



Getting Started

Note

Before using this information, be sure to read the general information under “Notices” on page 383.

Compilation date: June 27, 2004

© Copyright International Business Machines Corporation 2002, 2004. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this book	v
Summary of Changes	vii
Migrating	ix
How to send your comments	xv
Chapter 1. Planning	1
Chapter 2. Installing	3
Chapter 3. Installing WebSphere Application Server products	5
WebSphere Application Server packages for Version 5.1	7
Planning to install an e-business network	11
Single server topology.	13
Example: Choosing a topology for better performance	15
Queuing network.	16
Establishing multimachine environments	17
Setting up a multinode environment.	19
Putting it all together - a combined topology.	37
Running WebSphere Application Server across versions	39
Installing the product	43
Installing Network Deployment on AIX platforms	45
Installing Network Deployment on HP-UX platforms	61
Installing Network Deployment on Linux platforms	79
Installing Network Deployment on Solaris platforms	94
Installing Network Deployment on Windows platforms.	109
Installing additional features on a base Application Server	122
Installing additional features on the deployment manager	127
Platform-specific tips for installing and migrating.	131
Tips for installing the embedded messaging feature	171
Using the Launchpad to start the installation	173
Installing silently	174
responsefile	175
Automatically restarting server processes	176
firststeps command	182
Using the installation verification test	184
ivt command	184
Troubleshooting the installation	185
Creating multiple Version 5 configuration instances	193
wsinstance command	195
Creating servers in coexistence or multiple instance environments	201
Changing HTTP transport ports	202
Installing interim fixes, cumulative fixes, and fix packs	202
Using the update installer to update a base node	211
Using the update installer application to update a deployment manager node	214
updateSilent command	218
updateWizard command	230
Uninstalling interim fixes, cumulative fixes, and fix packs	239
Product version and history information	247
Uninstalling the product.	273
uninstall command	275

vpd.properties file	278
Uninstalling Network Deployment	282
Uninstalling base WebSphere Application Server	283
Uninstalling in a migration scenario	285
Uninstalling in a coexistence scenario	292
Uninstalling manually.	292
Reinstalling V5.1 after uninstalling and leaving the embedded messaging feature installed	310
Migrating and coexisting	311
Migration and coexistence overview	314
Configuration mapping during migration	316
Migrating configuration data	324
Coexistence support	355
Setting up Version 3.5.x and Version 5 coexistence	357
Setting up Version 4.0.x and Version 5.x coexistence	358
Setting up Version 5 coexistence	360
Port number settings in WebSphere Application Server versions	363
Default coexistence settings for port numbers	366
Configuring WebSphere Application Server after migration	371
XML parser for Java code	373
Configuring WebSphere Application Server for DB2 access	375
Installation: Resources for learning	377
Notices	383
Trademarks and service marks	385

About this book

This book is compiled from information center articles that are available at Web address: <http://publib.boulder.ibm.com/infocenter/ws51help/index.jsp>. The information center has many intertopic links that this book is not able to replicate in its subset of information. Although some effort was made to remove such links and replace them with pointers to information center articles, it is possible that some links were missed.

Links to information center articles that are not in this PDF do not function properly. Links to articles that are within this document do work correctly.

To view an article in the information center, go to the Web address and search for the title of the document.

Summary of Changes

This section describes what's new in installation and migration in Version 5.1.

- **New migration tools**

The WASPreUpgrade and the WASPostUpgrade migration tools in Version 5.1 are updated to work with V5.1. To migrate the configuration from another version of WebSphere Application Server, use the WASPreUpgrade tool from the migration/bin directory on the product CD-ROM. The tool from the previous release is not sufficient for migrating to V5.1.

There are several scripts that the installer program copies to the V5.0.x install_root/bin directory when you use the installation wizard to migrate a V5.0.x release to V5.1.

- The pre_uninst50ws and the post_uninst50ws tools prevent removing the configuration of a migrated node from the deployment manager when you uninstall.
- The pre_uninst502mq and the post_uninst502mq tools prevent deleting a messaging queue manager when you uninstall after migrating a configuration of V5.0.2 to V5.1.

- **New option on the uninstaller program**

When uninstalling V5.1 with embedded messaging, there is a prompt for deleting the embedded messaging code. You can leave the code to work with other installation instances that might use it.

- **New level of embedded messaging**

The release level of the embedded messaging feature, which is based on WebSphere MQ technology, is CSD04. Because installation instances on one machine share embedded messaging code, coexistence is affected by the presence of the embedded messaging feature. V5.1 with embedded messaging does not install on a V5.0.0 or V5.0.1 machine with embedded messaging at an earlier level. Stop the installation and upgrade the V5.0.x level to V5.0.2, which can coexist with V5.1. Or clear the selection of the embedded messaging feature and install V5.1 without embedded messaging.

- **New support for configuration instance creation**

You can create a configuration instance when the original installation location is read-only.

Migrating

Migrating is an activity in which you take advantage of existing materials. Migration tasks and tools help you upgrade the product and its prerequisites, reuse existing application components when feasible, and transfer administrative configurations from your past version to a current one.

Migration of WebSphere Application Server products is about leveraging the existing environment and applications and changing them to be compatible with the current product version.

Migration Redbook

Migrating to WebSphere V5: An End-to-End Migration Guide is available from the Redbooks Web site at <http://www.ibm.com/redbooks>. To locate the Redbook, search for the document number SG24-6910-01. The Redbook provides a broader coverage than the information center articles, including more detailed planning information for application migration and WebSphere Studio Application Developer tooling and samples. *Migrating Applications from WebSphere for z/OS V4 and V3.5 to V5* is also available from the Redbooks Web site. Search on document number SG24-7044-00 to locate this book.

Version 5.0.x migration **5.1+**

Migration from Version 5.0.x to Version 5.1 involves minimal change because both releases implement the Java 2 Platform, Enterprise Edition (J2EE) 1.3 specification. Version 5.0.x supports the Java 2 SDK Version 1.3.1. Version 5.1 supports the Java 2 SDK Version 1.4.1. V5.0.x applications can run on either release without change. V5.1 applications can run on either release without change if you compile applications on a V5.1 node with the compatibility options. See the "Changes for application developers" section for more information.

There are several considerations. There are new migration tools for V5.1 to help migrate federated nodes and nodes with the embedded messaging feature installed. Use the new tools when uninstalling a node you migrate or when uninstalling the V5.1 node, if you decide to keep using the previous node. One set of tools prevent the deployment manager from removing the node as a member of the cell. The other new set of tools prevents uninstalling the messaging queue manager and deleting the code for embedded messaging feature.

The WASPreUpgrade and the WASPostUpgrade migration tools are also updated for V5.1. Verify that you use the WASPreUpgrade tool from the migration directory on the CD-ROM if you intend to save a configuration from a previous product, install V5.1, and restore the configuration. You must use the V5.1 version of the WASPreUpgrade tool to save a configuration that you intend to add to a V5.1 node.

Version 4.x migration **5.1+**

Migration from Version 4.x to Version 5.1 involves minimal change because both releases implement the Java 2 Platform, Enterprise Edition (J2EE) specification. (V4 implements the J2EE 1.2 specification. V5.1 implements the J2EE 1.3 specification.) Most V4.x applications can run without change.

Despite this fact, knowledge of how the migration tools migrate Version 4.x applications is important. For example, extended messaging support service is now a component of WebSphere Business Integration Server Foundation, Version 5.1. Applications that use Version 4.x extended messaging support services require migration to use in an Integration Server system.

Version 3.5 migration: Moving to the J2EE model

V3.5.x users upgrading to V5 are moving to a platform that is fully compliant with J2EE specifications. J2EE technology clearly separates development and the creation of applications from application administration, deployment and management. Migration from V3.5 involves changes in application structures, development, and deployment.

5.1+ The migration tools assist in the transition from Version 3.5.x to Version 5.1 by migrating system configurations and creating J2EE artifacts, including J2EE security roles mapping. The migration tools create initial J2EE enterprise applications based on Version 3.5.x configurations. However, because of the significant change in application structures, plan to carefully test and tune migrated applications, using development and deployment tools, to determine exactly how the applications function in the new environment.

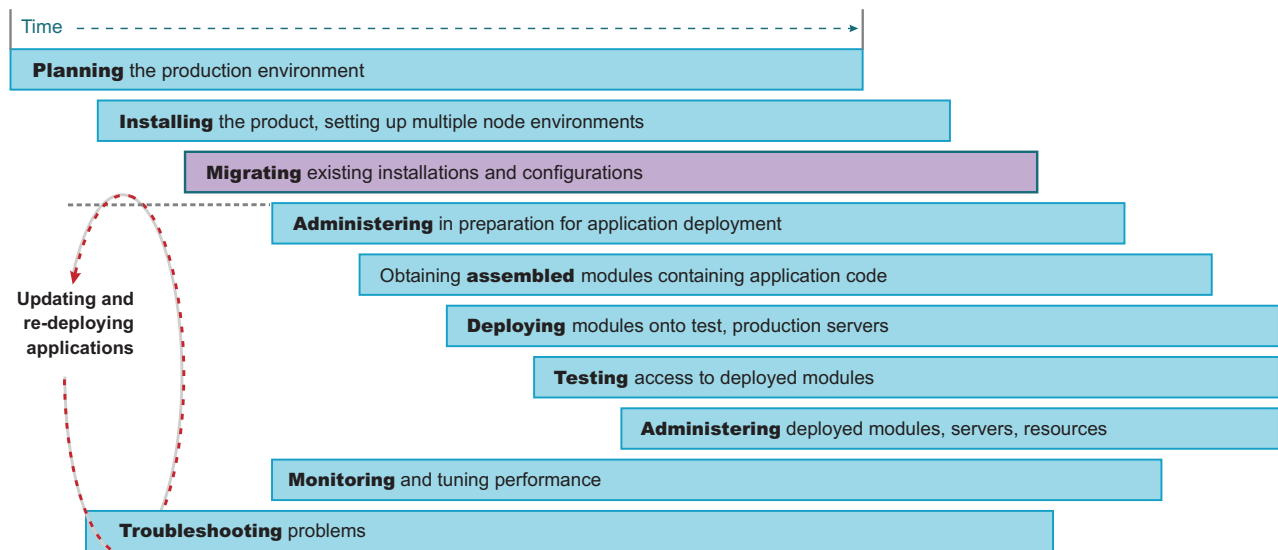
The J2EE model enables you to develop applications independently from their final deployment environment. This task separation simplifies the process of promoting an application from initial development through production, or moving an application from one server to another. The intent is to change only the application deployment parameters, and not the application code.

New assembly and deployment user roles and tools

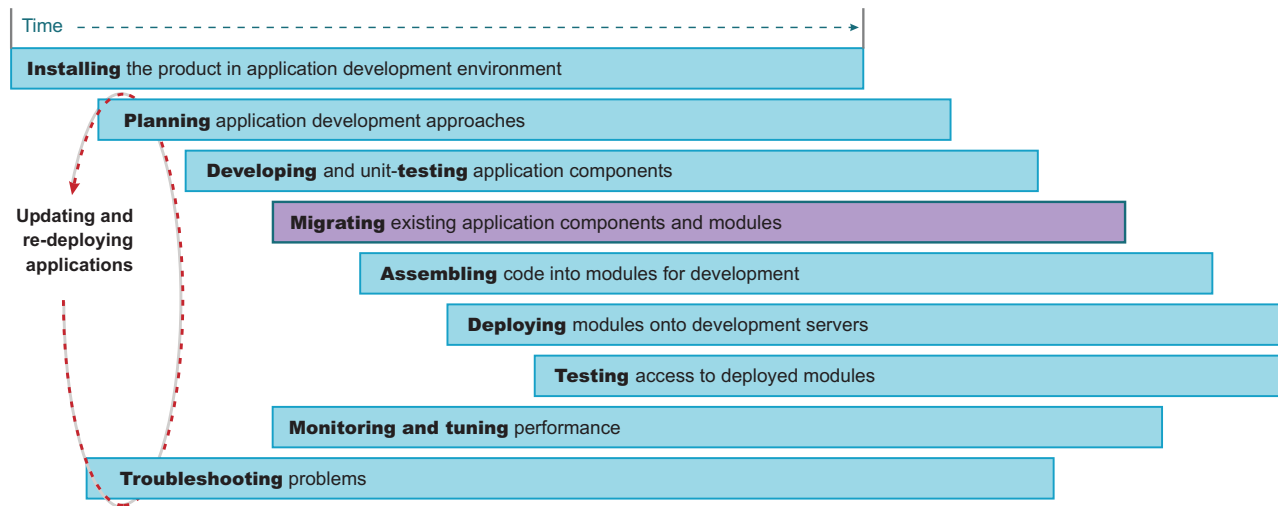
Version 3.5.x and Version 4.x users perform application tasks through the administrative console.

5.1+ Version 5 application settings are defined in J2EE deployment descriptors, by *application assemblers* using the assembly toolkit, which is available on the IBM WebSphere Application Server Toolkit (ASTK) CD-ROM. A *deployer* follows the instructions in the deployment descriptor to install the application into a particular server environment. (See).

A simple timeline of activities for Planning, Installer and Administrator roles.



A simple timeline of activities for an Application Developer role.



Version 5 administration

Version 5 administrators can install a Version 5 application onto the server and bind it to an environment. This capability enables administration at the application and module level. Administrators no longer must manage individual servlets, JSP files, or enterprise beans.

The relationship between applications and Application Servers has changed in the move to a J2EE base. After creating a J2EE application, deploy it by installing the application onto Application Servers through the administrative console. Through the administrative console, you can view, start, or stop deployed modules by the application to which they belong or by the Application Server on which they are deployed.

Support for J2EE resources. In addition to JDBC drivers and data sources, several resource types are added as the product transitioned to a J2EE base. Administrators now can configure URLs, the Java Message Service (JMS), and the JavaMail API. In each case, you can create a resource provider (such as a JMS provider) and then create resource factories for each provider (such as JMS destinations and JMS connections). The default JavaMail API provider does not appear in the administrative console because it is not configurable. You cannot create additional JavaMail API providers.

More information about the J2EE standard is available from the links in Installation: Resources for learning.

Role based security. Version 5 security is consistent with J2EE role-based security specifications. Deployment descriptors specify roles for an application. As the administrator installs the application, these roles are bound to users or to groups.

In the administrative console, a Security Center lets you perform all security tasks from a single location. Such tasks include everything from changing the binding information for roles in an application to setting Secure Sockets Layer (SSL) properties to enabling security. Application-specific security tasks are also supported, through the property sheets for each application.

Changes for application developers

Version 3.5.x and V4.x developers use the administrative console to create, edit, and view application configurations.

5.1 + V5.1 developers use the assembly toolkit, which is available on the IBM WebSphere Application Server Toolkit (ASTK) CD-ROM to package, edit, and view J2EE applications.

Packaging J2EE applications includes:

- Copying appropriate files into the enterprise archive (EAR) file, including classes, JSP files, HTML files, image files, and so on
- Defining deployment descriptor files for modules and applications

5.1 + In Version 5.1, the assembly toolkit is available on the IBM WebSphere Application Server Toolkit (ASTK) CD-ROM. The assembly toolkit lets you copy files with appropriate relative paths into the archive and provides a GUI method for defining deployment descriptors. Also, developers can set environment-specific binding information through the assembly toolkit. These bindings are defaults for the administrator to use when installing the application through the administrative console.

You can define IBM extensions to the J2EE specification, for example supporting servlets to be served by class name. These extensions are saved in an XML file that is separate from the standard J2EE deployment descriptor to ensure portability to other Application Servers.

5.1 + If your existing applications use IBM extensions from earlier product versions, it might be necessary for you to perform mandatory or optional migration to use the same kinds of extensions in Version 5.1.

Version 5.1 supports the Java 2 SDK Version 1.4.1. There are several implications for migration and coexistence. The V5.1 Network Deployment cell is a mixed node environment, where some Application Server nodes might be V5.0.x running Java 2 SDK 1.3.1 and some nodes might be V5.1 nodes running Java 2 SDK 1.4.1.

Applications compiled on V5.0.x nodes can also run on V5.1 nodes. The IBM SDK 1.4.1 in V5.1 provides this backward compatibility.

There is also a cross compiler option for the IBM SDK 1.4.1 that supports compiling code against a bootstrap and extension classes of a previous IBM SDK version. By default, the javac compiler compiles against the bootstrap and extension classes on its platform: the IBM SDK 1.4.1 compiles for 1.4 by default. You can use the IBM SDK 1.4.1 to compile for IBM SDK 1.3.1 compatible output. For example, a developer can issue this command on a V5.1 node to compile against a 1.3 target:

```
/ javac -target 1.3 -bootclasspath jdk1.3\lib\classes.zip \ -extdirs "" OldCode.java
```

The compiled code runs on a 1.3 Java virtual machine (JVM). The -target 1.3 parameter generates class files that are compatible with 1.3 JVMs. If you do not specify the -target option, the -bootclasspath option and the -extdirs option, the compiled code runs on 1.4 JVMs only. You must specifically compile for 1.3 JVMs on a V5.1 node, to avoid compiling against a Java 2 Platform API that is not present on a 1.3 JVM. A 1.4 application fails at run time on a V5.0.x node.

Migrating APIs and specifications

5.1 + Migrating APIs and specifications means moving to the current Java component level and to other technologies that IBM WebSphere Application Server, Version 5.1 supports.

Migrating APIs and specifications also includes moving to the most contemporary open specification levels. If your existing applications currently support different specification levels than those supported by this product version, updating at least some aspects of the applications to comply with the new specifications is probable.

5.1+ From V3.5.x to Version 5.1, the main migration areas include IBM extensions and the Java 2 SDK. In contrast, little migration is necessary to move from Version 4.x to Version 5.1.

How to send your comments

Your feedback is important in helping to provide the most accurate and highest quality information.

- To send comments on articles in the WebSphere Application Server Information Center
 1. Display the article in your Web browser and scroll to the end of the article.
 2. Click on the **Feedback** link at the bottom of the article, and a separate window containing an e-mail form appears.
 3. Fill out the e-mail form as instructed, and click on **Submit feedback**.
- To send comments on PDF books, you can e-mail your comments to: **wasdoc@us.ibm.com** or fax them to 919-254-0206.

Be sure to include the document name and number, the WebSphere Application Server version you are using, and, if applicable, the specific page, table, or figure number on which you are commenting.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

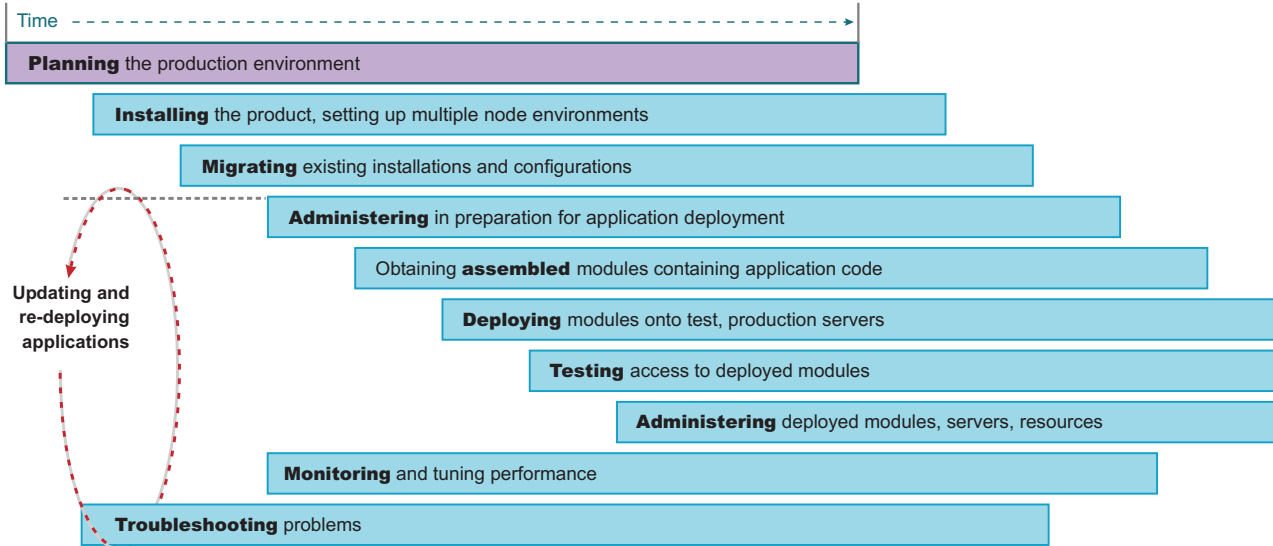
Chapter 1. Planning

Planning is an activity in which you make decisions that affect one or more subsequent activities.

Pre-installation planning helps IT architects and various administrators select and implement multiple node server environments comprised of one or more product installations.

Note the topic , as it is an essential task in a WebSphere Application Server Network Deployment environment.

A simple timeline of activities for Planning, Installer and Administrator roles.



Aside from the planning that precedes product installation, application developers and others encounter various decisions that are smaller in scale, but can have ramifications if not well planned. Those are the kind of decisions outlined under Planning.

For example, selecting a session tracking approach can affect application development, assembly, and administration downstream.

Whatever kind of planning you are doing, it is best to review the product documentation of the functional area you are using, to understand the key concepts underlying, as well as the specific product support and limitations in that area.

