Application installation

Unit objectives

After completing this unit, you should be able to:

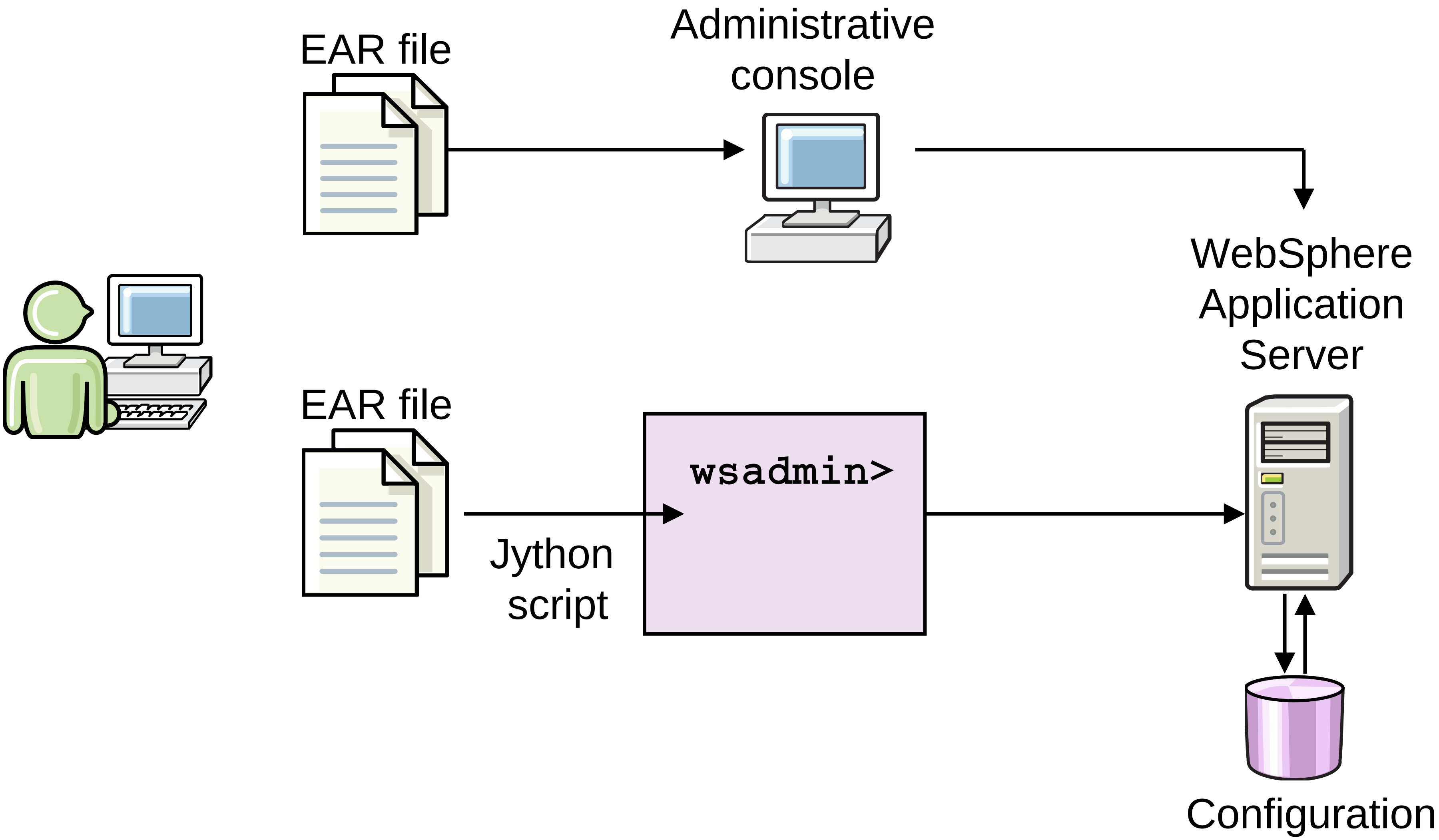
- Describe methods of installing enterprise applications in WebSphere Application Server
- Explain how fine-grained application updates work
- Describe enterprise application properties
- Enable monitored directories
- Deploy an application by using the monitored directory
- Use a profile-file-based configuration with monitored directories to deploy an application

Topics

- Application installation
- Application settings and interaction
- Monitored directory

Application installation

Installing enterprise applications



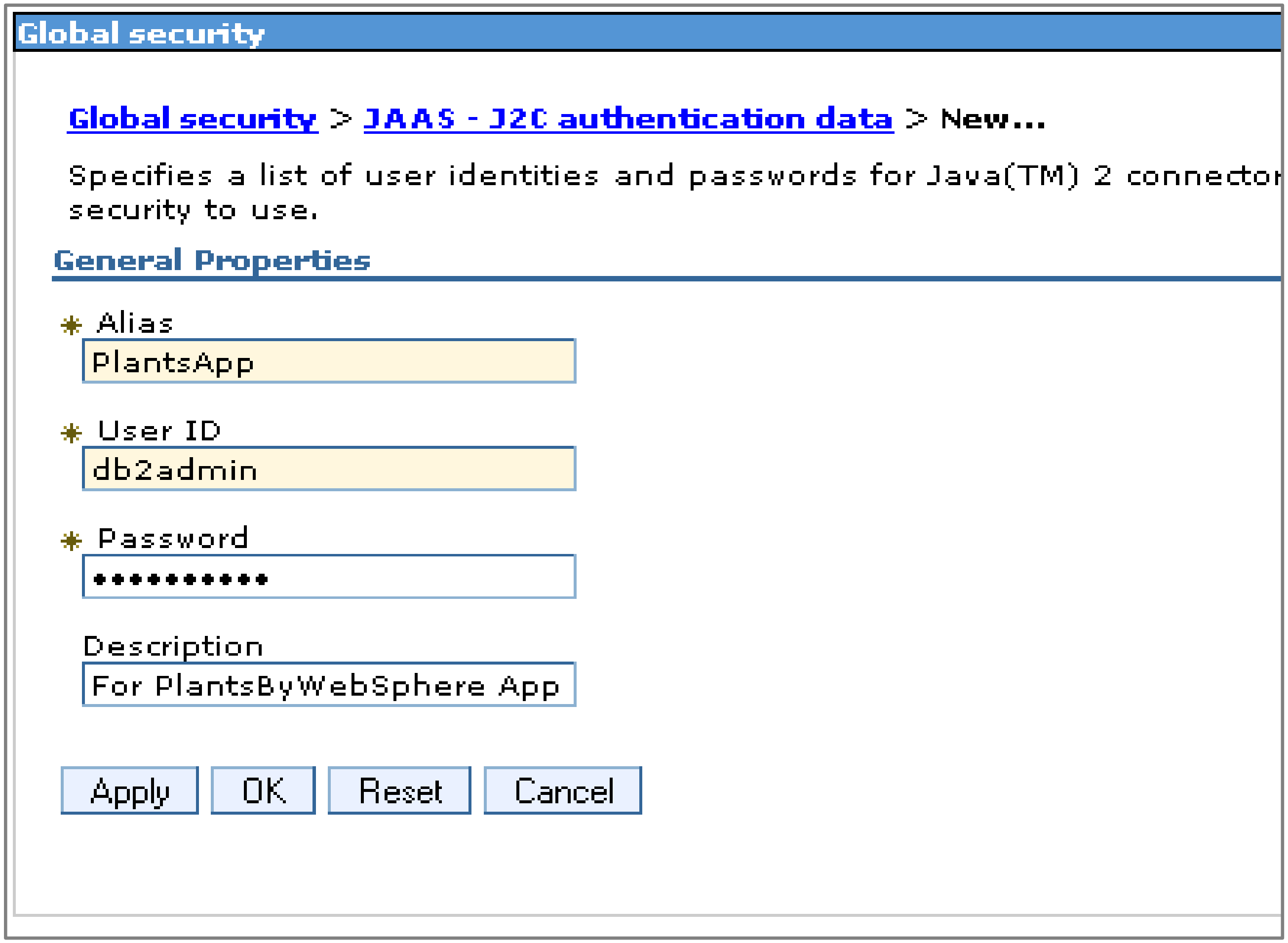
Installation tasks

- Configure the application environment as required
  - Variables, virtual hosts, class path, security
- Configure application resources
  - JDBC provider, data sources, JMS resources, or SIBus, if applicable
- Install the application
  - The default directory that is assigned to hold the EAR file before it is installed is `<profile_root>\<profile>\installableApps`
  - Most often, the application file you receive is an enterprise archive (`.ear`) file
- Manage static content
  - Web server serves files, not in EAR file
  - Leave static content in EAR file

Creating a J2C authentication alias

- Wizard can be reached from many pages in the administrative console
  - **Security > Global Security > Authentication > Java Authentication and Authorization Service > J2C authentication data**

- Provide
  - Alias name: console prefixes name with node name
  - User ID and corresponding password
  - Optional description
  - EJBs, data sources, JMS resources, and SIBus resources use them



Global security

Global security > JAAS - J2C authentication data > New...

Specifies a list of user identities and passwords for Java(TM) 2 connector security to use.

General Properties

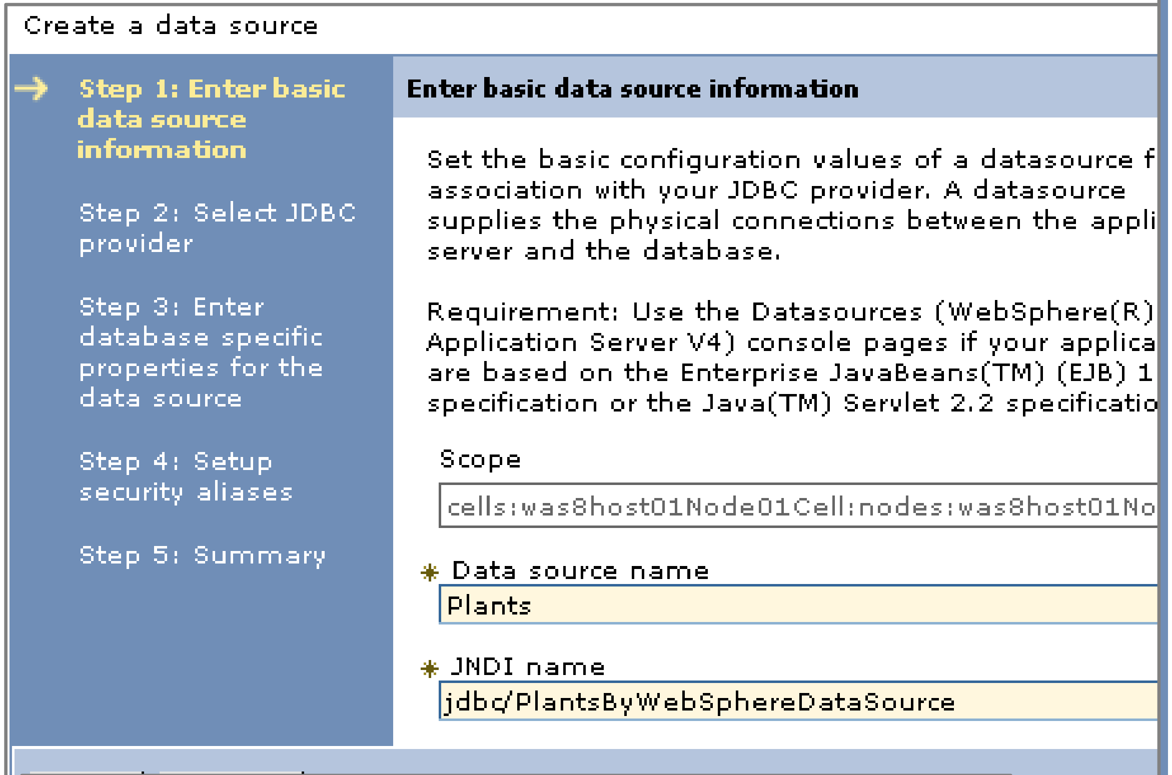
* Alias: PlantsApp

* User ID: db2admin

* Password: ••••••••••

Description: For PlantsByWebSphere App

Creating a data source (1 of 3)



Create a data source

- → **Step 1: Enter basic data source information**
- Step 2: Select JDBC provider
- Step 3: Enter database specific properties for the data source
- Step 4: Setup security aliases
- Step 5: Summary

Enter basic data source information

Set the basic configuration values of a datasource f association with your JDBC provider. A datasource supplies the physical connections between the appli server and the database.

Requirement: Use the Datasources (WebSphere(R) Application Server V4) console pages if your applica are based on the Enterprise JavaBeans(TM) (EJB) 1 specification or the Java(TM) Servlet 2.2 specificatio

Scope

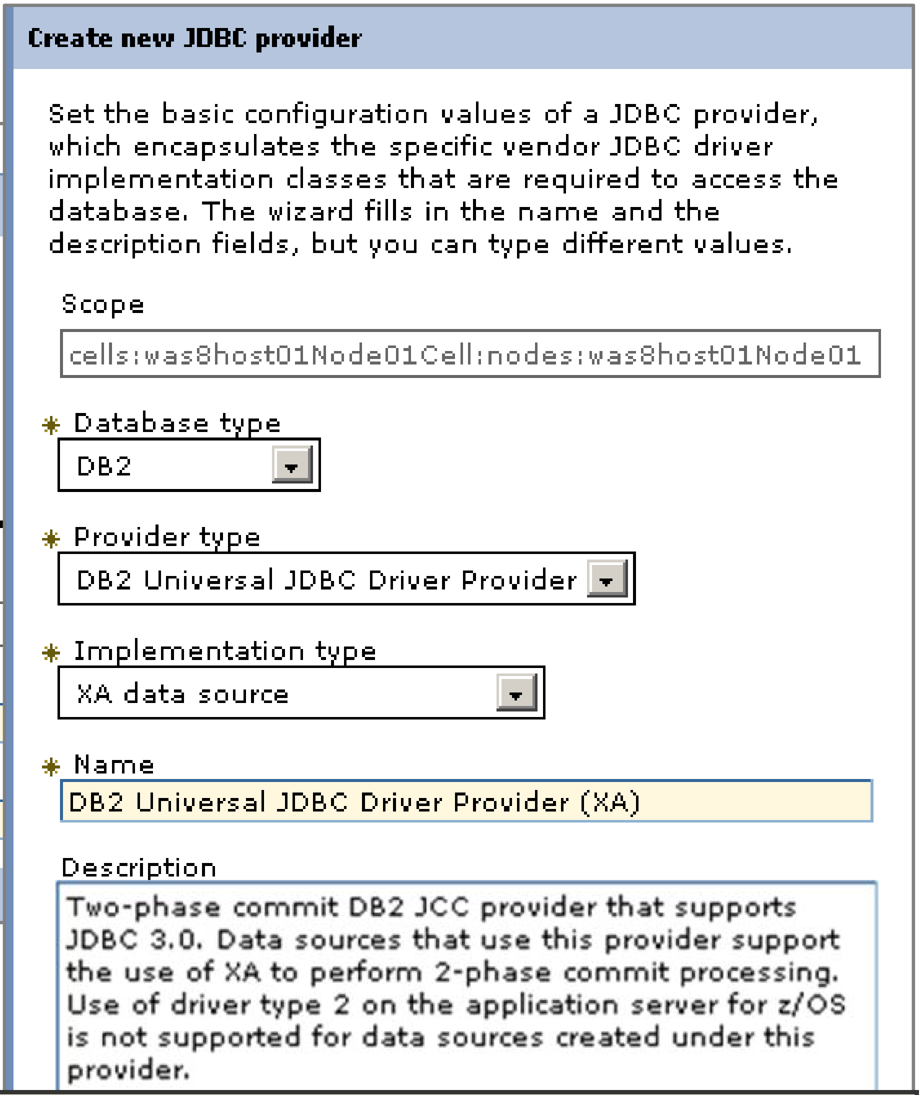
cells:was8host01Node01Cell:nodes:was8host01No

* Data source name

Plants

* JNDI name

jdbc/PlantsByWebSphereDataSource



Create new JDBC provider

Set the basic configuration values of a JDBC provider, which encapsulates the specific vendor JDBC driver implementation classes that are required to access the database. The wizard fills in the name and the description fields, but you can type different values.

