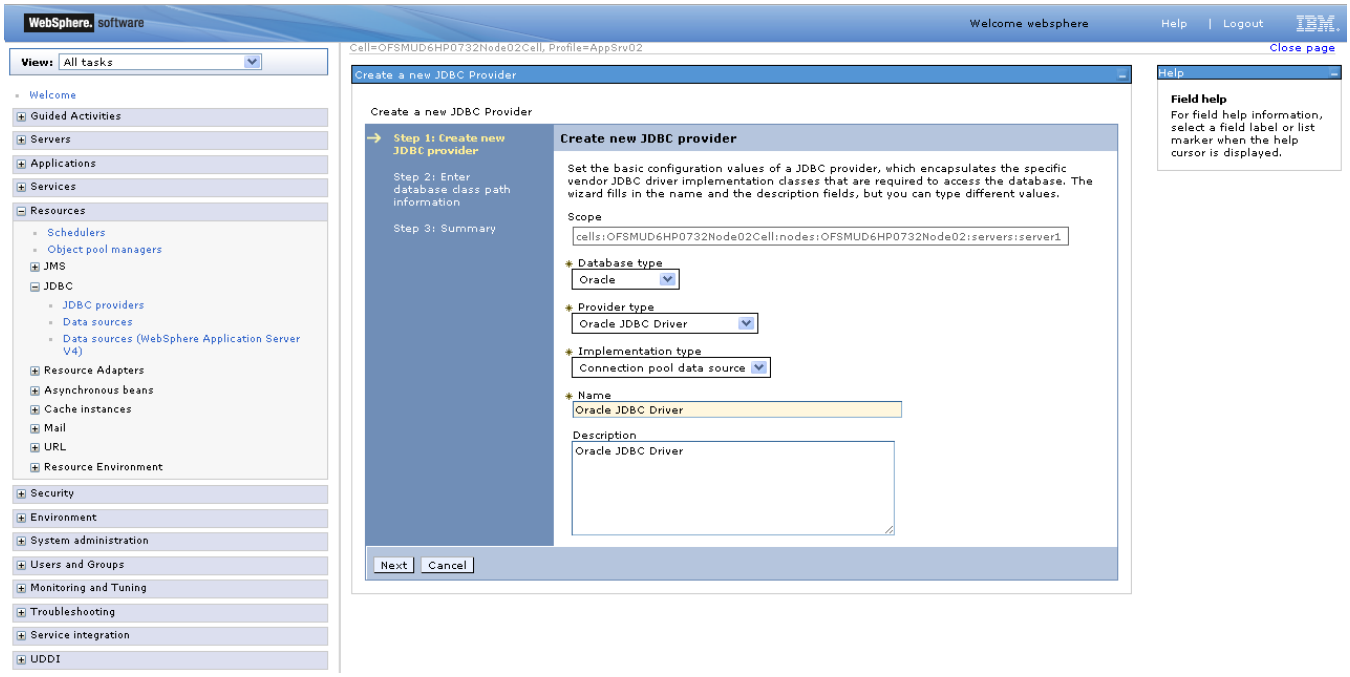


Figure 7.3

7) Click Next.
8) Enter the directory location as <FCDB BASE DIR>\system\build\extclasses\jars\oracle

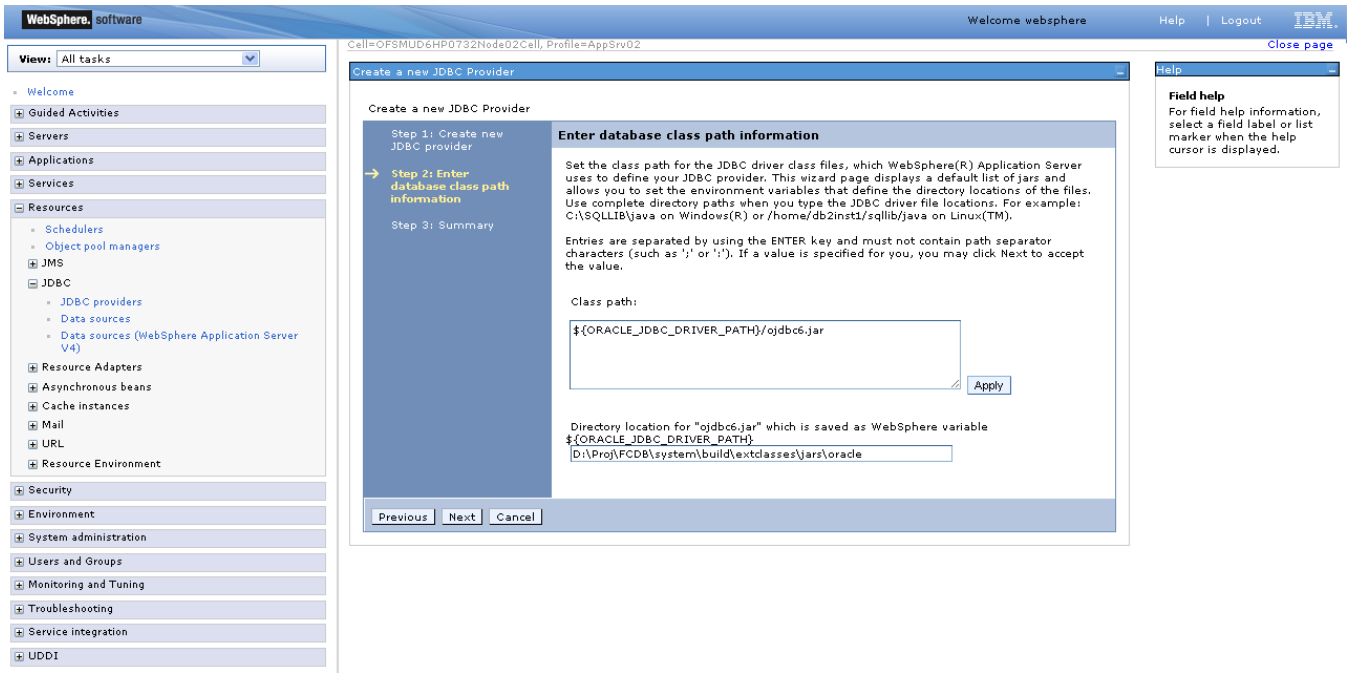


Figure 7.4

Note: Make sure oracle thin client has been installed successfully and set directory location for "ojdbc14.jar"

9) Click Next

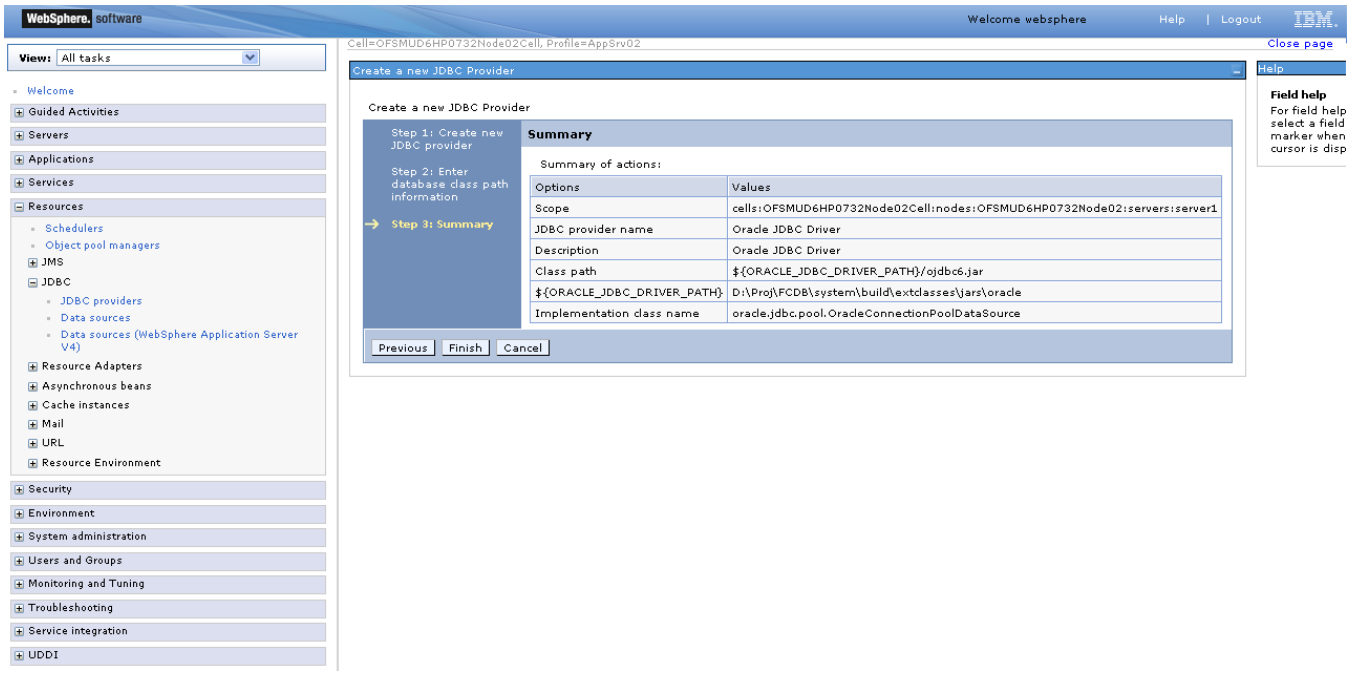


Figure 7.5

10) Click Finish

11) Click Save

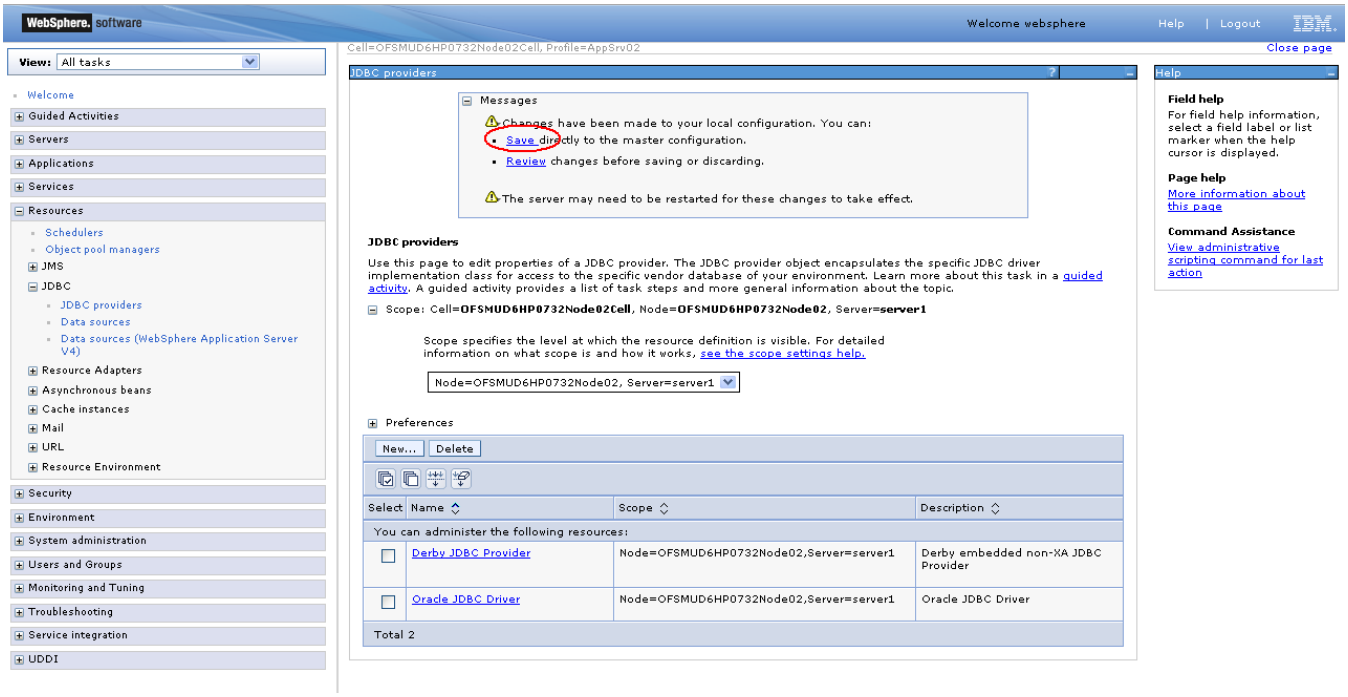


Figure 7.6

7.2 Creating Data Source

This activity will need to be repeated for all datasources (Application and EHMS)

e.g. A1,B1A1,B1AP etc.

The steps for both will be identical except where explicitly mentioned otherwise.

For creation of Data source:

1) Resources ->JDBC -> Data sources

   (For BT datasource Resources ->JDBC -> Data sources)

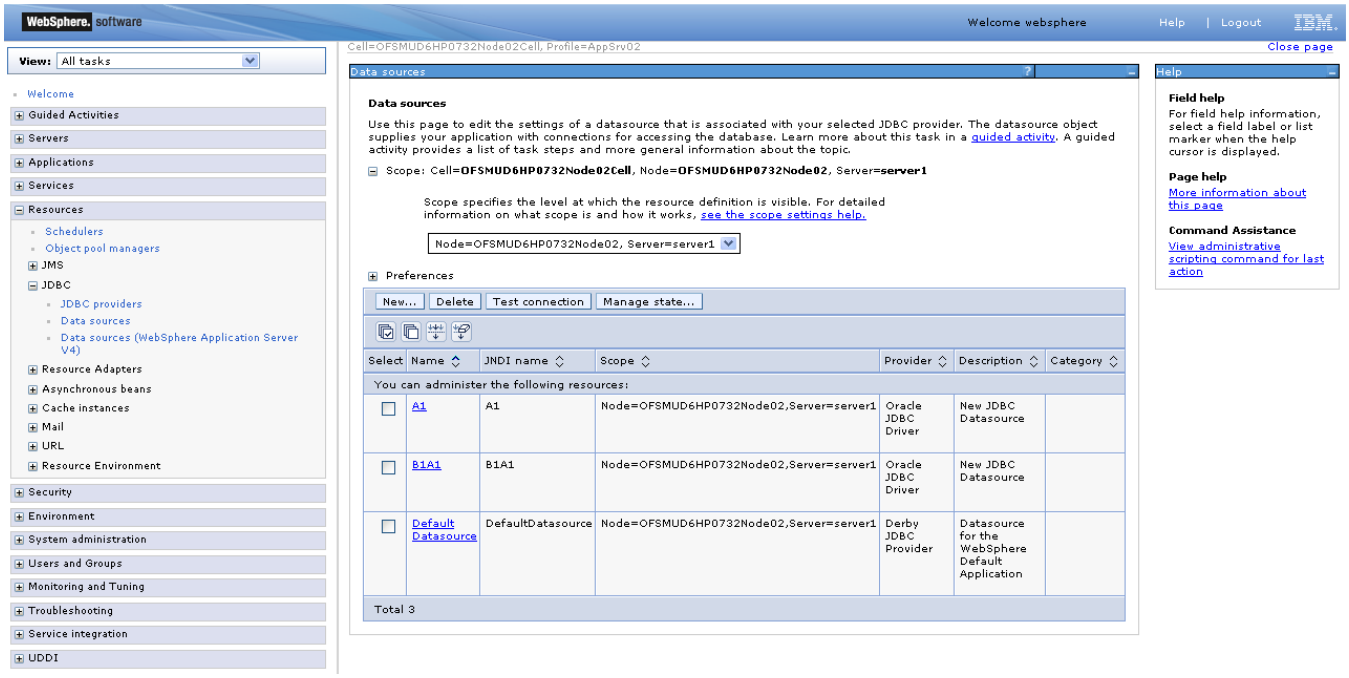


Figure 7.2.4

2) After click on New, On the below screen , put Data Source Name & JNDI name and Click on “NEXT”.

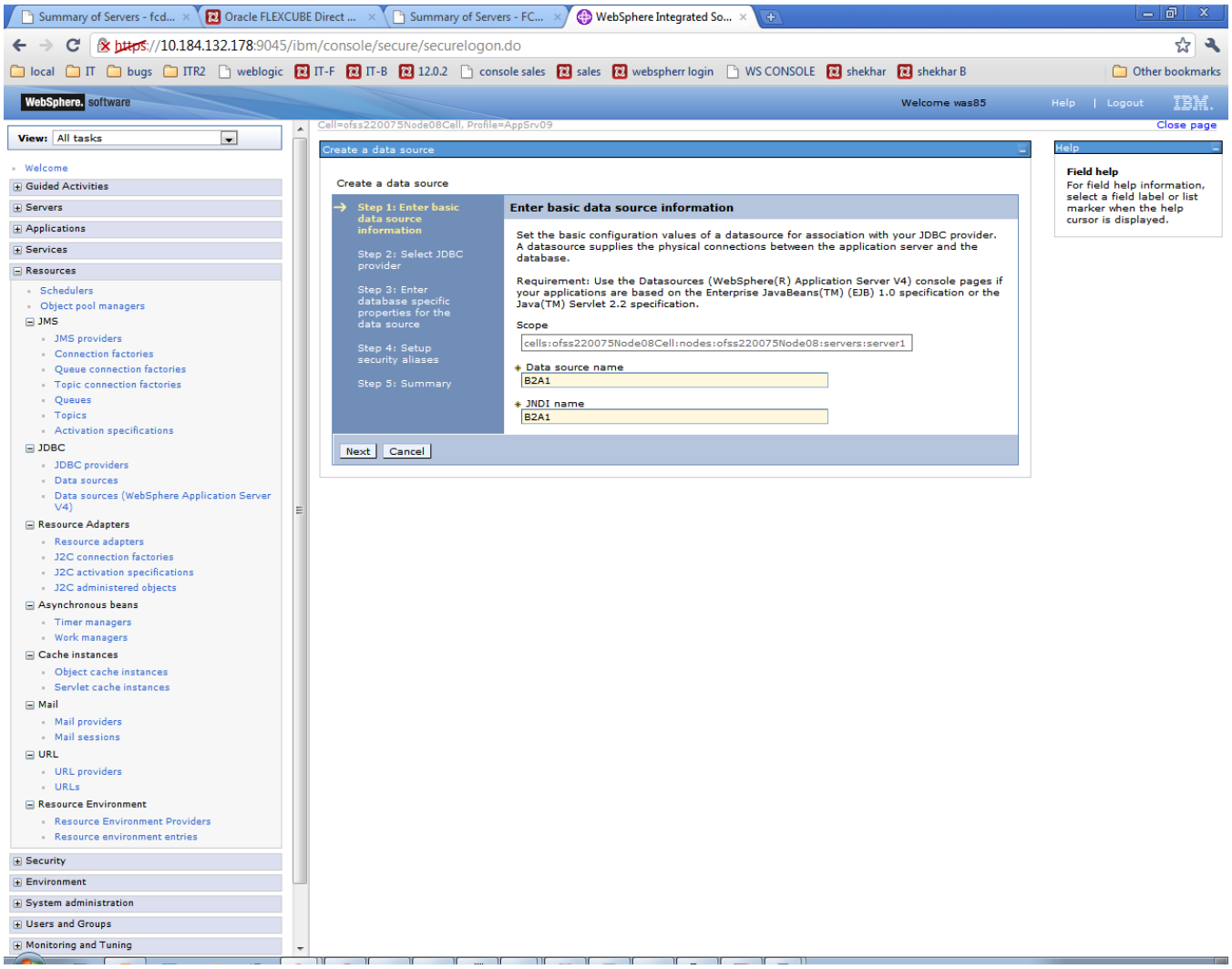
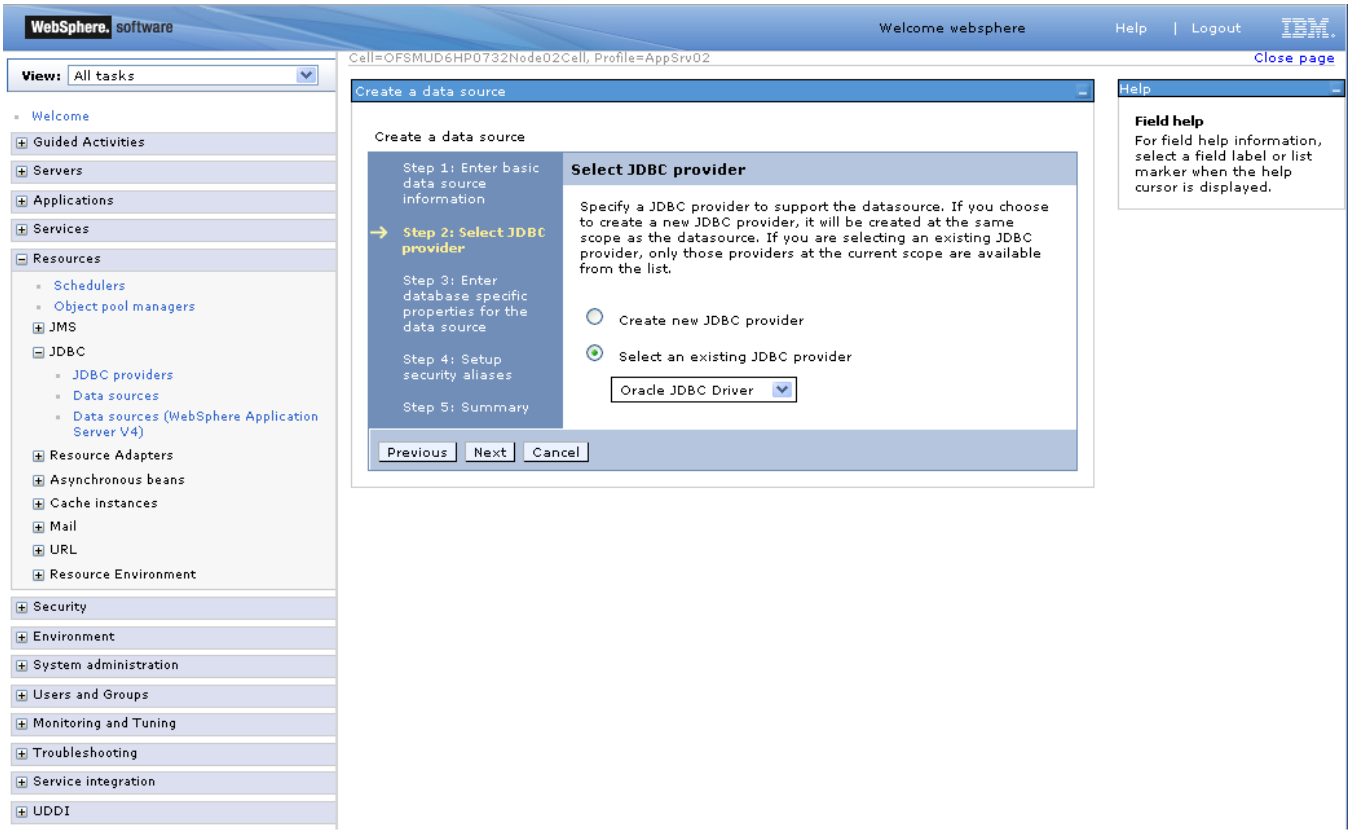