Chapter 2. Installing

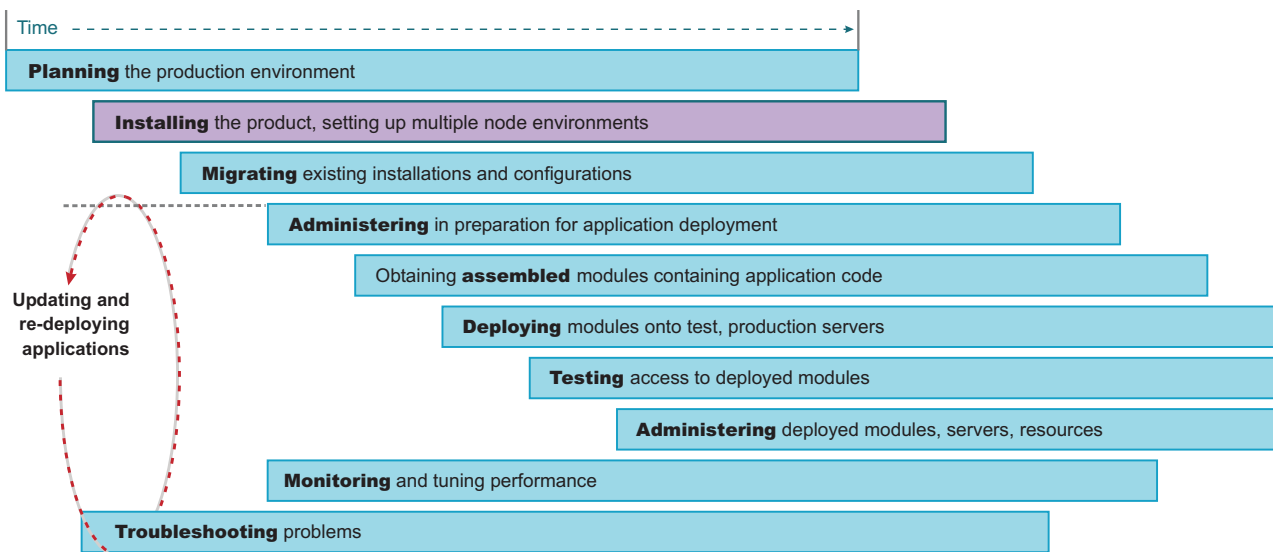
Installing is an activity in which you put software onto machines. Some installation can be performed after the initial product setup, such as installing the complimentary application clients provided with the product.

For Version 5.1.1 the Launchpad and the installer wizard use the language setting from the machine locale to determine the language for the interface.

Installation can be part of setting up a multiple node production environment, in which case significant planning is needed.

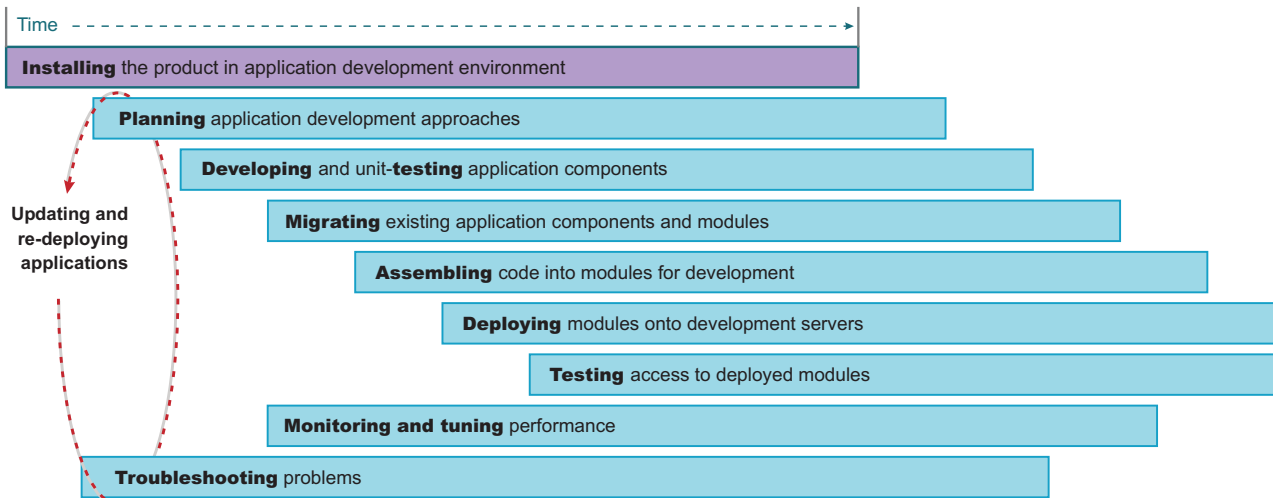
See "Establishing multimachine environments" for detailed setup instructions once you have installed WebSphere Application Server and WebSphere Application Server Network Deployment on various machines.

A simple timeline of activities for **Planning, Installer and Administrator** roles.



Application developers might perform quick, single machine installations to create a realistic server environment for unit-testing code artifacts and applications. This task can require little preparation beyond checking the software and hardware requirements.

A simple timeline of activities for an **Application Developer** role.



Chapter 3. Installing WebSphere Application Server products

This topic introduces installing IBM WebSphere Application Server Network Deployment, Version 5.x, including the tasks you need to perform.

To create a complete environment for installation, you must install any prerequisites and prepare the operating platform for installation. During installation, you can optionally migrate the configuration from a previous version of the product. After installation, you can run the installation verification test and bring up your Application Server.

Perform the following tasks to create a running version of the product on your machine.

1. Plan to install an e-business network, as described in “Planning to install an e-business network” on page 11

This task helps you plan to install an e-business network. It also describes interoperability considerations.

2. Install the product, as described in “Installing the product” on page 43.

This task helps you prepare your machine for installation and explains the different types of installation available to you. It includes information on:

- Using the Launchpad
- Deciding whether to migrate applications and the configuration from a previous version
- Deciding whether to coexist with a previous version
- Using the silent installation method

This task describes using the installation wizard. Through the installation wizard, you migrate applications and configurations from a previous version of WebSphere Application Server, coexist with the previous version, choose a full installation type or a custom installation type, and perform some initial configuration.

Migrate Enterprise, V5.0.x or Enterprise Edition, V4.x to WebSphere Business Integration Server Foundation, Version 5.1.

Table 1. When to migrate extended applications to Version 5.1

Migration from	Migrate to V5.1?		
	WebSphere Business Integration Server Foundation	WebSphere Application Server	Network Deployment
Enterprise Edition, V4.1.x	Yes	No. You disable programming model extensions in your migrated applications.	
Enterprise, V5.0.x	Yes		

Migrating to base or Network Deployment issues warnings that describe the parts of your configuration that are disabled. The migration tools record these warnings in a log file. If you have already migrated and disabled the extensions in your applications, you can enable the extensions again by installing Integration Server and selecting the migration option.

There is no Enterprise Edition in Version 3.5.x or Version 4.0.x. If you are migrating from a V3.5.x product or a V4.0.x product, migrate when you install the V5.1 base product or the V5.1 Network Deployment product. After migrating, install the WebSphere Business Integration Server Foundation product.

The Network Deployment node must be at the highest product level within its cell. The following table shows if a particular base product configuration is allowed within a Network Deployment node.

Table 2. Mixed version cells in Version 5.1

Application Server node	Network Deployment			
	V5.0.0	V5.0.1	V5.0.2	V5.1
V5.0.0	Yes	Yes	Yes*	Yes*
V5.0.1	No	Yes	Yes*	Yes*
V5.0.2	No	No	Yes	Yes
V5.1.x	No	No	No	Yes

* The base node cannot have the embedded messaging feature installed.

3. Verify the installation.

- a. Use the First Steps tool to run the Installation Verification Test, as described in “Using the installation verification test” on page 184.

The First Steps tool starts automatically at the end of the installation.

- b. Identify and correct any problems using the troubleshooting procedure, as described in “Troubleshooting the installation” on page 185.

4. Examine the results of migration, or prepare to perform a manual migration, as described in “Migrating and coexisting” on page 311.

This task describes exactly what is migrated during the automatic migration. It also describes how to perform a manual migration using the migration tools.

5. Prepare to migrate from an unsupported operating system, as described in “Migrating from an operating system that is no longer supported” on page 339.

This task describes a basic procedure to follow when migrating from one operating system to another.

6. Configure WebSphere Application Server after migration, as described in “Configuring WebSphere Application Server after migration” on page 371.

This task explains how to check migrated application and configuration information, to understand and configure exactly what you migrated.

- If you migrate from Version 3.5.x, examine the applications that you are moving. Make any necessary changes to the applications that are converted to Java 2 Platform, Enterprise Edition (J2EE) platform applications. The migration tools create the initial J2EE enterprise applications, based on Version 3.5.x configurations.
- If you migrate from Version 4.x, you have little to review. J2EE 1.2 enterprise archive (EAR) files in Version 4 work in Version 5 of WebSphere Application Server, which also supports the J2EE 1.3 specification.

7. Set up a multinode environment, as described in “Establishing multimachine environments” on page 17.

This task describes how to set up multiple nodes into a group or *cell* of Application Servers, with centralized configuration under the control of the deployment manager node and its distributed node agents.

8. Set up V3.5.x and V5 coexistence, as described in “Setting up Version 3.5.x and Version 5 coexistence” on page 357.

This task describes running WebSphere Application Server V3.5.5 and later editions with V5.x, on the same machine.

9. Set up V4.0.x and V5 coexistence, as described in “Setting up Version 4.0.x and Version 5.x coexistence” on page 358.

This task describes running WebSphere Application Server V4.0.2 and later editions with V5.x, on the same machine.

10. Set up V5.x coexistence, as described in “Setting up Version 5 coexistence” on page 360.

This task describes running multiple WebSphere Application Server product installations on the same machine.

11. Automatically restart WebSphere Application Server processes, as described in “Automatically restarting server processes” on page 176.
This task describes how to set up WebSphere Application Server processes for the operating system to monitor and restart.
12. Create multiple Version 5 instances on one machine, as described in “Creating multiple Version 5 configuration instances” on page 193.
This task describes creating multiple configuration instances from one installation. It also describes creating multiple servers in a coexistence or multiple instance environment, and changing port settings for HTTP transport to avoid conflicts.
13. Apply an interim fix, a cumulative fix, or a fix pack to an existing installation, as described in “Installing interim fixes, cumulative fixes, and fix packs” on page 202.
This task helps you apply maintenance to WebSphere Application Server products after you download the maintenance from the Support page.
14. Remove an interim fix, a cumulative fix, or a fix pack from an existing installation, as described in “Uninstalling interim fixes, cumulative fixes, and fix packs” on page 239.
This task helps you remove applied maintenance from WebSphere Application Server products.
15. Tune for performance. For best performance on any platform, see Tuning performance parameter index .

The WebSphere Application Server product is installed.

Uninstalling and reinstalling: See “Uninstalling the product” on page 273 for information about uninstalling and reinstalling any WebSphere Application Server product.

Deploying applications: Go to Quickly deploying Web components - Try it out! for a description of how to get started deploying applications.

WebSphere Application Server packages for Version 5.1

The WebSphere Application Server family of interoperable products provides a next-generation application server on an industry-standard foundation. The IBM WebSphere Application Server family is divided into the following packages for Version 5. Each edition addresses a distinct set of scenarios and needs.

WebSphere Application Server includes:

- **WebSphere Application Server Express** (not pictured below)
This edition is a lightweight server for static content, servlets, and JavaServer Pages (JSP) files, but does not support enterprise beans.
- **WebSphere Application Server**
This edition addresses the basic programming and run-time needs of desktop developers and single-server production scenarios. The run-time environment for this edition addresses standards-based programming for Web and component-based programming, as well as Web services. The administration model for this edition presumes a single-server environment with no clustering for failover or workload balancing, and with no centralized administration of multiple server instances. However, you can add a stand-alone node to a centrally administered network (the cell) at any time after installing the following product, Network Deployment that controls the cell.
- **WebSphere Application Server Network Deployment**
This edition addresses application servers running in multiple-server production scenarios. It provides centralized administration, as well as basic clustering and caching support.
- **WebSphere Business Integration Server Foundation**
IBM WebSphere Business Integration Server Foundation (previously IBM WebSphere Application Server Enterprise) builds on the industry leading IBM WebSphere Application Server Network Deployment to deliver a next generation composite application platform optimized for building service-oriented applications that extend and integrate your existing IT assets.

With the Version 5.1 release, IBM WebSphere Application Server Enterprise has been re-branded as WebSphere Business Integration and renamed to IBM WebSphere Business Integration Server Foundation to improve product alignment and positioning as the foundation for higher value IBM WebSphere Business Integration solutions.

The product value proposition and functionality remain unchanged. Future releases of IBM WebSphere Business Integration Server Foundation will focus on enhancing product integration capabilities, including maintaining and enhancing the process choreography and business rule features and adding new integration capabilities and easier-to-use tools.

IBM WebSphere Business Integration Server Foundation will continue to build on the IBM WebSphere Application Server and will maintain its commitment to J2EE and open standards.

IBM WebSphere Studio Application Developer Integration Edition Version 5.1 (name unchanged) is a comprehensive Eclipse-based composite application development environment for building service-oriented applications that extend and integrate your existing IT assets.

IBM WebSphere Studio Application Developer Integration Edition V5.1 is optimized for building integrated applications and business processes that deploy to IBM WebSphere Business Integration Server Foundation V5.1. WebSphere Studio Application Developer Integration Edition includes all of the features available in IBM WebSphere Studio Application Developer v5.1.1.

- **WebSphere Application Server for z/OS** (not pictured)

This edition integrates the WebSphere Application Server product and the Network Deployment product into a single package that leverages the unique qualities of service inherent to the z/OS platform. It addresses standards-based programming for Web and component-based programming, as well as Web services, and provides centralized administration of multiple server instances, as well as basic clustering and caching support.

5.1 +

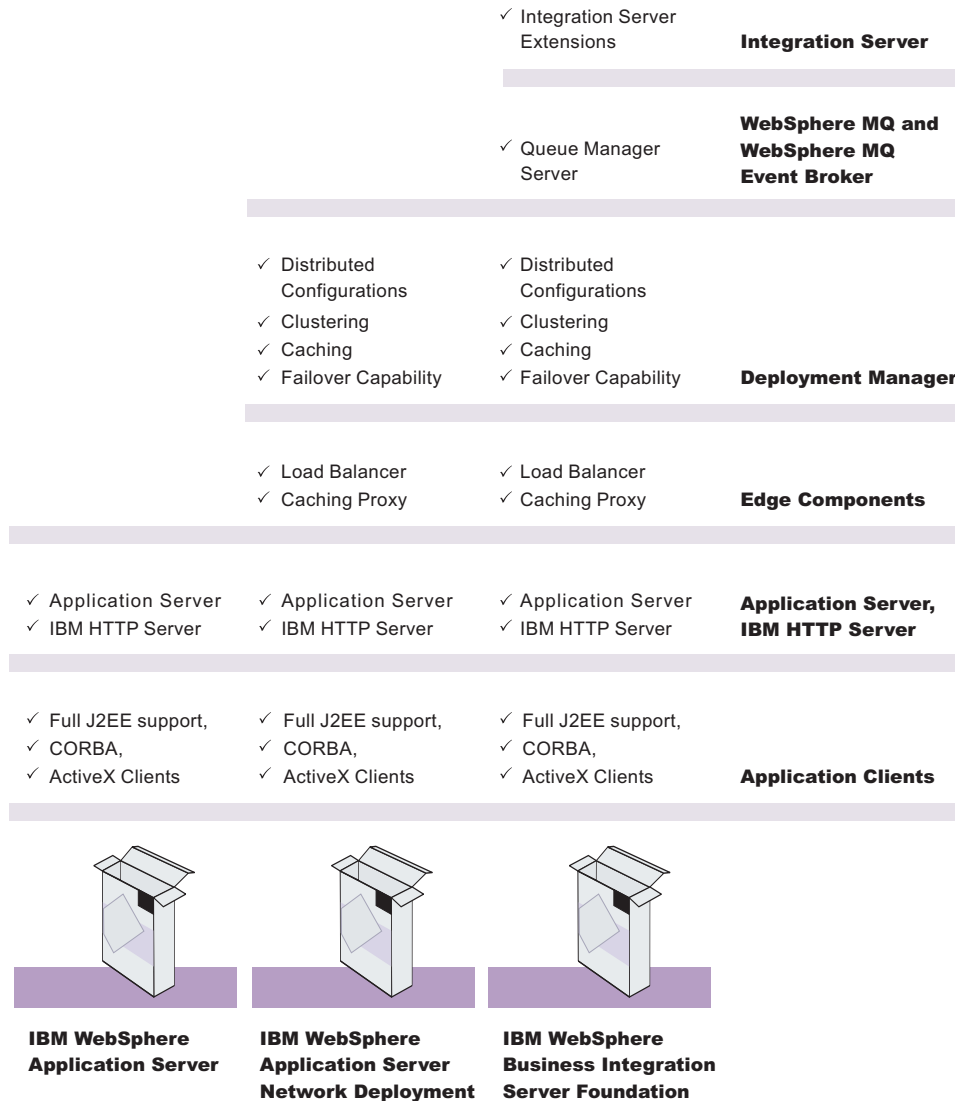


Figure 1. Installation images in WebSphere Application Server packages that are currently available for V5.1. The following packages are available for Version 5.1, which supports the Java 2 SDK 1.4.x specification.

Description of installation images

The WebSphere software platform for e-business starts with a foundation formed from Web application serving and integration. The WebSphere Application Server family lets you quickly, reliably and flexibly enable your business for the Web. It provides the core software to deploy, integrate and manage your e-business applications. WebSphere Application Server supports custom-built applications, based on integrated WebSphere software platform products, or on other third-party products. Such applications can range from dynamic Web presentations to sophisticated transaction processing systems.

IBM WebSphere Application Server, WebSphere Application Server Network Deployment, and WebSphere Business Integration Server Foundation are interoperable building blocks. The base product CDs are included in the Network Deployment package. The Network Deployment package CDs are included in the Integration Server package.

- The **Application Clients** installation image contains the Java 2 Enterprise Edition (J2EE) Application Clients support, the WebSphere Business Integration Server Foundation client programming model extensions, the CORBA C++ client (deprecated), the Applet client, and the ActiveX to Enterprise

JavaBeans (EJB) bridge client support. The image also contains the embedded messaging client support for the native Java Message Service (JMS) provider in WebSphere Application Server products.

- The **WebSphere Application Server** product installation image contains the core application server runtime, a native JMS provider (the embedded messaging feature), IBM HTTP Server, IBM Developer Kit, IBM Cloudscape, XML and XSL parsers, the deployment tool, the node agent for communicating to the deployment manager when it is part of a cell, and the external adapter library for proxy caching enablement.

The application assembly tool is no longer a feature of the base WebSphere Application Server product beginning with Version 5.1. The assembly toolkit replaces the application assembly tool. The assembly toolkit is available on the IBM WebSphere Application Server Toolkit CD in the product package. The assembly toolkit consists of the J2EE Perspective of the WebSphere Studio Application Developer product, including code generation capabilities.

- The **Deployment Manager** product installation image contains the deployment manager configured for use in departmental production computing scenarios. It includes the embedded messaging client feature, the Version 2 compliant universal description, discovery, and identification (UDDI) registry feature, and the Web services gateway feature. Although the Network Deployment package includes the base Application Server CD, installing the Network Deployment product does not install the base Application Server product. You must use the base product CD to install the base product.
- The **Edge components** installation image contains IBM HTTP Server, and edge of network support for the Load Balancer (Dispatcher) and Caching Proxy (edge caching), as well as support for network authentication and single sign-on.
- The **WebSphere Business Integration Server Foundation** installation image contains programming model extensions to the core Application Server, such as business rule beans and process choreographer, and deployment manager extensions for administering functions included in the programming model extensions. Integration Server supports an *umbrella installation*, which automatically installs the base product in the same installation procedure.

Although the Integration Server package includes the Network Deployment product CD, installing Integration Server does not install the Network Deployment product. You must use the Network Deployment product CD to install the Network Deployment product.

If you install the Integration Server product on a machine with an installed Network Deployment product, the Integration Server product extends the Network Deployment administrative console.

Although the Integration Server package includes a separate base WebSphere Application Server product CD, installing Integration Server on a clean machine automatically installs the base Application Server product in an umbrella installation.

If you install Integration Server on a machine with an installed base Application Server product, Integration Server extends the base Application Server product, while automatically installing any base product features that Integration Server requires.

- The **WebSphere MQ and WebSphere MQ Event Broker** installation images in Version 5.0 provide the non-embedded full-function WebSphere MQ Queue Manager for reliable, dynamically load balanced asynchronous messaging for more than 35 platforms.

Included with all packages are the Data Direct Technologies JDBC Drivers and the IBM WebSphere Application Server Toolkit (ASTK) compact disks (CDs).

Also included with the Version 5.1 base product and the Network Deployment product are the WebSphere Application Server, Version 5 for iSeries CDs. The Integration Server product for V5.1 is not supported on iSeries servers at this time.

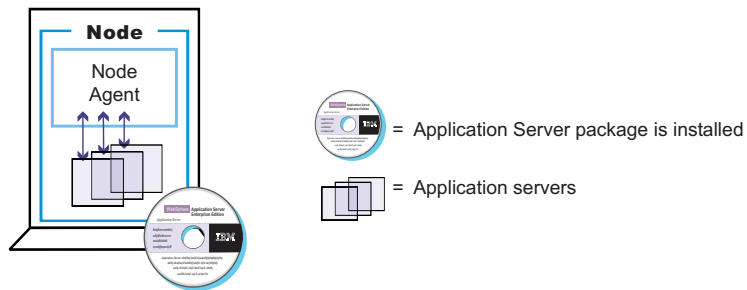
In addition, the Network Deployment package includes the IBM Directory Server and the DB2 Universal Database Enterprise Edition products.