Scope

cells:was8host01Node01Cell:nodes:was8host01Node01

* Database type

DB2

* Provider type

DB2 Universal JDBC Driver Provider

* Implementation type

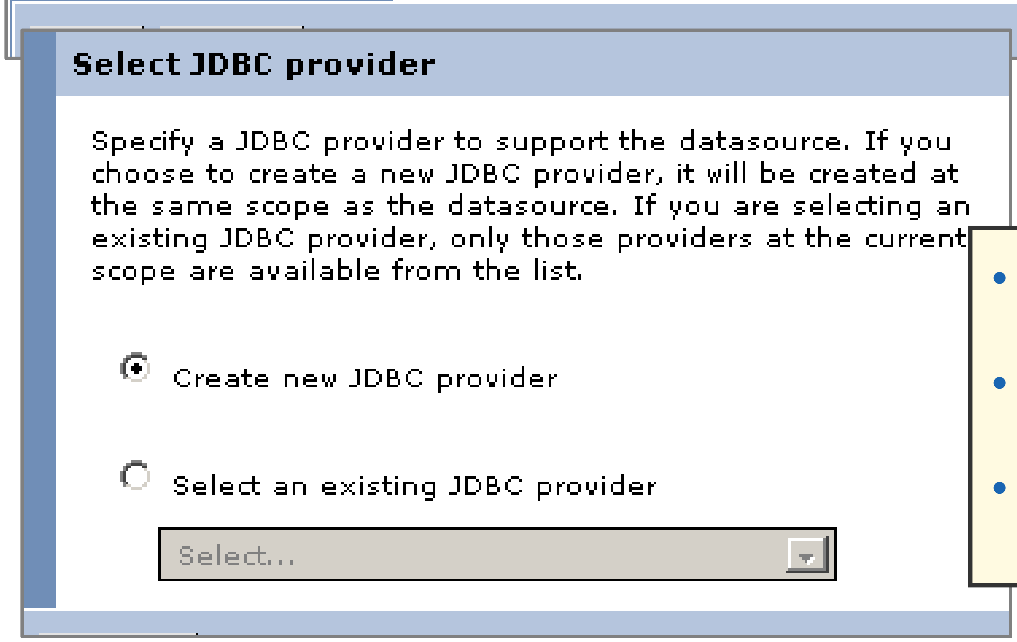
XA data source

* Name

DB2 Universal JDBC Driver Provider (XA)

Description

Two-phase commit DB2 JCC provider that supports JDBC 3.0. Data sources that use this provider support the use of XA to perform 2-phase commit processing. Use of driver type 2 on the application server for z/OS is not supported for data sources created under this provider.



Select JDBC provider

Specify a JDBC provider to support the datasource. If you choose to create a new JDBC provider, it will be created at the same scope as the datasource. If you are selecting an existing JDBC provider, only those providers at the current scope are available from the list.

● Create new JDBC provider

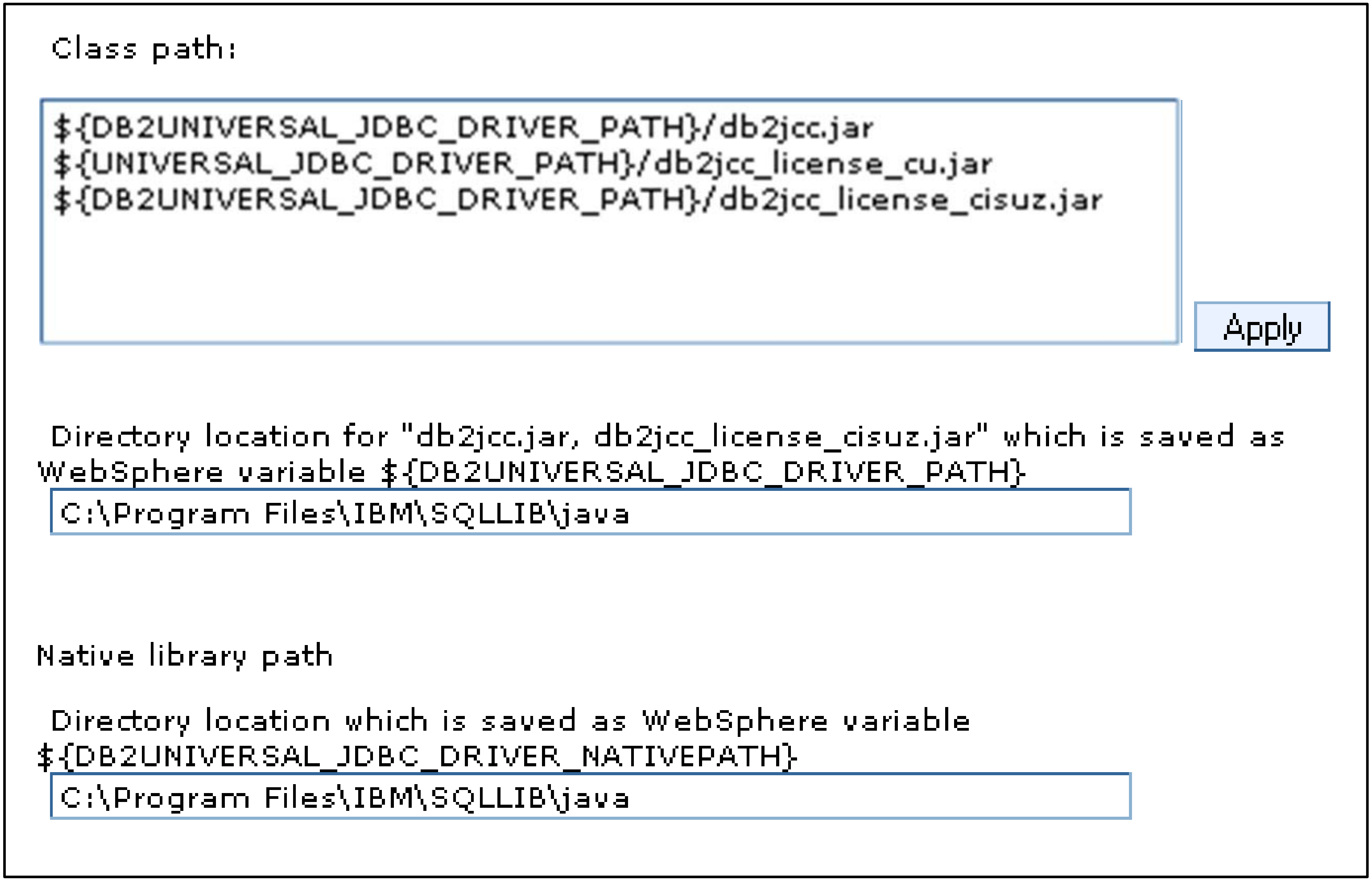
○ Select an existing JDBC provider

Select...

- Create JDBC provider before or while defining data sources
- One JDBC provider is needed for each database driver type
- JDBC providers can be defined at cell, node, server, or application scope (in an enhanced EAR file)

Creating a data source (2 of 3)

- JDBC driver paths can be defined in the wizard



Class path:

```
${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc.jar
${UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cu.jar
${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cisuz.jar
```

Directory location for "db2jcc.jar, db2jcc_license_cisuz.jar" which is saved as WebSphere variable ${DB2UNIVERSAL_JDBC_DRIVER_PATH}

C:\Program Files\IBM\SQLLIB\java

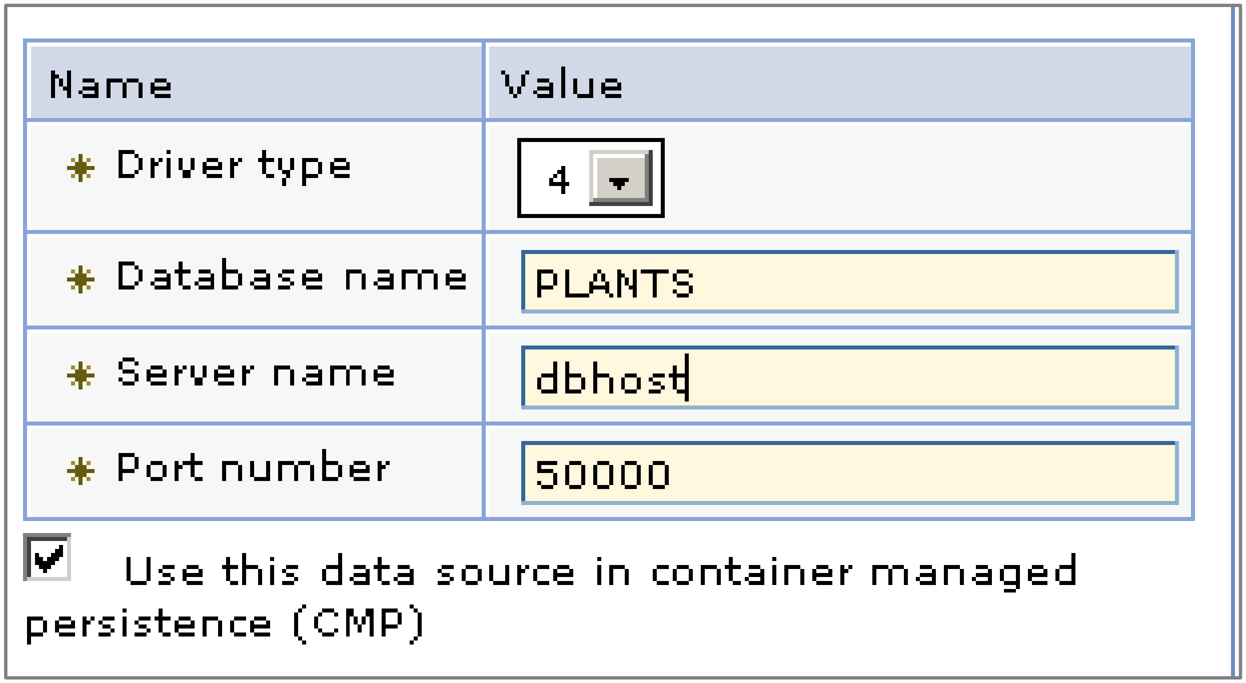
Native library path

Directory location which is saved as WebSphere variable ${DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH}

C:\Program Files\IBM\SQLLIB\java

Creating a data source (3 of 3)

- Provide the database-specific parameters:
  - Driver type
  - Database name
  - Database server name and communication port number
- Choose whether data source is going to be used with CMP beans

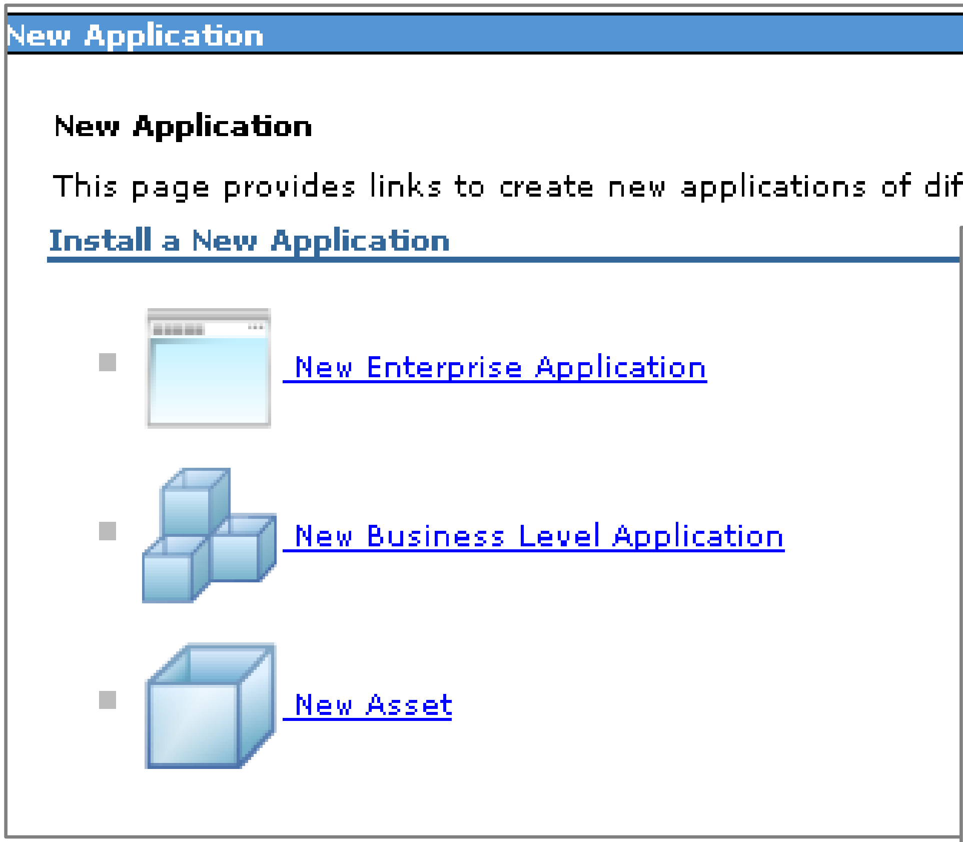


Name	Value
* Driver type	4
* Database name	PLANTS
* Server name	dbhost
* Port number	50000