Figure 7.2.5

3) On the below screen , select radio “Select an existing JDBC provider” and select the value from drop down and click on NEXT..



4) Fill value in the URL and click on NEXT.

URL format: jdbc:oracle:thin:@<DATABASE HOST>:<port>:<SERVICE_NAME>

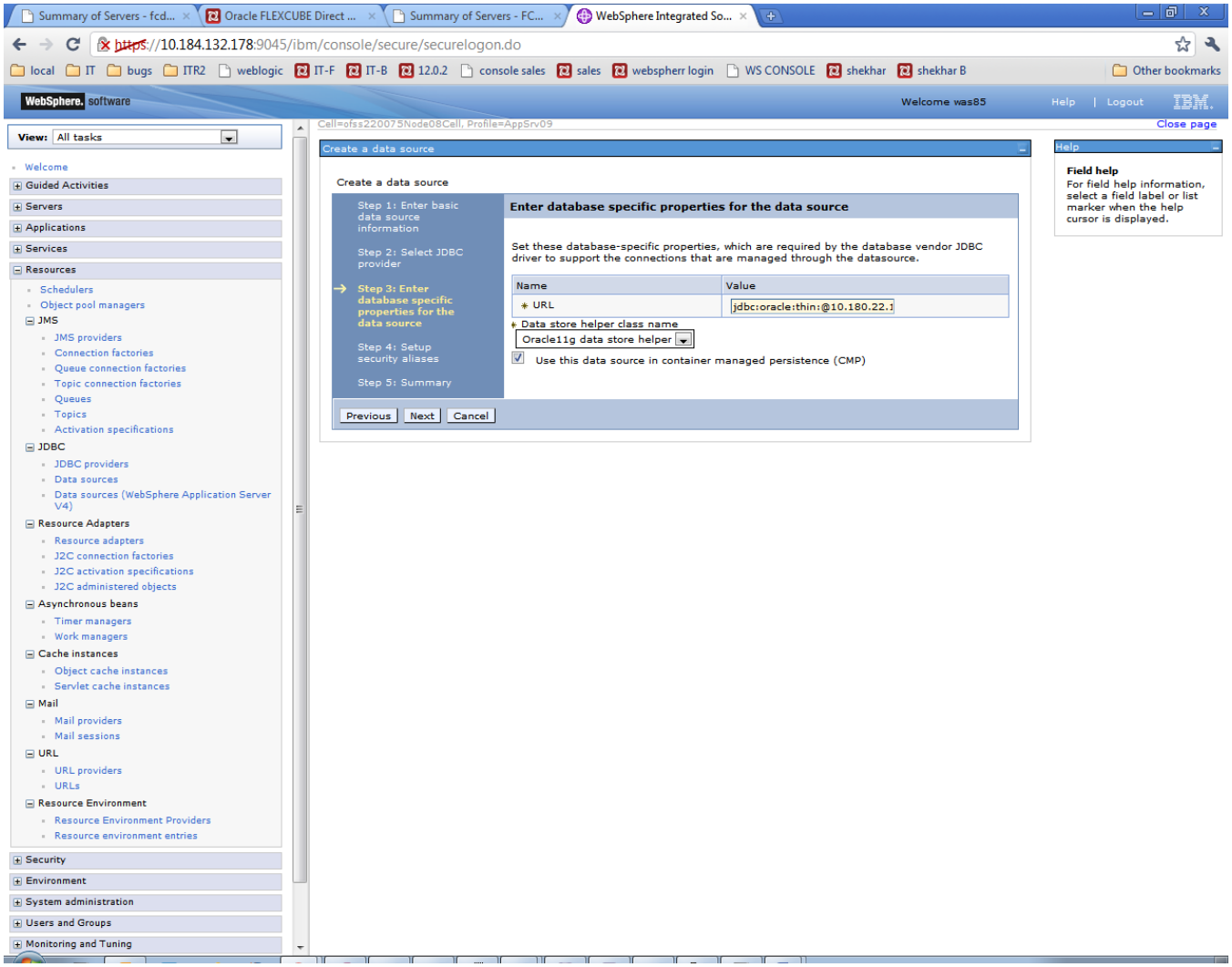


Figure 7.2.6

5) Setup Security aliases. Click ‘Next’

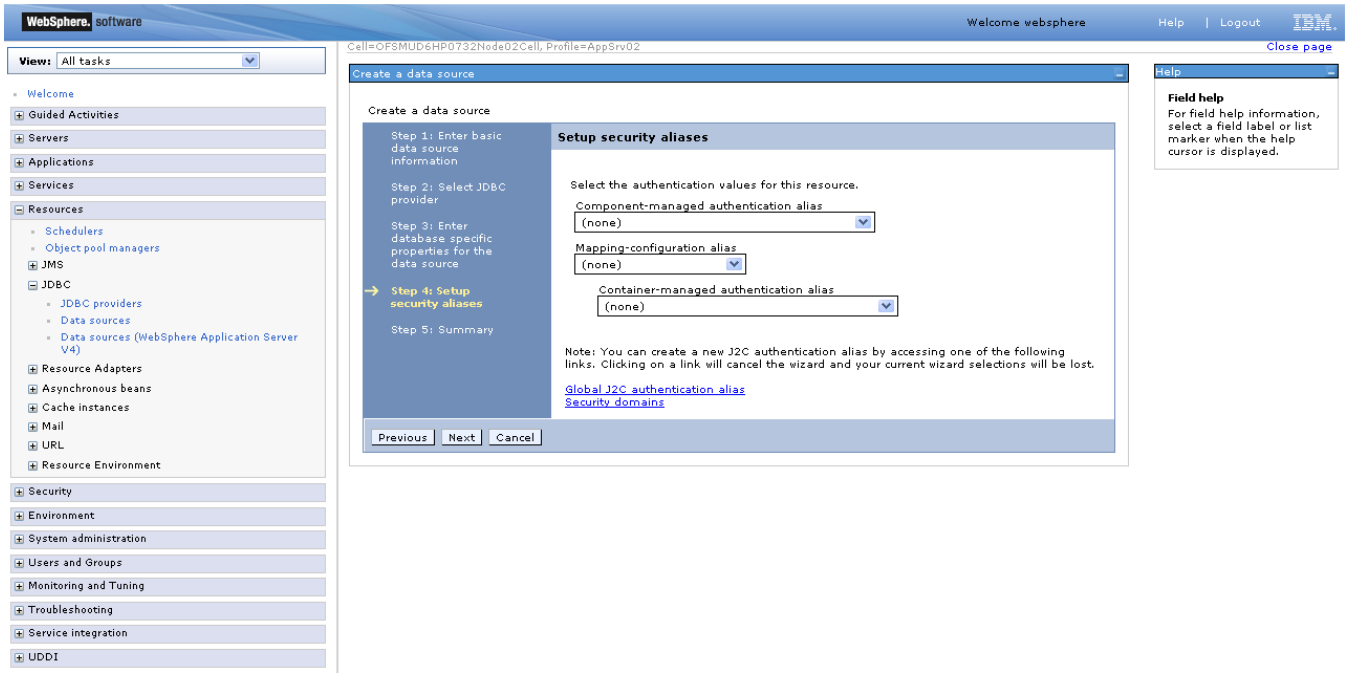


Figure 7.2.7

6) Review the Summary and click Finish

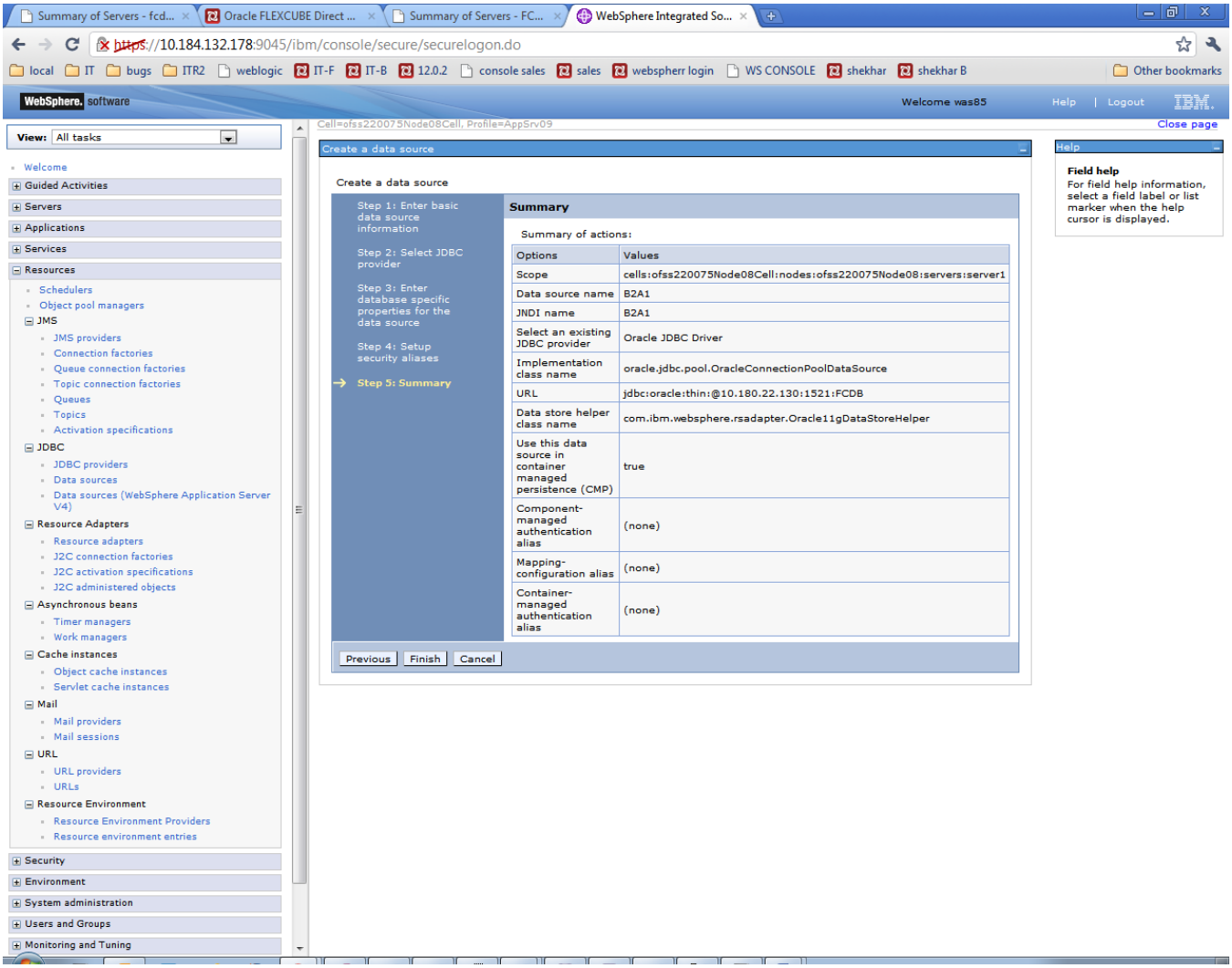


Figure 7.2.14

7) Click Save

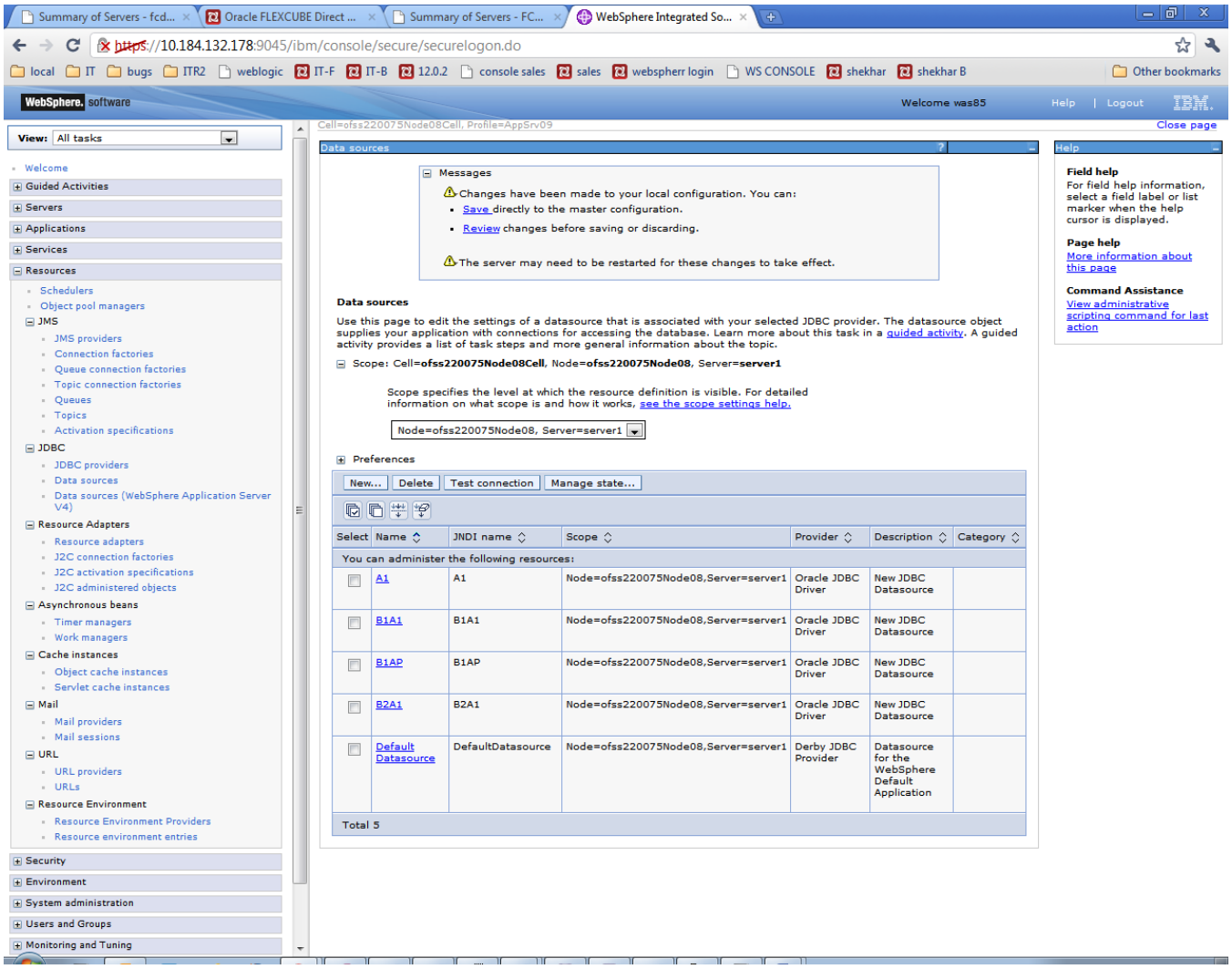


Figure 7.2.15

8) New DataSource will be created.