Planning to install an e-business network

Version 5 of the WebSphere Application Server family provides flexible configurations and deployment options for hosting Java 2 Platform, Enterprise Edition (J2EE) applications. The WebSphere Application Server package provides all required components:

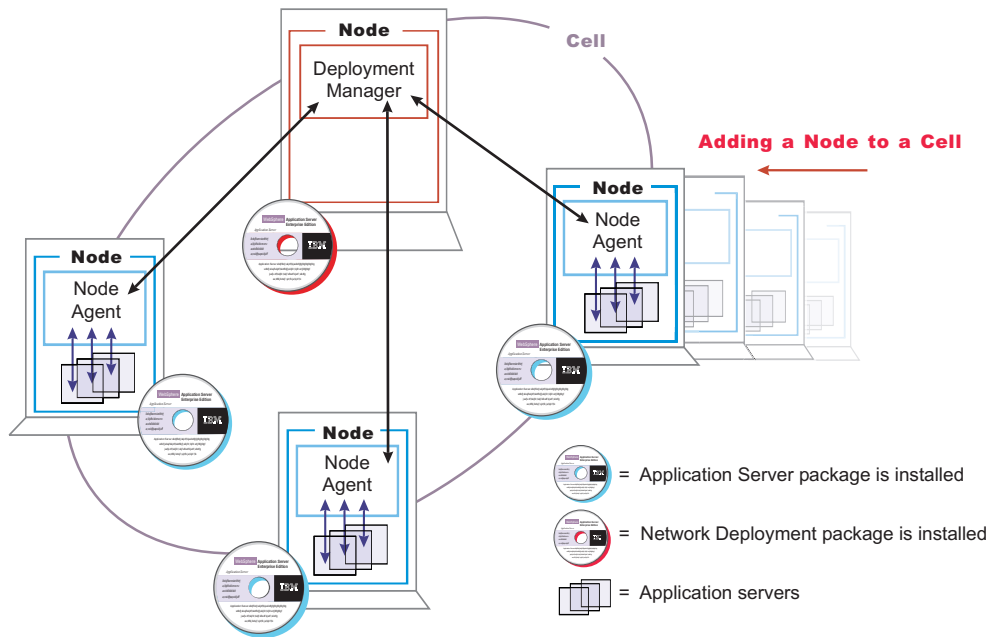
- The WebSphere Application Server
- A Web server
- A J2EE client

IBM WebSphere Application Server Package



The WebSphere Application Server Network Deployment package includes the core package and provides a smooth transition to deploying and managing applications in a distributed network environment. It includes productivity and scalability enabling components (Edge components). The deployment manager server lets you easily federate single nodes to its group, which it manages as a single image or *cell*, by using the nodeagent server process on each node.

IBM WebSphere Application Server Network Deployment Package



Tasks in the installation planning process appear below.

1. Plan the general scope of your network.

Review single machine topologies and how to establish a multimachine environment to learn about WebSphere Application Server, Version 5 product family scalability. With one command you can move an Application Server node from a stand-alone environment into a managed group of server processes, all part of a single-image cell, under the centralized control of the Version 5 deployment manager. Or, you can move an Application Server back to a stand-alone environment. You can start small and *scale up* to larger integrated topologies.

2. Review common multinode topologies.

Determine which scalable and reliable e-business environment is a best fit for your requirements as you learn where to install the typical components of a WebSphere Application Server environment. For example, decide whether to install WebSphere Application Server on the same server as the Web server, or whether to create a cluster of application servers on one or several machines.

Decide which WebSphere Application Server package you need. Understanding scalability factors and picking the right design can help you start with the correct package, and can help you achieve your business model objectives for e-business in a timely and cost-efficient manner.

3. Review common topologies for WebSphere Application Server Edge components, as described in the information center for Edge components.

5.1+ After planning how to install the WebSphere Application Server product, you can plan the installation of Edge components, which are included in the Network Deployment package and therefore, in the WebSphere Business Integration Server Foundation package. Edge components include the caching proxy and the load balancer component set, including the dispatcher.

You can find a link to the Edge components information center Web site in the *Installation: Resources for learning* topic.

4. Review the latest research in e-business patterns at the developerWorks Web site. You can find additional information on proven patterns of machine and product configurations, from thousands of actual IBM experiences.

You can find a link to the developerWorks Web site in the *Installation: Resources for learning* topic.

5. Plan for interoperability and coexistence.

Plan to have WebSphere Application Server interoperate with your other e-business systems, including other versions of WebSphere Application Server. *Interoperability* provides a communication mechanism for WebSphere Application Server nodes that are at different versions. *Coexistence* describes multiple versions or instances running on the same machine, at the same time.

Interoperability support enhances migration scenarios with more configuration options. It often is convenient or practical to interoperate during the migration of a configuration from an earlier Application Server version to a later one when some machines are at the earlier version and some machines are at the later version. The mixed environment of machines and application components at different software version levels requires interoperability and coexistence.

It is often impractical, or even physically impossible, to migrate all the machines and applications within an enterprise at the same time. Understanding multiversion interoperability and coexistence is therefore an essential part of a migration between version levels.

- a. Plan to run WebSphere Application Server across platforms.

Support of multiple versions is provided on all operating system platforms supported by WebSphere Application Server, Version 5.

- b. Plan to run WebSphere Application Server across versions.

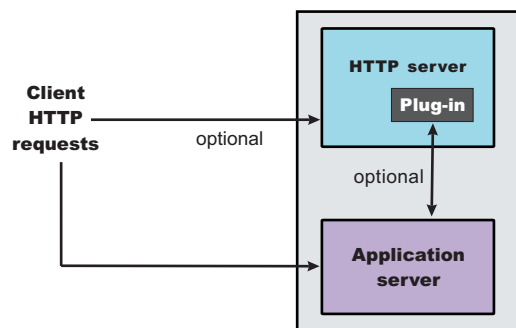
WebSphere Application Server, Version 5 is generally interoperable with WebSphere Application Server Versions 3.5.x and 4.0.x, although each version has specific requirements. However, the ability to run different versions of an Application Server in a configuration does not let you include Version 5 Application Servers in an existing administrative domain, or let you include Version 3.5.x or Version 4.0.x Application Servers in a Version 5 cell.

Reading the information referred to in this topic provides you with a good approach for implementing your WebSphere Application Server environment.

You are now ready for pre-installation planning and installation.

Single server topology

The following illustrations show examples of single server topologies. Each WebSphere Application Server product can run in a single server environment. The most common topology is a stand-alone base WebSphere Application Server product.

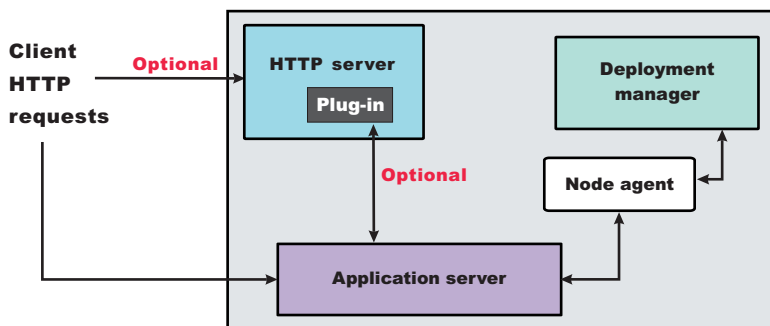


The WebSphere Application Server base product runs on a single machine. You can install the product in a stand-alone configuration or as part of a cell in a multimachine configuration. The stand-alone

configuration is typically for developer desktops or stand-alone production computing, which involve a single Application Server instance operating independently of any other applications.

You can install an installation image on any supported machine. You can configure the Application Server with the Administrative Console to host one or more applications.

The Network Deployment product can also run on a stand-alone machine or in a multimachine configuration, as described in other topology topics. The following illustration shows a typical developer environment for a Network Deployment product in a stand-alone configuration. This configuration is not recommended for a production environment unless you have a machine with the capacity for both products.



Multiple instances on one machine

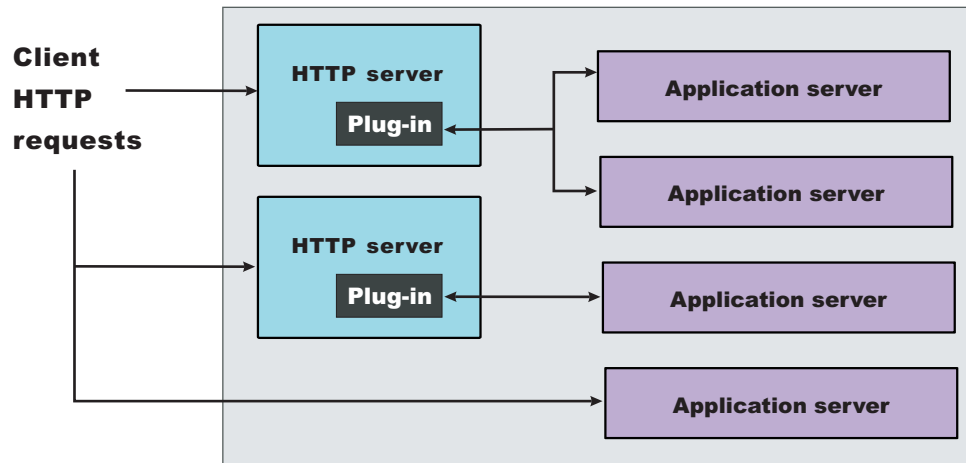
The base WebSphere Application Server product contains three main topologies:

- A single installation, as described previously
- Multiple installations in a coexistence environment
- A single installation with multiple configuration instances

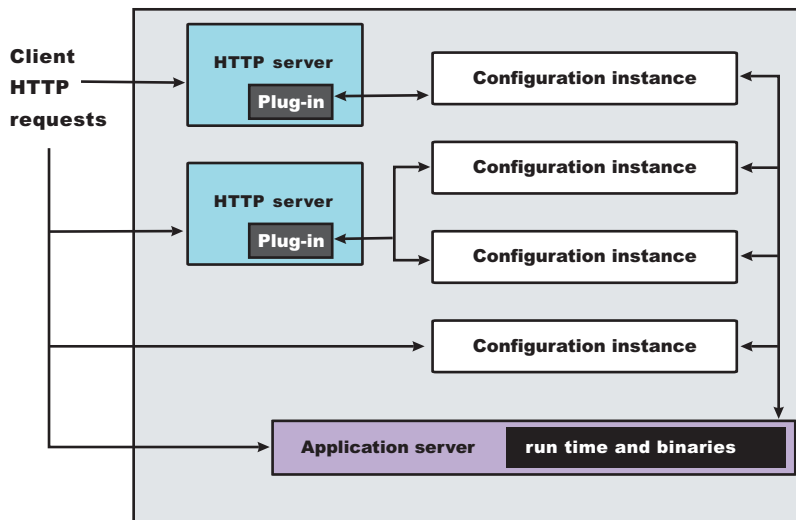
You can install the base WebSphere Application Server product one time as described previously, which creates a default server1 instance.

You can create more servers on a machine using the techniques described in “Creating servers in coexistence or multiple instance environments” on page 201.

You can install the base WebSphere Application Server product multiple times in separate directories. Each resulting installation instance is a fully functional Application Server. The following illustration shows an example of such a configuration.



You can also install the base WebSphere Application Server product one time and use the **wsinstance** command to create multiple configuration instances. Configuration instances are fully functional Application Servers that share the run time and command binaries of the initial product installation. The following illustration shows an example of such a configuration.



Example: Choosing a topology for better performance

WebSphere Application Server provides various Workload Management (WLM) topologies. The following case study uses two topologies, Topology A and Topology B to show how the type of topology you choose can affect performance. In the case study, Topology A can improve performance from 10%-20% more than Topology B. You can see performance increase by using the J2EE benchmark Trade, which is included in this release.

Topology A contains a Web server and a WebSphere Application Server plug-in to a cluster of WebSphere Application Servers. Each cluster member contains a Web container and EJB container. Topology B includes a Web server, a plug-in, and one Web container to a cluster of EJB containers. In both topologies, the Object Request Broker pass-by-reference is selected and the backend database is on a dedicated machine.

Topology A has an advantage because the Web container and EJB container are running in a single Java virtual machine (JVM). In Topology B, the Object Request Broker pass-by-reference option is ignored between the Web container cluster member and the EJB container member. In Topology A, the EJB container uses the same thread passed from the Web container. The request does not have to be passed from one thread in one JVM to another thread in another JVM.

Also, if the processor utilization of the cluster member machines is near 100% you can add more members. If the Web server box is not running at capacity and the Web container processing is not heavy, try freeing the processor on the other members by moving to Topology B.

In this test environment, Topology A had the advantage, however, many factors related to the application and environment can influence results.

Queuing network

WebSphere Application Server contains interrelated components that must be harmoniously tuned to support the custom needs of your end-to-end e-business application. These adjustments help the system achieve maximum throughput while maintaining the overall stability of the system. This group of interconnected components is known as a queuing network. These queues or components include the network, Web server, Web container, EJB container, data source, and possibly a connection manager to a custom back-end system. Each of these resources represents a queue of requests waiting to use that resource. Various queue settings include:

- IBM HTTP Server: **MaxClients** for UNIX and **ThreadsPerChild** for Windows NT and Windows 2000 systems described in Web server tuning parameters .
- Web container: **Maximum size** described in Thread pool settings, **MaxKeepAliveConnections** and **MaxKeepAliveRequests** described in HTTP transport custom properties .
- **Tuning Object Request Brokers** explained in Tuning application servers .
- Data source **connection pooling** is discussed in Connection pooling and **statement cache size** is explained in Data source settings .

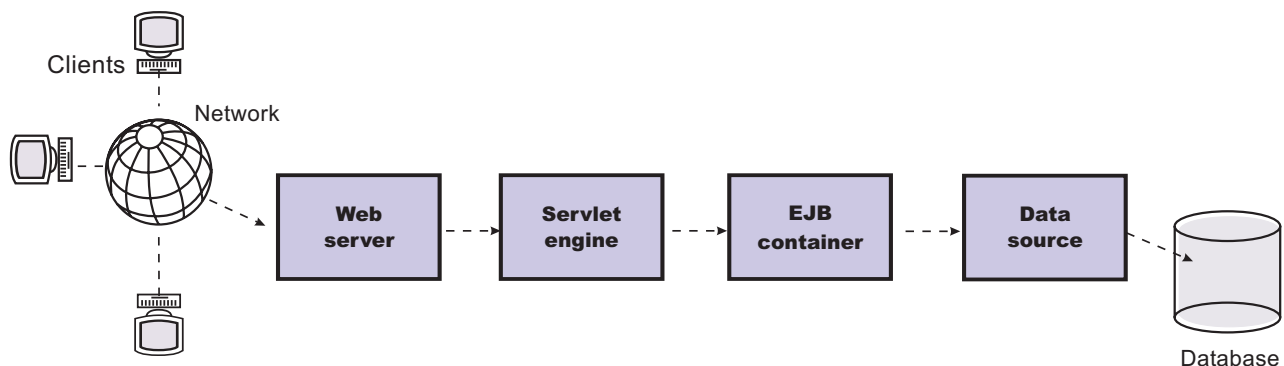


Figure Reference 1: WebSphere queuing network

Most of the queues that make up the queuing network are closed queues. A closed queue places a limit on the maximum number of requests present in the queue, while an open queue has no limit. A closed queue supports tight management of system resources. For example, the Web container thread pool setting controls the size of the Web container queue. If the average servlet running in a Web container creates 10MB of objects during each request, a value of 100 for thread pools limits the memory consumed by the Web container to 1GB.

In a closed queue, requests can be active or waiting. An active request is doing work or waiting for a response from a downstream queue. For example, an active request in the Web server is doing work,

such as retrieving static HTML, or waiting for a request to complete in the Web container. A waiting request is waiting to become active. The request remains in the waiting state until one of the active requests leaves the queue.

All Web servers supported by WebSphere Application Server are closed queues, as are WebSphere Application Server data sources. You can configure Web containers as open or closed queues. In general, it is best to make them closed queues. EJB containers are open queues. If there are no threads available in the pool, a new one is created for the duration of the request.

If enterprise beans are called by servlets, the Web container limits the number of total concurrent requests into an EJB container, because the Web container also has a limit. The Web container limits the number of total concurrent requests only if enterprise beans are called from the servlet thread of execution. Nothing prevents you from creating threads and bombarding the EJB container with requests. Therefore, servlets should not create their own work threads.

Establishing multimachine environments

Setting up a Network Deployment environment involves several steps performed on each of the systems that comprise the cell.

1. Optional: Start the administrative server from an earlier version of WebSphere Application Server.

If you intend to migrate the applications and configuration from an earlier version, you might need to start the administrative server so that the installation wizard can use XMLConfig to export the configuration data repository as it performs the automated migration that you can select in the next step.

Start the administrative server of WebSphere Application Server Standard Edition (Version 3.5.x) or WebSphere Application Server Advanced Edition (Versions 3.5.x or 4.0.x).

It is not necessary to start the administrative server for WebSphere Application Server Advanced Single Server Edition, Version 4. The migration tools use the XML configuration files directly.

2. Install the base WebSphere Application Server product on each machine to create a WebSphere Application Server node on the machine.

Install the base WebSphere Application Server product for the machine to become a node in the cell. The installation procedure is the same for a node that federates into a cell as it is for a stand-alone Application Server. You can install the base WebSphere Application Server product more than once on a single machine. Coexistence is supported. See “Setting up Version 5 coexistence” on page 360. There are several ways to federate a stand-alone Application Server installation instance into the deployment manager cell. See “Federating multiple Version 5 installation instances” on page 362.

You can install the base WebSphere Application Server product once and create multiple configuration instances on the machine, using the **wsinstance** command. You can also install the Network Deployment product once and create multiple configuration instances of the deployment manager. See “Creating multiple Version 5 configuration instances” on page 193. Each deployment manager configuration instance can federate stand-alone base WebSphere Application Server product installation instances, but a deployment manager cannot federate base product configuration instances.

Migrate applications, security settings, and the remaining configuration from WebSphere Application Server, Version 3.5.x and later, or WebSphere Application Server, Version 4.0.x and later. You can also choose to coexist with WebSphere Application Server, Versions 3.5.5 and later, or Versions 4.0.2 and later. Both migration and coexistence are described in the installation procedure for the base WebSphere Application Server product, which is available from the information center for the WebSphere Application Server product.

Stop the administrative server from the earlier version before you perform the installation verification test, which starts the new server. This avoids potential port conflicts.

3. Install the WebSphere Application Server Network Deployment product on a machine. Only one system hosts the deployment manager.

As the deployment manager federates base WebSphere Application Server nodes, it expands the cell that it manages. Although you can install a base WebSphere Application Server on the same machine as the deployment manager, it is not usual in a production environment unless you have a machine with the capacity to host both products.

The deployment manager is the central administrative manager. It does not install the base WebSphere Application Server product on other machines. The only functions supported in the Network Deployment installation are the deployment manager and its associated administrative programs.

Migrate applications, security settings, and the remaining configuration from WebSphere Application Server, Version 3.5.x and later, or WebSphere Application Server, Version 4.0.x and later. You can also choose to coexist with WebSphere Application Server, Versions 3.5.5 and later, or Versions 4.0.2 and later. Migration and coexistence are described in the installation procedure for the WebSphere Application Server Network Deployment product. See “Installing the product” on page 43.

4. Start the deployment manager process.

There are two ways to start the deployment manager:

- As a monitored process that restarts automatically if a failure occurs, as described in “Automatically restarting server processes” on page 176
- As an unmonitored process, which is how the startManager command starts the deployment manager

Run the startManager.sh or startManager.bat script from the /bin directory of the installation root of the deployment manager.

For production systems, running the deployment manager as a monitored process is recommended.

5. Run the addNode command script on every node that you plan to federate into the cell.

Alternatively, you can use the addNode.bat script.

The **addNode** command incorporates a base WebSphere Application Server product node into a deployment manager cell. You must run this tool on every system that you plan to make part of a Network Deployment cell. There are several parameters for the **addNode** command, but the most important are includeapps, the host name of the deployment manager node, the JMX connector type, and the JMX port of the deployment manager node.

For example:

```
./addNode.sh wasdoct 8889 -includeapps
```

The example adds the base node on which the command runs to the cell managed by the wasdoct deployment manager node, using the default SOAP JMX connector type at port 8889. (Port 8889 is a coexistence value for the SOAP port. On the wasdoct development machine, the base node uses the default SOAP port.) The command adds all applications on the base node into the cell configuration.

As it federates the base node in response to the **addNode** command, the deployment manager also instantiates the node agent process, nodeagent, on the Application Server node. If you installed the embedded messaging server feature on the base node, the deployment manager instantiates the Java Message Service (JMS) provider that WebSphere Application Server provides, as the jmsserver process on the base node.

Alternatively, you can use the administrative console of the deployment manager to add running Application Server nodes to the cell. See Using the administrative console .

6. Enable the appropriate level of security after the installation is complete, as described in Securing your environment after installation .

7. Develop and unit test application components, as described in Developing .

Load existing application components and modules into your development environment and debug them.

8. Assemble code into a main application module or enterprise archive (EAR) file, as described in Assembling or packaging .

9. Start all servers in the test environment, as described in Starting servers .

10. Deploy your applications in the test environment, as described in Deploying .

11. Test all applications thoroughly, as described in [Testing](#) .
Follow normal test procedures as you move the test environment into production. Review the information in the [Migrating](#) topic to understand what you must look for. In particular, review the table at the end of the topic that links you to specific recommendations and practices.
You must configure your migrating applications to verify that they migrate correctly, as described in [“Configuring WebSphere Application Server after migration”](#) on page 371.
12. Prepare and monitor the environment into which you deploy applications, as described in [Administering](#) .
13. Adjust application code, configurations, and system settings to improve performance, as described in [Tuning](#) .
14. Fix any known problems, as described in [Troubleshooting](#) or [problem determination](#) .
15. Set up your production system by configuring all server processes for monitoring by their operating systems, as described in [“Automatically restarting server processes”](#) on page 176.