☑ Use this data source in container managed persistence (CMP)

Installing a new application

- Select **Applications > New Application > New Enterprise Application**



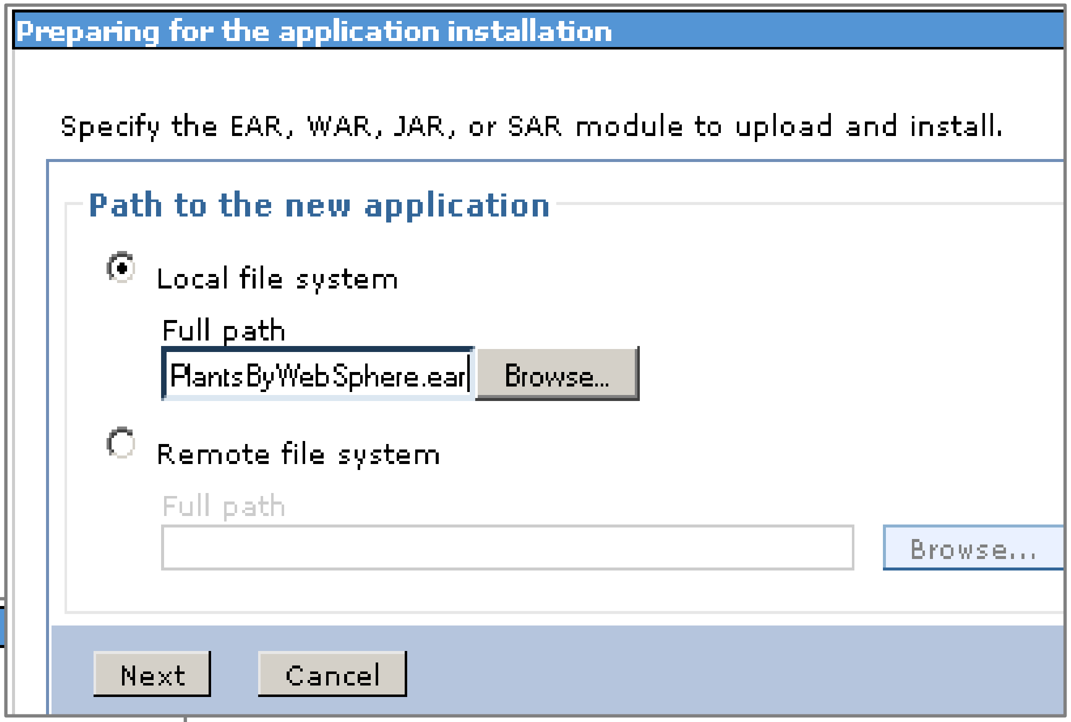
New Application

New Application

This page provides links to create new applications of different types.

Install a New Application

- New Enterprise Application
- New Business Level Application
- New Asset



Preparing for the application installation

Specify the EAR, WAR, JAR, or SAR module to upload and install.

Path to the new application

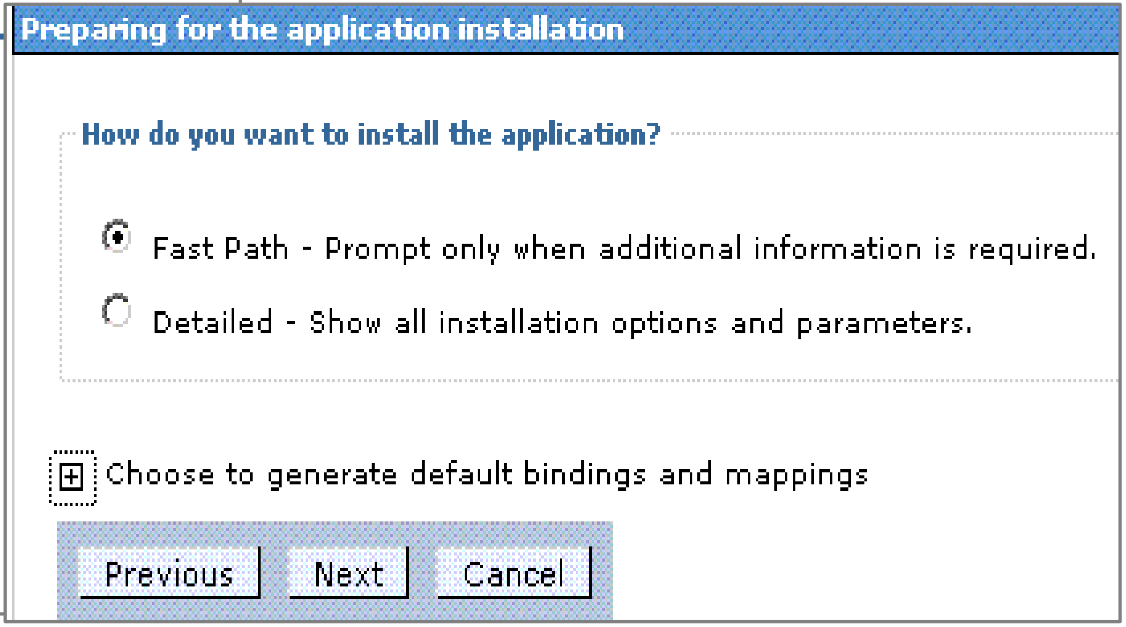
● Local file system

Full path: PlantsByWebSphere.ear [Browse...]

○ Remote file system

Full path: [Browse...]

[Next] [Cancel]



Preparing for the application installation

How do you want to install the application?

● Fast Path - Prompt only when additional information is required.

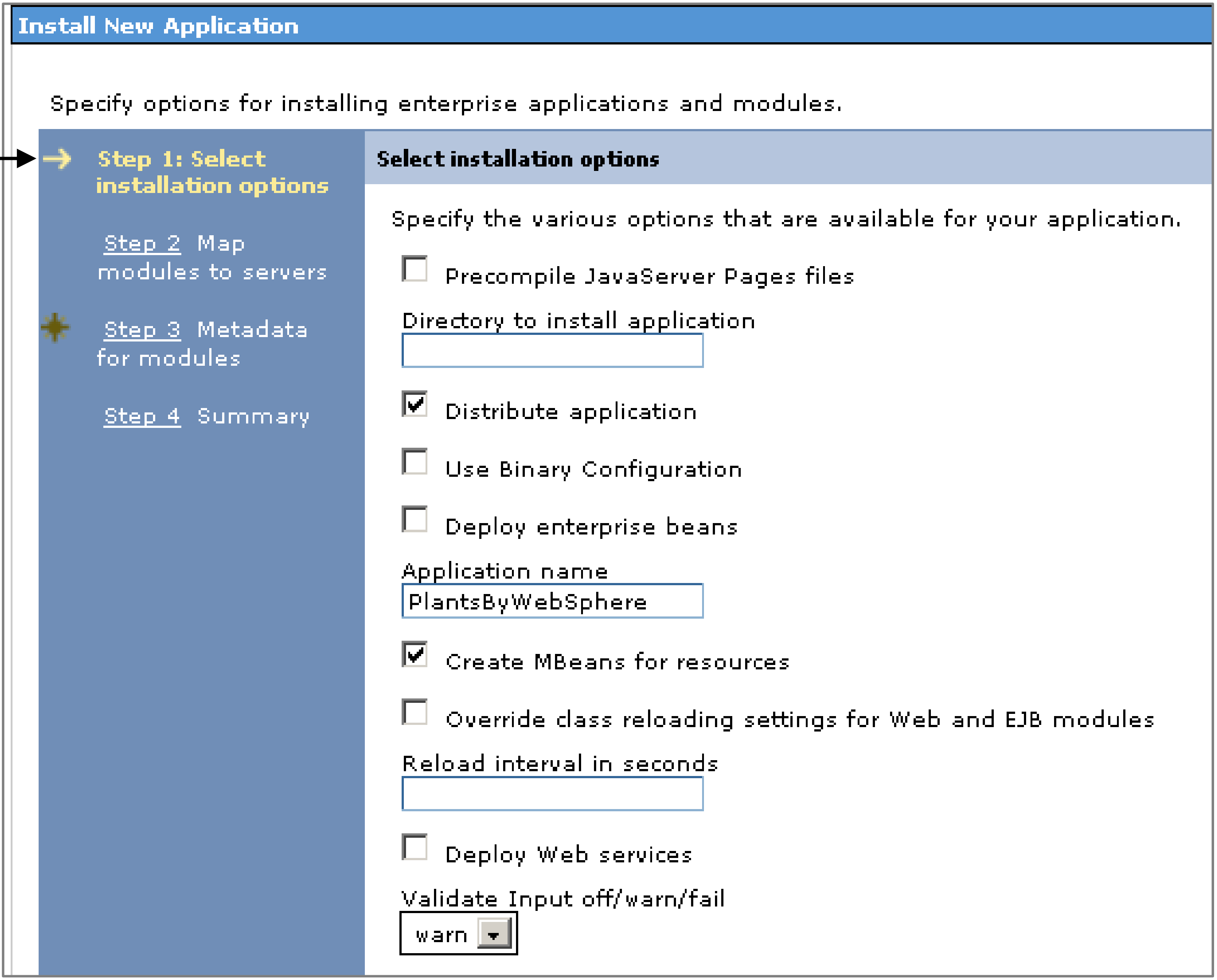
○ Detailed - Show all installation options and parameters.

⊞ Choose to generate default bindings and mappings

[Previous] [Next] [Cancel]

Example of fast path installation

Step 1: Select installation options



Install New Application

Specify options for installing enterprise applications and modules.

→ Step 1: Select installation options

Step 2 Map modules to servers

Step 3 Metadata for modules

Step 4 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
PlantsByWebSphere

☑ Create MBeans for resources

□ Override class reloading settings for Web and EJB modules

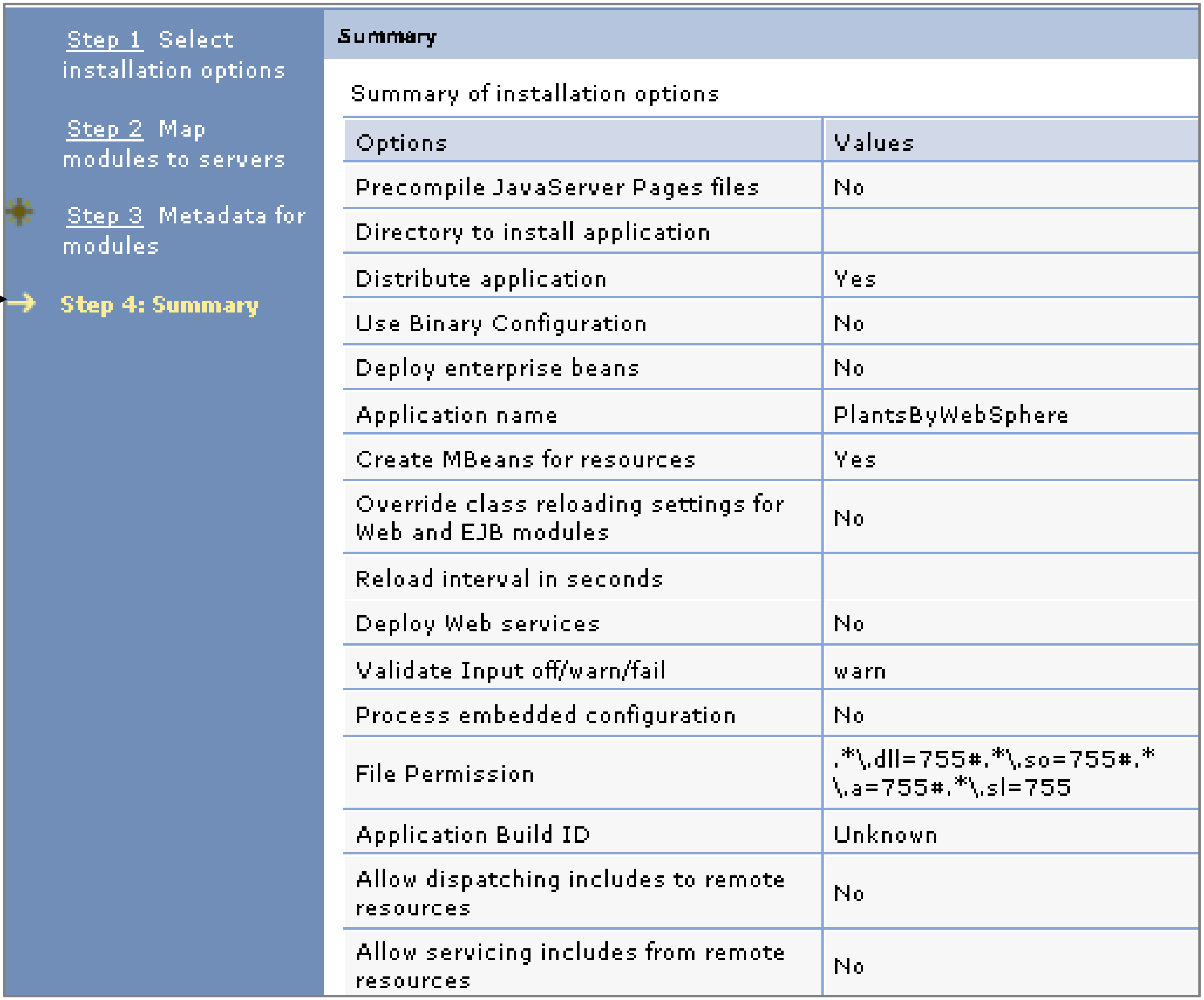
Reload interval in seconds

□ Deploy Web services

Validate Input off/warn/fail
warn

Example of detailed installation

Summary step



Step 1 Select installation options

Step 2 Map modules to servers

Step 3 Metadata for modules

→ Step 4: Summary

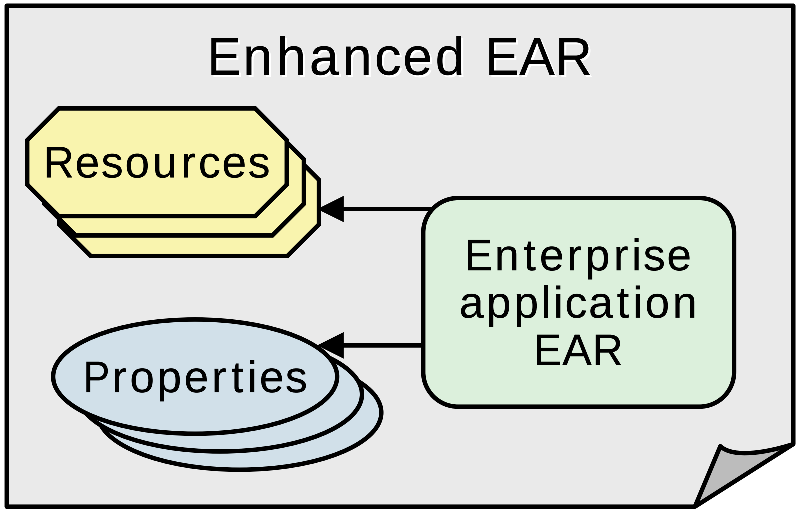
Summary

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	PlantsByWebSphere
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