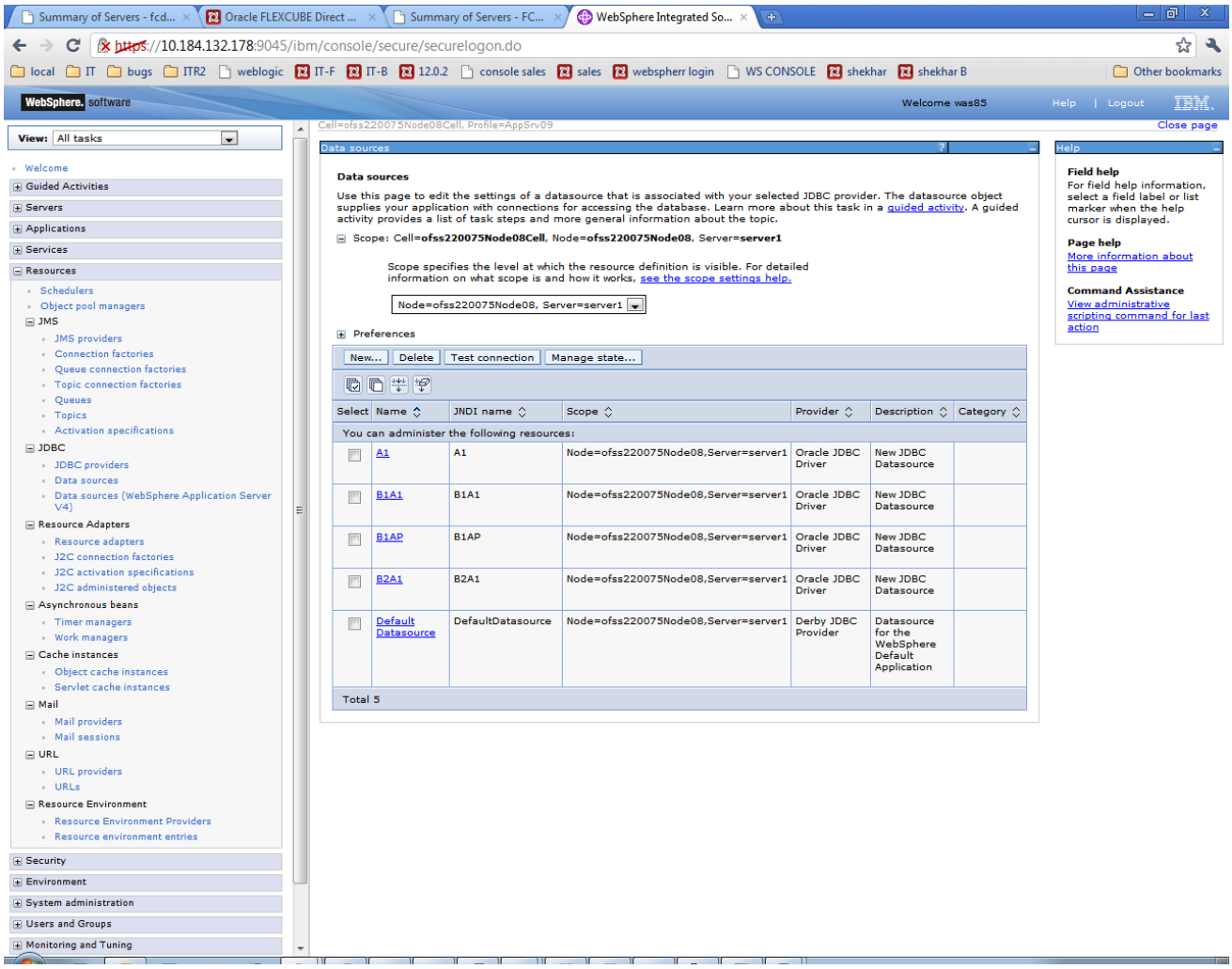
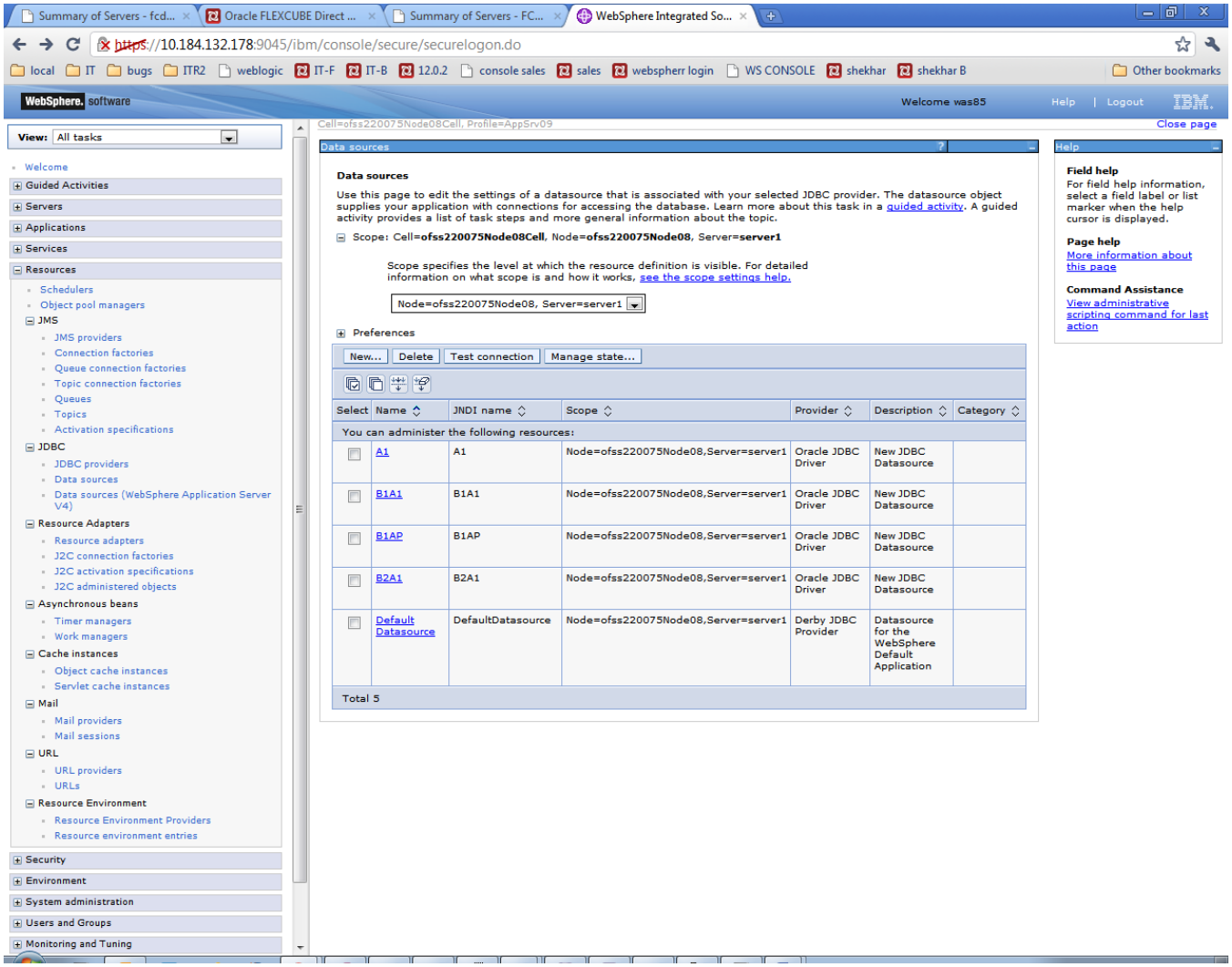


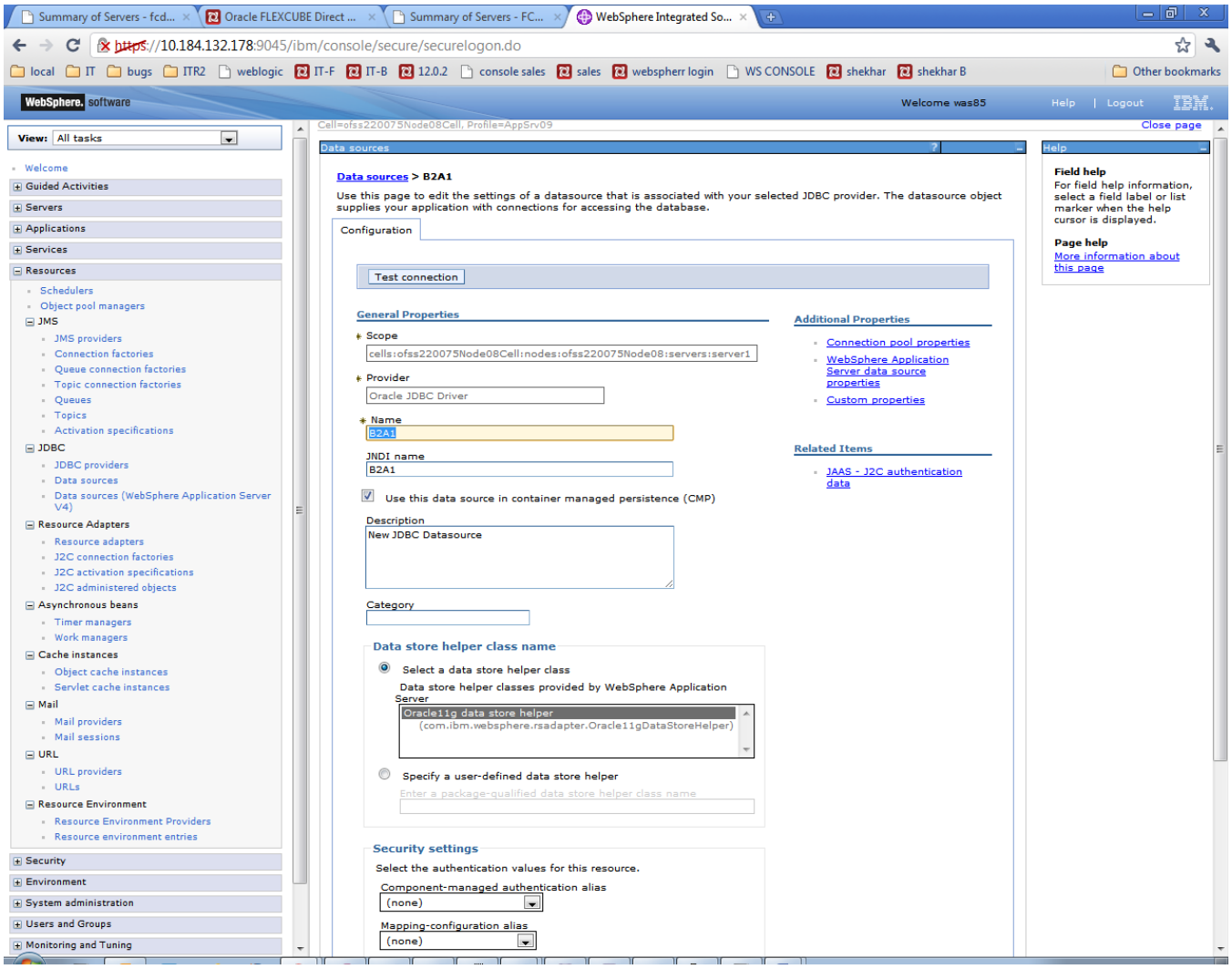
Figure 7.2.16

Steps to be followed for authentication of data.

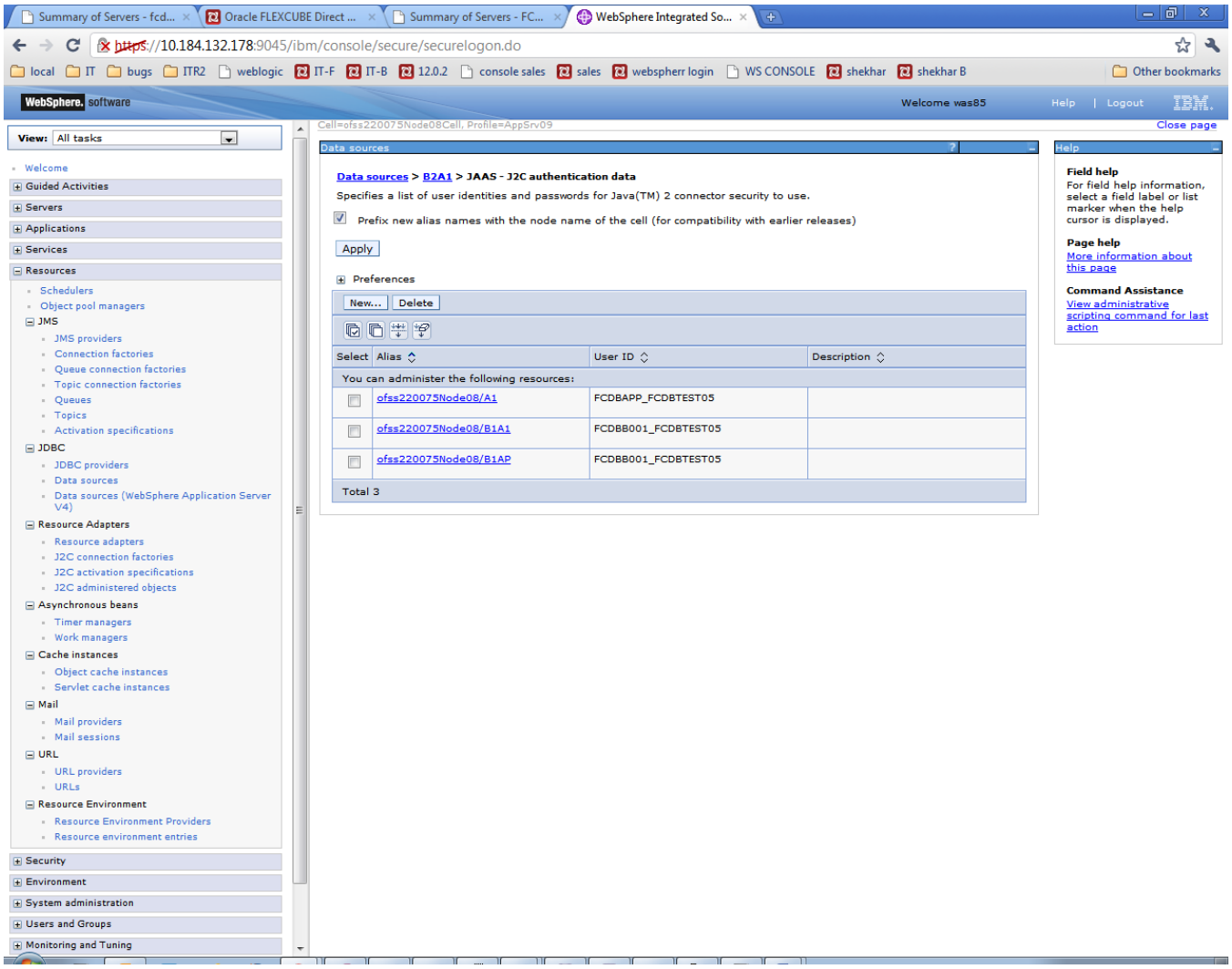
9) Click on newly created DataSource.



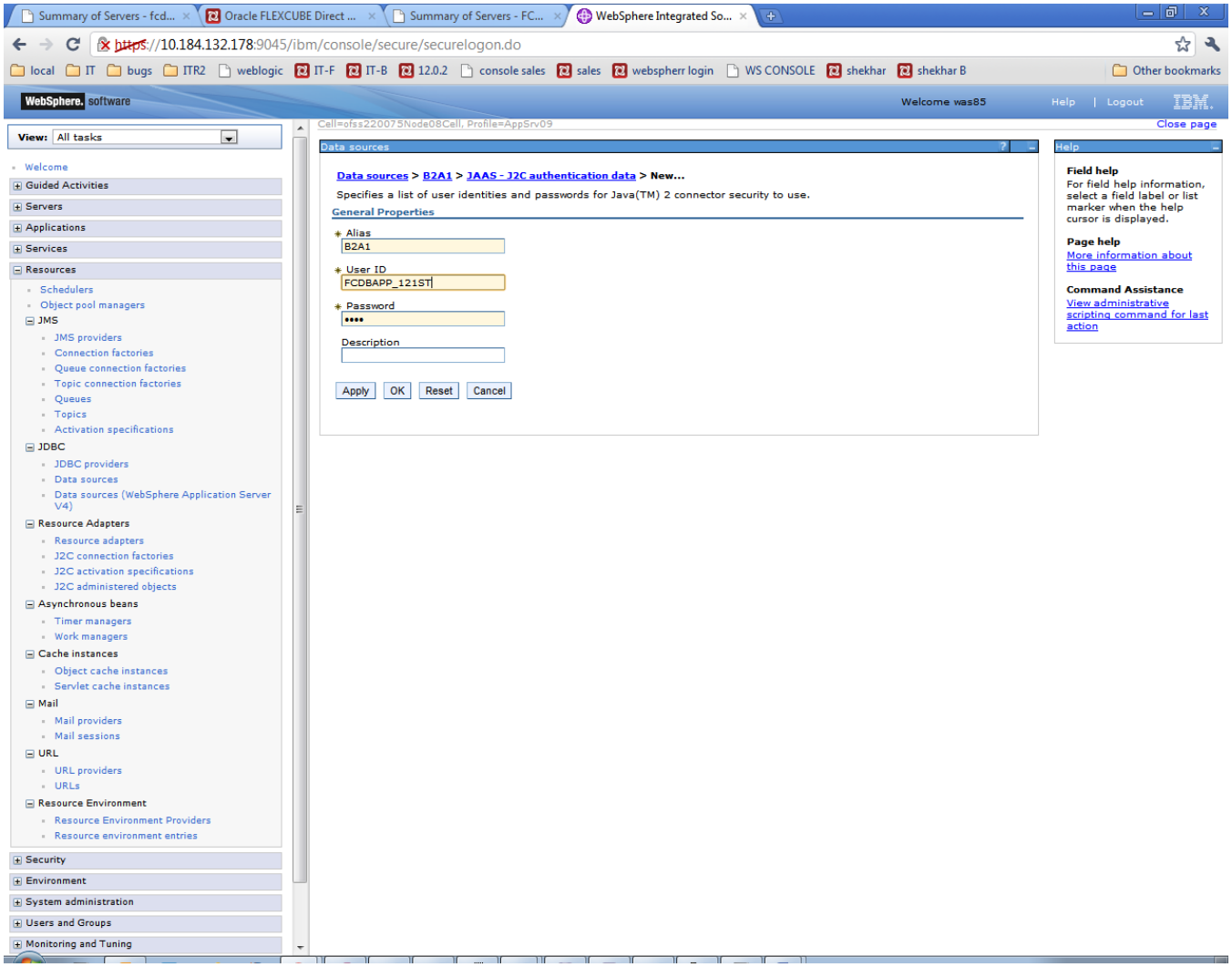
10) On configuration screen, click on ‘JAAS - J2C authentication data’ (under ‘Related Items’)



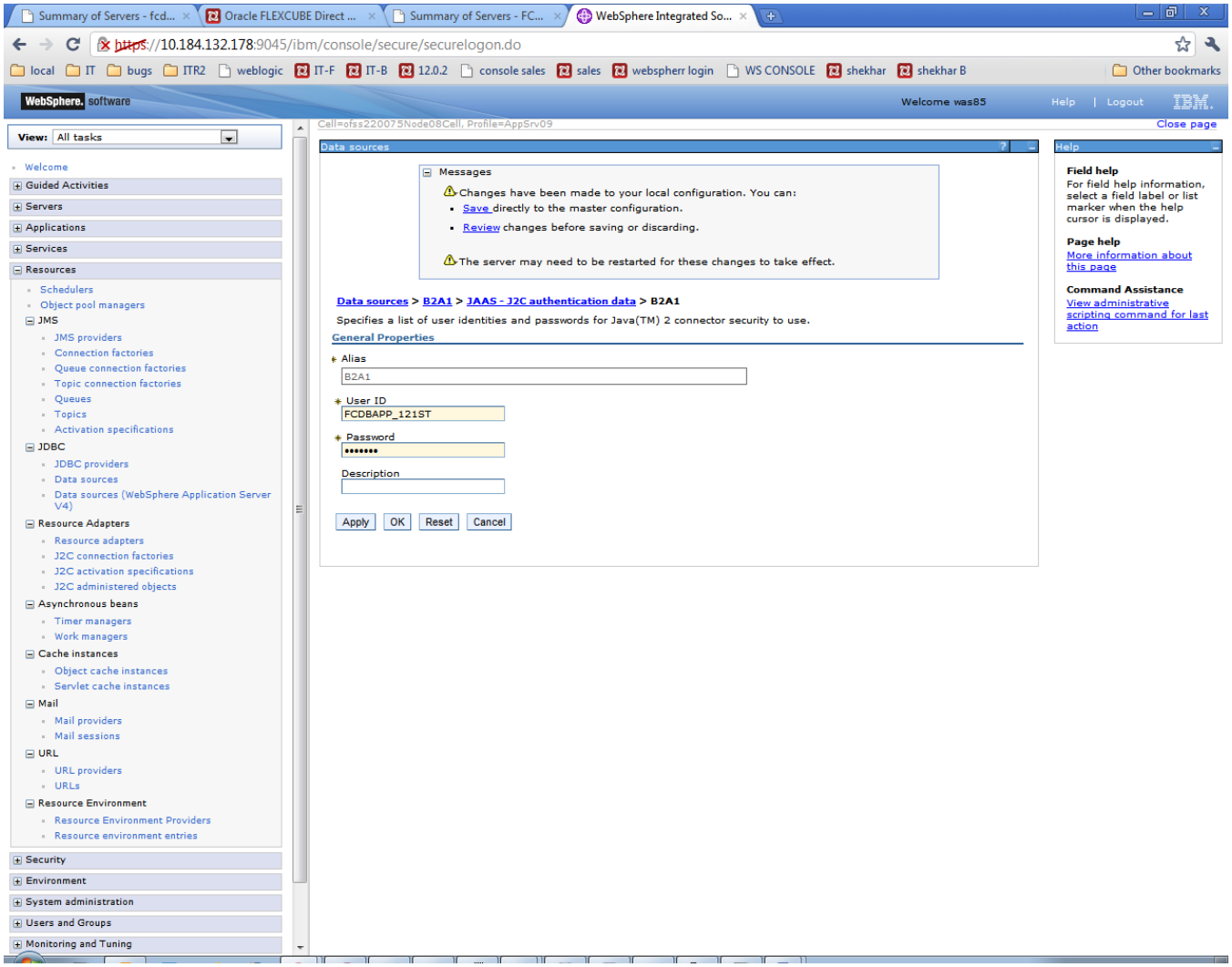
11) Click New


Summary of Servers - fcd...
Oracle FLEXCUBE Direct ...
Summary of Servers - FC...
WebSphere Integrated So...
https://10.184.132.178:9045/ibm/console/secure/securelogon.do
WebSphere. software
Welcome was85
Help | Logout
IBM.
Cell=ofss220075Node08Cell, Profile=AppSrv09
Close page
View: All tasks
Welcome
Guided Activities
Servers
Applications
Services
Resources
Schedulers
Object pool managers
JMS
JMS providers
Connection factories
Queue connection factories
Topic connection factories
Queues
Topics
Activation specifications
JDBC
JDBC providers
Data sources
Data sources (WebSphere Application Server V4)
Resource Adapters
Resource adapters
J2C connection factories
J2C activation specifications
J2C administered objects
Asynchronous beans
Timer managers
Work managers
Cache instances
Object cache instances
Servlet cache instances
Mail
Mail providers
Mail sessions
URL
URL providers
URLs
Resource Environment
Resource Environment Providers
Resource environment entries
Security
Environment
System administration
Users and Groups
Monitoring and Tuning
Data sources
Data sources > B2A1 > JAAS - J2C authentication data
Specifies a list of user identities and passwords for Java(TM) 2 connector security to use.
Prefix new alias names with the node name of the cell (for compatibility with earlier releases)
Apply
Preferences
New... Delete
Select Alias User ID Description
You can administer the following resources:
ofss220075Node08/A1 FCDBAPP_FCDBTEST05
ofss220075Node08/B1A1 FCDBB001_FCDBTEST05
ofss220075Node08/B1AP FCDBB001_FCDBTEST05
Total 3
Help
Field help
For field help information, select a field label or list marker when the help cursor is displayed.
Page help
More information about this page
Command Assistance
View administrative scripting command for last action