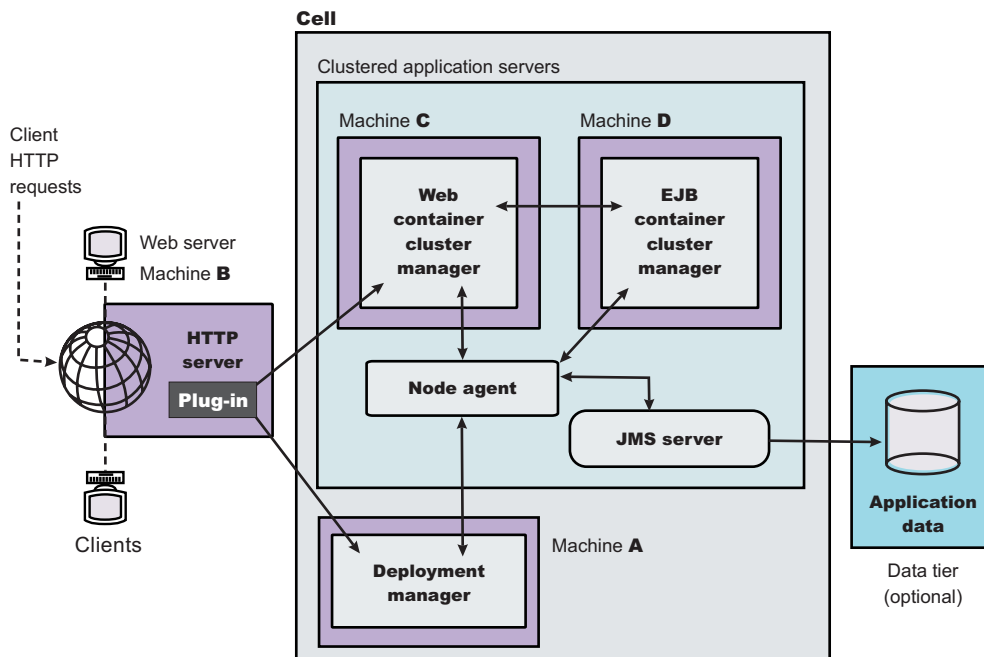
You can create a working WebSphere Application Server cell.

Use the administrative console as described in [Using the administrative console](#), or use other administrative tools, as described in [Welcome to System Administration](#), to observe and control the incorporated nodes, and the resources on these nodes. The console provides a central location for configuring, monitoring, and controlling all Application Servers on all nodes within the cell.

Setting up a multinode environment

A multinode environment places WebSphere Application Server processes on separate physical machines, under the central management of the deployment manager process, which groups Application Servers into its managed *cell*. This example topology shows how you can have a tier of Application Servers within the cell in addition to traditional tiers that can contain the Web server, databases, enterprise information systems, and other types of persistent storage.

The following illustration shows an example of multiple nodes and multiple tiers.



In this example, Web container cluster members (Machine C) are closer (in a network sense) to the HTTP server (Machine B), which improves their response to client requests. Application server processes that run enterprise beans (Machine D) are closer in network terms to application data, which is represented in an application by entity beans and stored in a database on the data tier.

Clustered Application Servers on Machines C and D help maximize resource use on each machine. Although the example shows two cluster members in each node, you can have as many as a machine can support. Cluster members in a multitiered topology provide process redundancy and use memory more efficiently than in similar topologies that host only single instances of Application Servers. Additional resources on the machines can improve application throughput and performance.

Introducing firewalls between each pair of tiers can provide the same level of security for entity beans as for application data.

A multitiered topology within the cell eliminates local Java virtual machine (JVM) optimizations that occur when the same cluster member runs both the Web container and the EJB container. This topology also introduces network latency, which tends to slow system performance. Although multitiered topologies provide more redundancy for Application Server processes, they also introduce more possible points of failure. The level of redundancy can also complicate maintenance.

The deployment manager on Machine A manages the configuration of cluster members and other Application Server processes on Machines C and D. The deployment manager coordinates all Application Server processes through node agent processes, each of which runs as the nodeagent server on a node.

Setting up environments such as those shown in the illustrations involves performing several steps on each computer comprising the deployment manager cell.

1. Install the Network Deployment product on Machine A, to make it the deployment manager node, using the product CD-ROM labeled, **Deployment Manager**, from the product package.

5.1+ Issue the **install** command on Linux platforms and UNIX platforms or the **Install.exe** command on Windows platforms. The command is in the platform root directory on the CD-ROM.

Machine A is the system for the deployment manager server (dmgr), which provides a centralized administrative console for the entire group, or cell, of application servers that it controls. Installing the Network Deployment product does not install the base Application Servers. The Network Deployment installation installs only the deployment manager process and its associated administrative programs. Install the Network Deployment product on only one computer system from the set that is to make up your administrative cell.

2. Install the base WebSphere Application Server product on Machines C and D to make them Application Server nodes, using the product CD-ROM labeled, **Application Server, IBM HTTP Server**, from the product package.

Issue the **install** command on Linux platforms and UNIX platforms or the **install.bat** command on Windows platforms. The command is in the platform root directory on the CD-ROM.

Aside from the one computer where you installed the deployment manager node, you must install the base WebSphere Application Server product on other computers that are to comprise the cell. The installation process is the same for an Application Server (server1) that you federate into a cell as it is for a stand-alone Application Server.

3. Start the deployment manager process on Machine A.

There are two ways to start the deployment manager:

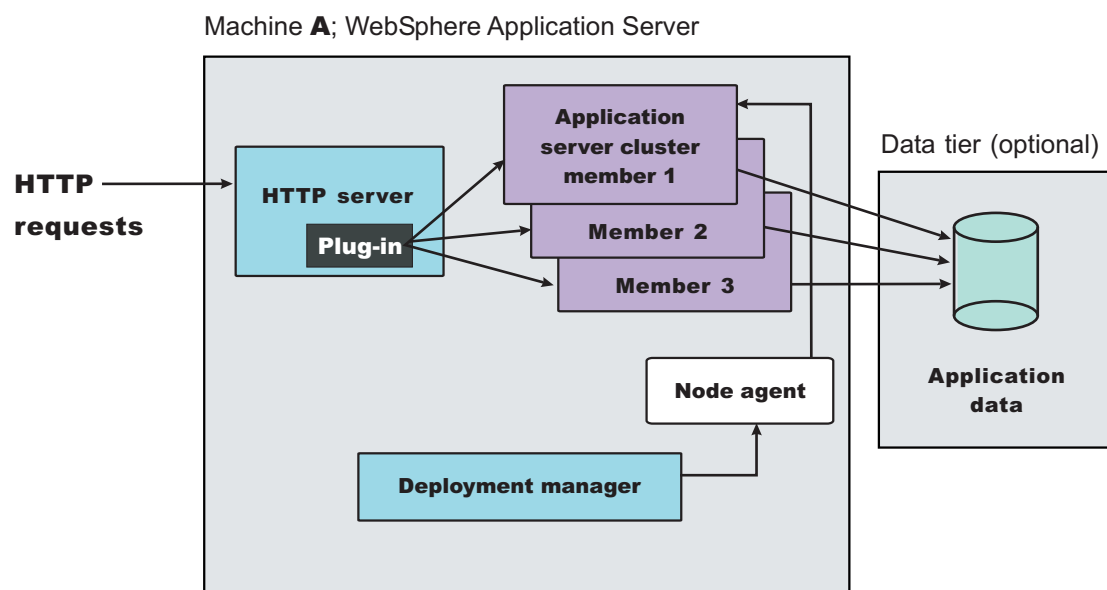
- As a monitored process that restarts automatically if a failure occurs, as described in “Automatically restarting server processes” on page 176
- As an unmonitored process, which is how the startManager command starts the deployment manager

4. Run the addNode command script on every node that you plan to federate into the cell, such as Machine C and Machine D.

Use the administrative console as described in Using the administrative console, or use other administrative tools, as described in Welcome to System Administration, to observe and control the incorporated nodes, and the resources on these nodes. The console provides a central location for configuring, monitoring, and controlling all Application Servers on all nodes within the cell.

Vertical scaling topology

Vertical scaling refers to setting up multiple application servers on one machine, usually by creating cluster members.



This topology illustrates a simple vertical scaling example, with multiple cluster members of an Application Server on Machine A. You can also implement vertical scaling on more than one machine in a configuration. Combine vertical scaling with other topologies to boost performance and throughput.

Typical use

Vertical scaling offers the following advantages:

- **Increased processing power efficiency.** An instance of an application server runs in a single Java virtual machine (JVM) process. However, the inherent concurrency limitations of a JVM process prevent it from fully utilizing the processing power of a machine. Creating additional JVM processes provides multiple thread pools, each corresponding to the JVM process associated with each Application Server process. This correspondence avoids concurrency limitations and lets the Application Server use the full processing power of the machine.
- **Load balancing.** Vertical scaling topologies can use the WebSphere Application Server workload management facility.
- **Process failover.** A vertical scaling topology also provides failover support among Application Server cluster members. If one Application Server instance goes offline, the other instances on the machine continue to process client requests.

Single machine vertical scaling topologies have the drawback of introducing the host machine as a single point of failure in the system. Vertical scaling on multiple machines avoids the single point of failure.

Instructions

To set up a vertical scaling topology, use the administrative console to configure a set of Application Server cluster members that reside on the same machine.

It is recommended that you plan vertical scaling configurations ahead of time. However, because vertical scaling does not require any special installation steps, you can implement vertical scaling whenever it is needed.

While you are deciding how many cluster members to create on a machine, take these factors into account:

- **The design of the application.** Applications that use more components require more memory, limiting the number of cluster members you can run on a machine.
- **The hardware environment.** Vertical scaling works best with plenty of memory and processing power. Eventually there is a point of diminishing returns on any machine, where the overhead of running more cluster members cancels out the benefits of adding them.

The best way to ensure good performance in a vertical scaling configuration is to tune a single instance of an Application Server for throughput and performance, then incrementally add cluster members. Test performance and throughput as you add each cluster member. Always monitor memory use when you are configuring a vertical scaling topology, so you do not exceed available physical memory on the machine.

Multimachine topology concepts

Multiple machine environments extend basic single machine WebSphere Application Server configurations by distributing the Application Server over multiple machines, increasing the overall processing power from one machine to contributions from all machines in the configuration.

The flow of data in a WebSphere Application Server environment starts with a Web server receiving requests and routing them to the Application Server for processing. A WebSphere Application Server node stores administrative configuration data in XML files. A database can hold application data for applications that require a place to store data, such as user session information. There are also one or more administrative clients, such as the administrative console, for manipulating configuration data.

Some of the reasons for creating WebSphere Application Server applications that run on multiple machine systems include:

- **Scalability.** Adding more machines increases processing power, which scales up the system to handle a higher client load than that provided by a basic, single machine configuration. Scalability pertains to the capability of a system to adapt readily to a greater or lesser intensity of use, volume, or demand. For example, a scalable system can efficiently adapt to work with larger or smaller networks performing tasks of varying complexity. Ideally, it is possible to handle any given load by adding more servers and machines, assuming each additional machine processes its fair share of client requests. Each machine should process a share of the total system load that is proportional to the processing power of the machine.

Consider these primary scalability factors when adopting a WebSphere Application Server topology:

- **Security.** Address certain security concerns by physically separating the Web server from the Application Server, by using firewalls.
- **Performance.** Maximize performance by ensuring the response time for transactions is as short as possible. You can use two general topologies to improve transaction performance:
 - *Vertical scaling*, in which you create additional Application Server processes on a single physical machine. The *Vertical scaling sample topology* topic describes a physical implementation for vertical scaling.
 - *Horizontal scaling*, in which you create additional application server processes on multiple physical machines to take advantage of the additional processing power available on each machine. You can also use WebSphere Application Server Edge Components, such as the Caching Proxy Edge component, and the Load Balancer component set (which includes the Dispatcher component), to implement horizontal scaling. The *Horizontal scaling with cluster members sample topology* and *Horizontal scaling with Network Dispatcher sample topology* topics describe physical implementations for horizontal scaling.
- **Throughput.** Add Application Server clusters to scale vertically or horizontally. Application server clusters can increase the number of concurrent transactions that the application can perform, to help process as many transactions as possible within a given time period.
- **Availability and failover support.** Avoid a single point of failure and maximize system availability by ensuring that the topology has some degree of process redundancy. High-availability topologies typically involve horizontal scaling across multiple machines. Vertical scaling can improve availability by creating multiple processes, but the machine becomes a point of failure.

A Dispatcher server performs intelligent load balancing to determine where to send a TCP/IP request. It can direct client HTTP requests to available Web servers, bypassing any that are offline. Another server can back up the Dispatcher server, to eliminate it as a single point of failure. Workload management of Application Servers and administrative servers also improves availability and failover support.

Failover support distributes client requests to the remaining servers, which ensures continued client access without significant interruptions. (In practice, failover is not entirely transparent to clients.)

WebSphere Application Server supports these methods of ensuring availability with multiple machines and applications:

- **Multiple tiers** The components of an application (the Web server, Application Servers, databases, and so forth) are physically separated on different machines.
- **HTTP server separation** The Web (HTTP) server is located on a different physical machine than the Application Server. You can redirect requests to Application Servers through a variety of methods.
- **Demilitarized zone (DMZ)** Firewalls create demilitarized zone machines, which are isolated from both the public Internet and other machines in the configuration. The DMZ scales security processing, which improves security and throughput in the application environment.
- **Multiple cells** Multiple WebSphere Application Server cells provide failover for clustering Application Servers and deploying applications.
- **Multiple applications** Multiple application instances can be on the same physical machine, or on more than one machine in the cell.
- **Maintainability.** Understand that the topology affects the ease with which you can update system hardware and software. For instance, using multiple WebSphere Application Server deployment

manager cells or horizontal scaling can make a system easier to maintain because you can take individual machines offline without interrupting other machines, running the application.

Maintainability sometimes conflicts with other topology considerations. For example, limiting the number of Application Server instances makes the application easier to maintain but can have a negative effect on throughput, availability, and performance.

- **Maintaining session state between client HTTP requests.** Consider that session state is important for stateful applications, or for applications that run on multiple machines or Application Server instances. You can share a session between multiple Application Server processes (cluster members) by saving the session state to a database. In addition, configuring a network dispatcher affects how the session state is maintained. This consideration does not apply if your application runs on a single Application Server instance or is completely stateless.
- **Shared data access.** Placing backend resources, such as databases, on different machines provides ease of use when sharing these resources.
- **Fault isolation.** Providing more robust failover support through a configuration that includes a degree of fault isolation, reduces the potential for failure of one server to affect other servers. Configurations that provide simple failover support are concerned only with individual server failures that have no effect on the performance of other servers. However, in some situations, a malfunctioning server can create problems for other servers that are otherwise functioning normally. For example, a failing server can consume more than its share of system and database resources, preventing other servers from gaining adequate access to these resources. You can configure WebSphere Application Server to provide fault isolation between different parts of a system.
- **Dynamic changes to configurations.** Modifying the system configuration without interrupting its operation enhances the manageability and flexibility of the system. For instance, administrators can add or remove cluster members to handle variations in the client load, change server characteristics and propagate the changes to its cluster members, temporarily stop servers for maintenance, and so forth.
- **Mixed Application Server versions.** Migrating a few machines and applications at one time is possible on certain multiple machine configurations. In addition to Version 5 Application Servers, your server configuration can include earlier version Application Servers, where you currently deploy all applications you intend to migrate. You can migrate applications to Version 5 and deploy them in stages. You can also easily upgrade system hardware and software. When combined with the ability to make dynamic changes to the configuration, you can use a configuration of mixed Application Server versions to upgrade an application or machine without any interruption of service.

The ability to run different versions of an Application Server in a configuration does not let you include Version 5 Application Servers in an existing administrative domain, nor does it let you include Version 3.5.x or Version 4.0.x application servers in a Version 5 cell.

You can configure multiple machines to add processing power, improve security, maximize availability, and balance workloads. WebSphere Application Server Network Deployment and WebSphere Application Server Edge Components provide clusters, workload management, and the Dispatcher to implement configurations that address these issues. These scaling techniques are generally combined to maximize benefits and minimize problems associated with multiple machine systems.

There are various ways to scale up the basic, single-machine WebSphere Application Server system to meet the needs of your organization. Some scaling techniques include:

- **Clusters.** Clusters are identical multiple Application Server copies. A cluster member is one Application Server in a cluster. The configuration of each cluster member is based upon the Application Server that you copied to create the cluster member. You can create all cluster members on the same physical machine, or on different machines. Using cluster members can improve the performance of a server, simplify its administration, and enable the use of workload management.
- **Workload management (WLM).** WLM enables both load balancing and failover, improving the reliability and scalability of applications deployed on the WebSphere Application Server server. Incoming processing requests from clients are transparently distributed among the cluster members of an Application Server.
- **Dispatcher.** The Dispatcher transparently redirects incoming HTTP requests from Web clients to a set of Web servers. Although the clients behave as if they are communicating directly with a given Web

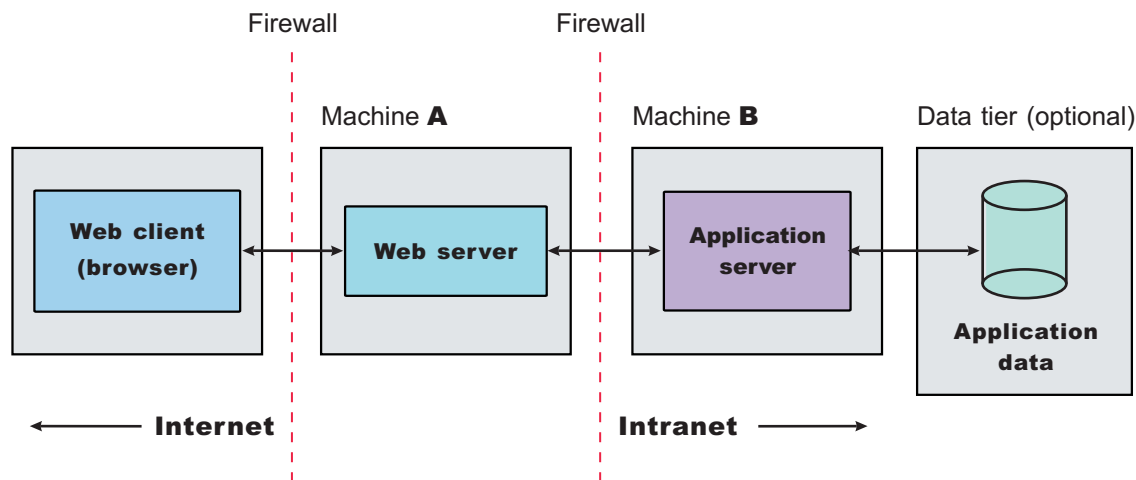
server, the Dispatcher is actually intercepting all requests and distributing them among all the available Web servers in the cluster. The Dispatcher can provide scalability, load balancing, and failover for Web servers.

Keep in mind that the techniques described above are not mutually exclusive. You can combine their basic elements in various ways.

Firewalls and demilitarized zone configurations:

Firewalls protect backend resources, such as databases in multiple machine systems. You can also use firewalls to protect Application Servers and Web servers from unauthorized outside access. A demilitarized zone (DMZ) configuration involves multiple firewalls that add layers of security between the Internet and critical data and business logic.

A wide variety of topologies are appropriate for a DMZ environment. Although WebSphere Application Server provides great flexibility in configuring DMZ topologies, the basic locations of elements in a simple DMZ topology follow:



The main purpose of a DMZ configuration is to protect the business logic and data in the environment from unauthorized access. A typical DMZ configuration includes:

- An outer firewall between the public Internet and the Web server or servers processing the requests originating on the company Web site.
- An inner firewall between the Web server and the Application Servers to which it is forwarding requests. Company data also resides behind the inner firewall.

The area between the two firewalls gives the DMZ configuration its name. Additional firewalls can further safeguard access to databases holding administrative and application data.

Comparison of DMZ configurations

Somehow, requests for applications that WebSphere Application Server manages must get from the Web server to the Application Servers, passing through firewalls. You can implement DMZ configurations for a wide variety of multitiered systems. WebSphere Application Server offers many configuration choices for accomplishing this goal. The following table summarizes benefits of each DMZ configuration option supported by the product. Criteria for each topology are described after the table.

An **X** represents an advantage.