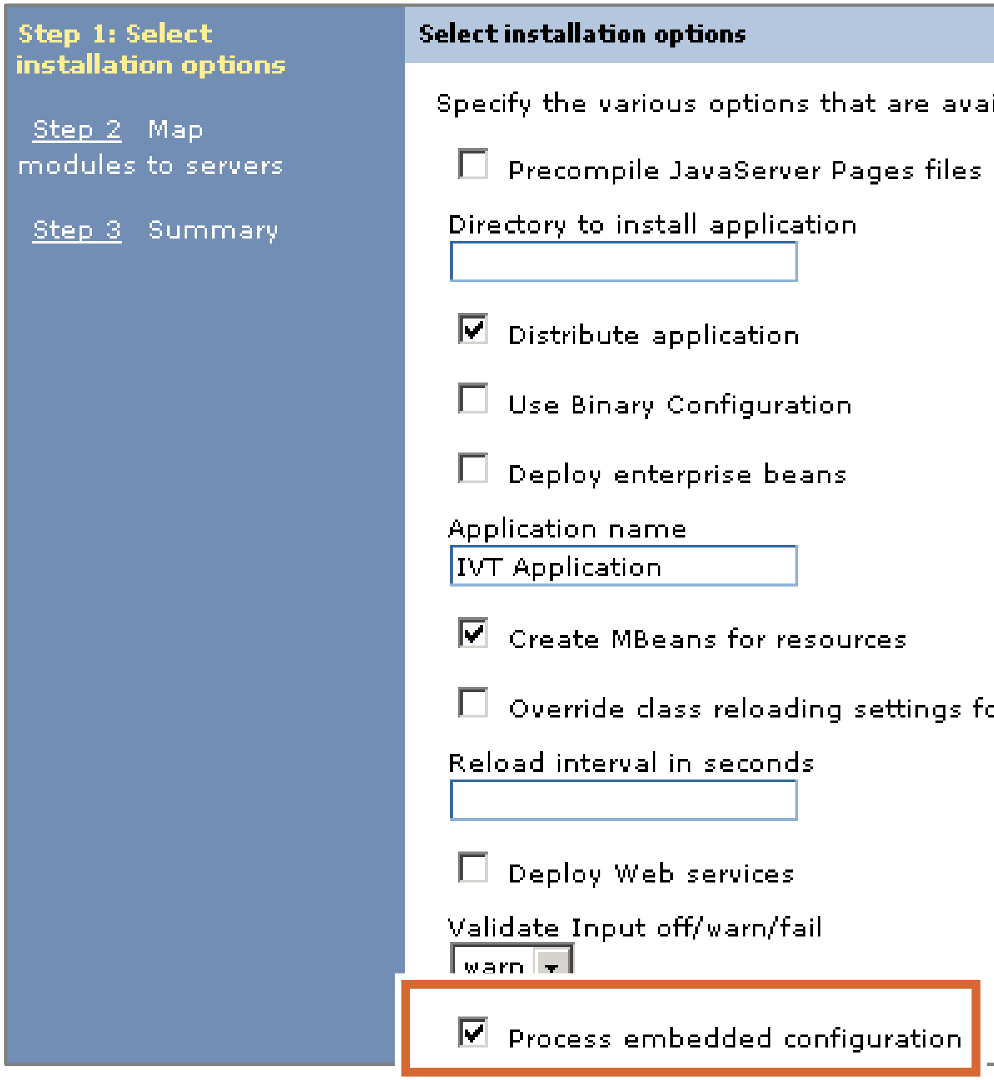
Enhanced EAR

- Enterprise archive that contains Java EE artifacts plus resource information necessary to install on WebSphere Application Server
  - JDBC resources (data sources)
  - Class loader
  - JAAS authentication aliases
  - Shared libraries
  - Virtual host information
- Benefits: improved productivity
  - Application resources and properties come with the application
  - Application installation process creates the necessary resources within the server or cluster
  - Moving application from one server to another also moves the resources



- Support integrated with the IBM Rational Development and Assembly and Deployment tools
  - Found on Deployment page of application deployment descriptor
- **Warning**: Can possibly cause problems if unintended application scoped resources are used in production
  - Enhancements can be removed or ignored during application installation

Removing enhancements



Step 1: Select installation options

Step 2 Map modules to servers

Step 3 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: IVT Application

☑ Create MBeans for resources

☐ Override class reloading settings for Web and EJB modules

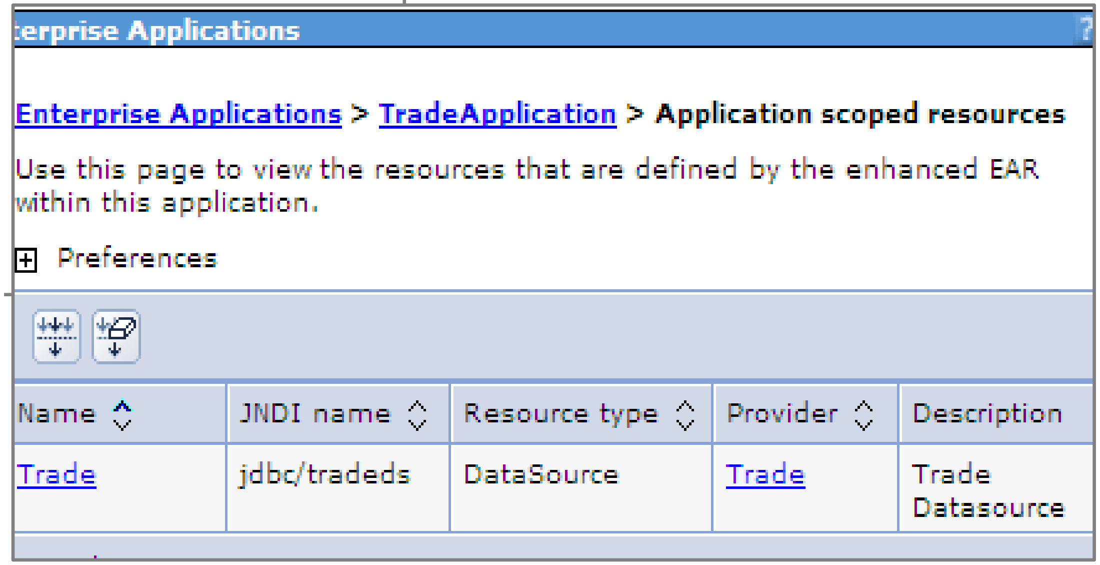
Reload interval in seconds

☐ Deploy Web services

Validate Input off/warn/fail: warn

☑ Process embedded configuration

- Resources can be ignored
  - Remove enhancements from EAR before deploying (preferred)
  - Clear **Process embedded configurations**
  - Is prechecked only if there are enhancements



terprise Applications

Enterprise Applications > TradeApplication > **Application scoped resources**

Use this page to view the resources that are defined by the enhanced EAR within this application.

⊞ Preferences

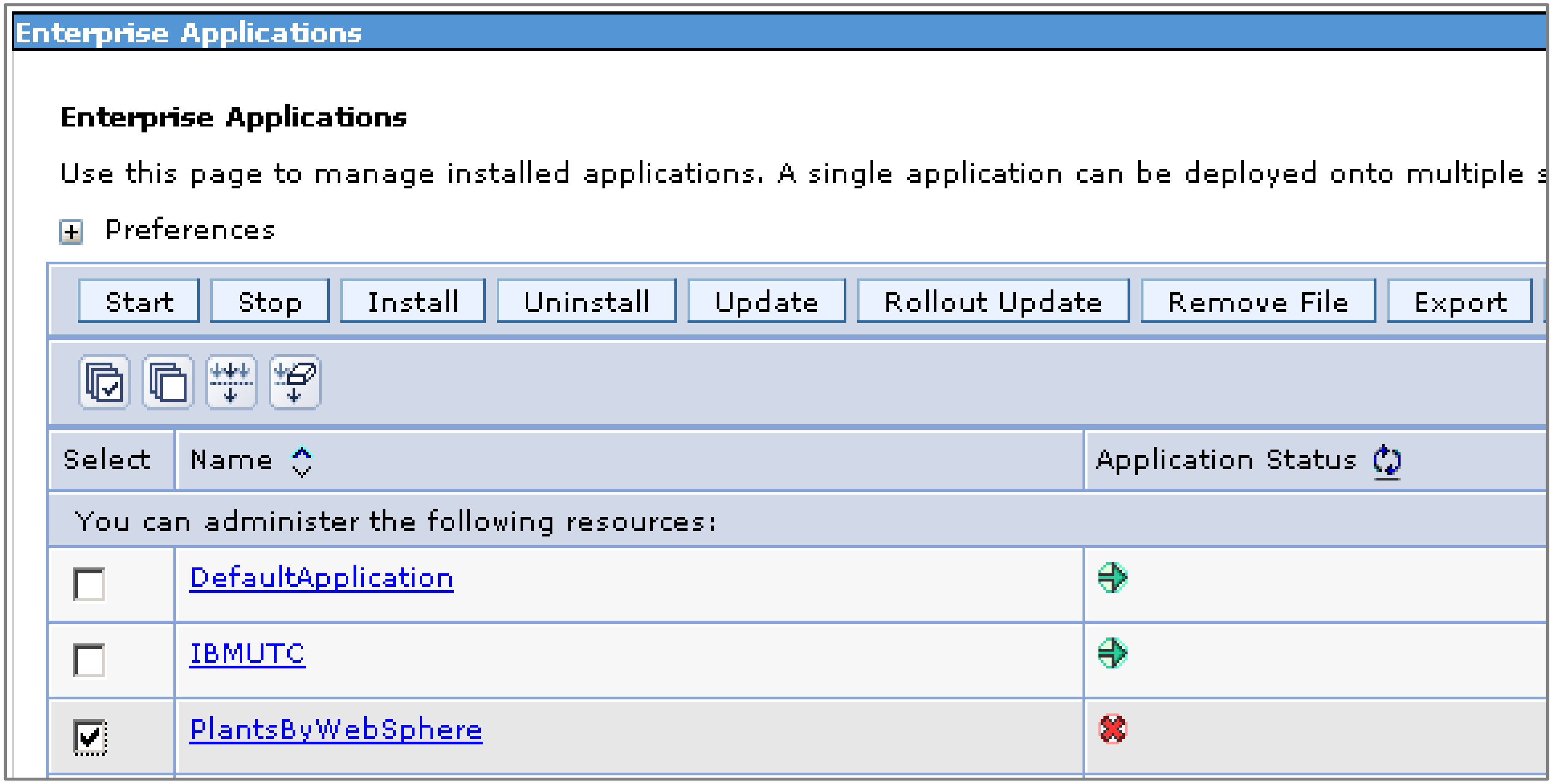
Name	JNDI name	Resource type	Provider	Description
Trade	jdbc/tradeds	DataSource	Trade	Trade Datasource

- Resources can be viewed, but not through the normal screens
  - Click **Application scoped resources** under the enterprise application

Application settings and interaction

Starting an application

- After the application is installed, you can select from a number of options to manage the application
  - Click **Applications > Application types > WebSphere enterprise applications**



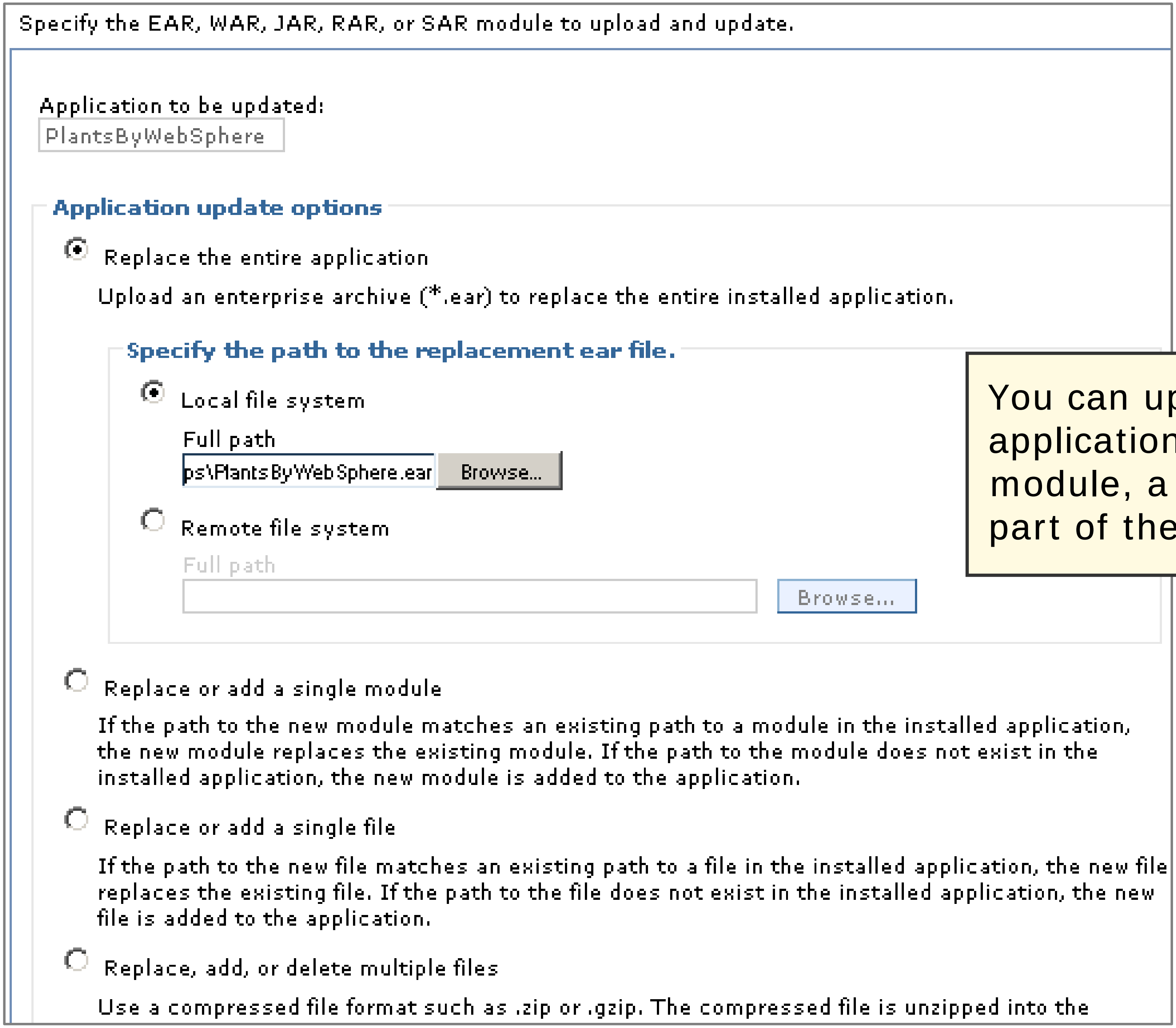
Enterprise Applications

Enterprise Applications

Use this page to manage installed applications. A single application can be deployed onto multiple s

Select	Name	Application Status
You can administer the following resources:		
□	DefaultApplication	➔
□	IBMUTC	➔
☑	PlantsByWebSphere	☒

Application update



Specify the EAR, WAR, JAR, RAR, or SAR module to upload and update.