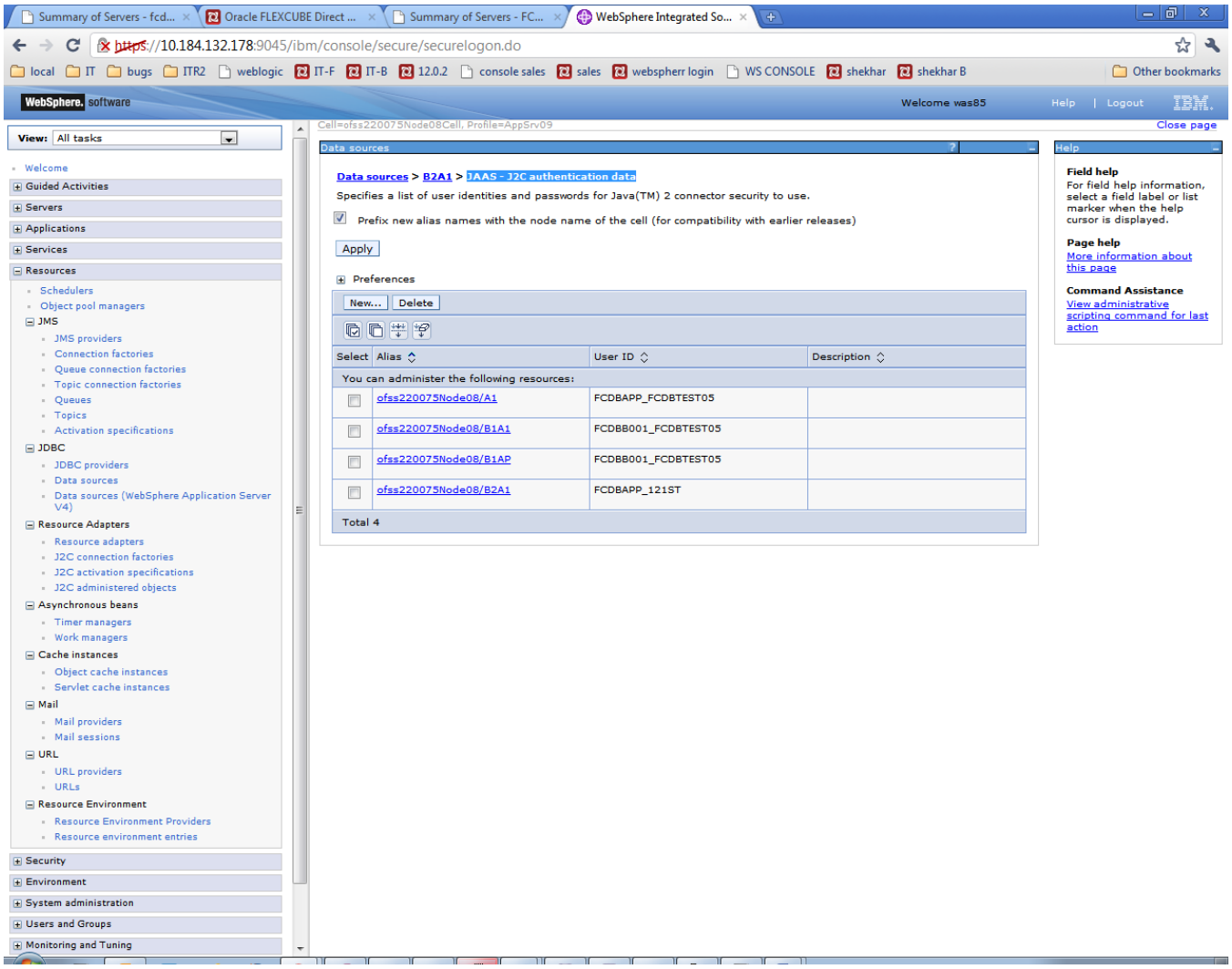
12) Enter Alias, DB user Id and Password. And click ‘Apply’.



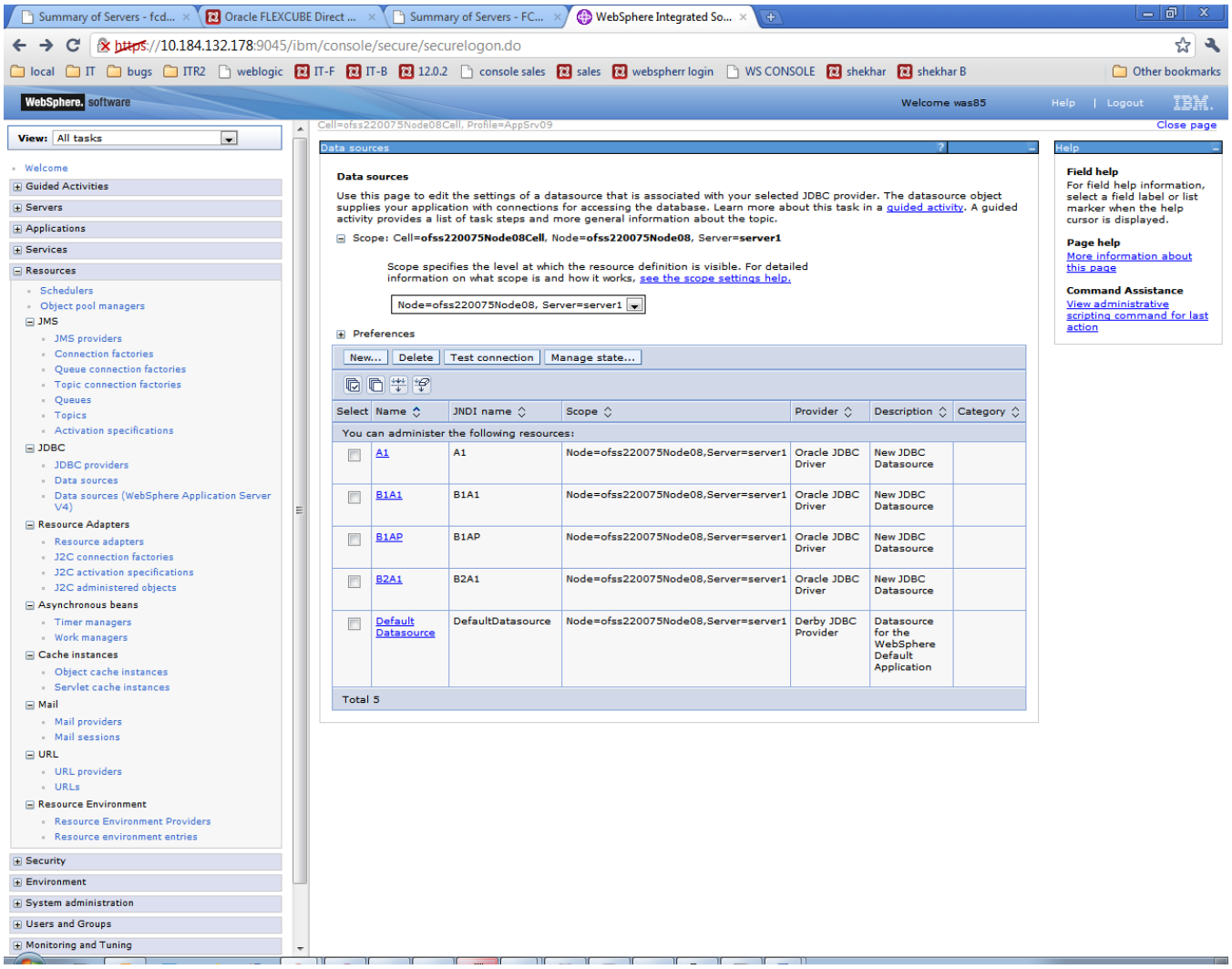
13) Click ‘Save’



14) Newly created JAAS - J2C authentication data will be displayed.



15) Go to the data source to which JAAS - J2C authentication data has to be applied.



16) On configuration screen, under Security Settings, select the proper alias for Component-managed authentication alias and Container-managed authentication alias and click Apply

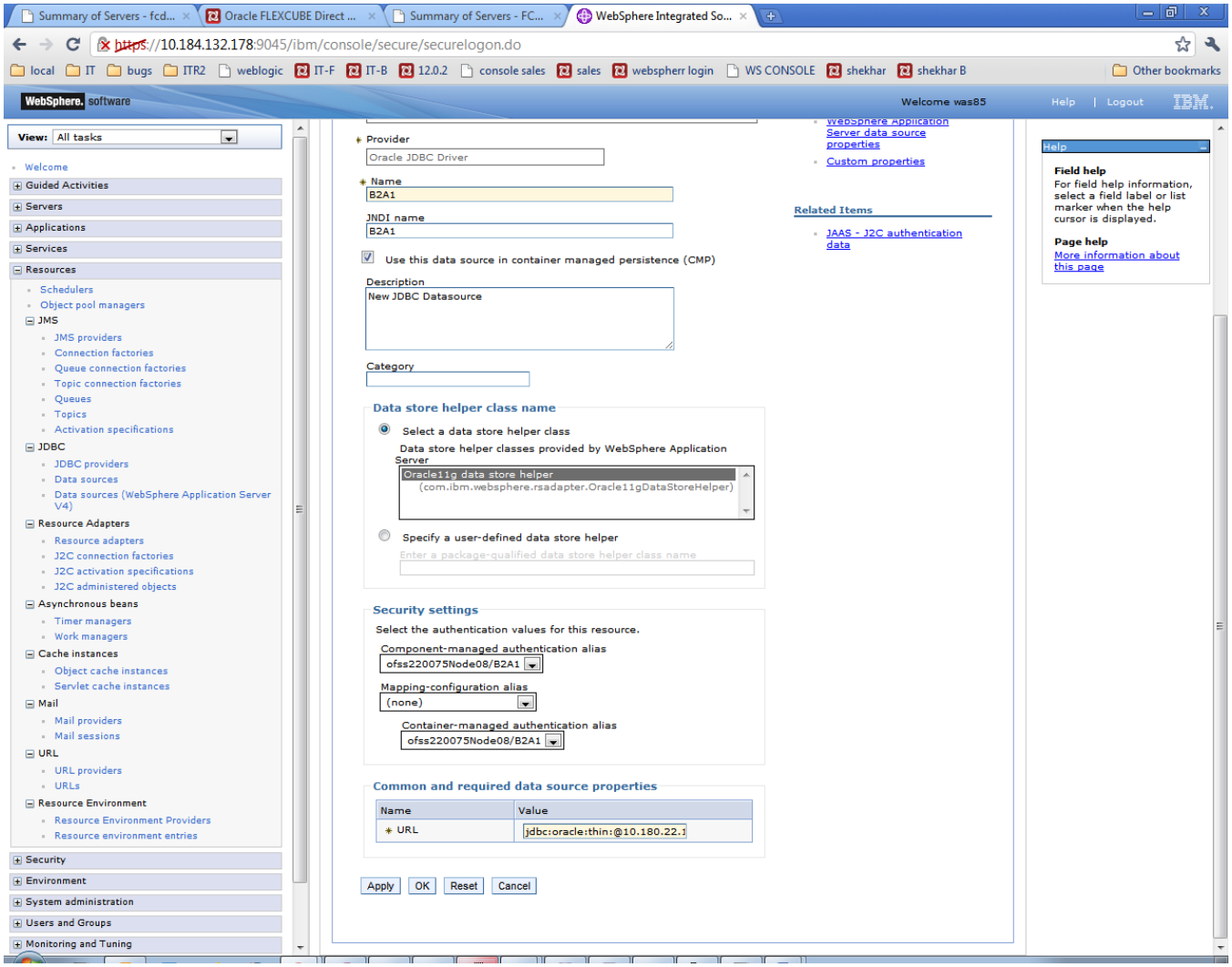
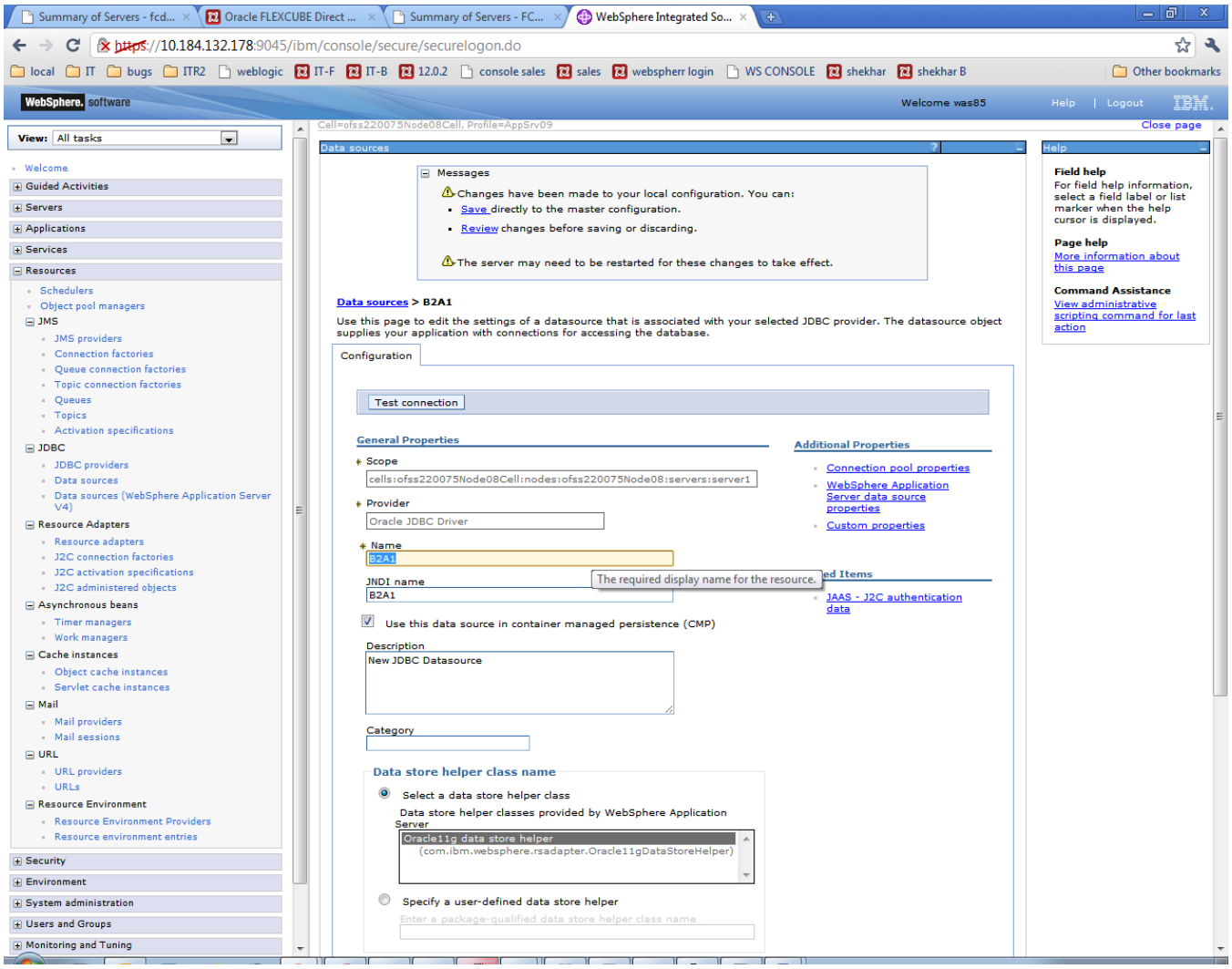
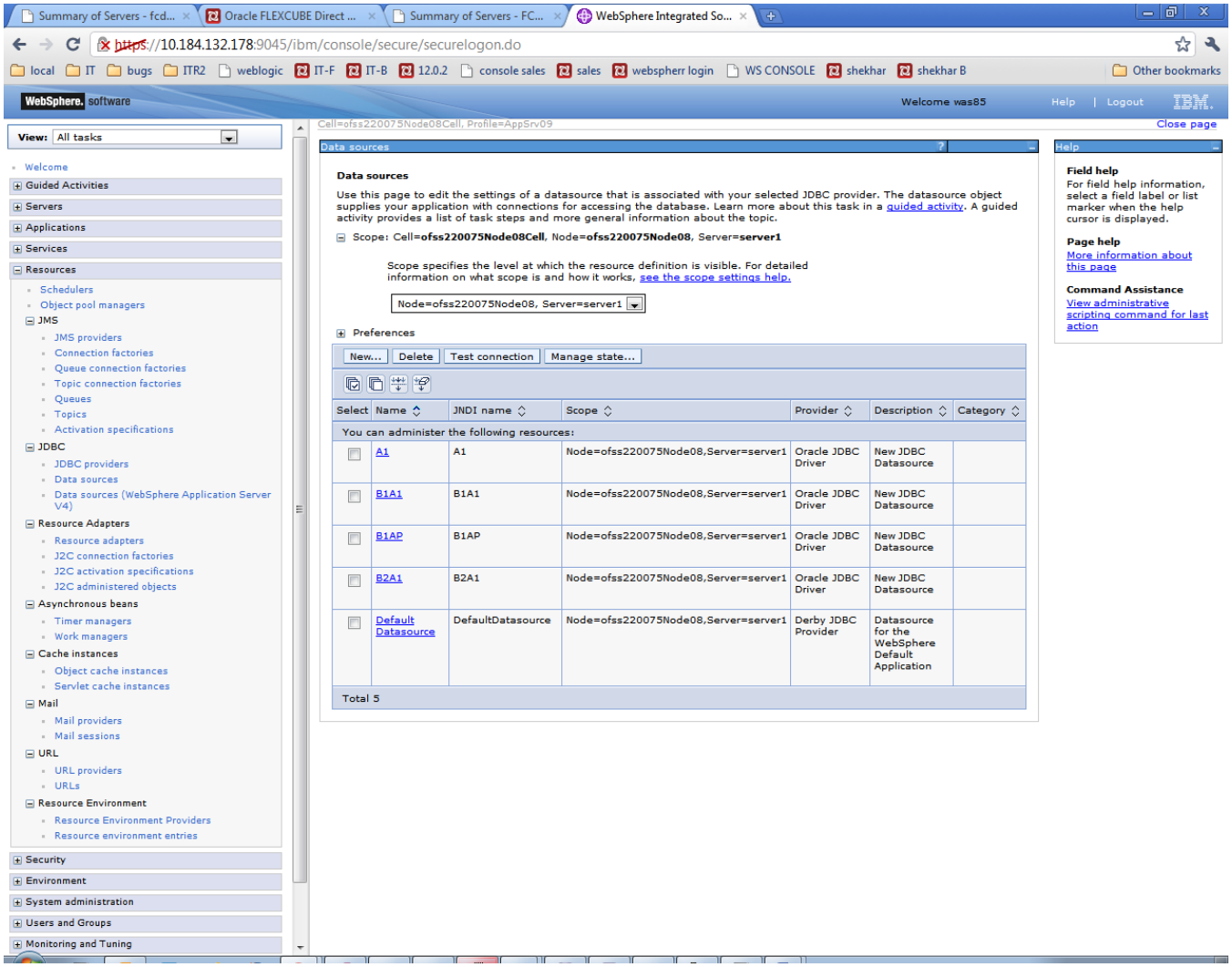