Benefit (X) or statistic	Remote HTTP	Reverse proxy
Compatible with product security	X	X
Avoids data access from DMZ	X	X
Supports Network Address Translation (NAT)	X	X
Avoids DMZ protocol switch		X
Allows encrypted link between Web server and Application Server	X	Depends on Web server
Avoids single point of failure	X	
Minimum firewall holes	One per Application Server, plus one if WebSphere Application Server security is used on the Web server machine	One

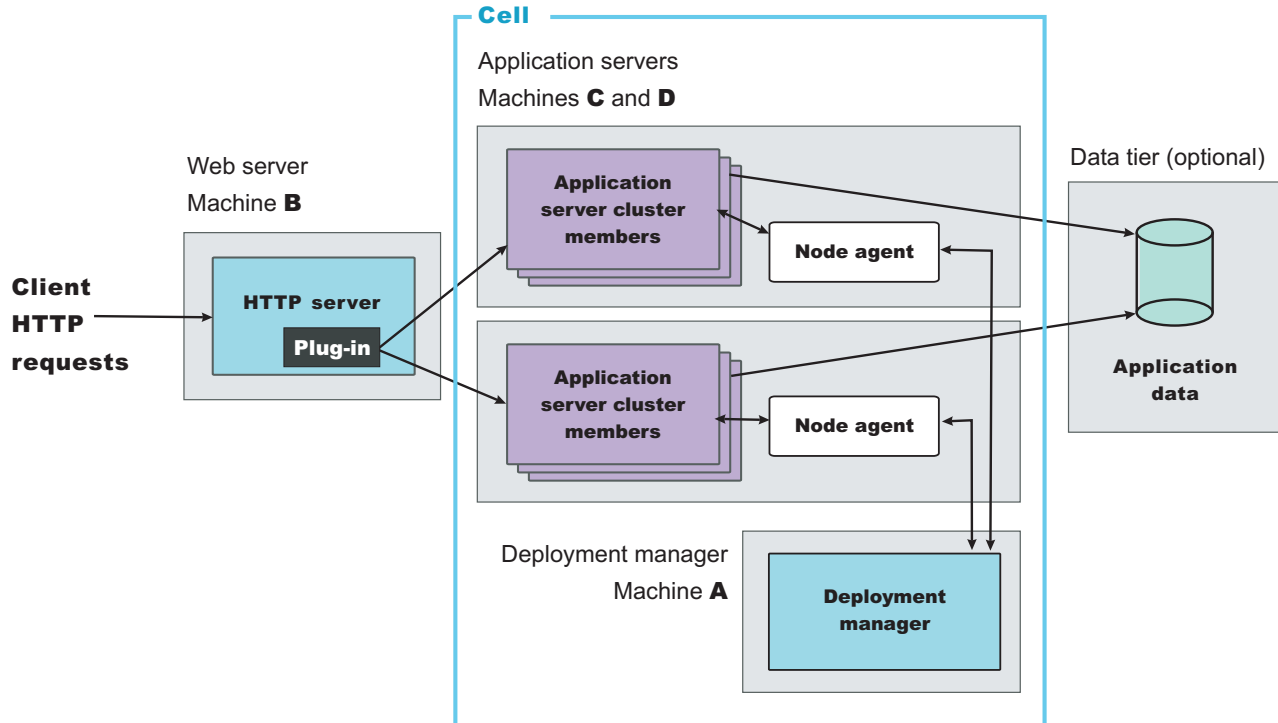
- **Works with product security.** WebSphere Application Server security protects applications and their components, by enforcing authorization and authentication policies. Configuration options compatible with product security are desirable because they do not necessitate alternative security solutions.
- **Avoids critical business data in the DMZ.** A DMZ configuration protects application logic and data, by creating a buffer between the public Internet Web site and the internal intranet, where Application Servers and the data tier reside. Desirable DMZ topologies do not have databases or application servers with critical business data in the DMZ.
- **Supports Network Address Translation (NAT).** A firewall product that runs NAT receives packets for one IP address, and translates the headers of the packet to send the packet to a second IP address. In environments with firewalls employing NAT, avoid configurations involving complex protocols in which IP addresses are embedded in the body of the IP packet, such as Java Remote Method Invocation (RMI) or Internet Inter-Orb Protocol (IIOP). These IP addresses are not translated, making the packet useless.
- **Avoids the DMZ protocol switch.** The Web server sends HTTP requests to Application Servers behind firewalls. It is simplest to open an HTTP port in the firewall to let the requests through. Configurations that require switching to another protocol, such as IIOP, and opening firewall ports corresponding to the protocol, are less desirable. They are often more complex to set up, and the protocol switching overhead can impact performance.
- **Allows an encrypted link between Web server and Application Server.** Configurations that support encryption of communication between the Web server and application server reduce the risk that attackers are able to obtain secure information by *sniffing* packets sent between the Web server and Application Server. A performance penalty usually accompanies such encryption.
- **Avoids a single point of failure.** A point of failure exists when one process or machine depends on another process or machine. A single point of failure is especially undesirable because if the point fails, the whole system becomes unavailable. When comparing DMZ solutions, a single point of failure refers to a single point of failure between the Web server and Application Server. Various failover configurations can minimize downtime and possibly even prevent a failure. However, these configurations usually require additional hardware and administrative resources.
- **Minimizes the number of firewall holes.** Configurations that minimize the number of firewall ports are desirable because each additional firewall port leaves the firewall more vulnerable to attackers.

Some solutions are faster than others, in terms of the number of client requests they can process per unit of time. Some solutions require little or no maintenance after you establish them, while others require periodic administrative steps, such as stopping a server and starting it again after modifying resources that affect the configuration. To learn about the necessary maintenance for a topology, review the instructions for setting up and maintaining that topology. Of course, if you can automate the necessary administrative steps through command line clients and scripting, this might not concern you.

Horizontal scaling topology

Horizontal scaling exists when there are members of an application server cluster on multiple physical machines. Having cluster members on several machines lets a single application span the machines, yet still present a single system image.

The following figure shows an example of horizontal scaling.



In this example, the Web server on Machine B distributes requests to clustered Application Servers on Machines C and D. Cluster members on Machines C and D are created in the same cluster.

You can combine a network dispatcher to distribute client HTTP requests with clustering, to reap the benefits of both types of horizontal scaling. The *Network dispatcher topology* topic describes this system configuration.

Typical use

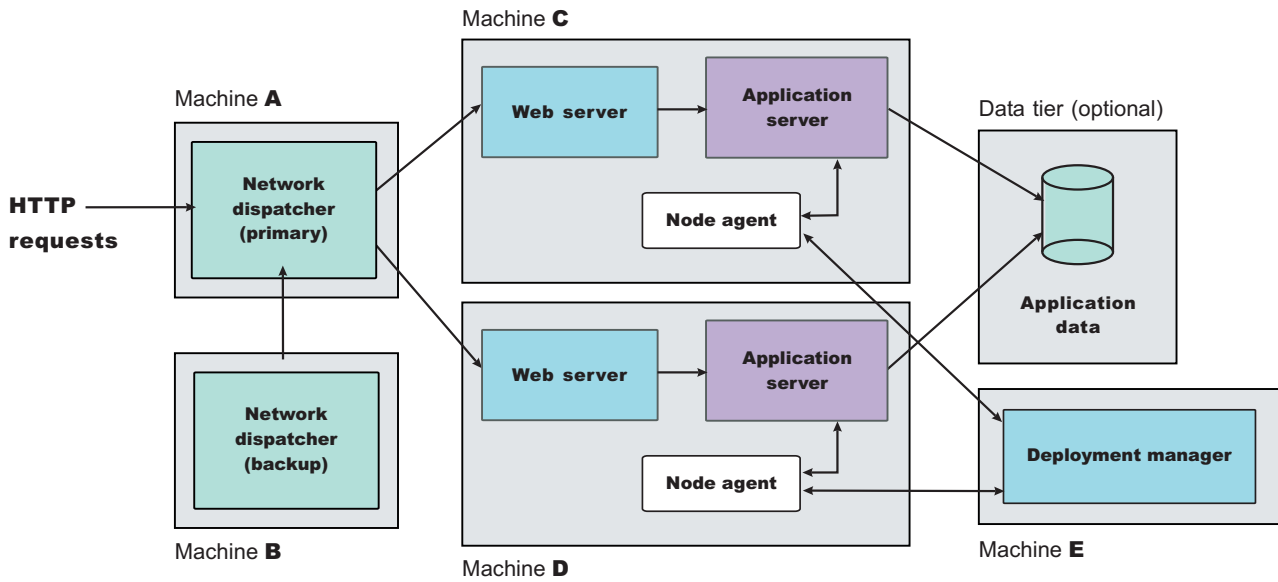
Horizontal scaling provides the increased throughput of vertical scaling topologies but also provides failover support. This topology lets you handle Application Server process failure and hardware failure without significant interruption to client service. You can also use horizontal scaling to optimize the distribution of client requests through mechanisms, such as workload management or remote HTTP transport.

Dispatcher

The Dispatcher component is part of the Load Balancer component set of the IBM WebSphere Application Server Edge Components product. The Edge components are included in WebSphere Application Server Network Deployment. The Dispatcher performs intelligent load balancing by using server availability, capability, workload, and other criteria you can define, to determine where to send a TCP/IP request. You can use the Dispatcher to distribute HTTP requests among Application Server instances that are running on multiple physical machines.

A simple Dispatcher topology

The following figure illustrates a simple horizontal scaling configuration that uses the Dispatcher to distribute requests among Application Servers on different machines.

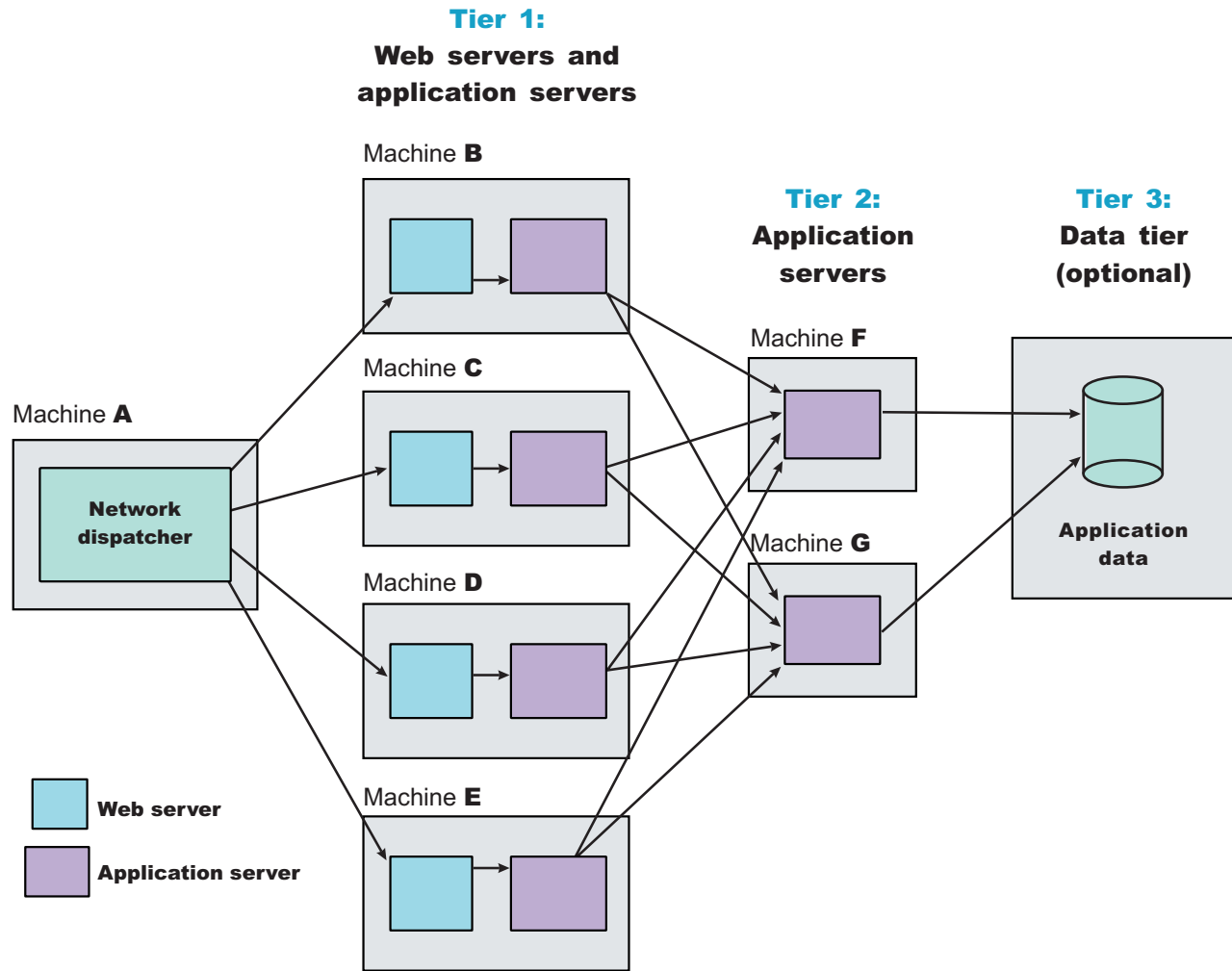


You normally configure a backup node for the Dispatcher machine, to eliminate it as a single point of failure. In this example, you can set up the backup Dispatcher node (Machine B) to take over if the primary Dispatcher node (Machine A) fails.

You can cluster Application Servers in this example from the same model, or configure them independently.

A more complex Dispatcher topology

The next figure shows a Dispatcher that distributes requests among several machines containing clustered Web servers and Application Servers. Backups are not shown.



The first tier Web server machines host servlet-based applications. The second tier Application Servers contain mostly enterprise beans that access application data and execute business logic. This approach lets you employ numerous, less powerful machines on the first tier and fewer but more powerful machines on the second tier.

Using the Dispatcher with firewalls

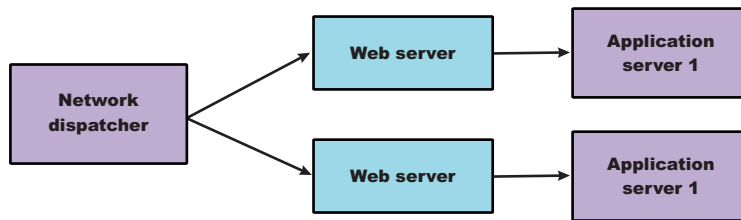
You can also use a load balancing product, such as the Dispatcher with demilitarized zone (DMZ) topologies. The Dispatcher simplifies creating a DMZ topology with two firewalls. One firewall protects the Web server from the public Web site. The second protects backend systems from the Web server in the DMZ, by using proxy services.

The Dispatcher machine is placed between the outside firewall and the cluster of Web servers that it serves. The outside firewall provides filtering to allow only HTTP and HTTPS traffic. The firewall to the backend systems (DBMS, CICS, SAP, and so on) handles non-HTTP protocols, such as IIOP and JDBC. The WebSphere Application Server can reside in the DMZ or on the same side of the firewall as the data tier.

The Dispatcher and session affinity

In a Dispatcher or similarly configured topology, you must associate a Web server with a separate, rather than clustered, Application Server to preserve affinity between the servers.

Supports affinity



Discussion

Adding a mechanism for the Dispatcher to distribute HTTP requests provides these advantages:

- Improves server performance, by distributing incoming TCP/IP requests - in this case, HTTP requests, among a cluster of servers.
- Increases the number of connected users.
- Eliminates the Web server as a single point of failure. You can also use this mechanism in combination with WebSphere Application Server workload management, to eliminate the Application Server as a single point of failure.
- Improves throughput, by letting multiple servers and CPUs handle the client workload.

Instructions

Set up machines containing Web servers and Application Servers for the topology you plan to implement.

Place the Dispatcher, or another load balancing product, in front of the Web server machines. See the Edge Component documentation for the Dispatcher, or the other load balancing product documentation. Instructions vary per product.

The load balancing product communicates with the Web server, which in turn communicates with Application Servers. The configuration involves setting up communications between the load balancing product and the Web server.

It does not matter to the Dispatcher whether the Web server is routing requests to an Application Server or processing them itself. Therefore, it is not necessary to perform any special configuration to make the load balancing product and Application Servers aware of one another. This lack of configuration is true with the Dispatcher, based on testing with IBM WebSphere Application Server. Results can vary with other load balancing products.

Web server separation

Web server separation is a topology that physically separates the Web (HTTP) server from the Application Servers, placing the Web server on a different machine in the configuration. Compared to a configuration where the Web server and the Application Servers are located on the same physical server, separating the Web server can improve application performance, provide better fault isolation, and enhance security. These topologies are often used with firewalls to create a secure demilitarized zone (DMZ) surrounding the Web server.

WebSphere Application Server provides these alternatives for physically separating the Web server from the Application Server:

- HTTP transport configurations
- Reverse proxy or IP forwarding configurations

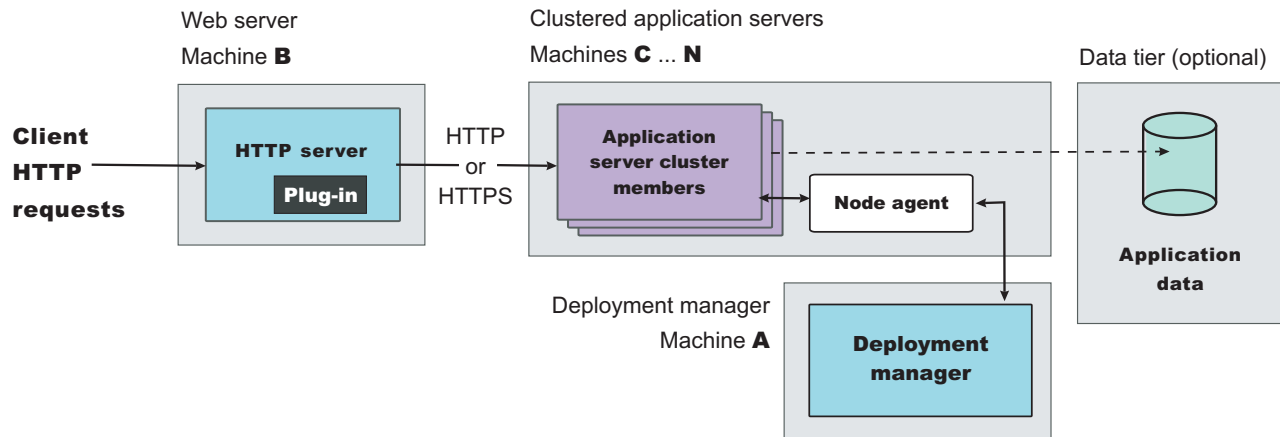
The following table summarizes advantages and disadvantages of each configuration. Criteria are explained after the table.

Topology	Secure Sockets Layer	Database password required?	Workload management	Network Address Translation	Performance	Administration
HTTP server separation	Yes	No	Yes	Yes	High	Manual
Reverse proxy	Yes	No	No	Yes	High	Manual

- **Secure Sockets Layer** Supports Secure Sockets Layer (SSL) security.
- **Database password required?** Requires a stored database user ID and password on the machine used by the database processes.
- **Workload management** Uses the workload management service to balance client workloads.
- **Network Address Translation** Supports Network Address Translation (NAT) firewalls. NAT firewalls receive packets for one IP address, translate the headers of the packets, and send the packets to a second IP address.
- **Performance** Compares the relative performance of each of these configurations.
- **Administration** Specifies whether to administer the configuration manually or through the administrative console.

These criteria give you a basis to compare the relative difficulty of administering each configuration.

HTTP Request routing: WebSphere Application Server can use HTTP to route requests from a Web server to Application Servers on remote machines.



In the diagram, Machine B hosts the Web server and the WebSphere Web server plug-in, and receives HTTP requests from clients. The Web server plug-in forwards the requests to an HTTP transport in the Application Server on Machine C, using HTTP or HTTPS protocol.

Variations on this configuration include vertical scaling of the application servers by creating new application servers on the same machine. Alternatively, for a horizontal scaling scenario, add machines (D, E,N) running application servers to the environment.

The Web server plug-in supports Network Address Translation (NAT) firewalls. See Default settings for port numbers for a list of default ports that WebSphere Application Server uses. See Default coexistence settings for port numbers for a list of default port settings in the coexistence panel. The installer program for the Application Server displays the coexistence panel when it detects another installation of Application Server.

Load balancing support

The Web server plug-in is fully integrated with WebSphere Application Server workload management and the clustering facility. HTTP transport balances loads within a cluster.

If you are using a third party load balancer in between the plug-in and the Application Server for HTTP requests that are not subject to an existing affinity, you can override the round robin algorithm that determines where to route a request. To override this algorithm, specify the IP address of your third party load balancer on the `ClusterAddress` element within the `ServerCluster` tag in `plugin-cfg.xml` file . Specifying this IP address delegates load balancing for the servers within this cluster to the third party load balancer.

- **Load balancing between Application Servers.** You can configure the Web server plug-in to forward requests from each URL to a different application server and its cluster members, enabling manual load balancing. For instance, you can forward URLs that generate a large number of requests to application servers on more powerful machines.
- **Load balancing among cluster members.** The Web server plug-in automatically distributes requests among members of a cluster that is defined to respond to a single URL. The method for selecting which cluster member handles a particular request combines a round-robin selection policy with server affinity. If session persistence is not enabled, which is the default, requests are distributed among all available cluster members using a strict round-robin policy. Each cluster member gets the next request in turn. The only exception is when a cluster member is added or restarted. See the failover support information later in this topic for details.

If session affinity is enabled, requests are distributed as follows:

- The Web server plug-in distributes the first request of each session and all requests that are not associated with a session, as if session persistence is not enabled. They are distributed using a round-robin policy, except when cluster members are added or restarted.
- The Web server plug-in attempts to distribute all requests associated with a particular session to the same cluster member. Different sessions are assigned to different cluster members of the Application Server.

Be aware that there is no guarantee that the same cluster member is used for all requests within a session. You cannot always maintain session affinity in situations where the number of available cluster members changes during the lifetime of a session. The Session Manager session clustering facility ensures that session state is not lost if requests are switched to another cluster member during a session. In any case, applications that require available session information across multiple client invocations must store session information in a database.

Failover support

The Web server plug-in automatically handles failover and changes in the number of available cluster members.

- If a cluster member is stopped or unexpectedly fails, all subsequent requests are distributed among the remaining cluster members. The unavailable cluster member is skipped.
- If a cluster member is added or restarted, the system automatically begins to distribute requests to it. The next several requests are dispatched to that cluster member before HTTP resumes its normal methods for distributing requests to the cluster members of an Application Server, based on whether session persistence is enabled. See the load balancing support information for details.