Application to be updated:
PlantsByWebSphere

Application update options

● Replace the entire application

Upload an enterprise archive (*.ear) to replace the entire installed application.

Specify the path to the replacement ear file.

● Local file system

Full path
ps\PlantsByWebSphere.ear

○ Remote file system

Full path

○ Replace or add a single module

If the path to the new module matches an existing path to a module in the installed application, the new module replaces the existing module. If the path to the module does not exist in the installed application, the new module is added to the application.

○ Replace or add a single file

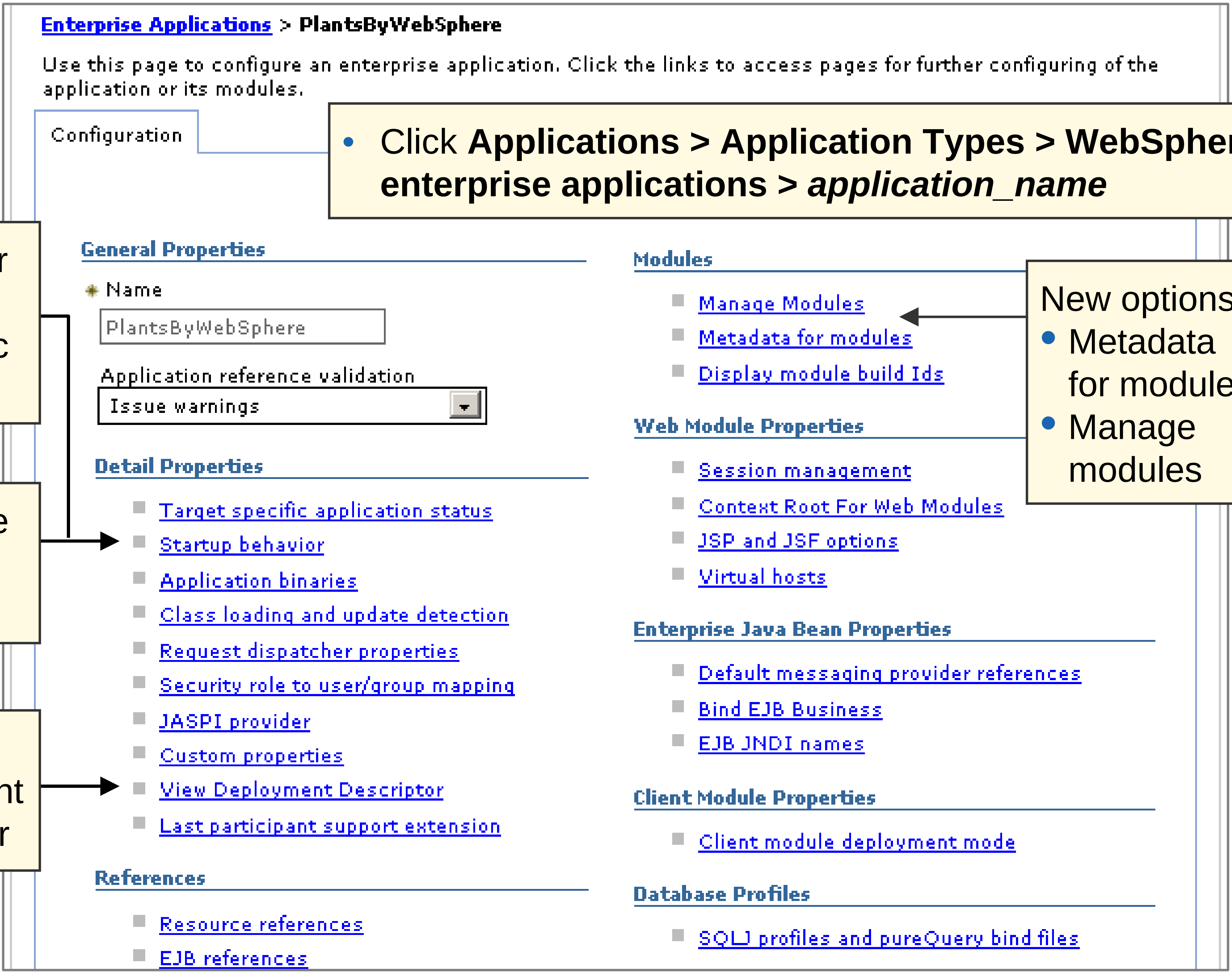
If the path to the new file matches an existing path to a file in the installed application, the new file replaces the existing file. If the path to the file does not exist in the installed application, the new file is added to the application.

○ Replace, add, or delete multiple files

Use a compressed file format such as .zip or .gzip. The compressed file is unzipped into the

You can update the full application, a single module, a single file, or part of the application

Other application configuration settings



Enterprise Applications > PlantsByWebSphere

Use this page to configure an enterprise application. Click the links to access pages for further configuring of the application or its modules.

Configuration

General Properties

* Name

PlantsByWebSphere

Application reference validation

Issue warnings

Detail Properties

- Target specific application status
- Startup behavior
- Application binaries
- Class loading and update detection
- Request dispatcher properties
- Security role to user/group mapping
- JASPI provider
- Custom properties
- View Deployment Descriptor
- Last participant support extension

References

- Resource references
- EJB references

Modules

- Manage Modules
- Metadata for modules
- Display module build Ids

Web Module Properties

- Session management
- Context Root For Web Modules
- JSP and JSF options
- Virtual hosts

Enterprise Java Bean Properties

- Default messaging provider references
- Bind EJB Business
- EJB JNDI names

Client Module Properties

- Client module deployment mode

Database Profiles

- SQLJ profiles and pureQuery bind files

- Click **Applications > Application Types > WebSphere enterprise applications >** ***application_name***

Enable or disable automatic start

Configure startup behavior

View deployment descriptor

New options:

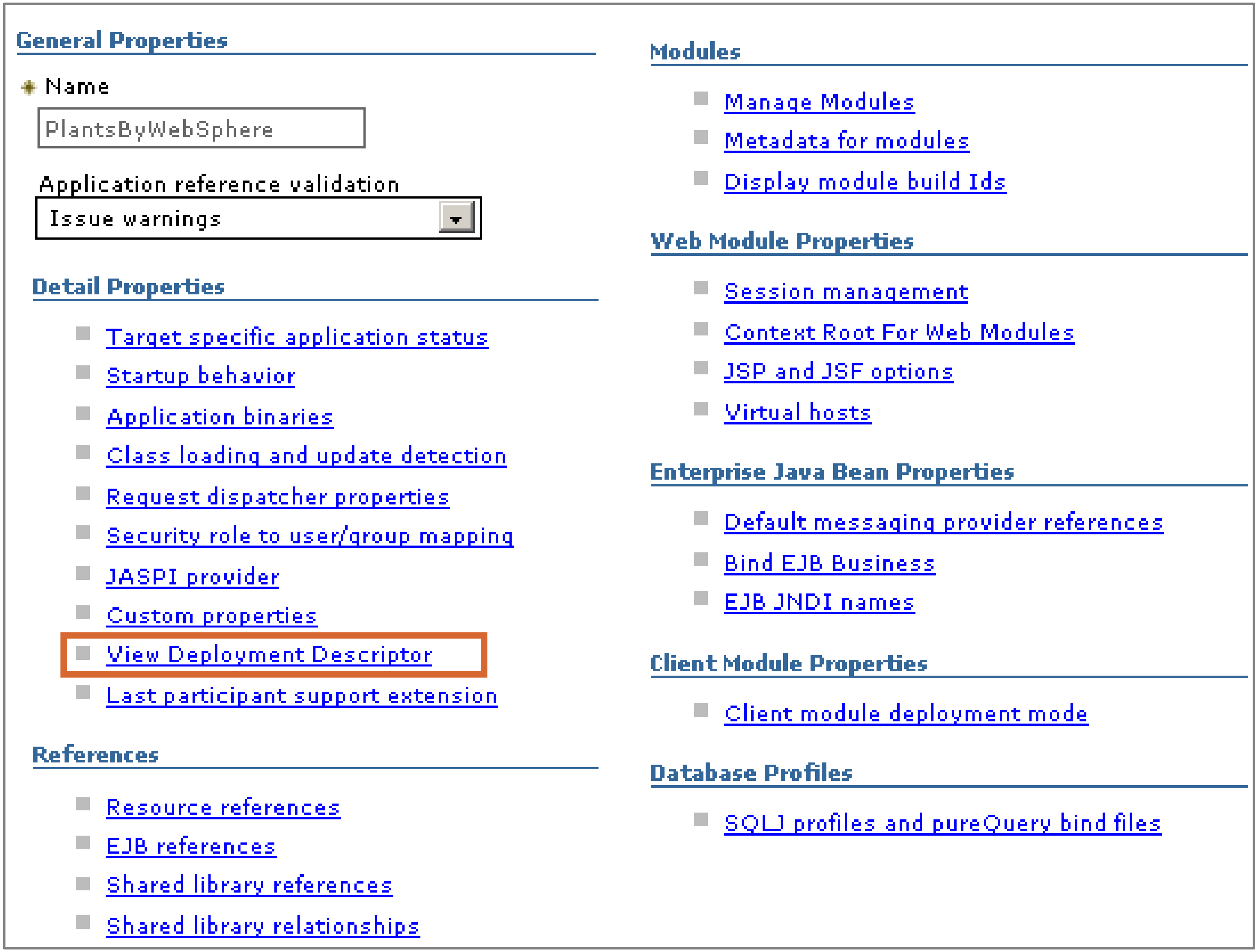
- Metadata for modules
- Manage modules

Application startup behavior and auto start

- Start up behavior of an application
  - The values set affect how quickly an application starts and what occurs when an application starts
  - Click **Applications > Application Types > WebSphere enterprise applications >** ***application_name*** **> Startup behavior** in the console navigation tree to configure startup behavior settings

- Automatic starting of an application
  - By default, an installed application starts automatically when the server starts on which the application is deployed
  - Click **Applications > Application Types > WebSphere enterprise applications >** ***application_name*** **> Target specific application status** to configure auto startup

View the application deployment descriptor



General Properties

* Name

PlantsByWebSphere

Application reference validation

Issue warnings

Detail Properties

- Target specific application status
- Startup behavior
- Application binaries
- Class loading and update detection
- Request dispatcher properties
- Security role to user/group mapping
- JASPI provider
- Custom properties
- View Deployment Descriptor
- Last participant support extension

References

- Resource references
- EJB references
- Shared library references
- Shared library relationships

Modules

- Manage Modules
- Metadata for modules
- Display module build Ids

Web Module Properties

- Session management
- Context Root For Web Modules
- JSP and JSF options
- Virtual hosts