Figure 6.2.17

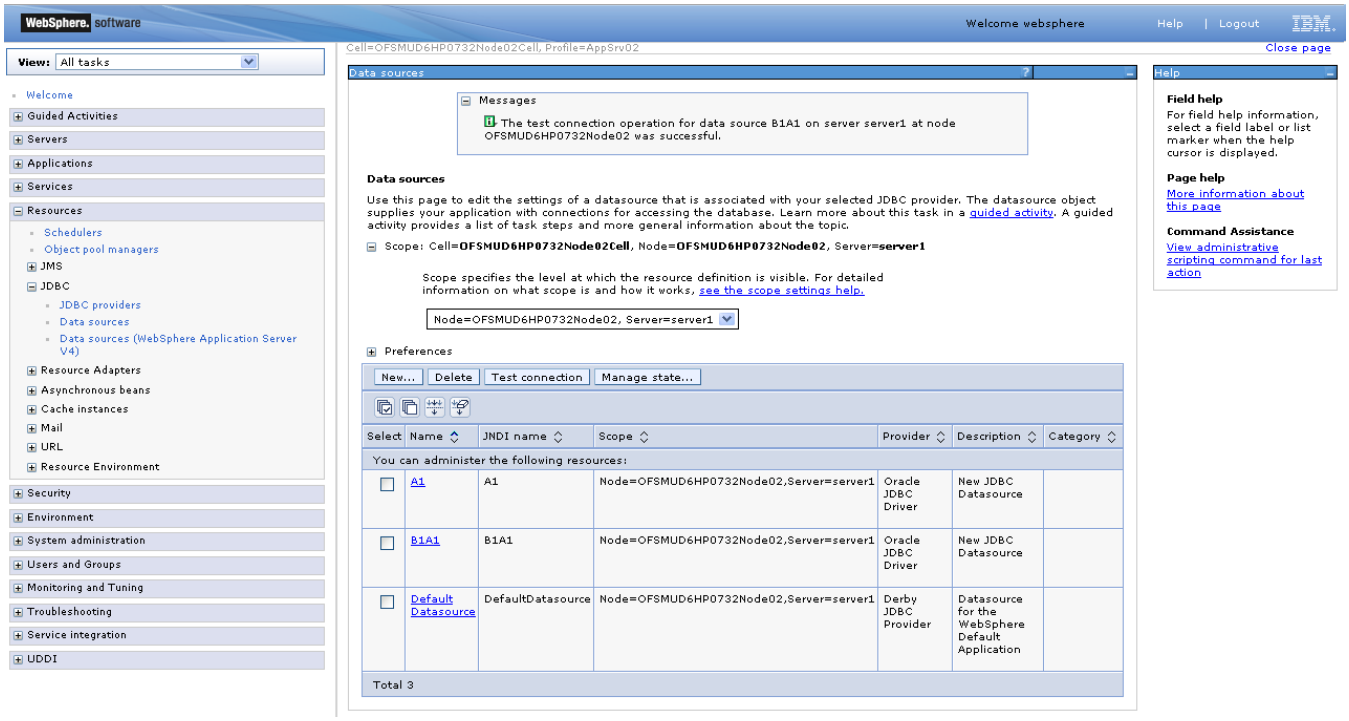
17) Click ‘Save’



18) Test the new created DataSource. JDBC >Data sources>select data source and click on Test connection



Success screen will be displayed (shown underneath) stating the test connection operation for database at node is successful.



8. Integration of WebSphere MQ6 with WAS

Only those modules which require deployment on Message Queue e.g. the Bulk module and configuration of JMS on the server need to follow the setup steps mentioned below for integration of Web sphere MQ6 with Weblogic server.

Pre-requisites:

Integration of WebSphere MQ6 including creation of queue manager, queues etc must be complete as per the reference doc attached "***Oracle_FLEXCUBE_Direct_Banking_Websphere_MQ_6***".

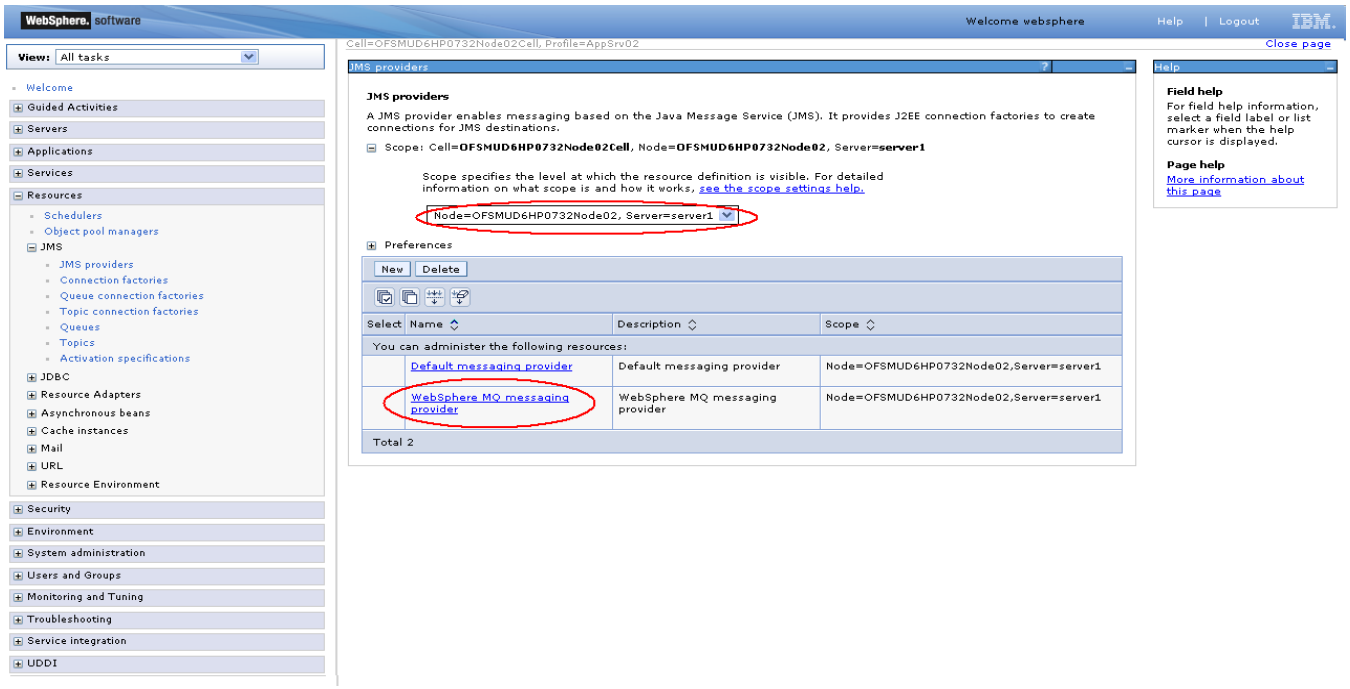
Path of the directory containing following jars should be given in websphere variable - MQ_INSTALL_ROOT

- com.ibm.mq.jar
- com.ibm.mqjms.jar
- com.ibm.mq.jmqi.jar
- com.ibm.mq.commonservices.jar
- dhbcore.jar
- fscontext.jar
- Providerutil.jar

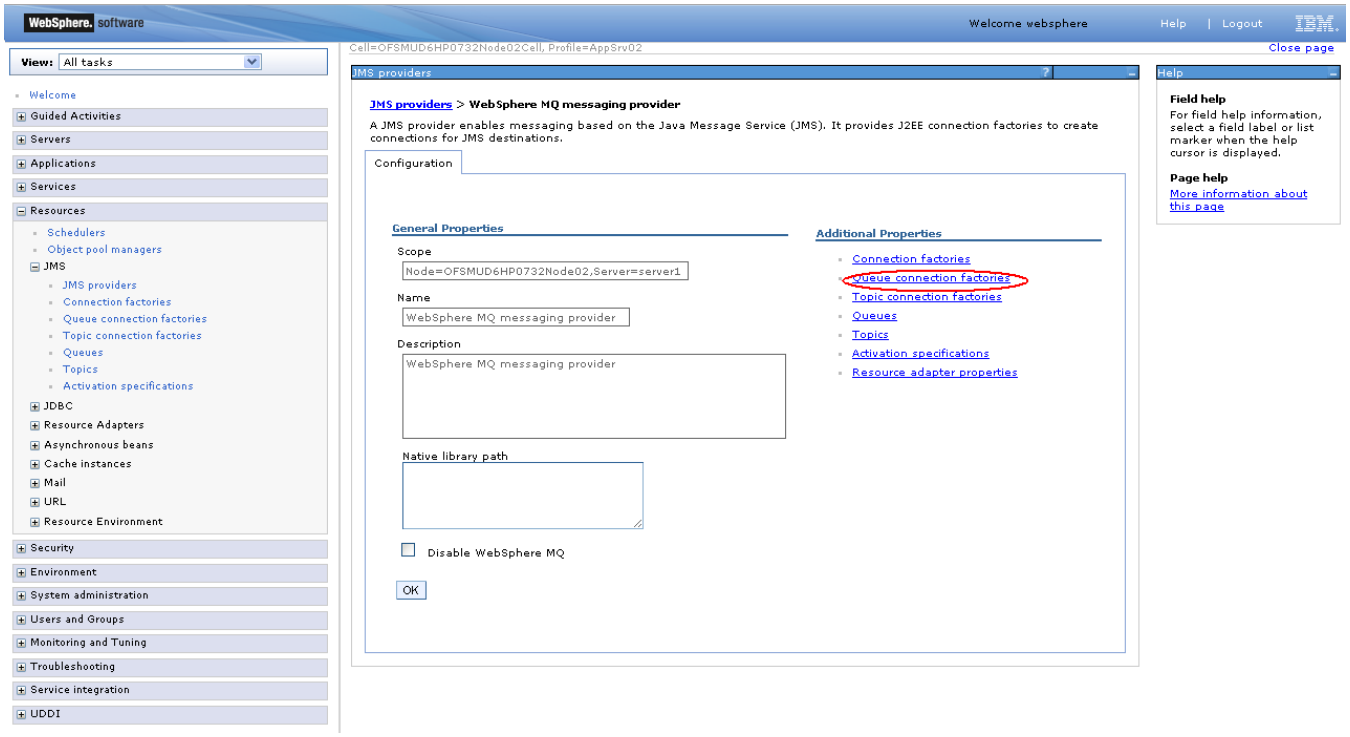
8.1 Create JMS Module

1) Click on "*Resources→JMS Providers"*

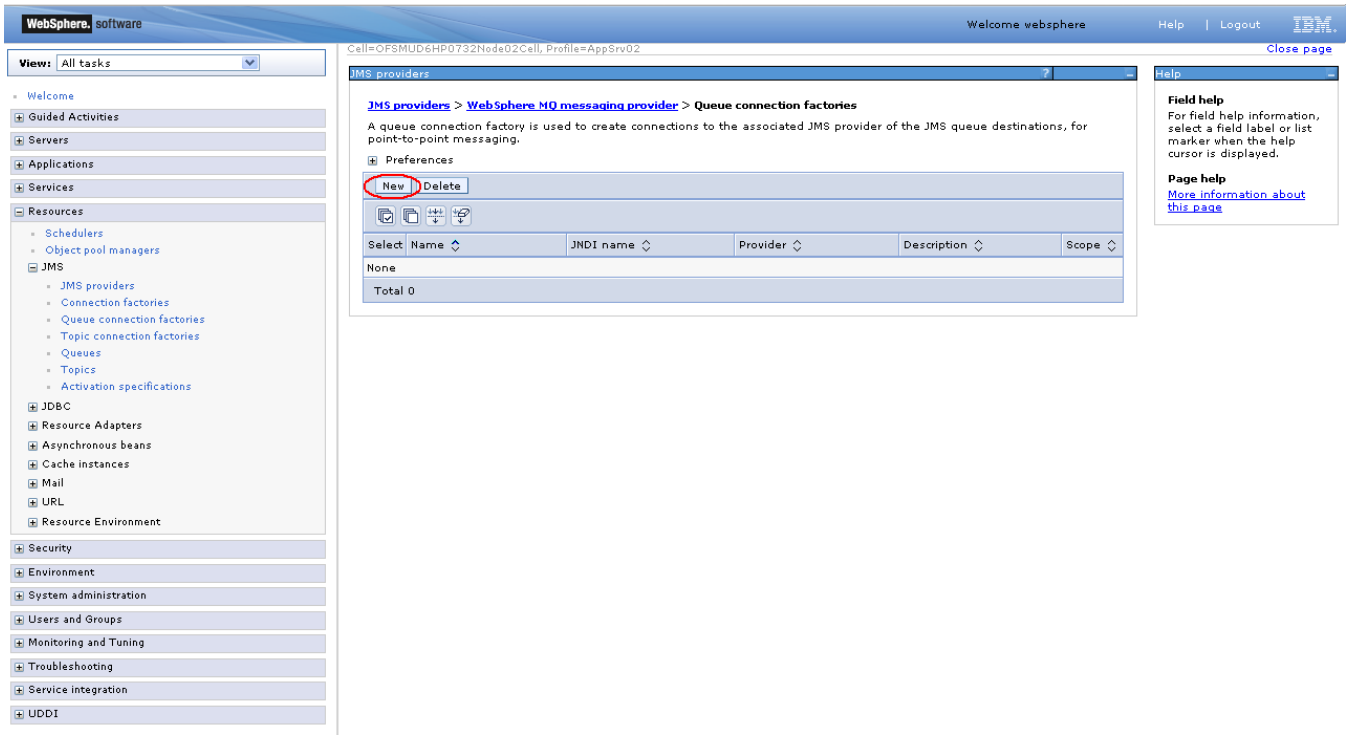
Select the scope from the dropdown as shown in below screenshot and then click on "Websphere MQ messaging provider"



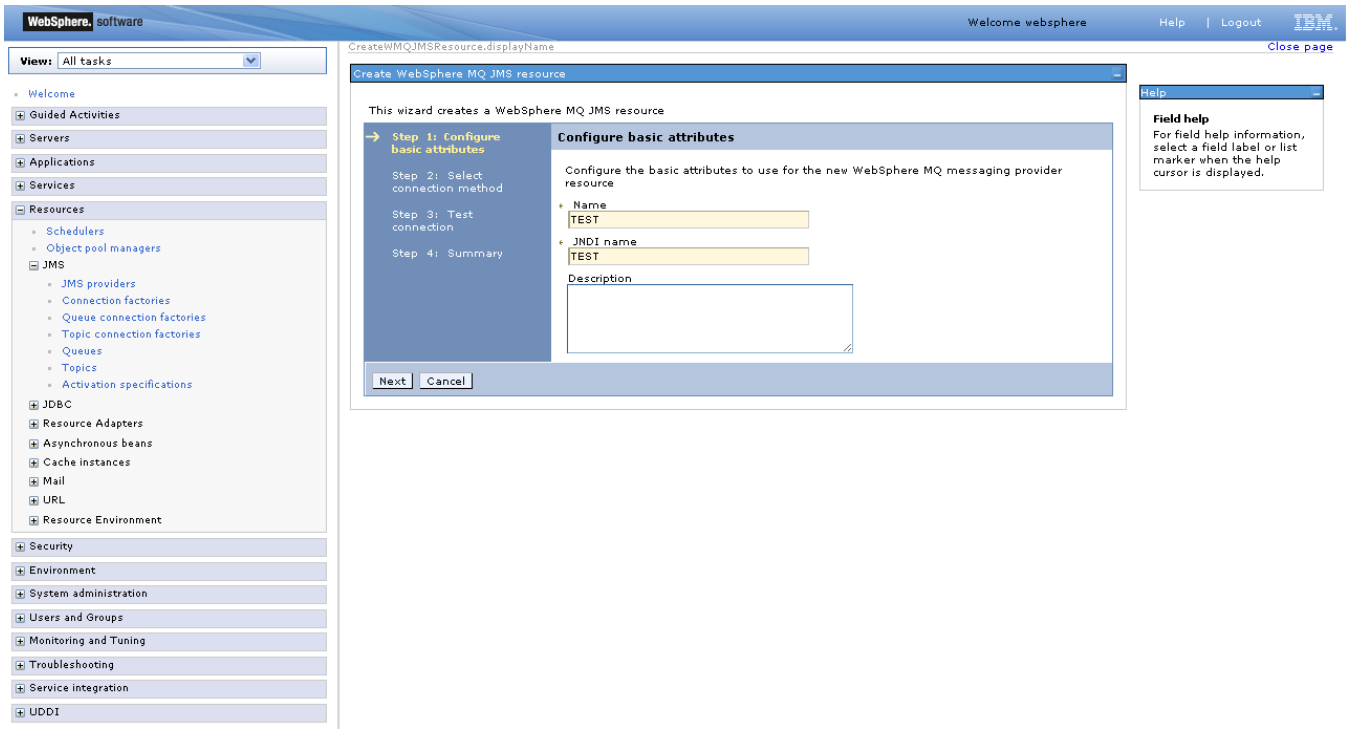
2) Click on “Queue connection factories”



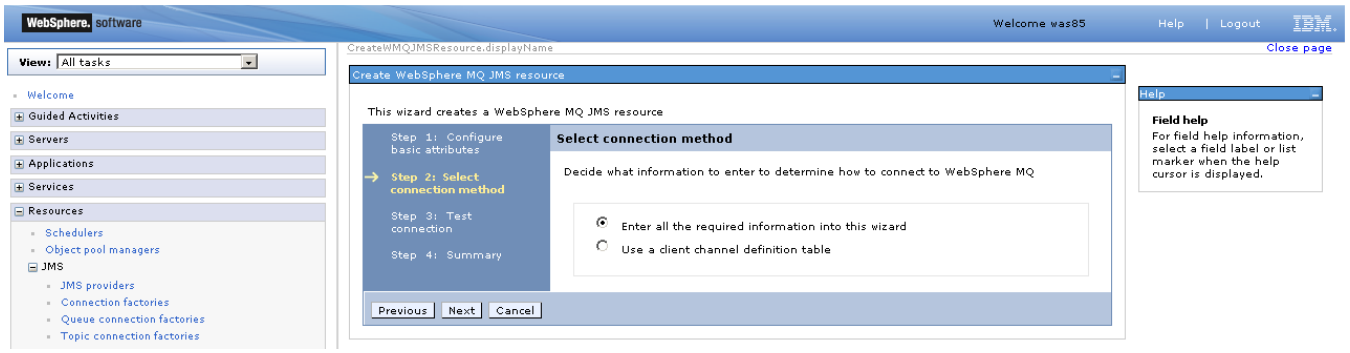
3) On “Queue connection factories screen” click on new.