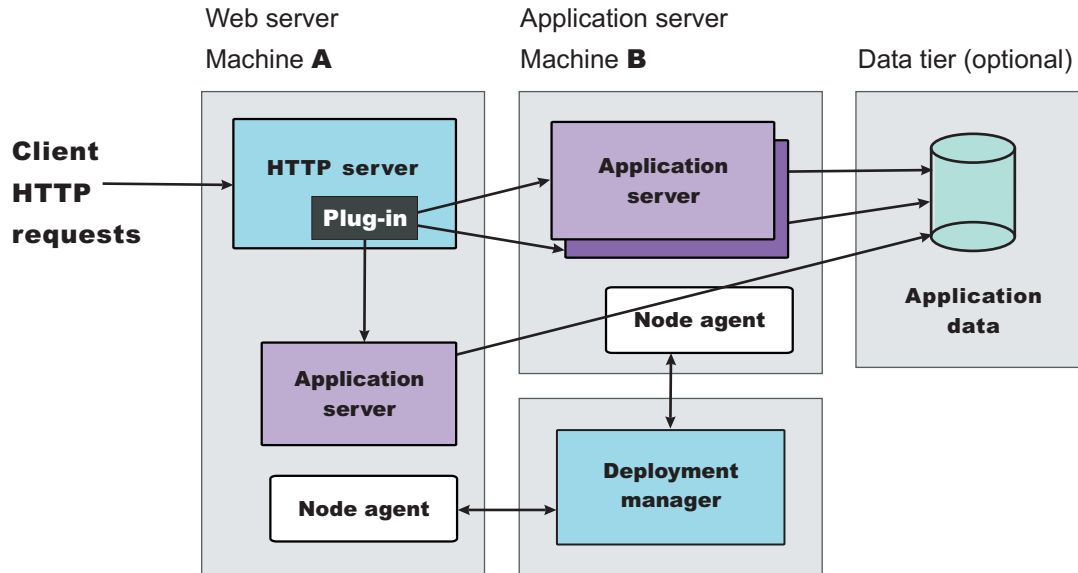
Typical use

The Web server plug-in has the following advantages:

- Supports load balancing and failover.
- Requires no requirements for database access through the firewall.
- Supports WebSphere Application Server security.
- Supports Secure Sockets Layer (SSL) encryption for communications between the Web server and the Application Server.
- Supports Network Address Translation (NAT) firewalls.
- Supports relatively fast performance.

The Web server plug-in has the disadvantage of requiring at least one firewall port, more if multiple Application Server cluster members are configured, or WebSphere Application Server security is used on the machine hosting the Web server.

A variation of the Web server plug-in topology occurs when an instance of the Application Server runs on the same machine as the Web server. Such a configuration is illustrated in the following figure.



Such configurations can direct client requests to additional application server cluster members on other machines. This example redirects client requests to both the Application Server instance running on Machine A and the cluster members running on Machine B. You can administer all Application Server instances from the deployment manager node, which can exist on Machine A, Machine B, or another machine. The deployment manager communicates with the node agent server process on each machine, to coordinate configuration changes.

Typical use

This topology is recommended only in situations where hardware limitations prevent you from hosting the Web server on a dedicated machine.

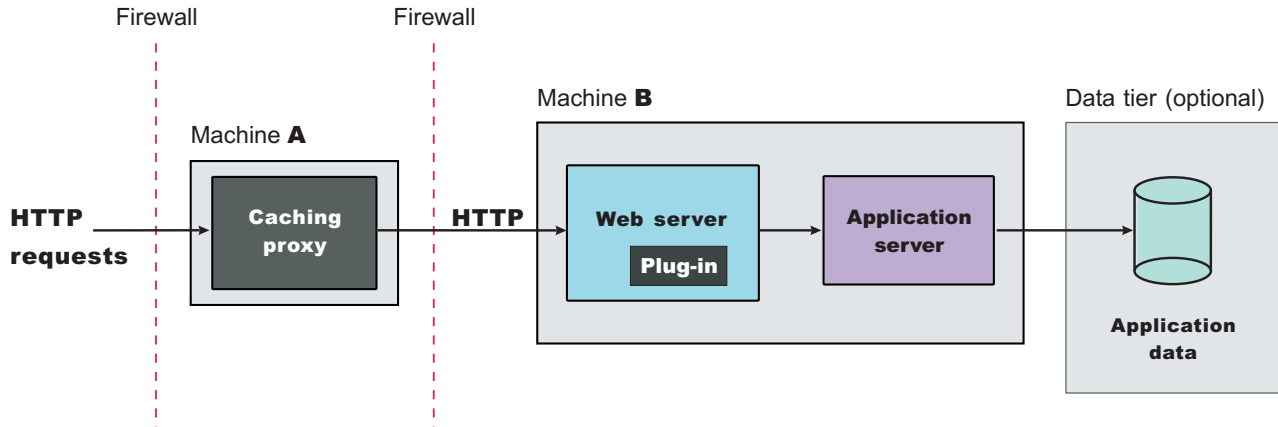
In many production environments, one set of servers is configured to run Web servers and another set of servers is configured to run Application Servers. This configuration lets you add capacity in a production environment. You might also use this topology to more fully replicate a production configuration in a test environment. This topology provides a means of load distribution between a machine hosting both the Web server and Application Server, and machines hosting just the Application Server.

You can also use this topology to distribute the workload in situations where there are a limited number of machines.

Reverse proxy (IP forwarding):

Reverse proxy, or *IP-forwarding* topologies use a reverse proxy server, such as the Caching Proxy in WebSphere Application Server Edge Components, to receive incoming HTTP requests and forward them to a Web server. The Web server forwards the requests to the Application Servers for actual processing. The reverse proxy returns completed requests to the client, hiding the originating Web server.

The following figure shows a simple reverse proxy topology.



In this example, a reverse proxy resides in a demilitarized zone (DMZ) between the outer and inner firewalls. It listens on an HTTP port, typically port 80, for HTTP requests. The reverse proxy then forwards such requests to an HTTP server that resides on the same machine as WebSphere Application Server. After the requests are fulfilled, they are returned through the reverse proxy to the client, hiding the originating Web server.

Typical use

Reverse proxy servers are typically used in DMZ configurations to provide additional security between the public Internet and the Web servers (and application servers) servicing requests. A reverse proxy product used with WebSphere Application Server must support Network Address Translation (NAT) and WebSphere Application Server security.

Reverse proxy configurations support high performance DMZ solutions that require as few open ports in the firewall as possible. The reverse proxy capabilities of the Web server inside the DMZ require as few as one open port in the second firewall, potentially two if using Secure Sockets Layer (SSL) - port 443.

Advantages of using a reverse proxy server in a DMZ configuration include:

- The reverse proxy server does not need database access through the firewall.
- The reverse proxy configuration supports WebSphere Application Server security and NAT firewalls.
- The basic reverse proxy configuration is well known and tested in the industry, resulting in less customer confusion than other DMZ configurations.
- The reverse proxy configuration is reliable and its performance is relatively fast.
- The reverse proxy configuration eliminates protocol switching, by using the HTTP protocol for all forwarded requests.
- The reverse proxy configuration does not affect the configuration and maintenance of an application deployed on WebSphere Application Server.
- The reverse proxy server uses only one HTTP firewall port for requests and responses.

The reverse proxy configuration is also a disadvantage in some environments where security policies prohibit using the same port or protocol for inbound and outbound traffic across a firewall.

Disadvantages of using a reverse proxy server in a DMZ configuration include the following:

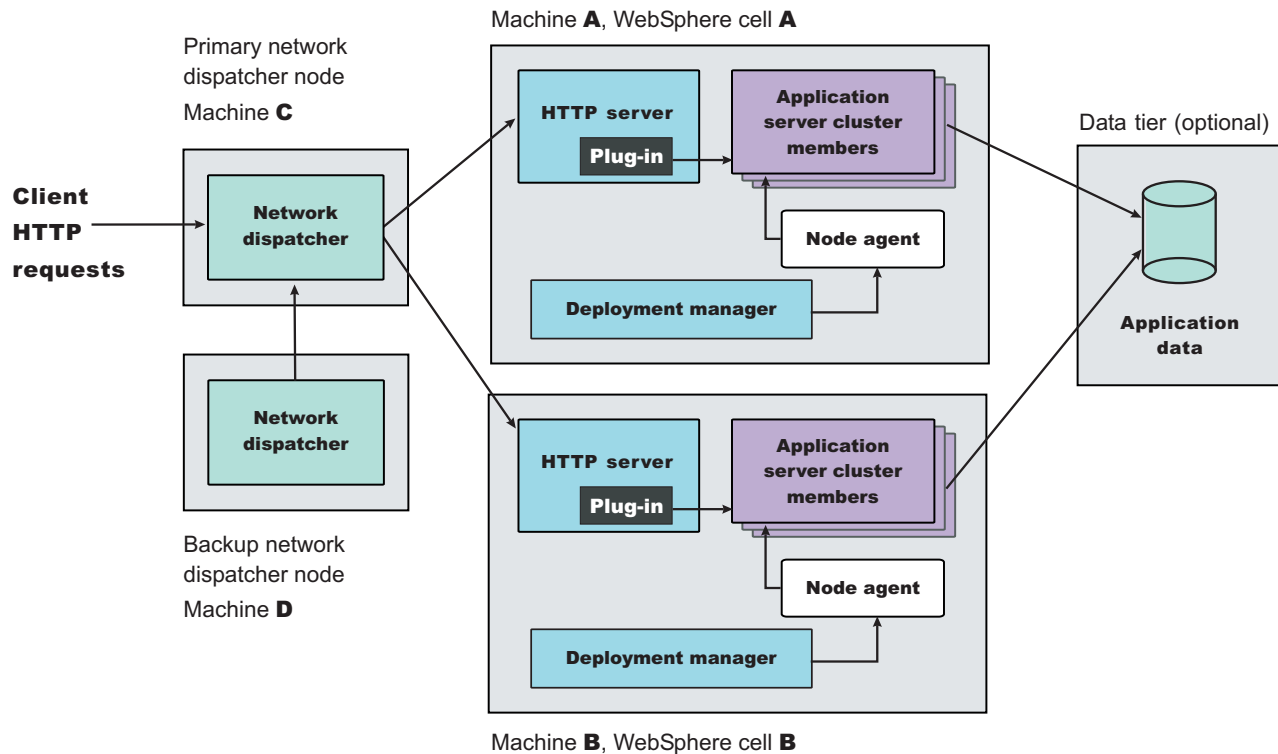
- The presence of a reverse proxy server in a DMZ is not suitable for some environments.
- The reverse proxy configuration requires more hardware and software than similar topologies that do not include a reverse proxy server, which makes it more complicated to configure and maintain.
- The reverse proxy server does not participate in WebSphere Application Server workload management.

Instructions

Implementation specifics are determined by the reverse proxy server. Refer to the documentation for the product you are using. No additional WebSphere Application Server administration is required for the reverse proxy server, although you might need it for other elements of the reverse proxy topology.

Multiple deployment manager cells

The following figure shows an example of how you can implement an application over multiple WebSphere Application Server deployment manager cells.



The example application runs simultaneously in two cells, each hosted on a different physical machine (Machines A and B). Network Dispatcher is used to distribute incoming HTTP requests between the two cells, presenting a single image of the application to clients. A backup Network Dispatcher node provides failover support.

Installing enterprise applications on the cluster in each deployment manager node ensures that identical versions of the application run in each cluster member. However, you administer each cell independently. Each cell has its own set of XML configuration files.

You can also run a different version of the application in each cell cluster. Because the cells are isolated from one another, you can also run different versions of the WebSphere Application Server software in each cell. For example, you can have a Version 5 cell and a Version 4 domain interoperating in your network.

Typical use

Topologies that incorporate more than one cell have the following advantages:

- Isolation of hardware failure. If one cell goes offline due to hardware problems, the others can still process client requests.
- Isolation of software failure. Running an application in two or more cells isolates any problems that occur within a cell, while the other cells continue to handle client requests. This isolation is helpful in a variety of situations:
 - When rolling out a new application or a revision of an existing application.

You can bring the new application or revision online in one cell, and test it in a live situation while other cells continue to handle client requests with the production version of the application.

- When deploying a new version of the WebSphere Application Server software.

You can bring the new version into production, and test it in a live situation without interrupting service.

- When applying fixes or patches to the WebSphere Application Server software.

You can take each cell offline to upgrade it, without interrupting the application.

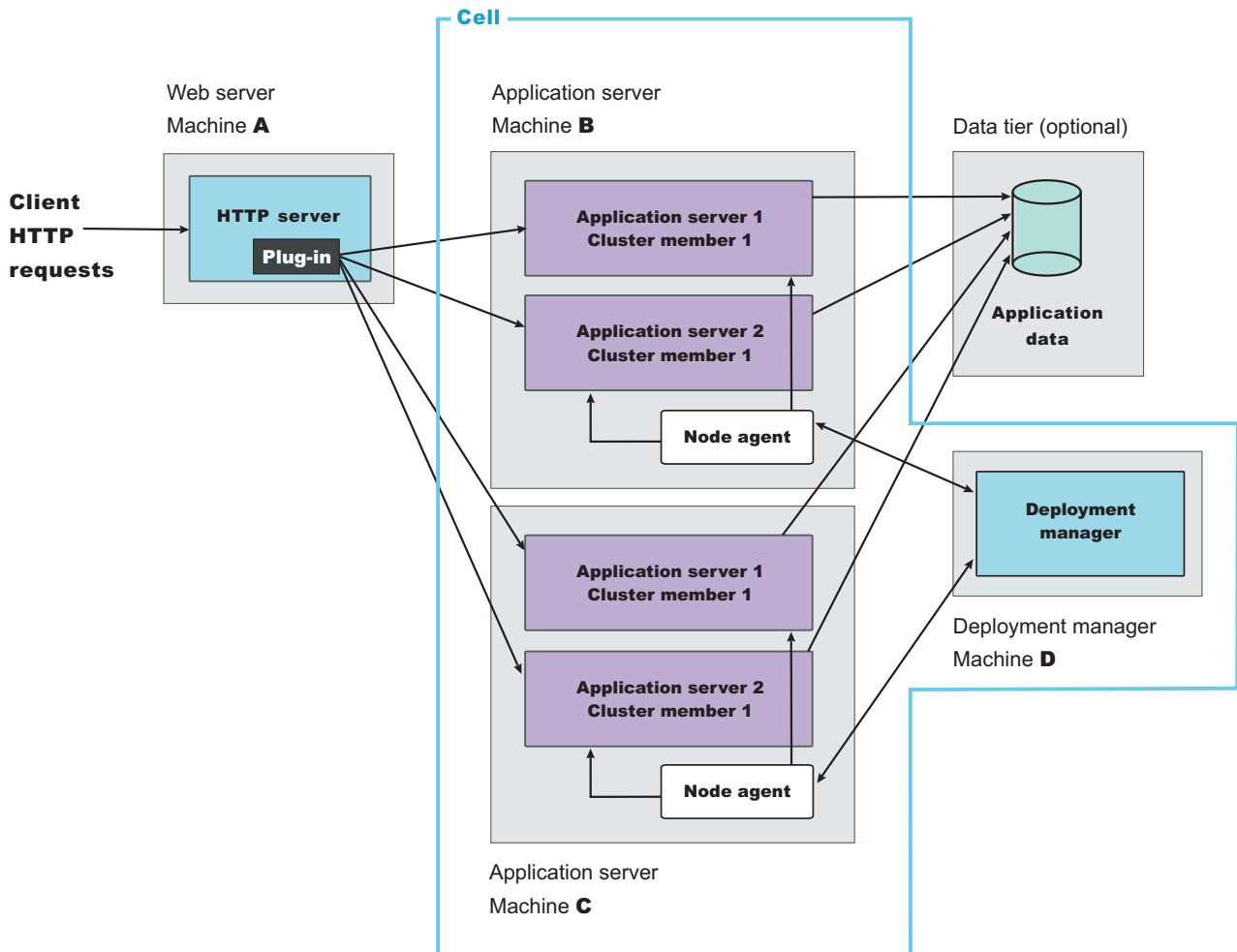
If an unforeseen problem occurs with the new software, using multiple cells can prevent an outage to an entire site. You can also rollback to a previous software version more quickly. You can handle hardware and software upgrades on a cell-by-cell basis during off-peak hours.

Using multiple cells has several drawbacks:

- Deployment is more complicated than for a single administrative cell.
- Multiple cells require more administration effort because each cell is administered independently. Use scripts to standardize and automate common administrative tasks to reduce this problem.

Multiple Application Servers within a node

The following figure shows a topology in which cluster members of more than one Application Server are hosted on a physical node.



The example topology is a variation of the basic horizontal scaling topology. Cluster members are not hosted on just a single machine, but are distributed throughout all of the machines in the system. In this

example, a cluster member is hosted on Machines B and C. Machine A serves as the Web server for the application and distributes client requests to the Application Server cluster members on each node.

Typical use

Hosting cluster members of multiple Application Servers within a node provides the following benefits:

- Improved throughput. Clustering an Application Server enables it to handle more client requests simultaneously.
- Improved performance. Hosting cluster members on multiple machines lets cluster members have full use of the processing resources on their machines.
- Hardware failover. Hosting cluster members on multiple nodes isolates hardware failures and provides failover support. Redirect client requests to the Application Server cluster members on other nodes if one node goes offline.
- Application server failover. Hosting cluster members on multiple nodes also isolates application software failures and provides failover support if a cluster member stops running. Redirect client requests to cluster members on other nodes.
- Process isolation. If one Application Server process fails, its cluster members on the other nodes are unaffected.

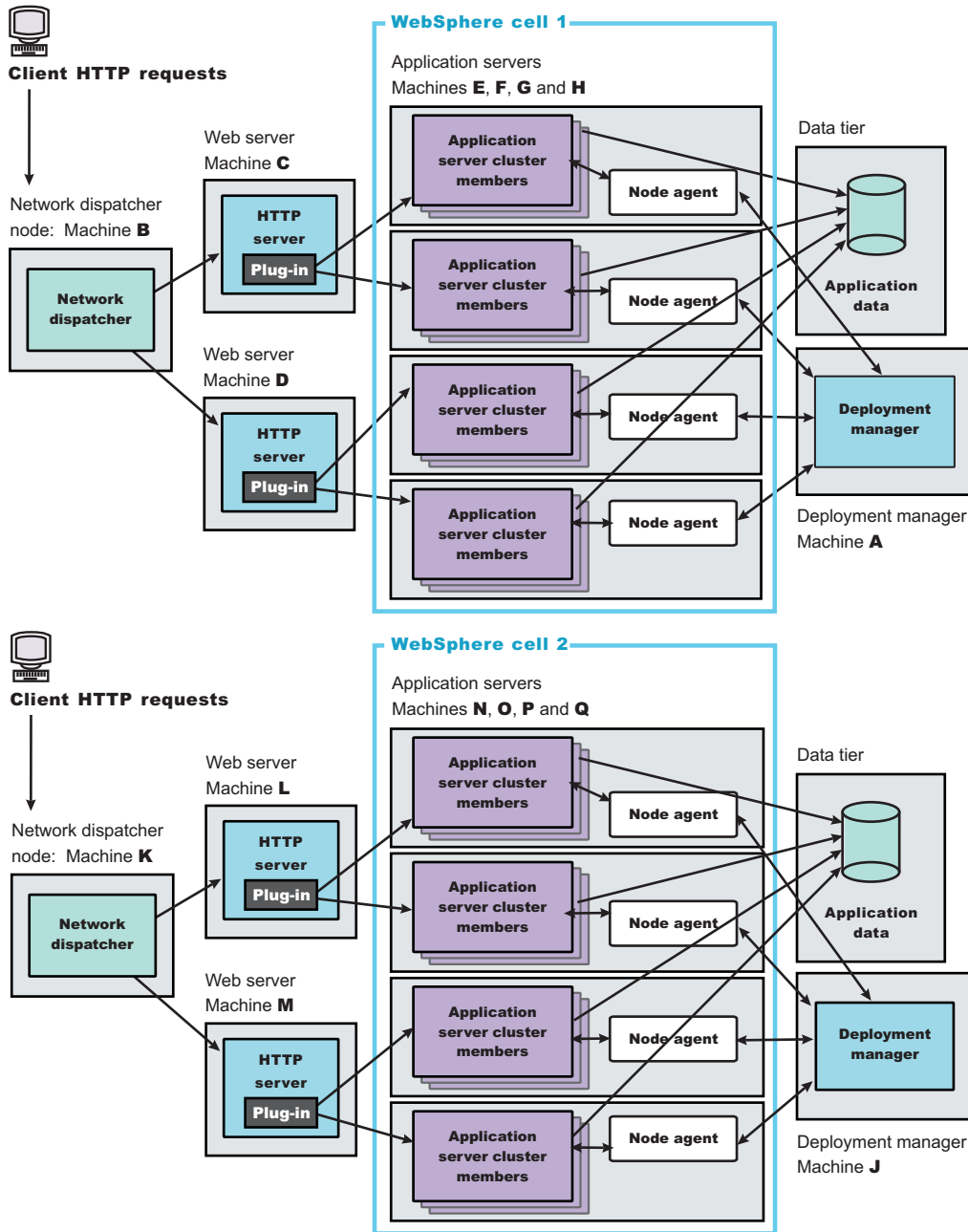
A drawback of this topology is more complex maintenance. You must maintain cluster members of each Application Server on multiple machines.

Having more than one Application Server on a stand-alone base node

It is possible to use the administrative console of a stand-alone WebSphere Application Server node to create new servers on the node. You must install the administrative console application on each new server to administer the new server. Each Application Server on a stand-alone base node must have its own administrative console.

Putting it all together - a combined topology

An example of a topology that combines the best elements of the other topologies discussed in this section is shown in the following figure.



This topology combines elements of several different basic topologies:

- Two deployment manager cells
- A deployment manager node to manage each cell (Machine A in cell 1; Machine J in cell 2)
- Two Network Dispatcher nodes (Machine B in cell 1; Machine K in cell 2)
- Two HTTP servers for each cell (Machines C and D in cell 1; Machines L and M in cell 2)
- Four Application Server nodes for each cell (Machines E, F, G, and H in cell 1; Machines N, O, P, and Q in cell 2)
- The use of cluster members for both vertical and horizontal scaling. In the example topology, each node hosts three cluster members; in practice, the number of cluster members is limited by the computing resources of each node.
- A data tier for each cell.