Enterprise Java Bean Properties

- Default messaging provider references
- Bind EJB Business
- EJB JNDI names

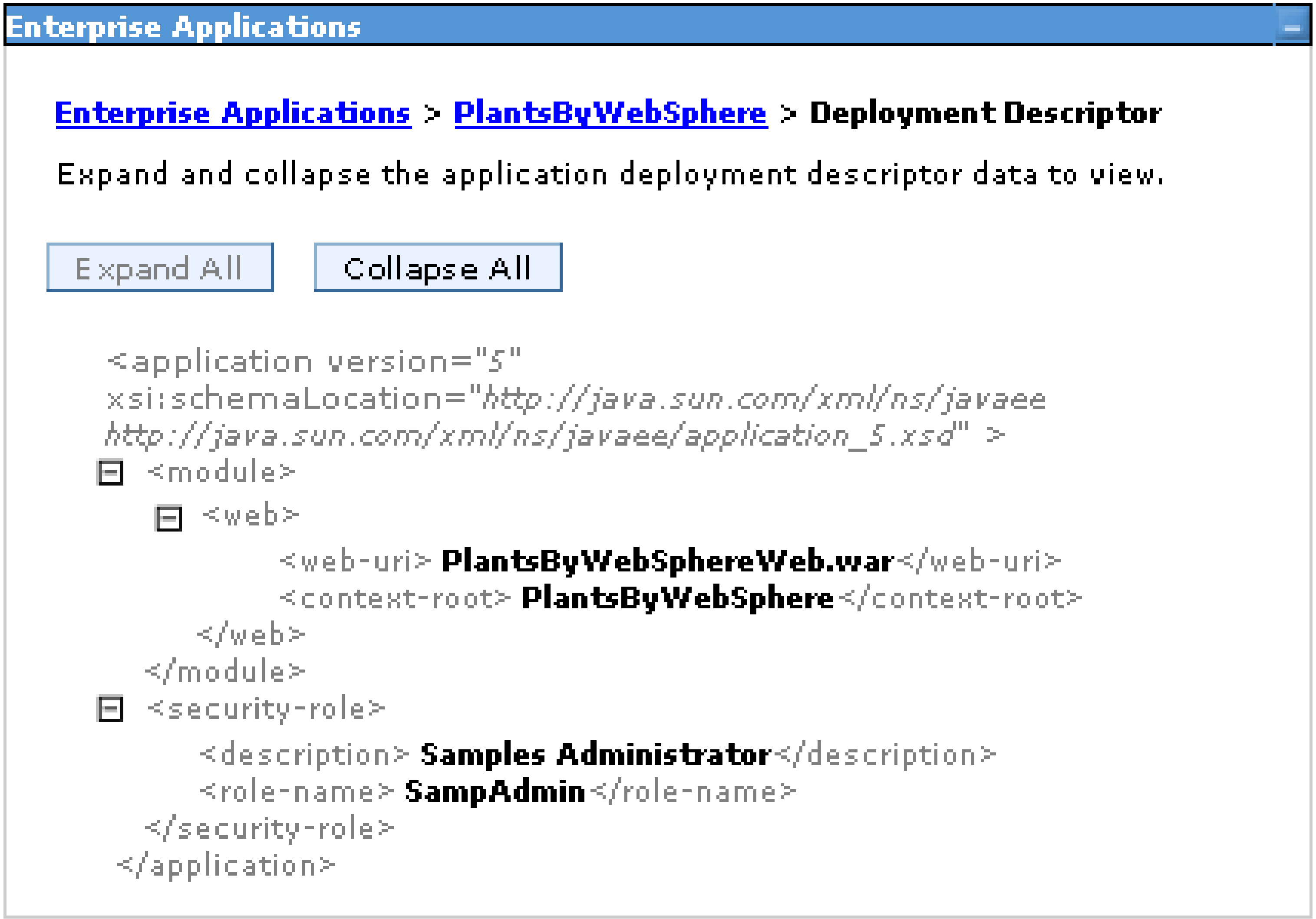
Client Module Properties

- Client module deployment mode

Database Profiles

- SQLJ profiles and pureQuery bind files

The application deployment descriptor



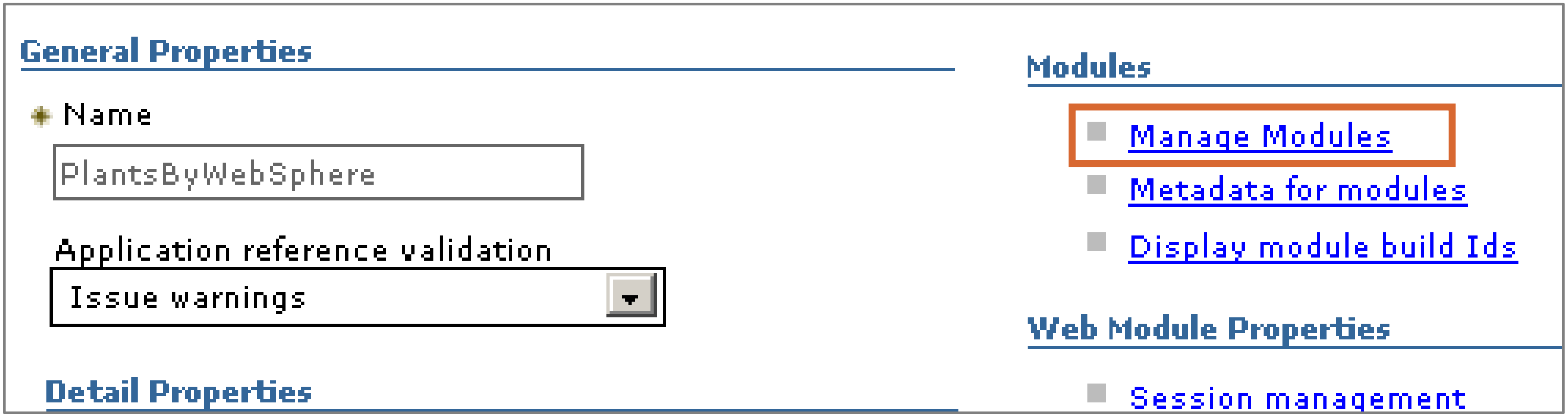
Enterprise Applications > PlantsByWebSphere > Deployment Descriptor

Expand and collapse the application deployment descriptor data to view.

```
<application version="5"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/application_5.xsd" >
⊟ <module>
   ⊟ <web>
         <web-uri> PlantsByWebSphereWeb.war</web-uri>
         <context-root> PlantsByWebSphere</context-root>
     </web>
  </module>
⊟ <security-role>
     <description> Samples Administrator</description>
     <role-name> SampAdmin</role-name>
  </security-role>
</application>
```

Manage modules (1 of 4)

- To view the web or EJB deployment descriptors for an enterprise application
  - Click **Manage Modules**



General Properties

* Name

PlantsByWebSphere

Application reference validation

Issue warnings

Detail Properties

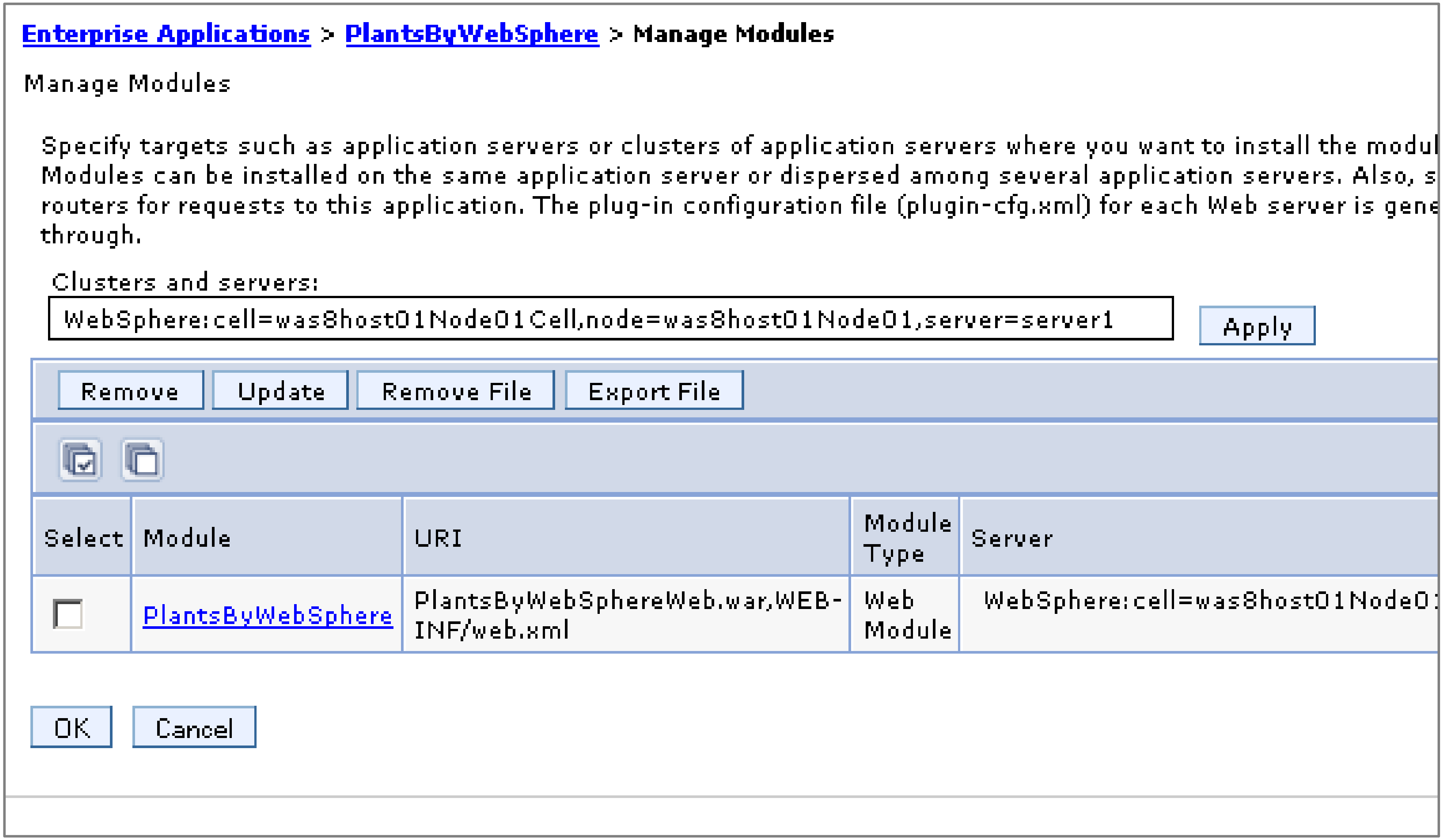
Modules

- Manage Modules
- Metadata for modules
- Display module build Ids

Web Module Properties

- Session management

Manage modules (2 of 4)



Enterprise Applications > PlantsByWebSphere > Manage Modules

Manage Modules

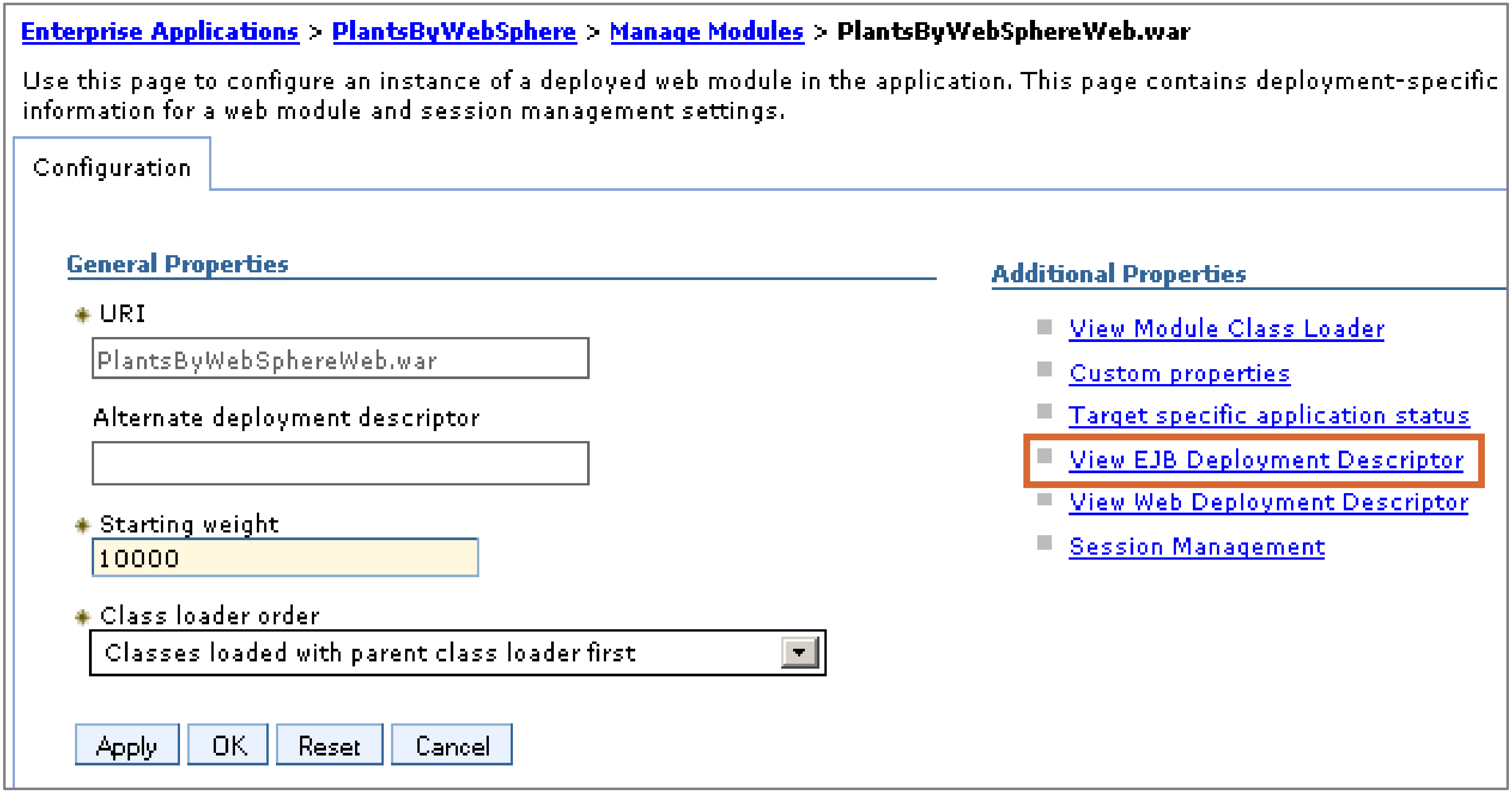
Specify targets such as application servers or clusters of application servers where you want to install the modul
Modules can be installed on the same application server or dispersed among several application servers. Also, s
routers for requests to this application. The plug-in configuration file (plugin-cfg.xml) for each Web server is gene
through.

Clusters and servers:

WebSphere:cell=was8host01Node01Cell,node=was8host01Node01,server=server1

Select	Module	URI	Module Type	Server
□	PlantsByWebSphere	PlantsByWebSphereWeb.war,WEB-INF/web.xml	Web Module	WebSphere:cell=was8host01Node0

Manage modules (3 of 4)



Enterprise Applications > PlantsByWebSphere > Manage Modules > PlantsByWebSphereWeb.war

Use this page to configure an instance of a deployed web module in the application. This page contains deployment-specific information for a web module and session management settings.