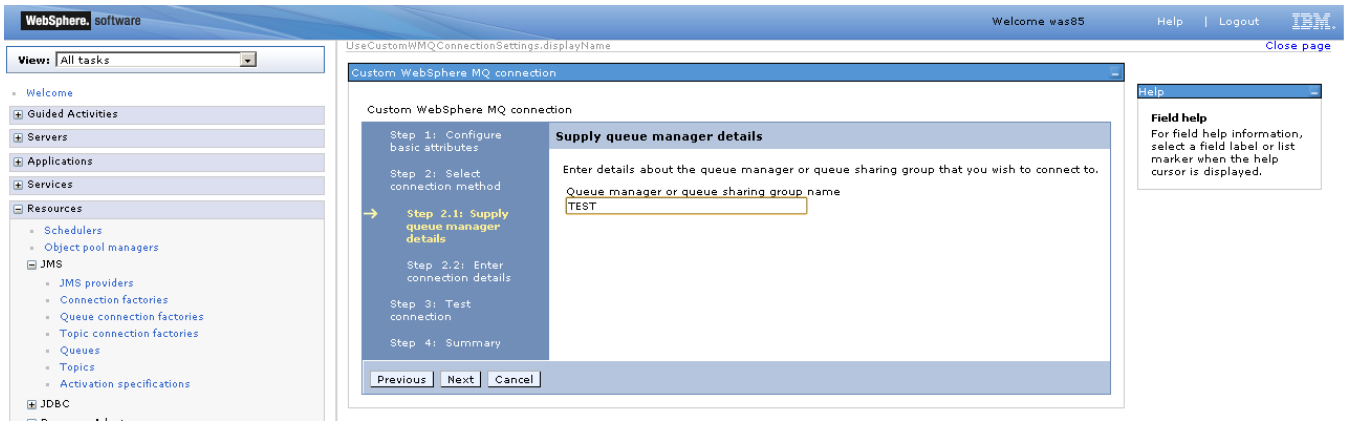
4) Enter the name of the Connection factory and JNDI name and click next



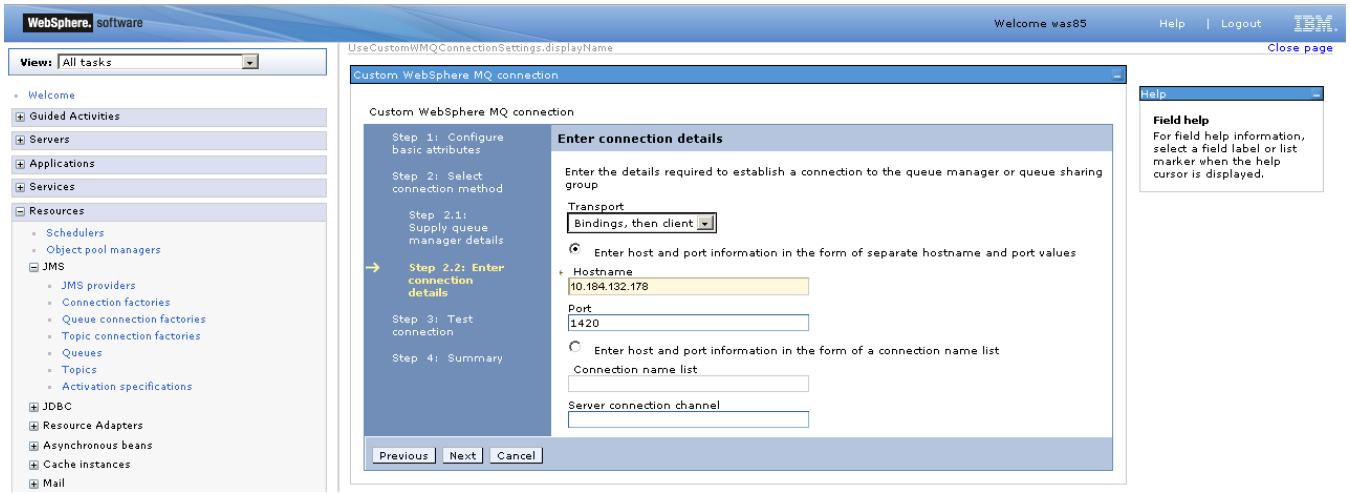
5) On “Select Connection method” screen, select *Enter all the required information into this wizard* option and click next



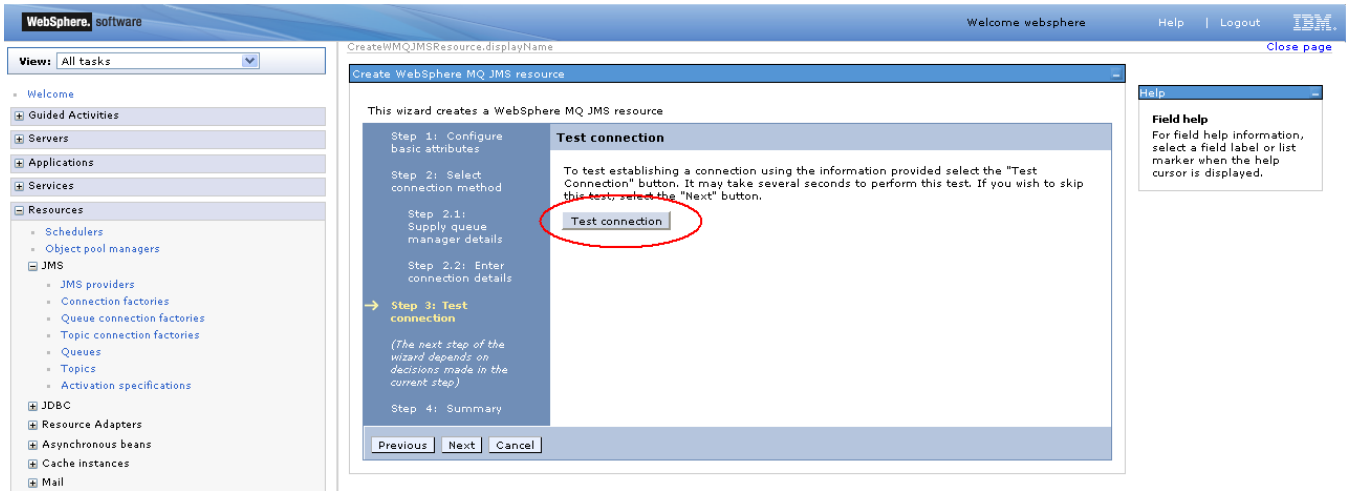
6) On “Supply queue manager details” screen enter the name of queue manager. Please note that this name must be same as queue manager name on websphere MQ. Now click next



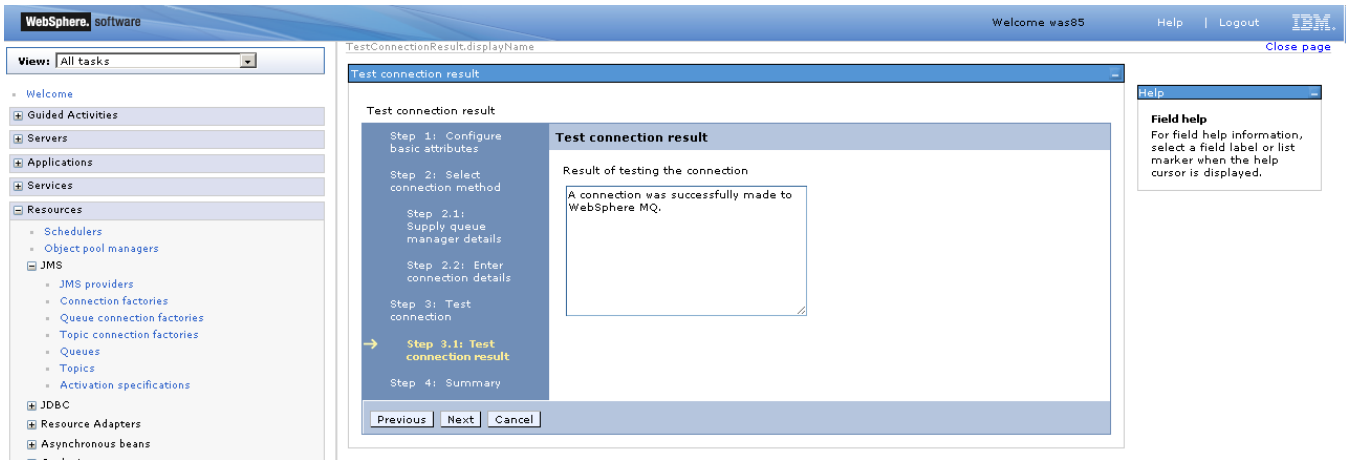
7) On “Enter connection details” screen enter the hostname or IP of the MQ host and port no and click next



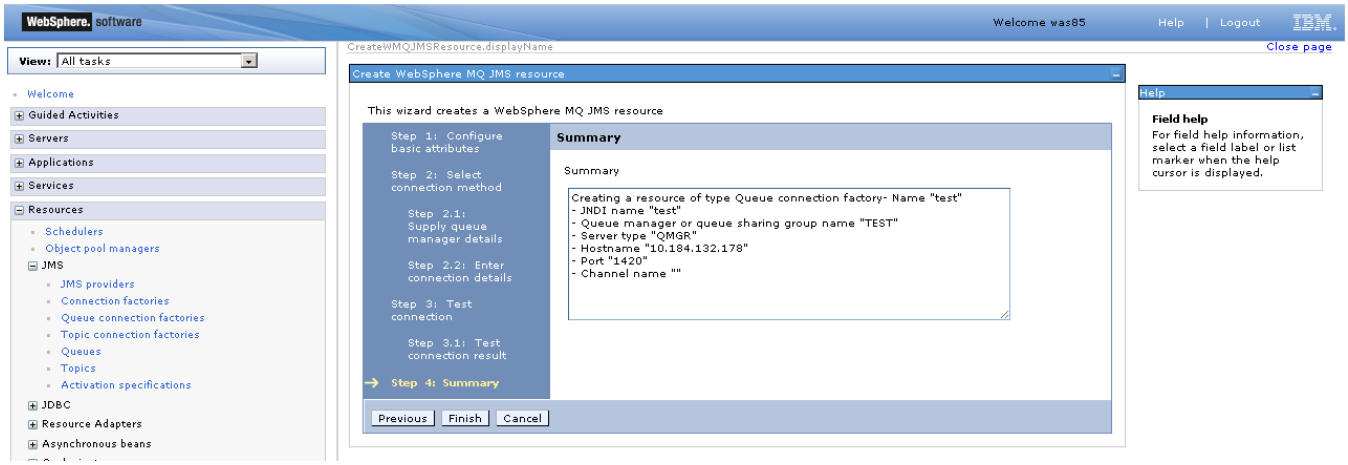
8) On ‘Test connection” screen click on test connection button



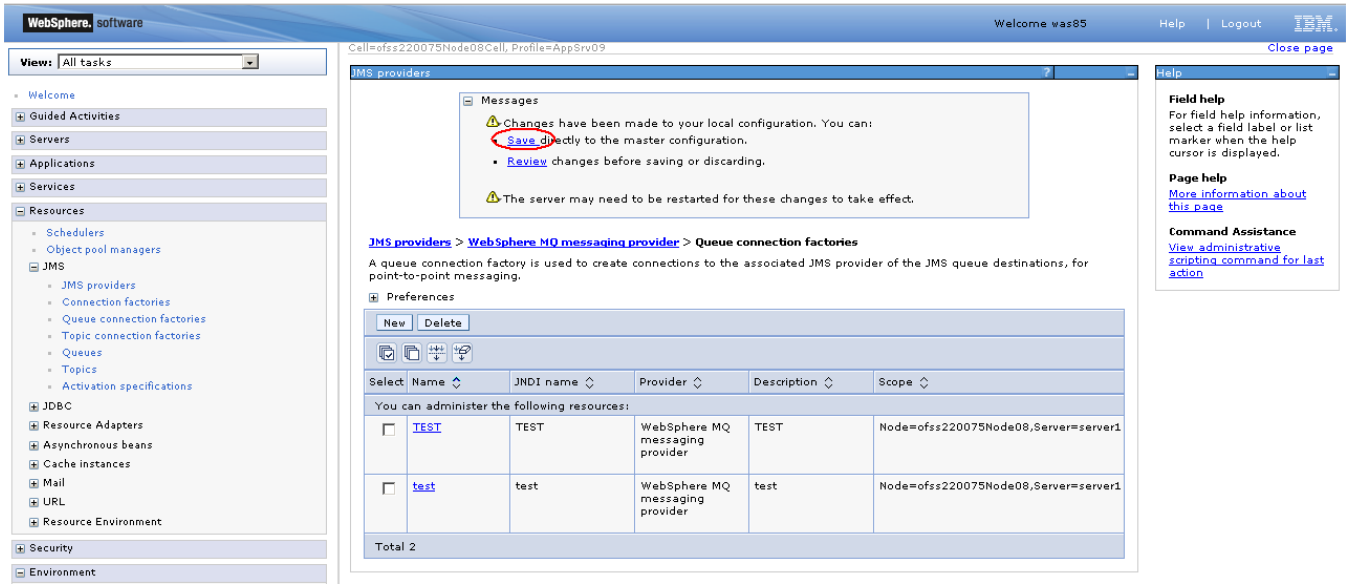
9) If everything entered on the previous screens is correct then it will display success message as shown below otherwise it will give an error.



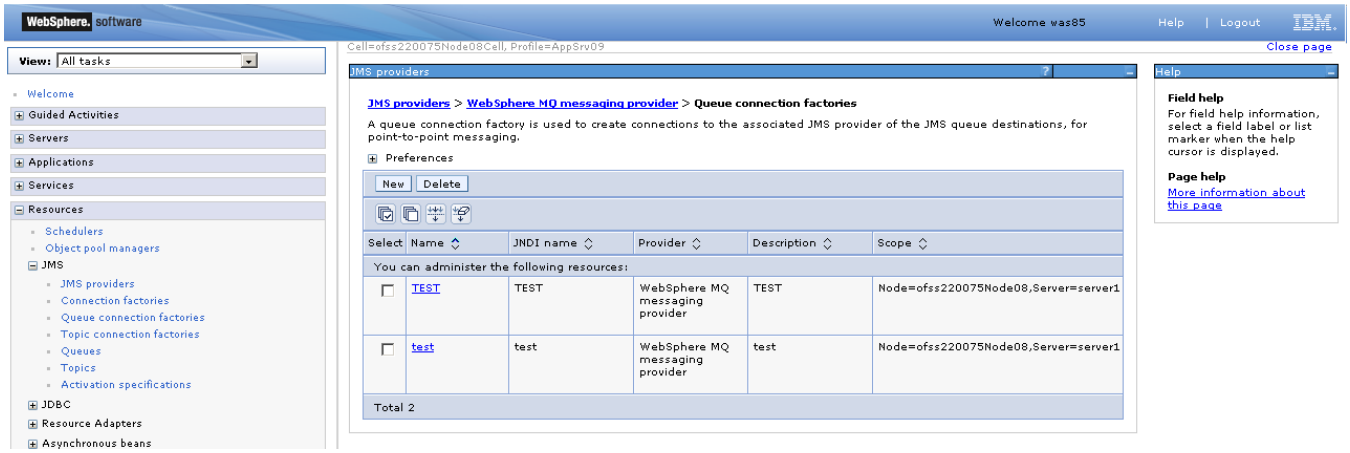
10) On summary screen click finish



11) On the screen shown below click save.



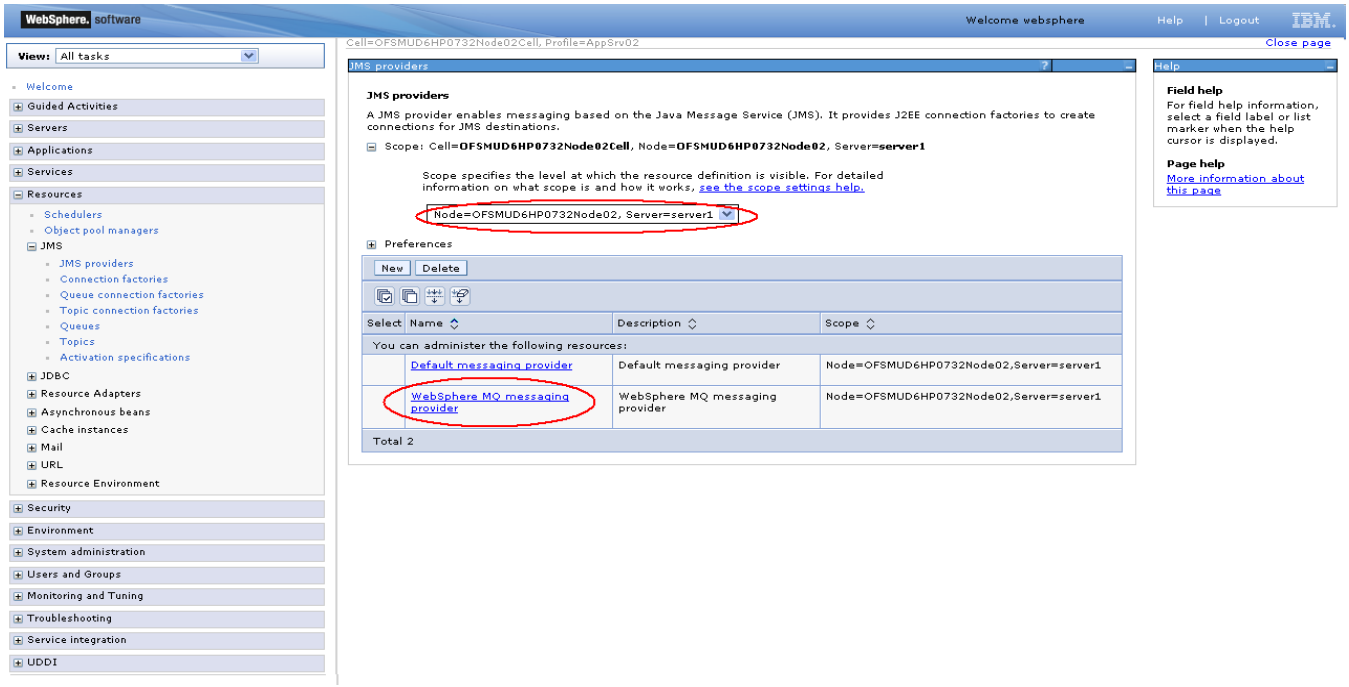
12) Queue connection factory is now created and will be shown in the list as shown in the screenshot below:



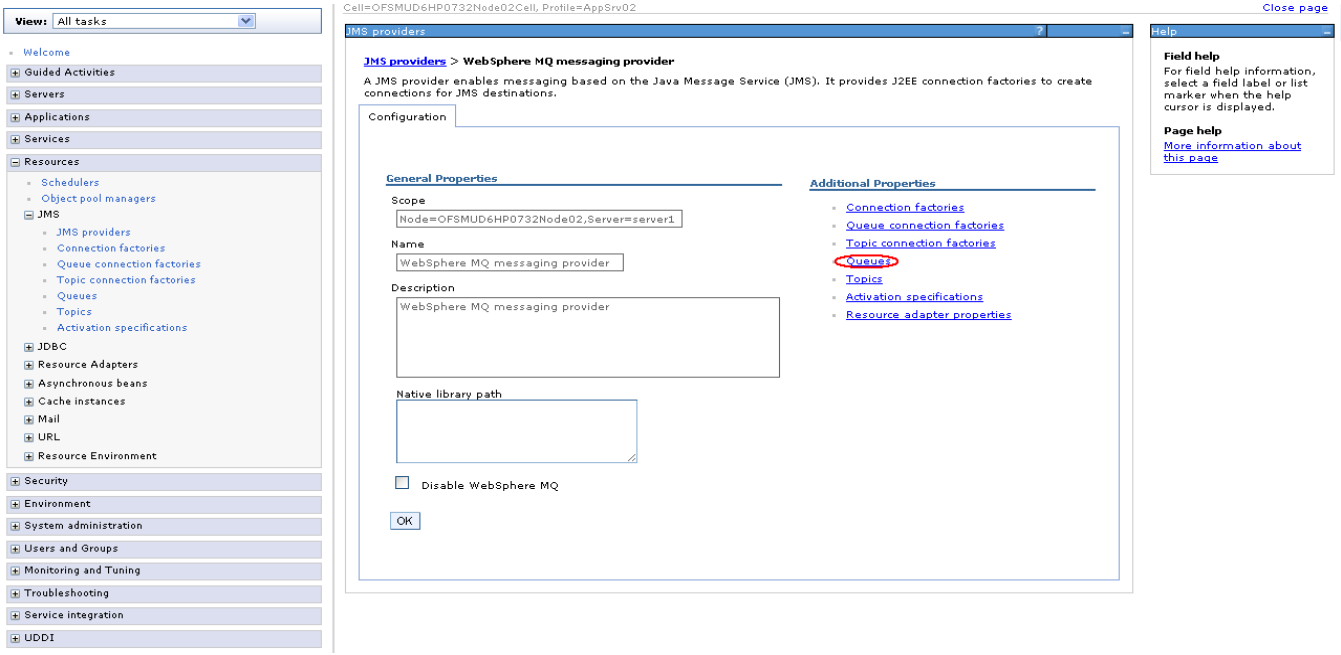
8.2 Creating Destinations

1. Click on "*Resources→JMS Providers*"

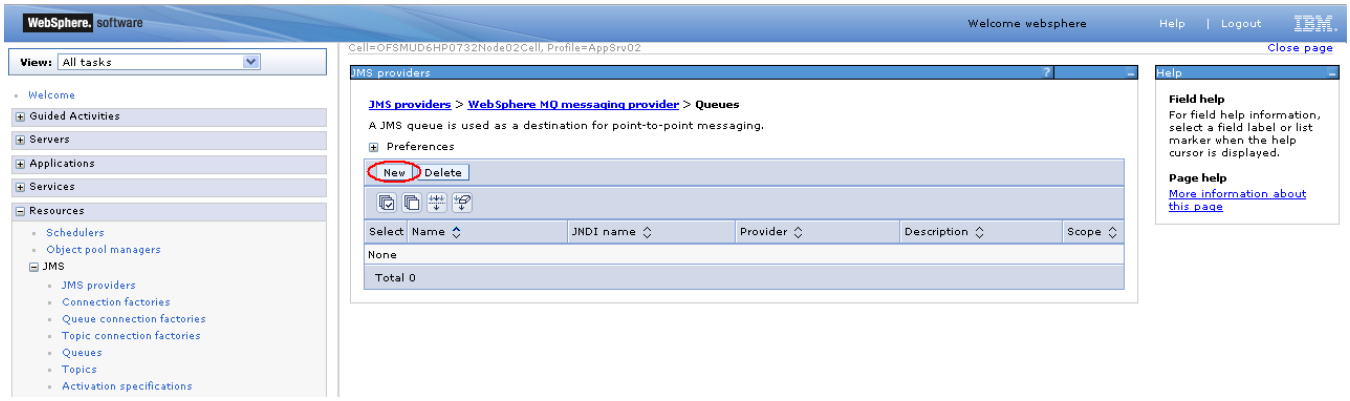
Select the scope from the dropdown as shown in below screenshot and then click on "Websphere MQ messaging provider"



2. Click on "Queues"



3. Click on New

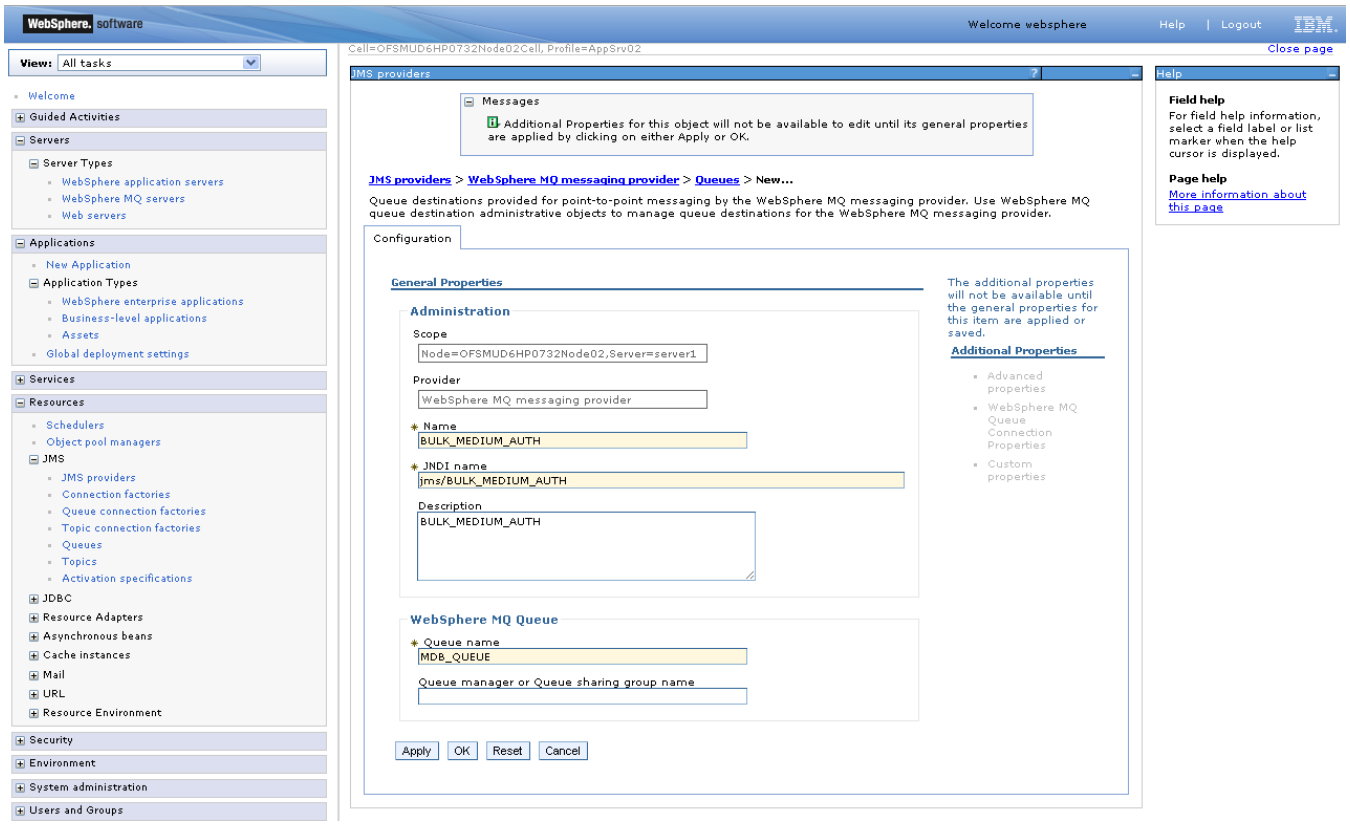


4. Specify following details and save settings.

   NAME: *<request_queue_name>*

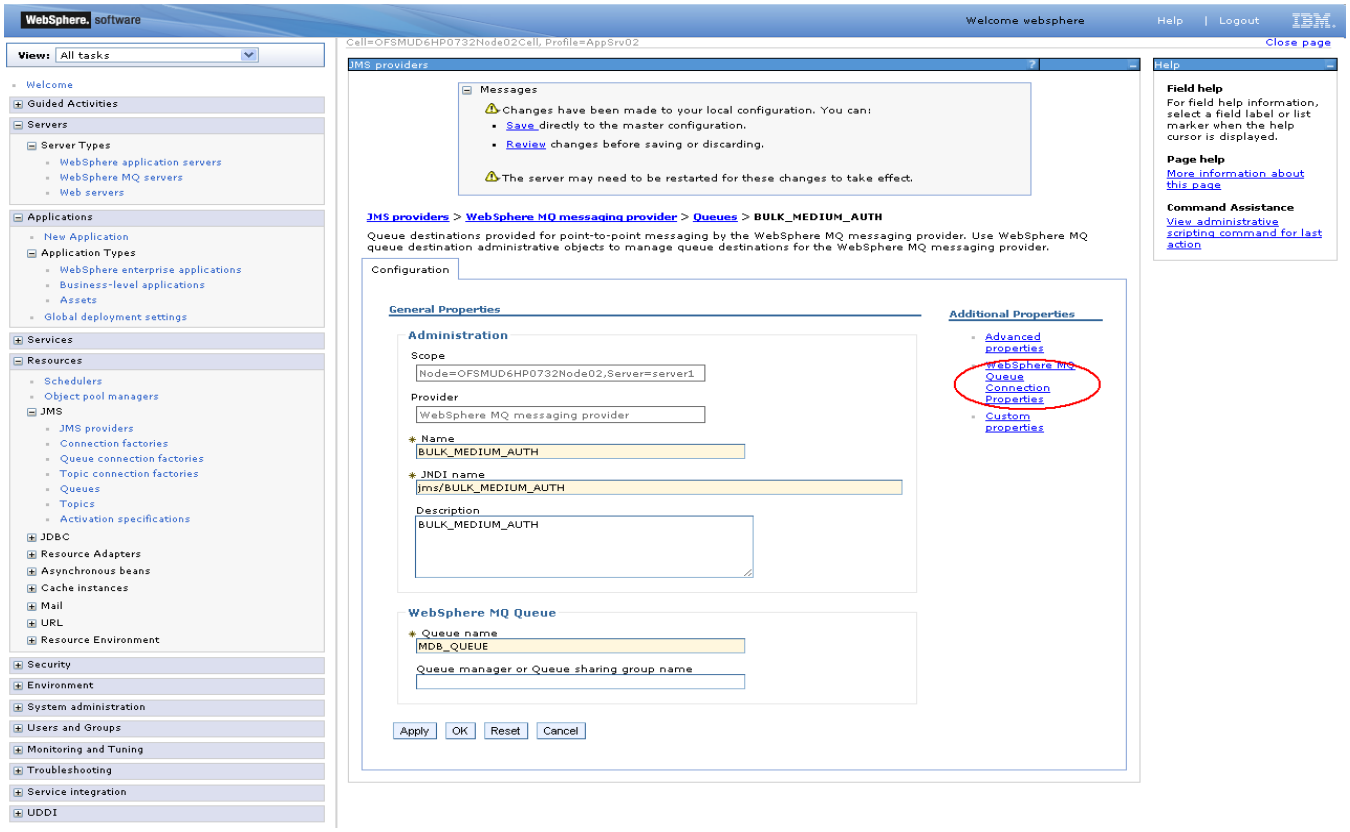
   JNDI NAME: *<request_queue_JNDI_name>*

   Websphere MQ Queue Name: *<base_queue_name>*

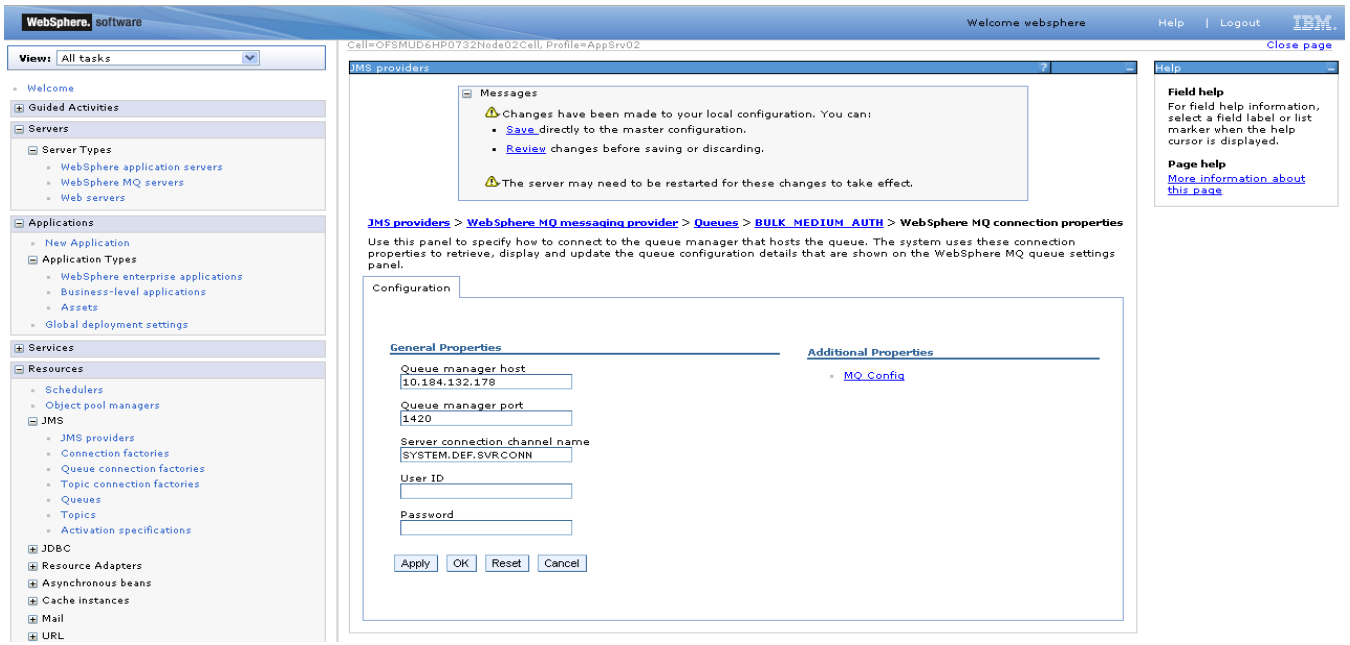


5. Click on apply

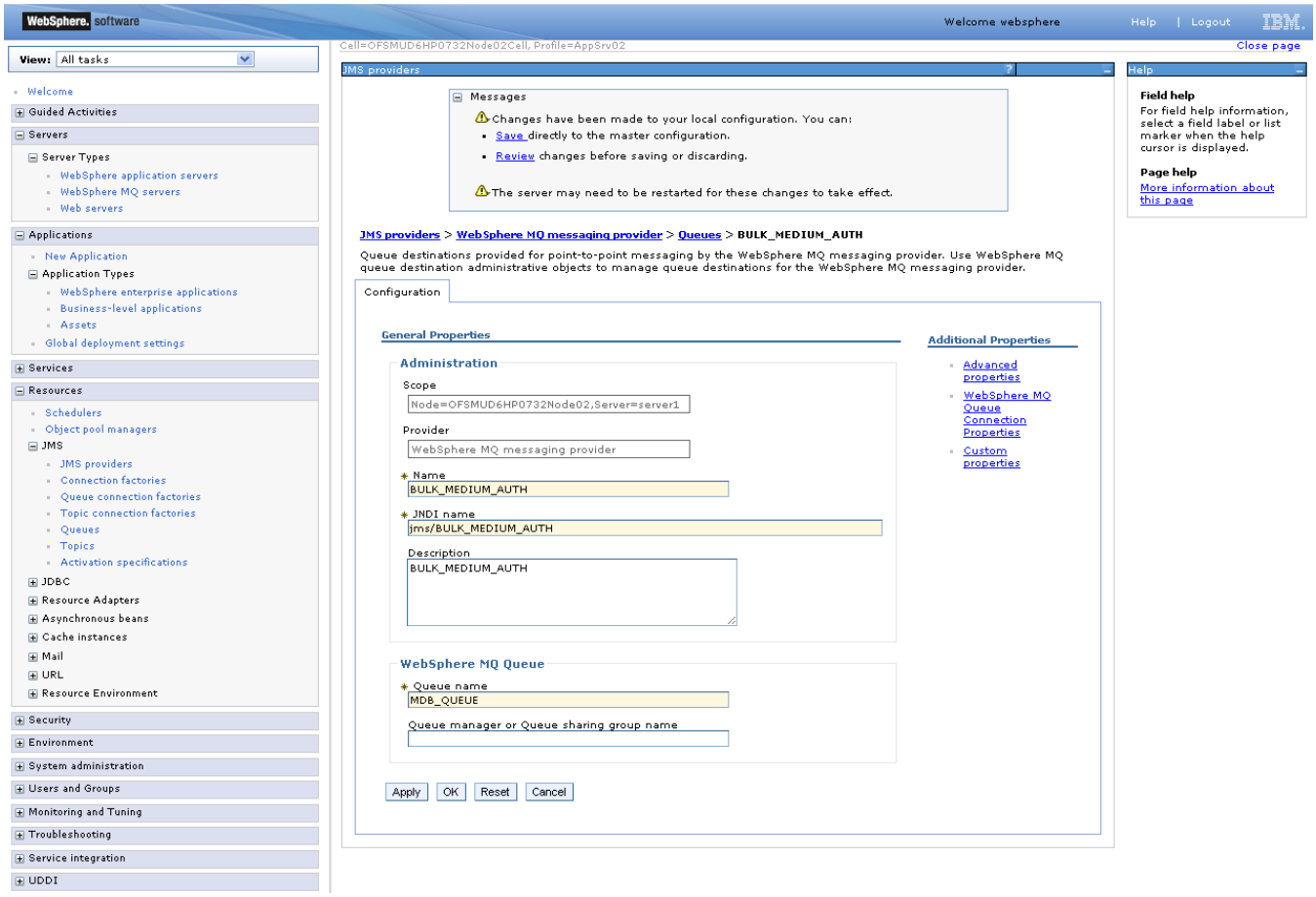
6. Click on “Websphere MQ queue connection properties” as shown in the screenshot below.



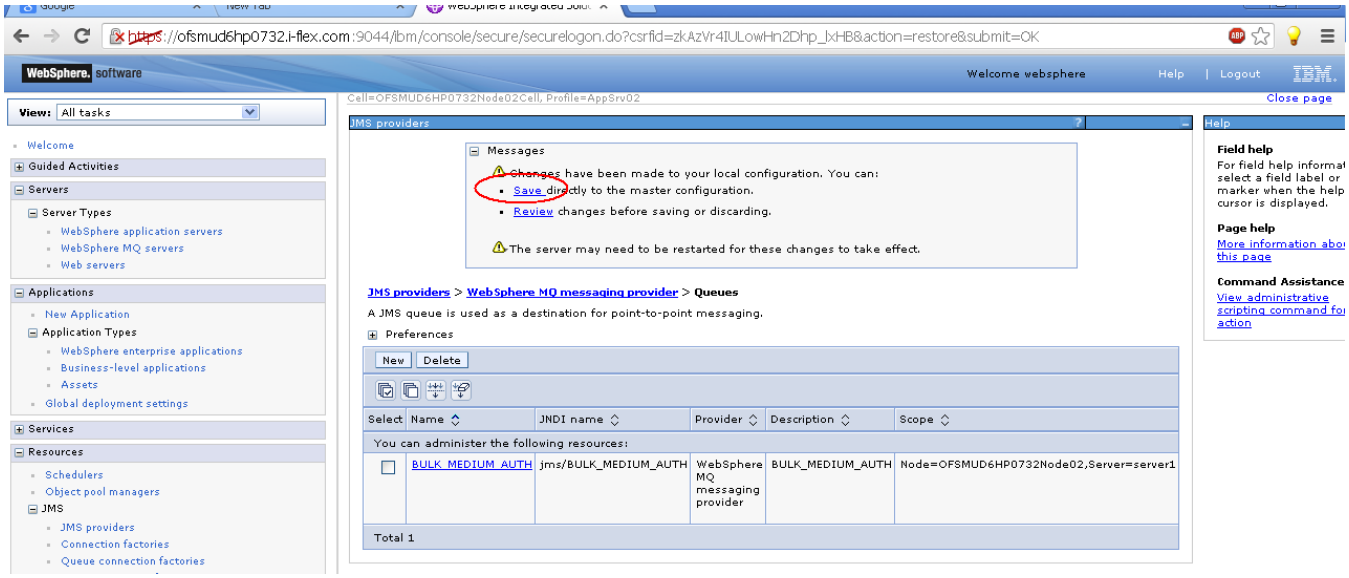
7. Enter queue manager hostname, queue manager port number and server connection channel name and click OK.