Typical use

This topology is designed to maximize throughput, availability, and performance. It incorporates the best practices of the other topologies discussed in this section:

- Having more than one Network Dispatcher node, HTTP server, and application server in each cell eliminates single points of failure.
- Multiple cells provide both hardware and software failure isolation, especially when upgrades of the application or the Application Server software are rolled out. You can handle hardware and software upgrades on a cell-by-cell basis during off-peak hours.
- Horizontal scaling uses both clustering and a Network Dispatcher to maximize availability and eliminate single points of process and hardware failure.
- Application performance is improved by using several techniques:
 - Hosting Application Servers on multiple physical Machines to boost the available processing power.
 - Creating multiple smaller cells instead of one large cell. There is less interprocess communication in a smaller cell, which lets you devote more resources to processing client requests.
 - Using cluster members to vertically scale Application Servers on each node, which makes more efficient use of machine resources.
- Applications with this topology can use several workload management techniques. In this example, you can perform workload management through one or both of the following:
 - Using the WebSphere Application Server Network Deployment workload management (WLM) facility to distribute work among Application Server cluster members.
 - Using Network Dispatcher to distribute client HTTP requests to each Web server.

For example, an application can manage workloads at the Web server level with Network Dispatcher and at the Application Server level with WebSphere Application Server workload management. Using multiple workload management techniques in an application provides finer control of load balancing.

Regardless of which workload management techniques are used in the application, Network Deployment participates in workload management to provide failover support.

In this topology, users notice an interruption only when an entire cell is lost. If this situation occurs, the active HTTP sessions are lost for half of the clients. The system can still process HTTP requests, although its performance is degraded.

The combined topology has several drawbacks:

- Deployment is more complicated. WebSphere Application Server software and application files must deploy in each cell, which is not the case for applications that run only in a single cell.
- Multiple cells require more administration effort because each cell is administered independently. Reduce this problem by using scripting to standardize and automate common administrative tasks.

Running WebSphere Application Server across versions

WebSphere Application Server Version 5.1 is generally interoperable with WebSphere Application Server Versions 3.5.x, 4.0.x, and 5.0.x. However, there are specific requirements to address for each version. Make the following changes to support interoperability between versions.

1. Apply required interim fixes.

Table 3. Interim fixes to apply to Version 3.5.x

Interim fix	Version 3.5.3	Version 3.5.4	Version 3.5.5	Version 3.5.6
PQ51387	Apply	Apply		
PQ60074	Apply	Apply	Apply	Apply
PQ60335			Apply	
PQ63548	Apply	Apply	Apply	Apply

Table 4. Interim fixes to apply to Version 4.0.x

Interim fix	Version 4.0.1	Version 4.0.2	Version 4.0.3
PQ60074	Apply	Apply	
PQ60336	Apply	Apply	
PQ63548	Apply	Apply	Apply

Table 5. Interim fixes to apply to Version 5.0.x client

Interim fix	Version 5.0.0	Version 5.0.1	Version 5.0.2
PQ81989	Apply	Apply	Apply

All fixes are available on the Support site for WebSphere Application Server products. There is a link to the Support site for WebSphere Application Server products at the bottom of each information center topic. Scroll all the way to the bottom of each page to see the link.

Interim fix PQ51387

A naming client fix that supports Version 3.5.x naming client access to the V5.0.x or V5.1.x name server.

Interim fix PQ60074

An Object Request Broker (ORB) fix that supports V5.0.x or V5.1.x naming client access to the Version 3.5.x or 4.0.x name server. A down-level client has no problem accessing a V5.0.x or V5.1.x name server, even when using corbaloc.

Interim fix PQ60335

An ORB fix to reconcile `java.math.BigDecimal` and other class differences in IBM Software Development Kits 122 and 131.

Note: This fix does not apply to IBM Software Development Kits on the Solaris Operating Environment.

Interim fix PQ60336

An ORB fix to reconcile `java.math.BigDecimal` and other class differences in IBM Software Development Kits 130 and 131.

Note: This fix does not apply to IBM Software Development Kits on Solaris platforms.

Interim fix PQ63548

A fix to correct problems that might occur when passing embedded valueTypes between WebSphere Application Server releases.

The best solution is to upgrade all your installations to the latest release and PTF levels, such as Versions 3.5.7 or 4.0.4, which do not require this fix. If this solution is not possible, apply the fix to your version.

Symptoms include `org.omg.CORBA.MARSHAL` exceptions when passing embedded valueTypes across the versions. Sometimes, other symptoms might mask `org.omg.CORBA.MARSHAL` exceptions, which makes them difficult to identify.

If symptoms reoccur in spite of the fix, re-export existing IORs.

Interim fix PQ81989

An interim fix to upgrade the Software Development Kit (SDK) used by the Version 5.0.x client. The evolution of a number of core classes causes interoperability errors between a WebSphere Application Server, Version 5.0.x client and a Version 5.1 server.

You might see the following message when running an interoperability scenario between a WebSphere Application Server, Version 5.0.x client and a WebSphere Application Server, Version 5.1 server:

```
java.rmi.MarshalException:
CORBA MARSHAL 0x4942f89a No;
nested exception is: org.omg.CORBA.MARSHAL:
```



```

Unable to read value from underlying bridge :
Invalid start_value valuetag: c
minor code: 4942F89A
completed: No

```

A number of core classes evolved between Software Development Kit (SDK) 1.3.x and SDK 1.4.x. You can experience problems interoperating with WebSphere Application Server, Version 5.1, which is the first WebSphere Application Server release to use SDK 1.4.x.

The recommended response is to upgrade the 5.0.x SDK 1.3.1 to a newer Service Release (SR). The SDK Service Release update is available at: <http://www-1.ibm.com/support/docview.wss?uid=swg24006169>.

2. Follow the required guidelines.

Table 6. Guidelines to apply for Version 3.5.x and Version 4.0.x

Guideline	Version 3.5.x	Version 4.0.x
1	Apply	
2	Apply	Apply
3	Apply	
4	Apply	
5	Apply	Apply
6	Apply	Apply
7	Apply	Apply

Guideline 1

Use the context of the lowest common denominator when interoperating at the naming level. For example, always use the 3.5.x context `com.ibm.ejs.ns.jndi.CNInitialContextFactory` when a client or server is at 3.5.x. For later versions, use the current `com.ibm.websphere.naming.WsnInitialContextFactory` context.

Guideline 2

Make required naming changes to support V3.5.x or V4.0.x client access to V5.0.x or V5.1.x enterprise beans. This issue is new, introduced by V5. There are several ways to make it work, such as:

- Updating the `namebindings.xml` file if you do not use the V5.0.x or V5.1.x migration tools, but have installed V3.5.x or V4.0.x applications on V5.0.x or V5.1.x. To allow V3.5.x or V4.0.x client access to the applications, add additional information to the bind information in the V5.0.x or V5.1.x namespace to make the old JNDI names work. Add the information to the `namebindings.xml` configuration file at the cell level using the administrative console.

Note: Applications that you migrate to V5.0.x or V5.1.x during installation, or that you migrate using the WASPreUpgrade and WASPostUpgrade migration tools, already have this update.

After federating an application server node into a deployment manager cell, you cannot use the migration tools. To use these tools again, remove the node from the cell, use the tools, and add the node to the cell again.

- Calling V5.0.x or V5.1.x enterprise beans directly using the naming path that includes the server on which the enterprise beans are running, such as `cell/node/server1/some/ejb/name`, for example.
- Using the Version 4.0.x client `java:/comp` location to find V5.0.x or V5.1.x enterprise beans. (You cannot use the command from a V3.5.x client.)

Guideline 3

Ensure that programs performing a JNDI lookup of the `UserTransaction` interface, use an `InitialContext` that resolves to a local implementation of the interface. Also ensure that such programs use a JNDI location appropriate for the enterprise bean version.

Prior to the EJB 1.1 Specification, the JNDI location of the UserTransaction interface was not specified. Earlier versions up to and including Version 3.5.x do not use the EJB 1.1 Specification. They bind the UserTransaction interface to a JNDI location of `jta/usertransaction`.

Version 4, and later releases, bind the UserTransaction interface at the location defined by the EJB 1.1 Specification, which is `java:comp/UserTransaction`.

Version 5.0.x and Version 5.1.x no longer provide the earlier `jta/usertransaction` binding within Web and EJB containers to applications at a J2EE specification level of 1.3 or later, to enforce use of the newer UserTransaction interface. For example, EJB 2.0 applications can use only the `java:comp/UserTransaction` location.

Guideline 4

Be aware of limitations when calling WorkLoad Management (WLM)-enabled enterprise beans.

Some clients cannot call WLM-enabled enterprise beans on remote clusters when there is a local WLM-enabled enterprise bean of the same name. If there is a cluster local to the client with the same enterprise bean as the remote cluster that the client is trying to call, the client ends up talking to the local cluster. The following table lists supported combinations of clients calling WLM-enabled enterprise beans on remote application servers.

All clients at Version:	Server at Version:	Supported interoperability
3.5.6	5.0.x, 5.1.x	Yes
4.02, 4.03	5.0.x, 5.1.x	Yes
5.0.x, 5.1.x	5.0.x, 5.1.x	Yes
5.0.x, 5.1.x	3.5.x	No
5.0.x, 5.1.x	4.02, 4.03	Yes

Guideline 5

Be aware of administrative console limitations.

You cannot use the administrative interfaces for V5.0.x or V5.1.x to administer a V3.5.x or V4.0.x administrative server. Likewise, you cannot use a Version 3.5.x or Version 4.0.x administrative console to administer a V5.0.x or V5.1.x environment. If you use the administrative console on a remote machine to administer V3.5.x or V4.0.x WebSphere Application Server domains, migrating any of the nodes or domains to V5.0.x or V5.1.x renders the remote administration console ineffective for administering any V5.0.x or V5.1.x environment.

Guideline 6

Use Secure Sockets Layer Version 3 (SSL v3) when interoperating with Version 3.5.x for secure connections. You cannot use Common Secure Interoperability Version 2 (CSlv2) for interoperability, because Versions 3.5.x and 4.0.x do not support CSlv2.

Guideline 7

The evolution of a number of core classes causes interoperability errors between a V3.5.x or V4.0.x WebSphere Application Server client and a Version 5.1 Application Server.

You might see an error message similar to the following example when running an interoperability scenario between a WebSphere Application Server V3.5.x or V4.0.x client and a Version 5.1 Application Server:

```
java.rmi.MarshalException: CORBA MARSHAL 0 No; nested exception is:
org.omg.CORBA.MARSHAL: Unable to read value from underlying bridge :
Custom marshaling of RMI:java.lang.Throwable:
F8678B4F4D2EB705:D5C635273977B8CB not compatible with local class
(local class not custom marshal capable) minor code : 0
completed: No
```

The exception is provoked by the evolution of the class you are trying to marshal. For example, the `readObject()` and `writeObject()` methods are inconsistent at the sender and the receiver.

The problem is provoked by an enterprise bean `CreateException` exception, as a result of its inheritance from the `Throwable` object, which evolves between Software Development Kit (SDK) 1.3.1 and SDK 1.4.0.

A number of core classes evolve between SDK 1.3.x and SDK 1.4.x. Therefore, you see the problems when you interoperate with WebSphere Application Server, Version 5.1, which is the first release to use SDK 1.4.x.

Migrate WebSphere Application Server, Version 3.5.x to Version 4.0.0 and upgrade to WebSphere Application Server, Version 4.0.7 to solve the problem.

Or upgrade WebSphere Application Server, Version 4.0.x to Version 4.0.7 or later to solve the problem.

See “Federating multiple Version 5 installation instances” on page 362 for information about mixed version cells, where a Version 5.1 deployment manager cell can include Version 5.0.x base nodes.

This information is dynamic and might be augmented by information in technical articles that are available on the IBM DeveloperWorks WebSphere site. Check the site for the latest information.

Installing the product

This topic describes how to install WebSphere Application Server products as the root user on a Linux platform or on a UNIX-based platform, or from a user ID that belongs to the administrator group on a Windows platform.

You cannot install the product correctly as a non-root user on a Linux or UNIX-based operating system platform, or from a user ID on a Windows platform that is not part of the administrators group. Some steps of the installation procedure on a Windows platform require the user to belong to the administrator group and have the following advanced user rights, *Act as part of the operating system* and *Log on as a service*. If you back up the product CD-ROM on a Linux or UNIX platform, do so as root. Backup copies made from non-root users do not preserve the correct file attributes and do not work.

IBM Support offers tutorials on various WebSphere Application Server installation topics. See WebSphere education on demand: Installation best practices tutorials for more information. One topic describes installing new or coexisting versions.

5.1+ WebSphere Business Integration Server Foundation is the V5.1 level of the Enterprise product. WebSphere Business Integration Server Foundation, Version 5.1 extends Network Deployment V5.1. After migrating the underlying product to V5.1, do not reinstall the V5.0.x Enterprise product. Version 5.0.x of the Enterprise product does not extend Version 5.1 of the WebSphere Application Server Network Deployment product.

Read through this topic and its related topics to prepare for installation and to make yourself familiar with installation options, before you start to use the installation tools. Review the prerequisite requirements on the IBM WebSphere Application Server supported hardware, software, and APIs Web site to get started.

After verifying prerequisites, read these topics before installing the product:

- Platform-specific tips for installing and migrating
- Tips for installing the embedded messaging feature
- Migrating and coexisting

- Using the Launchpad to start the installation
- Installing silently

If you encounter a problem such as not having enough disk space or temporary space or not having prerequisite packages on your Linux or UNIX-based system, cancel the installation, make the required changes, and restart the installation to pick up the changes that you made.

The installation uses InstallShield for Multiplatforms (ISMP) to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

The Installation wizard does not support hot keys, such as Alt-Y. You can tab to the Yes key and press **Enter** to select it, for example.

Although you can create multiple servers on a base WebSphere Application Server node, the servers all share one set of configuration files. Changes that you make to one server affect the others. There is a possibility that configuration documents might become corrupted. Use configuration instances to create multiple servers, each with its own set of configuration files. Or install the Network Deployment product to create and manage multiple base Application Servers.

Install the base product before installing the Network Deployment product when installing both products on the same machine. The embedded messaging feature that is included in the default installation requires this specific order of installation.

Table 7. Installation tip

Operating platform	Tip
All platforms	Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component

5.1 + Open this topic in the information center for the WebSphere Business Integration Server Foundation product to learn how to install the Integration Server installation image and extend the multimachine environment. The CD-ROM for the Integration Server installation image is labeled, **Integration Server**.

This topic describes installing the Network Deployment product, using the installation image on the product CD-ROM labeled, **Deployment Manager**. Use a CD-ROM labeled, **Application Server, IBM HTTP Server**, in the Network Deployment product package to install the base WebSphere Application Server product. Open this topic in the information center for the base WebSphere Application Server product to learn how to install the base WebSphere Application Server product.

5.1 + The **Deployment Manager** CD-ROM is available in both the Network Deployment product package and the WebSphere Business Integration Server Foundation product package. Use this information center topic to install the Network Deployment product, regardless of which package the CD-ROM comes from. Open this topic in the information center for the Integration Server product to learn how to install the Integration Server product and extend the multimachine environment. The CD-ROM for the Integration Server product is labeled, **Integration Server**.

This topic is available in Adobe PDF format, on the product CD-ROMs, and online in an information center that is available from the IBM WebSphere Application Server Web site at <http://publib.boulder.ibm.com/infocenter/ws51help/index.jsp>. When possible, access the most current version of this information by selecting the information center. The information center displays in the language of your machine locale.

The Launchpad tool lets you access the product overview, the `readme.html` file, and installation guides. You also use the Launchpad during the installation procedure to install the product. See “Using the Launchpad to start the installation” on page 173 for more information.

The Installation wizard performs the following actions:

- Automatically checks prerequisites
- Looks for a previous WebSphere Application Server installation, to determine whether to display the Migration panel or the Coexistence panel during the installation

Install WebSphere Application Server Network Deployment on your platform. Use one of the following procedures:

- Install Network Deployment on AIX platforms, as described in “Installing Network Deployment on AIX platforms.”
- Install Network Deployment on HP-UX platforms, as described in “Installing Network Deployment on HP-UX platforms” on page 61.
- Install Network Deployment on Linux platforms, as described in “Installing Network Deployment on Linux platforms” on page 79.
- Install Network Deployment on Solaris platforms, as described in “Installing Network Deployment on Solaris platforms” on page 94.
- Install Network Deployment on Windows platforms, as described in “Installing Network Deployment on Windows platforms” on page 109.

The Installation wizard configures the product. It is not necessary to perform further configuration at this time.

You have now successfully installed WebSphere Application Server Network Deployment and the features that you selected.

Installing Network Deployment on AIX platforms

This topic describes how to install the WebSphere Application Server Network Deployment product as the root user on an AIX operating system platform.

You cannot install the product correctly from a user ID that does not belong to the administrator group. Some steps of the installation procedure require the administrator group ID to have the following advanced user rights, *Act as part of the operating system* and *Log on as a service*.

If you encounter a problem such as not having enough space or not having the right prerequisite updates on your system, cancel the installation, make the required changes, and restart the installation.

The installation uses InstallShield for Multiplatforms (ISMP) to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

Installation programs in WebSphere Application Server products

5.1 +

Product	Link to installation procedure	Label on product CD
(Base) WebSphere Application Server	Base product installation	Application Server, IBM HTTP Server
Network Deployment	Network Deployment installation (which links to this procedure!)	Deployment Manager
WebSphere Business Integration Server Foundation	Integration Server installation	Integration Server Disk 1 and Integration Server Disk 2

Use the Network Deployment installation image to manage a multimachine environment, where you have installed the base product on different machines and want to manage the Application Servers in a group, or *cell*. If you buy the Network Deployment product, you also get the base product in the package.

WebSphere Business Integration Server Foundation is the V5.1 level of the Enterprise product. WebSphere Business Integration Server Foundation, Version 5.1 extends the base WebSphere Application Server V5.1 product. After migrating the underlying product to V5.1, do not reinstall the V5.0.x Enterprise product. Version 5.0.x of the Enterprise product does not extend Version 5.1 of the base WebSphere Application Server product nor does it extend the V5.1 Network Deployment product.

5.1 + If you buy the WebSphere Business Integration Server Foundation product, you also get the Network Deployment product and the base product in the package.

Creating multiple Application Servers on a single machine

Although you can create multiple servers on a base WebSphere Application Server node, the servers all share one set of configuration files. Changes that you make to one server affect the others. Configuration documents might become corrupted. The **wsinstance** command can create multiple configuration instances. Each instance is a stand-alone Application Server with its own set of configuration files. Or install the Network Deployment product to create and manage multiple base Application Servers.

Order of installation

5.1 + Install the base product before installing the Network Deployment product when installing both products on the same machine. Install the Network Deployment product before the Integration Server product that extends the Network Deployment product. You can install the Integration Server product before the base product. The Integration Server product can install the base product in what is known as an *umbrella* installation but Integration Server cannot install the Network Deployment product. Some features of the base product cannot be installed by Integration Server.

The embedded messaging feature that is included in the default installation requires that you install base before Network Deployment when installing both on the same machine. Otherwise, the order does not matter. See the following installation tip for more information:

Table 8. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component

This topic is available in Adobe PDF format on the product CD-ROM and online in an information center. The information center always has the most current information. The information center displays in the language of your machine locale if possible.

The installation guide PDF is updated occasionally. Download a new installation guide from:

5.1 + ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf

The Launchpad tool lets you access the product overview, the `readme.html` file, and the installation guide. See “Using the Launchpad to start the installation” on page 173.

You also use the Launchpad during the installation procedure to install the product. The installation program performs the following actions:

- Checks prerequisites automatically
- Looks for a previous WebSphere Application Server installation, to determine whether to display the Migration panel or the Coexistence panel during the installation

After using the Launchpad to view product documentation, perform the following procedure. The procedure includes several preliminary steps before actually launching the installer program.

1. Stop all WebSphere Application Server-related Java processes on the machine where you are installing the product.
2. Stop any Web server process such as the IBM HTTP Server, if you are extending the base product.
3. **5.1+** Install the prerequisite Java130.rte.lib version 1.3.0 on AIX Version 5.2 maintenance level 1. You must install Java130.rte.lib Version 1.3.0 to verify that the embedded messaging feature installs correctly. To download a copy of Java 1.3.0:
 - a. Go to the <http://www-106.ibm.com/developerworks/java/jdk/aix/index.html> Web page.
If the link is unavailable, search for IBM Developer Kit for AIX.
 - b. Click **Download and service information** in the Contents box.
 - c. Click **Java 1.3.0** from the **Java Version** column in the table.
 - d. (Optional) Register for a user ID and password.

To correct an existing problem:

- a. Uninstall the following components, if present:
 - WebSphere Application Server
 - WebSphere MQ (MQSeries)
 - WebSphere MQ (MQSeries) classes for Java and WebSphere MQ (MQSeries) for Java Message Service
 - IBM HTTP Server
 - WebSphere Embedded Messaging Publishing and Subscribe Edition
 - b. Clean up the system registry and edit the `vpd.properties` file on AIX to remove any entry related to WebSphere Application Server products. Entries related to WebSphere Application Server products begin with:
 - **WSB** for the base WebSphere Application Server, V5.0.x product
 - **WSB..51** for the base WebSphere Application Server, V5.1 product
 - **WSC** for the WebSphere Application Server Application Clients
 - **WSE** for the Enterprise product, V5.0.x
 - **WSE..51** for WebSphere Business Integration Server Foundation, V5.1
 - **WSN** for the Network Deployment product, V5.0.x
 - **WSN..51** for the Network Deployment product, V5.1
 - **WSM** for the WebSphere MQ product
 - c. Reinstall the WebSphere Application Server product in an umbrella installation of the WebSphere Business Integration Server Foundation product.
4. **5.1+** Install the prerequisite `x1C.rte` 6.0 run-time code on AIX 5.2 maintenance level 1. You must install the `x1C.rte` 6.0 run-time code, which is a prerequisite of GSKit7. Download the run-time code as a fix from the AIX Support site.
 5. Provide adequate disk space.