Configuration

General Properties

* URI

PlantsByWebSphereWeb.war

Alternate deployment descriptor

* Starting weight

10000

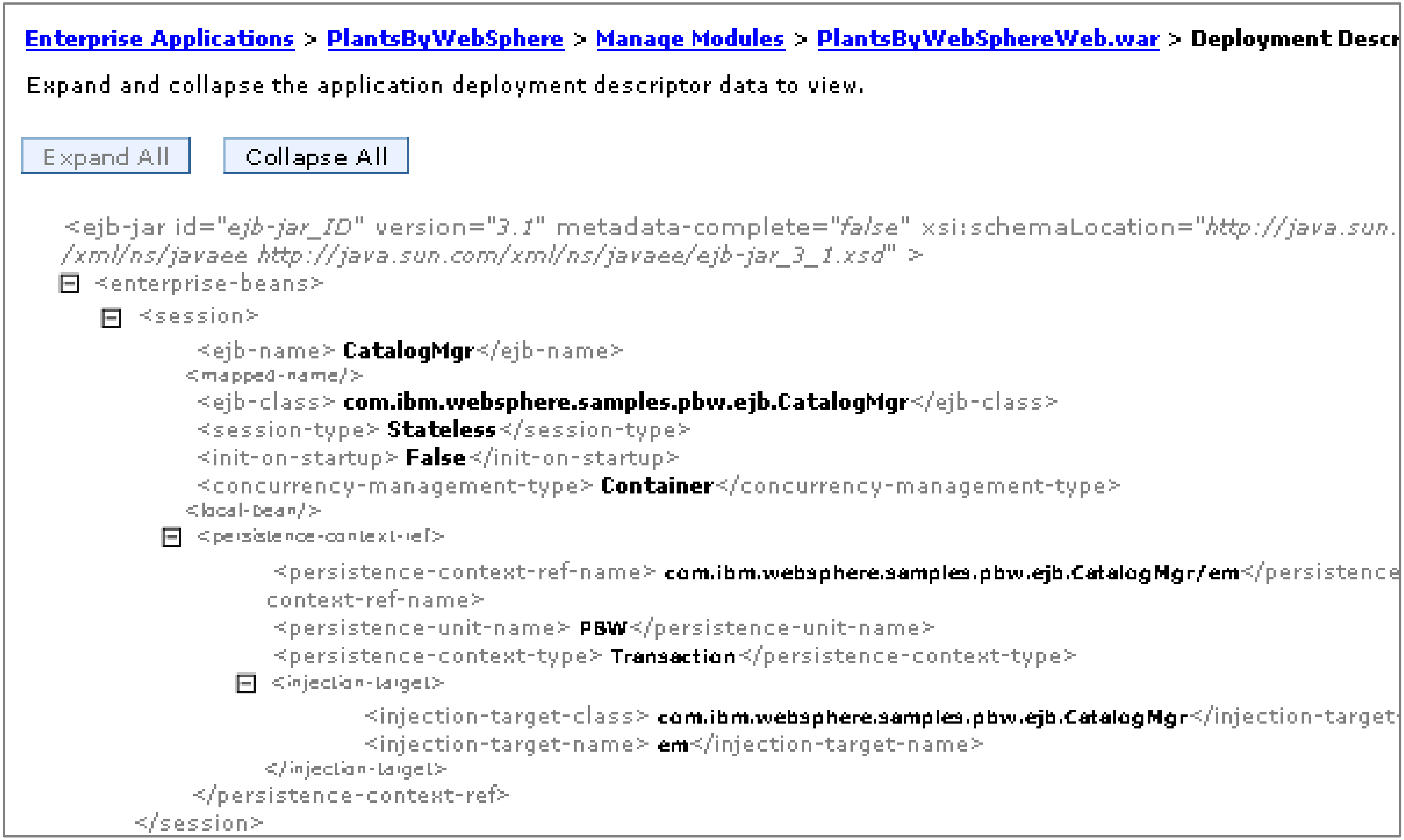
* Class loader order

Classes loaded with parent class loader first

Additional Properties

- View Module Class Loader
- Custom properties
- Target specific application status
- View EJB Deployment Descriptor
- View Web Deployment Descriptor
- Session Management

Manage modules (4 of 4)



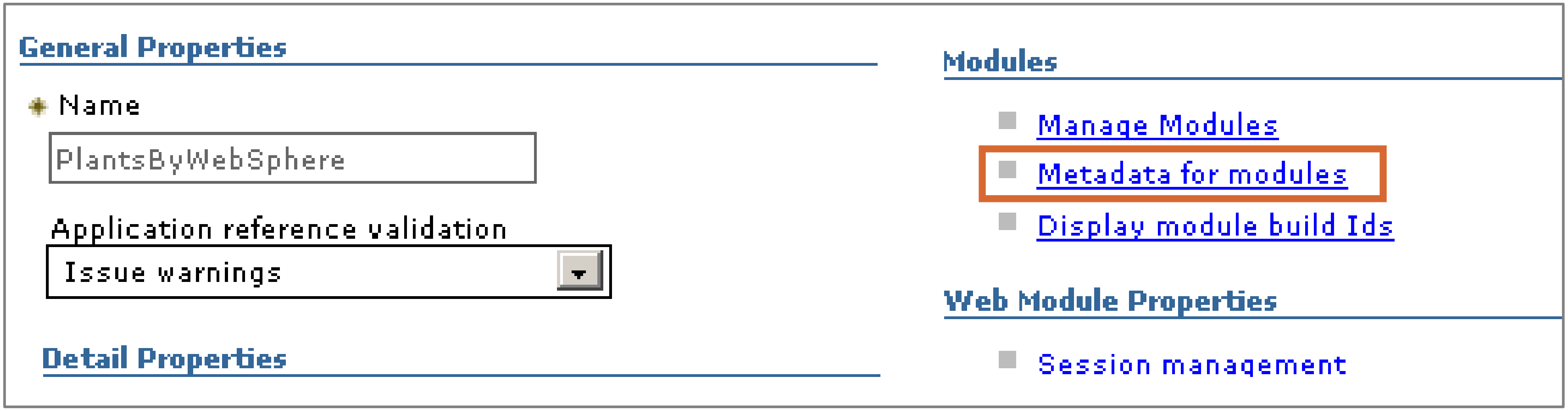
Enterprise Applications > PlantsByWebSphere > Manage Modules > PlantsByWebSphereWeb.war > Deployment Descr

Expand and collapse the application deployment descriptor data to view.

```
<ejb-jar id="ejb-jar_ID" version="3.1" metadata-complete="false" xsi:schemaLocation="http://java.sun.
/xml/ns/javaee http://java.sun.com/xml/ns/javaee/ejb-jar_3_1.xsd" >
⊟ <enterprise-beans>
   ⊟ <session>
        <ejb-name> CatalogMgr</ejb-name>
       <mapped-name/>
        <ejb-class> com.ibm.websphere.samples.pbw.ejb.CatalogMgr</ejb-class>
        <session-type> Stateless</session-type>
        <init-on-startup> False</init-on-startup>
        <concurrency-management-type> Container</concurrency-management-type>
       <local-bean/>
      ⊟ <persistence-context-ref>
           <persistence-context-ref-name> com.ibm.websphere.samples.pbw.ejb.CatalogMgr/em</persistence
           context-ref-name>
           <persistence-unit-name> PBW</persistence-unit-name>
           <persistence-context-type> Transaction</persistence-context-type>
         ⊟ <injection-target>
              <injection-target-class> com.ibm.websphere.samples.pbw.ejb.CatalogMgr</injection-target
              <injection-target-name> em</injection-target-name>
           </injection-target>
       </persistence-context-ref>
   </session>
```

Metadata for modules (1 of 2)

- This option enables you to either allow or ignore metadata that is coming from annotations in source code



General Properties

* Name

PlantsByWebSphere

Application reference validation

Issue warnings

Detail Properties

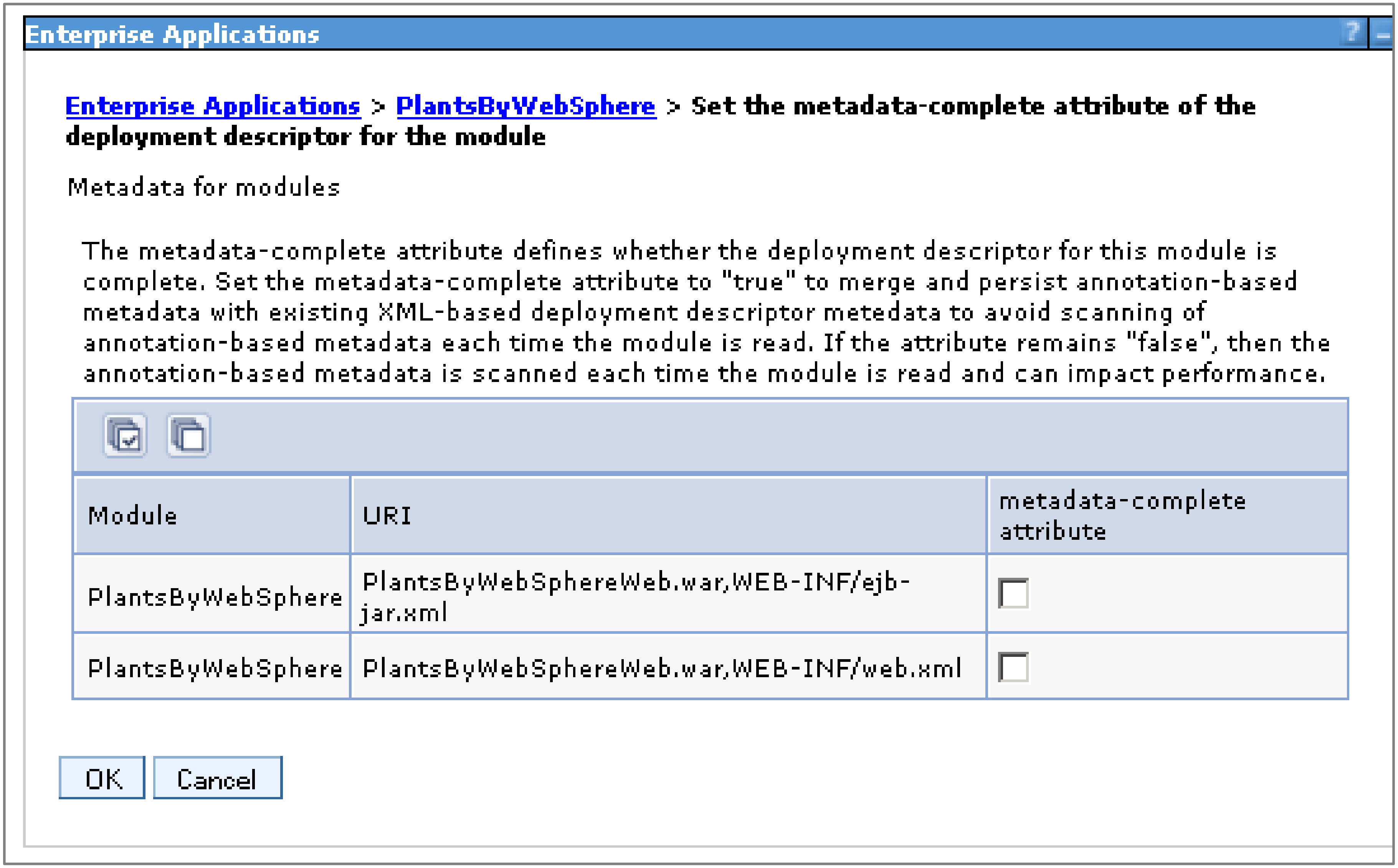
Modules

- Manage Modules
- Metadata for modules
- Display module build Ids

Web Module Properties

- Session management

Metadata for modules (2 of 2)



Enterprise Applications > PlantsByWebSphere > Set the metadata-complete attribute of the deployment descriptor for the module

Metadata for modules

The metadata-complete attribute defines whether the deployment descriptor for this module is complete. Set the metadata-complete attribute to "true" to merge and persist annotation-based metadata with existing XML-based deployment descriptor metedata to avoid scanning of annotation-based metadata each time the module is read. If the attribute remains "false", then the annotation-based metadata is scanned each time the module is read and can impact performance.

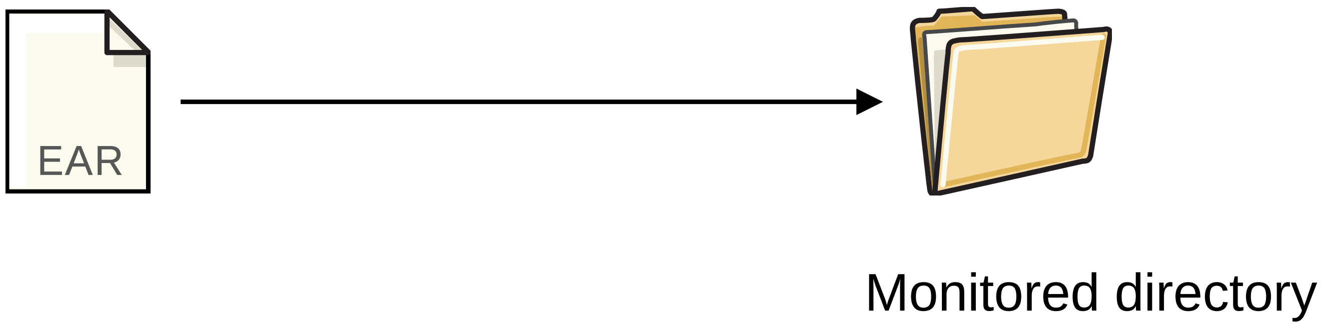
Module	URI	metadata-complete attribute
PlantsByWebSphere	PlantsByWebSphereWeb.war,WEB-INF/ejb-jar.xml	□
PlantsByWebSphere	PlantsByWebSphereWeb.war,WEB-INF/web.xml	□

Monitored directory

Overview of monitored directory deployment

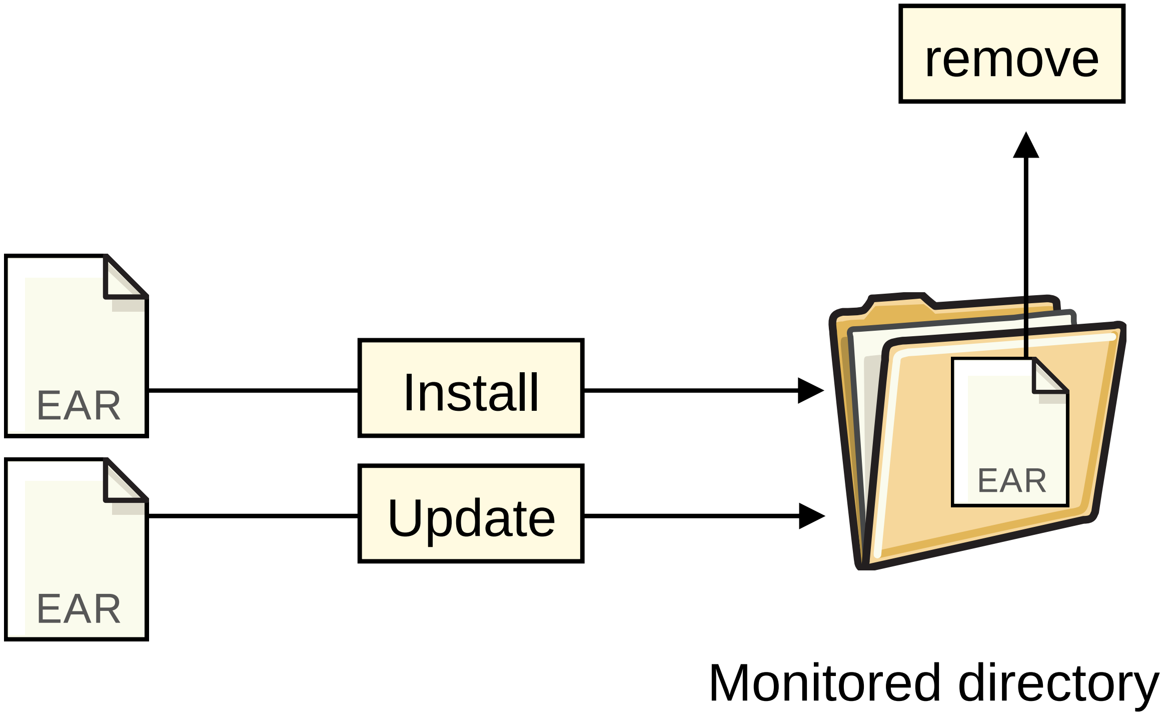
- A simple and fast way to install, update, and uninstall applications without:
  - The administrative console
  - Rational Application Developer
  - wsadmin
  - A specially configured environment