8. Again click OK on the screen shown below.



9. Click on save as shown in the below screen.



10. Repeat the above steps for response queue with following details:

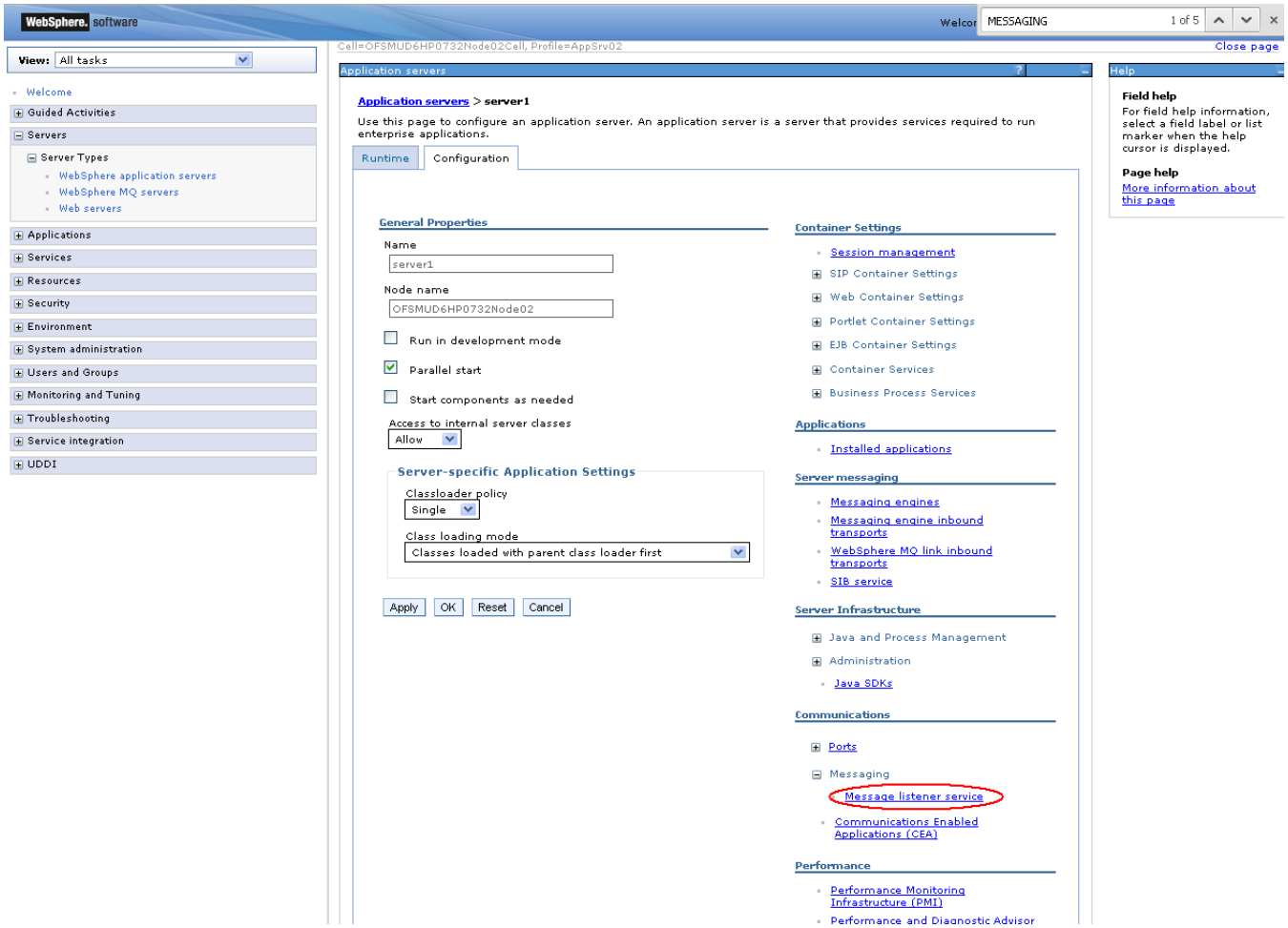
    NAME: <*response_queue_name*>

    JNDI NAME: < *response _queue_JNDI_name*>

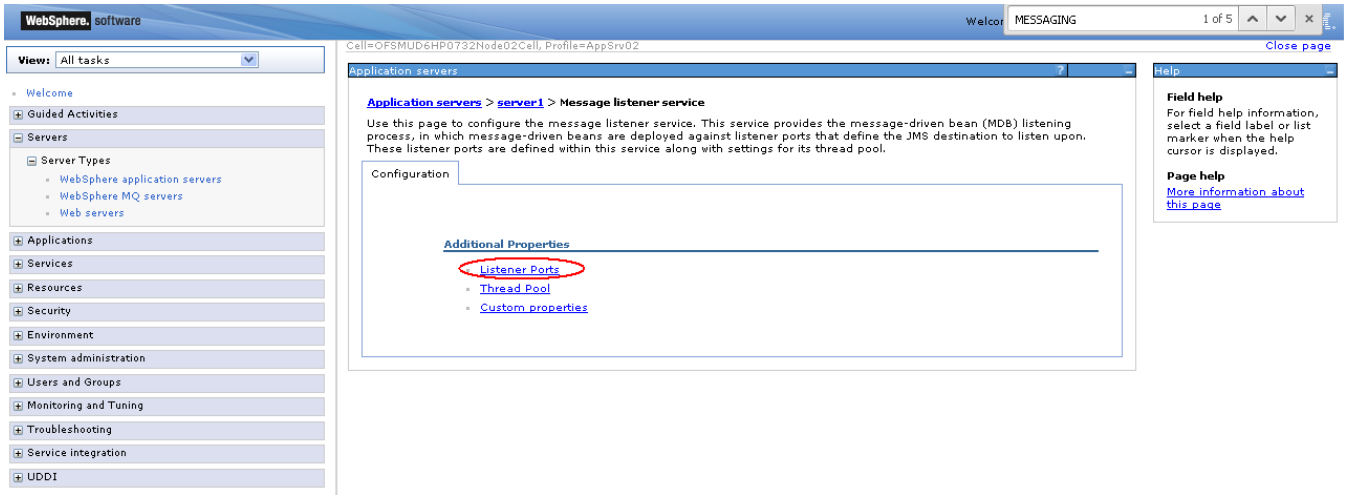
    BASE QUEUE NAME: <*base_queue_name*>

8.3 Creation of Listeners

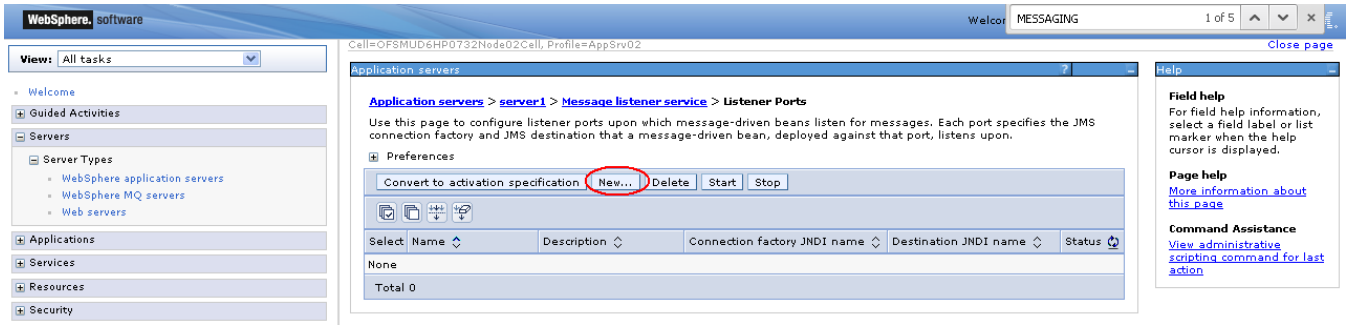
a) " Click on Server→ Websphere application Server → Server name → Click on Messaging > Message Listener Service"



b) Click on Listener ports.



c) Click “New” as shown in the screen below.

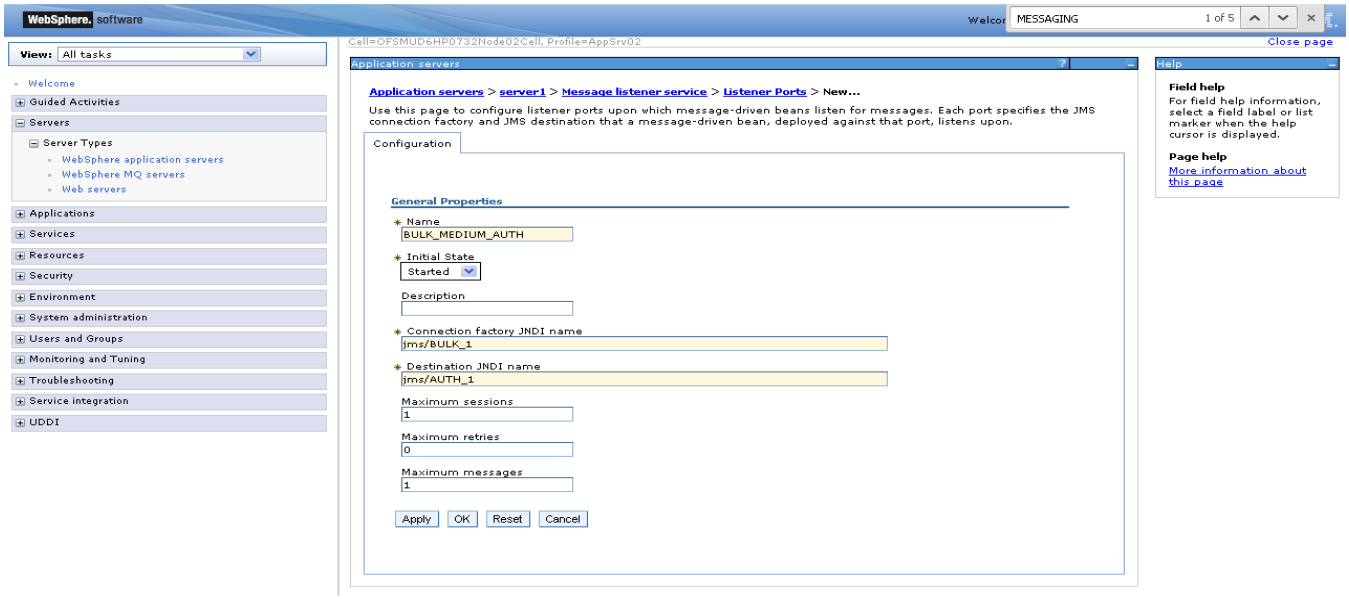


d) Enter details as mentioned below table

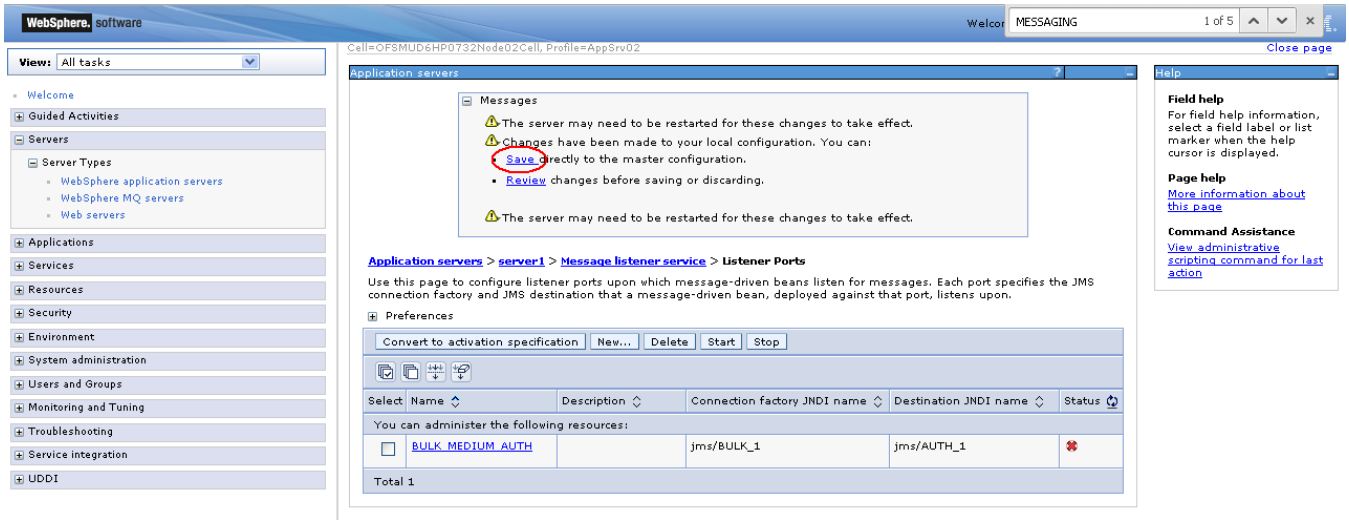
Connection factory JNDI: < *connection_factory_JNDI_name* >

Destination JNDI name: <response_queue_JNDI_name>

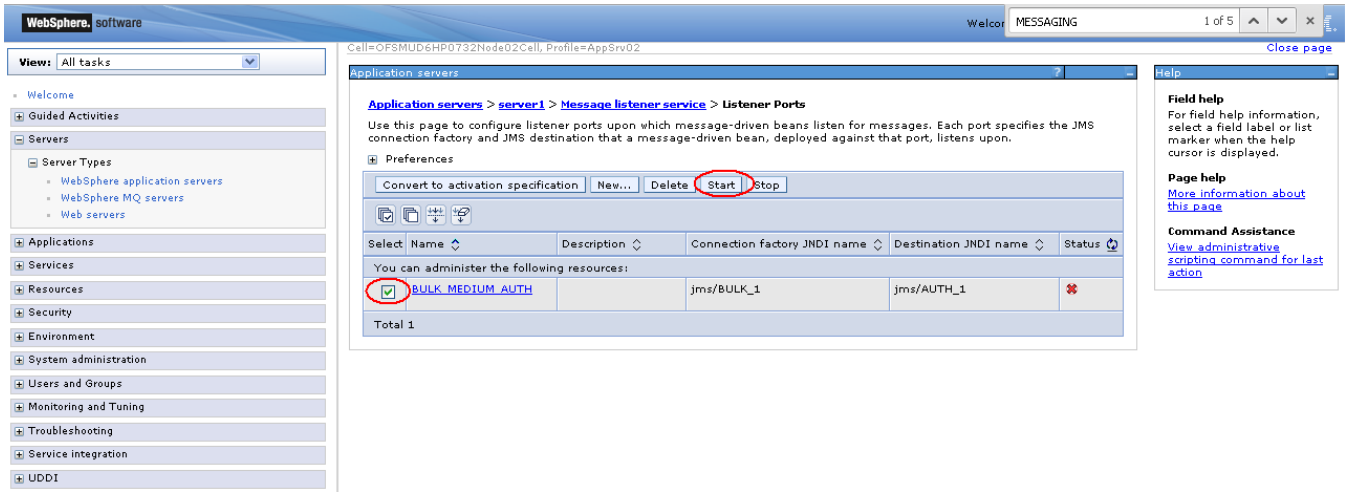
After entering all required information click OK.



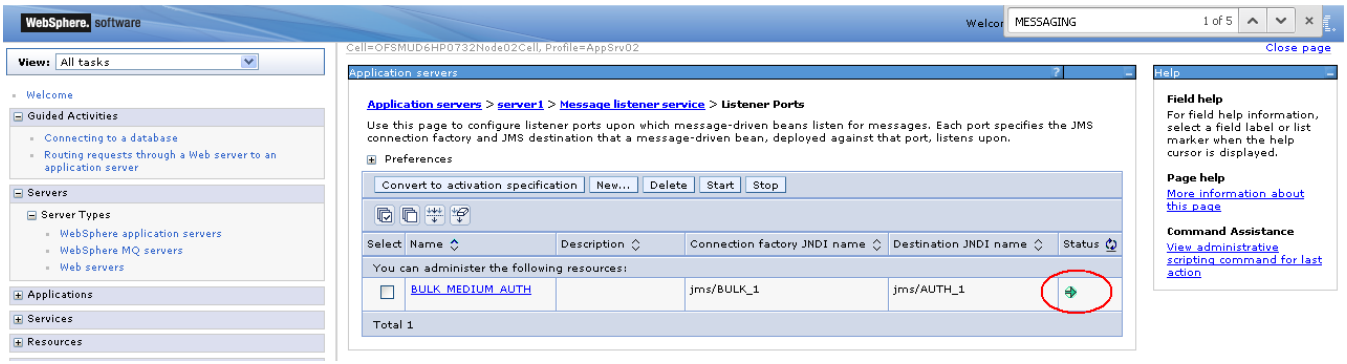
e) Now click on save as shown in the below screen.



f) Now select the listener and click on start as shown in the screenshot below.



g) If the listener started successfully then the status of the listener will turn green as shown in the screenshot below and if it is not started it will display the error and the status will remain red.



Third Party Setup

For third party Listeners, destination queues and connection factories should be created separately with respective names.

Connection Factory: TPCF

Request Queue name: THIRDPARTYINQ

Response Queue name: THIRDPARTYOUTQ

Two listeners should be created for In and Out Queue.

Bulk Setup

For Bulk file upload, following destination queues, connection factory and message listeners have to be created:

Connection factory JNDI name (as in btqueuesetup table): e.g. jms/BULK_1

Following queues have to be configured (JNDI name as in btqueuesetup table) for each step:

Name	JNDI Name	Base queue name
AUTH	jms/AUTH_1	AUTH
AUTHREJECT	jms/AUTHREJECT_1	AUTHREJECT
COMPLETE	jms/COMPLETE_1	COMPLETE
DECRYPT	jms/DECRYPT_1	DECRYPT
PREPROCESS	jms/PREPROCESS_1	PREPROCESS
PROCESS	jms/PROCESS_1	PROCESS
RESPONSE	jms/RESPONSE_1	RESPONSE
UPLOAD	jms/UPLOAD_1	UPLOAD

For each destination, there will be a message listener:

Name	Connection factory JNDI name	Destination JNDI Name
BULK_MED_AUTH	jms/BULK_1	jms/AUTH_1
BULK_MED_AUTHREJ	jms/BULK_1	jms/AUTHREJECT_1
BULK_MED_COMPLETE	jms/BULK_1	jms/COMPLETE_1
BULK_MED_DECRYPT	jms/BULK_1	jms/DECRYPT_1
BULK_MED_PREPROCESS	jms/BULK_1	jms/PREPROCESS_1
BULK_MED_PROCESS	jms/BULK_1	jms/PROCESS_1
BULK_MED_RESPONSE	jms/BULK_1	jms/RESPONSE_1
BULK_MED_UPLOAD	jms/BULK_1	jms/UPLOAD_1

Update following properties in MSTPROPERTIES table of FCDB.

- FCAT.SEND.FACTORY.LOOKUP.NAME=<connection_factory_JNDI_name>
- FCAT.SEND.QUEUE.LOOKUP.NAME=<request_queue_JNDI_name>
- FCAT.SEND.QUEUE.NAME=<request_queue_JNDI_name>
- FCAT.SEND.PORT.NO=<Queue_Manager_port>
- FCAT.SEND.SERVER.NAME=< Queue_Manager_server_ip>
- FCAT.SEND.QMANAGER.NAME=<Queue_Manager_Name>
- CLONE.Q.CLASS.NAME=com.iflex.fcat.queue.JFQalMQJMS
- FCAT.SEND.CONTEXT.FACTORY=com.ibm.websphere.naming.WsnInitialContextFactory
- FCAT.SEND.PROVIDER.URL=iiop://< FCDB_SERVER _IP>:<FCDBSERVER_BOOTSTRAP_PORT>

8.4 Deployments on application Server

Deploy the following jars on application server:

ClonedQueueReceiverMDB_jar: to listen response queue.

ThirdPartyReceiverMDB_jar: to listen third party response queue.

TPT_sim.jar

Bulk Deployables:

- ✓ BulkEventHandlerEJB.jar
- ✓ StepName_MDB.jar

  StepName is Log queue name as in btqueuesetup table.

  e.g UPLOAD_MDB.jar

9. DB Creation

For completing the DB setup kindly refer to the document –
“Oracle_FLEXCUBE_Direct_Banking_Database_Setup“