- Tasks can be accomplished by copying archive files in or out of a monitored directory
  - The application must be an EAR, JAR, WAR, or SAR



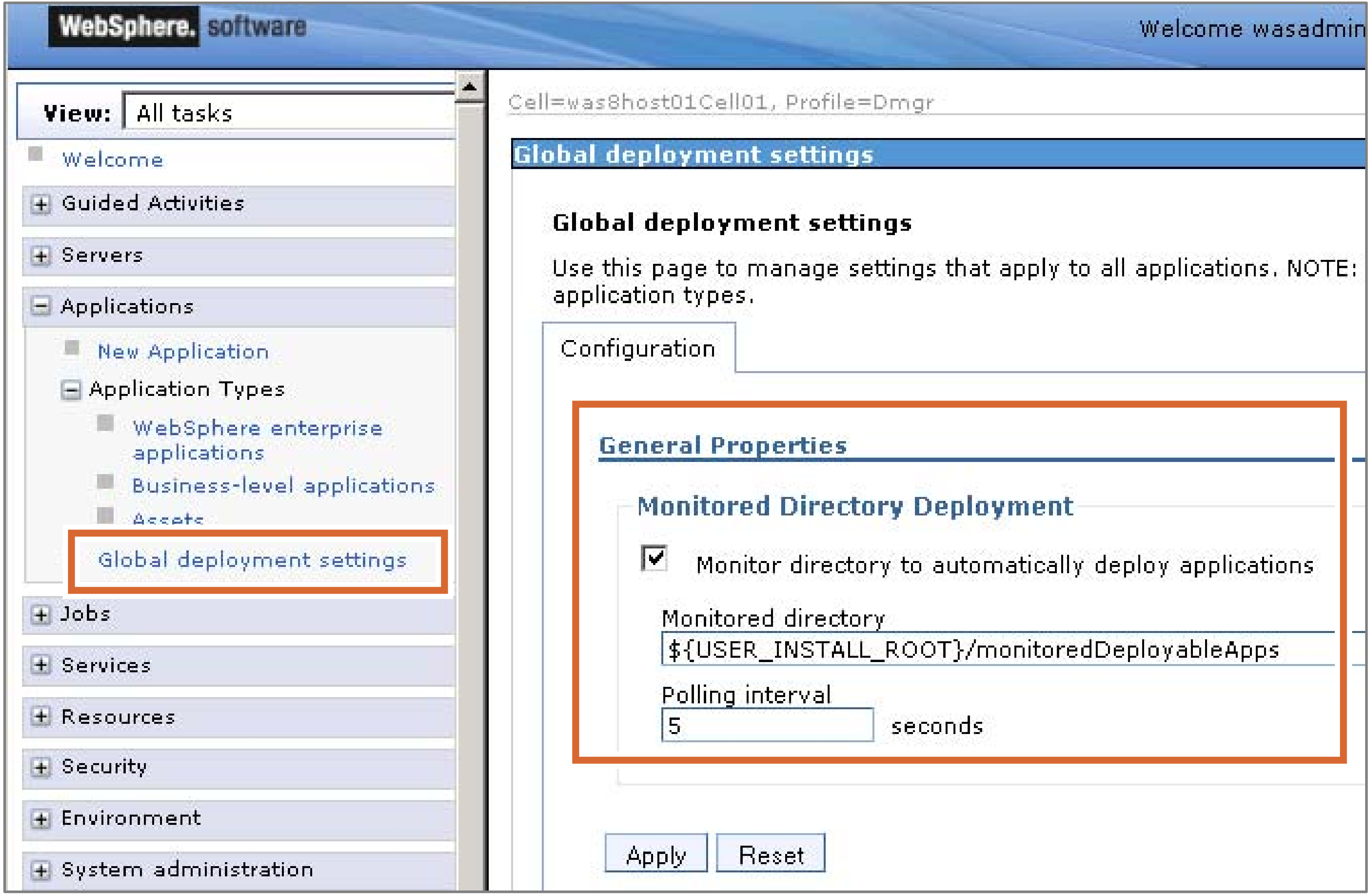
Supported tasks

- Install: place an archive file into the monitored directory
- Uninstall: remove an archive file from the monitored directory
- Update: move or copy a new archive file with the same name as an existing archive file in the monitored directory



Enabling the monitored directory

- Disabled by default in both stand-alone federated environments
  - Click **Applications > Global deployment settings**



Cell=was8host01Cell01, Profile=Dmgr

Global deployment settings

Global deployment settings

Use this page to manage settings that apply to all applications. NOTE: application types.

Configuration

General Properties

Monitored Directory Deployment

☑ Monitor directory to automatically deploy applications

Monitored directory
${USER_INSTALL_ROOT}/monitoredDeployableApps

Polling interval
5 seconds

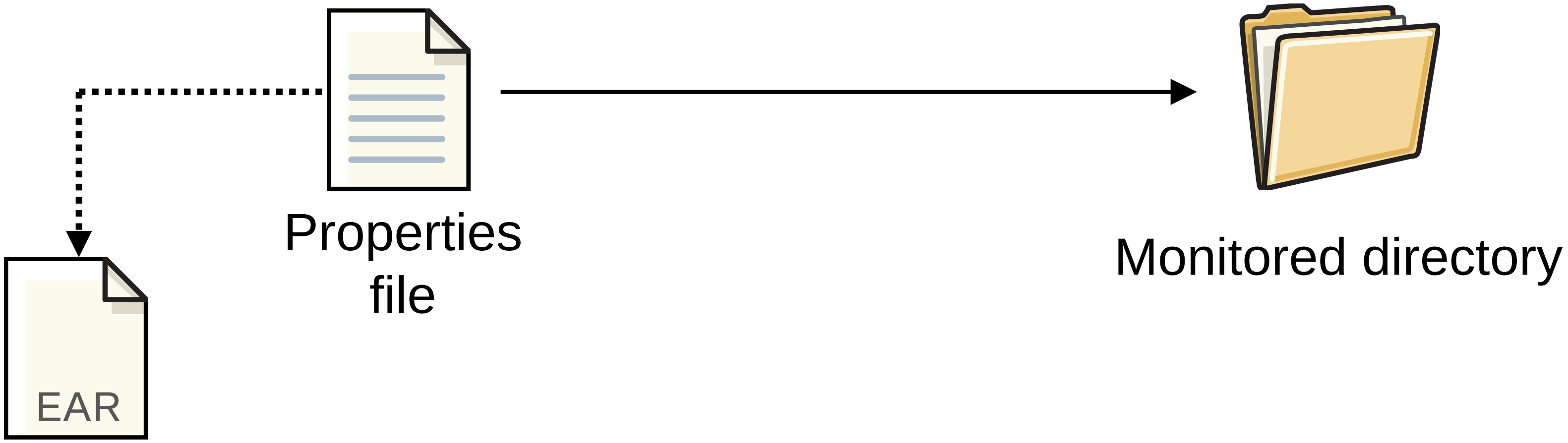
Notes about the monitored directory

- Within the existing directory structure, it is possible to create specific server, cluster, and even node directories
  - These additional structures allow tasks to be directed to specific elements of your environment
- Directory location:
  - Stand-alone: <*profile_root*>/monitoredDeployableApps/servers/<*servername*>
  - Federated: <*dmgr_profile*>/monitoredDeployableApps/servers/<*servername*>
- For clusters:
  - Create a clusters directory: .../monitoredDeployableApps/clusters/<*clustername*>
- For servers with the same name on federated nodes:
  - Applications are deployed to all servers with the same name
  - Create: .../monitoredDeployableApps/nodes/nodename/servers/<*servername*>

Drag-and-drop properties files

- The standard drag-and-drop approach lacks the ability to do anything but the default
  - There is no ability to customize a deployment in any way

- Drag-and-drop technique also supports properties file based configuration
  - A property file can define which EAR file to install, and also configure any of the necessary attributes



Properties file based configuration

- Provides a group of administrative commands
- Manage system configuration
- Troubleshoot configuration issues
- Replicate configuration properties across profiles, nodes, servers, or applications
- Use properties file on monitored directory deployment to deploy applications
- Introduced in WebSphere Application Server V7.0

Steps to use properties file to deploy applications (1 of 2)

- Step 1: create a properties file that defines deployment options
  - Use properties file based configuration command to create an application properties file
  - Extract application properties to a file with version 7 output format

```
AdminTask.extractConfigProperties('[-propertiesFileName
  myApp.props -configData Deployment=MyApplication]')
```

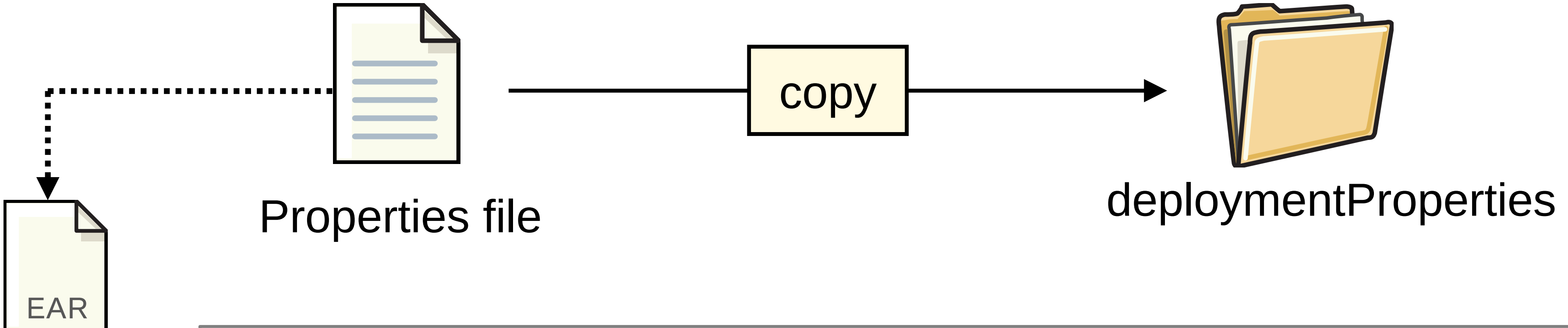
  - Extract application properties to a file with simple output format

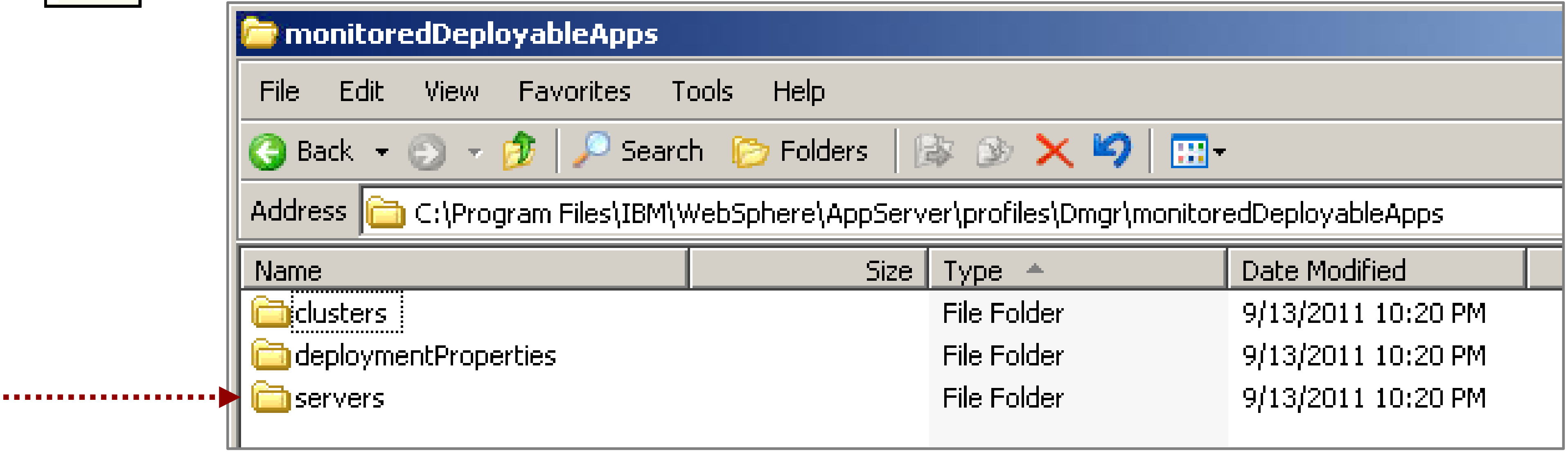
```
AdminTask.extractConfigProperties('[-propertiesFileName
  myApp.props -configData Deployment=MyApplication -
  options [[SimpleOutputFormat true]]]')
```

  - Note: the monitored directory installation process uses only the properties that relate to an application
  - Create the file manually

Steps to use properties file to deploy applications (2 of 2)

- Step 2: verify that the targeted server or cluster member is running
- Step 3: verify that monitored directory deployment is enabled
- Step 4: copy the properties file to the `deploymentProperties` directory





C:\Program Files\IBM\WebSphere\AppServer\profiles\Dmgr\monitoredDeployableApps

Name	Size	Type	Date Modified
clusters		File Folder	9/13/2011 10:20 PM
deploymentProperties		File Folder	9/13/2011 10:20 PM
servers		File Folder	9/13/2011 10:20 PM

Unit summary

Having completed this unit, you should be able to:

- Describe methods of installing enterprise applications in WebSphere Application Server
- Explain how fine-grained application updates work
- Describe enterprise application properties
- Enable monitored directories
- Deploy an application by using the monitored directory
- Use a profile-file-based configuration with monitored directories to deploy an application

Checkpoint questions

1. True or false: You can update a single module or part of an application in the console.

2. True or false: The default startup behavior for an application is to automatically start when the server starts.

3. True or false: Monitored directory is supported in a federated environment.

Checkpoint answers

1. True. You can update a single module or part of an application.

2. True. The default startup behavior for an application is to automatically start when the server starts.

3. True. Monitored directory is supported in a federated environment.

Exercise 6

Installing an application

Exercise objectives

After completing this exercise, you should be able to:

- Use the administrative console to install an application
- Use a web browser to test the application
- Use the drag-and-drop function to deploy an application