5.1+ The Network Deployment product requires the following disk space:

370 MB for the `/usr/WebSphere/DeploymentManager` directory

The installation root directory contains the Network Deployment product.

150 MB for the `/tmp` directory

The temporary directory is the working directory for the installation program.

A message about free space occurs when less than 4 MB of free space remains after starting the installation. The InstallShield for MultiPlatforms (ISMP) program displays a message about using the `-is:tempdir` parameter to identify an alternate temporary space directory.

Ignore any `-is:tempdir` message. The `-is:tempdir` parameter is not supported. Cancel the installation, allocate at least 150 MB of free space in the `/tmp` directory, and start the installation again.

520 MB

A significant amount of the required space is temporarily required during installation and is released after installation.

5.1+ The Installation wizard displays required space for individual features on the Feature selection panel. The Installation wizard also warns you if you do not have enough space to install the product. If you plan to migrate applications and the configuration from a previous version, verify that application objects have available disk space. As a rough guideline, plan for space equal to 110 percent of the size of the application objects:

- **For Version 3.5.x:** The size of application Java archive (JAR) files, Web archive (WAR) files, and servlet files
- **For Version 4.0.x:** The size of enterprise archive (EAR) files
- **5.1+ For Version 5.0.x:** The size of EAR files

6. **5.1+** Define the user groups and the user needed for the embedded messaging feature.

A user ID of the name **mqm**, with a primary group of **mqm**, owns the directories and files that contain the resources associated with the embedded messaging feature:

- a. Use the System Management Interface Tool (smit) to create the **mqm** and **mqbrkrs** user groups:
 - 1) Click **Security & Users > Groups > Add a Group**.
 - 2) Set the name field to **mqm** and create the user group.
 - 3) Create the **mqbrkrs** user group in the same way.
- b. Use smit to create the **mqm** user:
 - 1) Click **Security & Users > Users > Add a User**.
 - 2) Set the name field to **mqm** and create the user.
 - 3) Click **Security & Users > Change a Users Password > Passwords** to add a password to the new user ID.
- c. Use smit to add the **mqm** and **root** users to the **mqm** group:
 - 1) Click **Security & Users > Users > Change / Show Characteristics of a User**.
 - 2) Enter **mqm** in the User Name field.
 - 3) Add **mqm** to the Group SET field, which is a comma-separated list of the groups to which the user belongs.
 - 4) Add **root** to the **mqm** group and the **mqbrkrs** group.

The recommended user ID for running the **jmsserver** process is **root**. If you do run the **jmsserver** process under another user ID, add that user ID to the **mqm** and **mqbrkrs** groups. User IDs longer than 12 characters cannot be used for authentication with the embedded WebSphere JMS provider.

The **mqm** user starts the **jmsserver** for general JMS support and the WebSphere embedded broker for WebSphere Application Server topic connections.

7. Log out and back in to pick up the secondary user groups, **mqm** and **mqbrkrs**, for **root**. Use **ssh** instead of **telnet** to log in or set up the display environment for a real **root** login on AIX systems by running the following command after logging on:

```
su -
```

In a normal **root** login, issue the command **su**. For a real **root** login, issue the command **su -**.

Display settings for a normal **root** login are automatic. For a real **root** login, you must set your display environment properly to successfully view the installation wizard. Otherwise, you see a message about Preparing Java(tm) Virtual Machine..., and seven rows of dots, but no installation GUI and no further messages. Refer to the documentation for your AIX platform to determine proper display settings.

Use the **id -a** command or the **groups** command to see defined groups for **root**. If **mqm** and **mqbrkrs** are not in the list that is returned, you cannot install the embedded messaging feature:

```
[root@wasdoc2 root]# id -a
uid=0(root) gid=0(root)
groups=0(root),1(bin),2(daemon),
```


3(sys),4(adm),6(disk),10(wheel),500(mqm),501(mqbrkrs)

```
[root@wasdoc2 root]# groups
root bin daemon sys adm disk wheel mqm mqbrkrs
```

- Allocate adequate disk space for the embedded messaging feature if you are planning to install the feature. The installation locations for the embedded messaging feature are fixed as shown in the following table, which lists the locations for the base messaging functions and the messaging broker functions for publish/subscribe messaging.

Table 9. Space requirements for the embedded messaging feature on AIX

Component	Base code	Broker code	Base data	Broker data
Path	/usr/ mqm	/usr/ opt/ wemps	/var/ mqm	/var/ wemps
Client subfeature	15 MB	15 MB	5 MB	N/A

- Prepare to install the embedded messaging feature with WebSphere MQ. The embedded messaging feature is based on the IBM WebSphere MQ product. The feature and the product each provide a Java message service (JMS) function that supports queues for point-to-point messaging and topics for publish and subscribe messaging. You can install the deployment manager with the embedded messaging client feature with or without the WebSphere MQ product on the same machine. To support both the embedded messaging feature and the WebSphere MQ product on the same machine, the WebSphere MQ product must be at a certain fix level and must have several of its features installed.

If you already have WebSphere MQ installed, you can configure it as the JMS provider. Otherwise, you can install the embedded messaging feature during the installation of the base Application Server. You can also install the WebSphere MQ product or another JMS provider after you install the WebSphere Application Server products.

On a machine where you want application servers to run messaging applications that use a JMS provider on another host, install IBM WebSphere Application Server or the deployment manager with the embedded messaging client feature.

Even though you might decide now to install only the embedded messaging feature, you can install the WebSphere MQ product later and use the IBM WebSphere MQ product as the JMS provider instead.

- 5.1+** Verify that you have upgraded to WebSphere MQ 5.3 with the CSD04 update to install embedded messaging on a machine where you already have WebSphere MQ installed. Determine if your WebSphere MQ 5.3 installation is at the required level by running the **mqver** utility provided by WebSphere MQ.

The required level as indicated by mqver is shown below:

```
Name:      WebSphere MQ
Version:   530.4 CSD04
...
```

- 5.1+** Verify that you have installed the required WebSphere MQ 5.3 features to install embedded messaging on a machine where you already have WebSphere MQ installed. When installing the embedded messaging client feature, the required MQ feature is *Java messaging*.

If you attempt to install the embedded messaging feature when WebSphere MQ is already installed, the level of WebSphere MQ must be Version 5.3 with the required MQ features. Otherwise, the installation of the embedded messaging feature fails with prerequisite check errors.

- 5.1+** Create and mount a journalized file system called /var/mqm for your messaging working data. Use a partitioning strategy with a separate volume for embedded messaging or WebSphere MQ data to isolate system activity from the potentially high volume of messaging work that can build up in the /var/mqm directory.

13. **5.1+** Create separate file systems for log data in the `var/mqm/log` directory and error files in the `var/mqm/errors` directory. Store log files on a different physical volume from the embedded messaging queues, which are in the `var/mqm` directory. This ensures data integrity in the case of a hardware failure. If you are creating separate file systems, allow the following minimum free space:

30 MB `/var/mqm`

20 MB `/var/mqm/log`

4 MB `/var/mqm/errors`

The `/var` file system stores all the security logging information for the system and stores the temporary files for e-mail and printing. Therefore, it is critical that you maintain free space in `/var` for these operations. If you do not create a separate file system for messaging data, and `/var` fills up, all security logging stops on the system until free space is available in `/var`. Also, e-mail and printing do not work without some available free space in `/var`.

You have the same options for creating file systems for the embedded messaging feature as you do for WebSphere MQ. For example, if you cannot install the embedded messaging options in the required file system (for example, if it is too small), you can do one of the following *before* installing embedded messaging:

- Create and mount a new file system for the installation directory.
- Create a new directory anywhere on your machine, and create a symbolic link from the required installation directory to the new directory. For example, on AIX:

```
mkdir /bigdisk/mqm
ln -s /bigdisk/mqm /usr/mqm
```

14. **5.1+** Unmount file systems with broken links to avoid `java.lang.NullPointerException` errors. Unmount file systems with broken links before installing.

The installation of the cumulative fix can fail with the following error when broken links exist to file systems:

An error occurred during wizard bean change notification:

```
java.lang.NullPointerException
  at com.ibm.wizard.platform.aix.AixFileUtils.
    getFileSystemData(AixFileUtils.java:388)
  at com.ibm.wizard.platform.aix.AixFileUtils.
    getPartitionDataWithExecs(AixFileUtils.java:172)
  at com.ibm.wizard.platform.aix.AixFileUtils.
    getPartitionData(AixFileUtils.java:104)
  at com.ibm.wizard.platform.aix.AixFileServiceImpl.
    getPartitionNames(AixFileServiceImpl.java:397)
...

```

Use the `df -k` command to check for broken links to file systems. Look for file systems that list blank values in the *1024-blocks size* column. Columns with a value of "-" (dash) are not a problem. The following example shows a problem with the `/dev/lv00` file system:

```
> df -k
Filesystem      1024-blocks      Free %Used    Iused %Iused Mounted on
/dev/hd4         1048576      447924  58%      2497    1% /
/dev/hd3         4259840     2835816  34%        484    1% /tmp
/proc            -             -      -         -      - /proc
/dev/lv01        2097152     229276  90%       3982    1% /storage
/dev/lv00
/dev/hd2         2097152     458632  79%      42910    9% /usr
iw031864:/cdrom/db2_v72_eee_aix32_sbcs
```

The `/proc` file system is not a problem. The `iw031864:/cdrom/db2_v72_eee_aix32_sbcs` file system is a definite problem. The `/dev/lv00` file system is also a likely problem. Use one of the following commands to solve this problem:

```
> umount /cdrom/db2_v72_eee_aix32_sbcs
> umount /cdrom
```

If the problem continues, unmount any file systems that have blank values, such as the `/dev/1v00` file system in the example. If you cannot solve the problem by unmounting file systems with broken links, reboot the machine and begin this procedure again.

15. Verify that prerequisites and corequisites are at the required release levels. Although the Installation wizard checks for prerequisite operating system patches with the `prereqChecker` application, review the prerequisites on the IBM WebSphere Application Server supported hardware, software, and APIs Web site if you have not already done so. Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Some operating systems that were not supported at the time that this product was shipped on CD-ROM might now be supported. You might receive a message from the `prereqChecker` program that an operating system is not supported when, in fact, the operating system is supported.

Always consult the IBM WebSphere Application Server supported hardware, software, and APIs Web site to determine whether your operating system is supported when you receive a message from the `prereqChecker` program. The Web site lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. After confirming that your operating system is supported and that you have installed all necessary patches, you can click **Next** to continue an installation when you receive an error message from the `prereqChecker` program.

16. Remove any entries from the `/usr/bin/jitk.db` file if you have uninstalled WebSphere Application Server Enterprise Edition Version 4.1. Remove any remaining artifacts from an uninstalled Enterprise Edition V4.1 system to prevent the display of the Coexistence panel or the Migration panel during installation.

The Installation wizard might display the Migration panel or the Coexistence panel even though you have uninstalled WebSphere Application Server Version 4.1. You can prevent the Installation wizard from recognizing a previously deleted V4.1 Application Server by removing the following entry from the `/usr/bin/jitk.db` file:

```
WebSphere Application Server 4.1
```

Remove other Version 4.x entries for WebSphere Application Server products that are no longer on your system.

17. Verify the system **cp** command when using emacs or other freeware.

If you have emacs or other freeware installed on your AIX operating platform, verify that the system **cp** command is being used.

- a. Type `which cp` at the command prompt before running the installation program for the WebSphere Application Server product.
- b. Remove the freeware directory from your PATH if the resulting directory output includes freeware. For example, if the output is similar to this `/opt/freeware/bin/cp` message, remove the directory from the PATH.
- c. Install the WebSphere Application Server product.
- d. Add the freeware directory back to the PATH.

If you install with a **cp** command that is part of a freeware package, the installation might appear to complete successfully, but the Java 2 SDK that the product installs might have missing files in the `install_root/java` directory.

Missing files can destroy required symbolic links. If you remove the freeware **cp** command from the PATH, you can install the Application Server product successfully.

Perform the following step to verify that the Java 2 SDK is working correctly.

18. Verify the SDK on the WebSphere Application Server CD. Perform the following steps to verify that the SDK on the product CD-ROM is working correctly.
 - a. Change directories to the `/aix/WAS/jdk/java/bin` directory on the product CD-ROM. For example:

```
cd /mnt/aix/WAS/jdk/java/bin
```
 - b. Verify the SDK version. Type the following command:

```
./java -version
```

The command completes successfully with no errors when the SDK is intact.

19. Select the Installation wizard method or the silent installation method but do not start the installation yet. The installer program has two interfaces, the Installation wizard and a silent command-line installation.

Performing the installation with the wizard

You can start the Installation wizard using the Launchpad or directly using the **install** command.

The default installation method is to open a command window and issue the command to start the Launchpad tool. Click the **Install the product** option on the Launchpad. (See “Using the Launchpad to start the installation” on page 173.)

This option launches the Installation wizard, which is an InstallShield for MultiPlatforms (ISMP) application. This action launches the Installation wizard in the language of your machine locale unless no translation exists for your locale, in which case you receive the English version.

A short delay occurs before the ISMP wizard displays. You do not need to click the Install the product option more than once to cause the wizard to display. The delay is particularly noticeable on x-windows platforms.

You can also start the Installation wizard using the **/cdrom/aix/install** command, where **/cdrom** is the mount point for the product CD-ROM and **aix** is the platform directory.

Performing a silent installation

You can perform a silent installation using the **-options responsefile** parameter with the command method:

```
fully_qualified_CD_pathname/install  
-options fully_qualified_HDD_pathname/responsefile
```

```
# /cdrom/aix/install -options /tmp/my_responsefile
```

Start the silent installation with a fully qualified path to the options response file. Otherwise, the Installation wizard starts.

A silent installation causes the installation program to read your responses from the options response file, instead of from the wizard interface. You must customize the *responsefile* before installing silently. See Customizing the Network Deployment options response file.

After customizing the file, you can issue the command to silently install. See “Installing silently” on page 174.

After issuing the command, the following text displays:

```
# .....  
.InstallShield Wizard
```

```
Initializing InstallShield Wizard...
```

```
Searching for Java(tm) Virtual Machine...
```

The silent installation runs without displaying status to the window:

- Press **Enter** when the installation is complete to return to the command prompt.
- Press **Enter** at any time to return from a background job to the command prompt.

You can change the **-W launchPRTBean.active** option in the response file to display the Registration panel to indicate the completion of a silent installation on a local system with a graphical user interface.

To determine the status of the silent installation, review the installation logs in the *install_root/logs* directory or in the */tmp* directory. See “Troubleshooting the installation” on page 185 for more information about log files.

Silent installation is particularly useful if you install the product often.

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard.

Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize their options.

Asynchronous and synchronous command lines

After running the **install** command, the command line returns synchronously. A synchronous **install** command returns the command line after the installation is complete.

You can start the installation asynchronously with the installation process and its children processes all running as background processes. Consult your operating system documentation to learn how to issue asynchronous commands. After running the **install** command, the command line returns immediately.

Do not misinterpret an asynchronous command line to mean that the installation has finished when the command prompt returns. Although the command line returns, either the Installation wizard or a silent installation might still be in progress.

Installing with a network file system mount

If you must use an NFS mount, see the following installation tip:

Table 10. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Platform-specific issues when dealing with network file systems

20. Insert the product CD labeled, *Deployment Manager*.
21. Open a shell window to mount the product CD-ROM, as described in “Mounting a CD-ROM drive on AIX” on page 60. Use the same shell window throughout the installation procedure.
22. Start the installation with the **/cdrom/aix/launchpad.sh** command, where **/cdrom** is the mount point for the product CD-ROM and **aix** is the platform directory on the CD-ROM. You can also start the installation directly with the **/cdrom/aix/install** command. The following example shows a fully qualified call to the installer program:

```
fully_qualified_CD_pathname/install
```

```
# /cdrom/aix/install
```

The readme link in the Launchpad is to the `readme.html` file in the CD root directory. The *Getting Started* document that contains installation information is in the `docs` directory on the CD.

Download the current version of the *Getting Started* document from:

5.1 + ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize the options.

23. Click **Next** to continue. The license agreement displays.
The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.
24. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist.
The prerequisites checker program recognizes the official supported level for AIX 5.2 as ML1 + APAR IY44183. AIX 5.2 ML2 is also supported. If you receive a message from the prerequisites checker program that AIX 5.2 ML2 is an unsupported operating system, ignore the message and continue the installation.
Update your operating system if it is missing required file sets. If you receive a message that a file set is missing, such as file set X11.fnt.ucs.ttf, cancel the installation, update the operating system, and restart the installation.

As the base WebSphere Application Server product version changes, its prerequisites and corequisites change. Updating your database, Web server, Software Development Kit (SDK), and other software is probably necessary.

The base WebSphere Application Server product simplifies migrating product prerequisites, by providing the option to install a complimentary Web server and SDK on your supported operating system. You can uninstall back-level prerequisites and let the Installation wizard install current versions.

If the wizard finds a previous version of WebSphere Application Server, it prompts you to migrate applications and the configuration from the previous version, or to coexist with it. If it finds more than one previous version, the Installation wizard lists them for you to select which one to migrate.

Migrating or coexisting with an existing WebSphere Application Server node that AIX does not recognize.

In some cases, the InstallShield for MultiPlatforms (ISMP) program might not detect a previously installed version of WebSphere Application Server because of a failure to read the registry keys on AIX. You can force the migration and coexistence panel to display, by starting the installation with an option on the `/cdrom/aix/install` command.

For example, use this command:

```
./install -W previousVersionDetectedBean.previousVersionDetected="true"
```

You can also force the appearance of the coexistence panel to change conflicting port number assignments. For example, force the coexistence panel to appear using this command:

```
./install -W coexistenceOptionsBean.showCoexistence="true"
```

On either panel, identify the location of the existing product instance to cause it to be recognized.

25. Choose whether to install additional features or to install the product again, when there is a previous installation of the same level product. You can add features at any time, by running the installation wizard again.

This installation wizard panel appears when the installer program detects a previous installation at the same product level. The panel lets you select whether to add features to the existing installation, or perform a new installation to another directory.

If you intend to install additional features, follow this procedure to avoid component regression problems:

- a. Uninstall any interim fixes.
 - b. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.
 - c. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
 - d. Log off as root and back on.
 - e. Install new features.
 - f. Install the most current fix pack.
 - g. Install the most current cumulative fix.
 - h. Install any interim fixes to bring the node back to its previous fix level.
 - i. Use the administrative console on the Network Deployment node to synchronize all node agents.
26. Choose to migrate applications and the configuration from a previous version, to coexist with another version, or to neither coexist or migrate. Click **Next** to continue. See "Migrating and coexisting" on page 311 for more information.

All WebSphere Application Server products on a single machine share some of the code in the embedded messaging feature, if installed. The required level of the embedded messaging feature for V5.1 (CSD04) is not the same as for V5.0.0 or V5.0.1. The required level of the embedded messaging feature for V5.1 is the same as for V5.0.2.

If you attempt to install V5.1 on a machine where a version of the embedded messaging feature is at a release level earlier than CSD04, the installer program displays the message log in a panel. The message that you see is similar to one of the messages in the following example:

MQSeries or WebSphere MQ server at an earlier release than required to support embedded messaging is already installed on the system.
 Unsupported earlier maintenance level of MQSeries or WebSphere MQ detected.
 Unsupported earlier release of MQSeries client or WebSphere MQ client detected.
 Unsupported maintenance level of MQSeries client or WebSphere MQ client detected.
 Software conflict with MQSeries JMS SupportPac MA88 detected.

To correct the problem, perform one of the following actions:

- Upgrade the full MQSeries or WebSphere MQ product to WebSphere MQ at the latest level that supports embedded messaging (CSD04).
- **5.1+** See Installing WebSphere embedded messaging as the JMS provider for more information.
- Uninstall the existing MQSeries or WebSphere MQ product if MQSeries or WebSphere MQ is not required on this system and reinstall the WebSphere Application Server product. Select the embedded messaging feature.

The MQSeries JMS SupportPac MA88 problem is slightly different. Uninstall the MQSeries JMS SupportPac MA88 and reinstall the WebSphere Application Server product, selecting the embedded messaging feature. The function provided by SupportPac MA88 is included in the embedded messaging feature.

You can upgrade the WebSphere Application Server product to V5.0.2 before migrating it to V5.1 to avoid any problem with an incorrect level of the embedded messaging feature. See “Upgrading a Version 5.0.0 or Version 5.0.1 product to Version 5.0.2” on page 362.

You can also perform the procedure for migrating V5.0.0 or V5.0.1 with embedded messaging to V5.1. See “Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1” on page 338.

To share embedded messaging in a coexistence environment, the node names for each installation must be unique, so that each installation has a message queue manager that is named uniquely. To migrate V5.0.2 to V5.1, the node names must be identical. Therefore, the queue manager names are also identical, if you are migrating from V5.0.2 to V5.1.

To prevent losing the queue manager when you uninstall V5.0.2 (or V5.1), you must create a dummy queue manager before uninstalling one of the WebSphere Application Server versions. A series of migration topics in “Migrating and coexisting” on page 311 describe how to migrate after the installation.

- **5.1+** The first rule of migration is to migrate after you install WebSphere Business Integration Server Foundation, if you are planning to install the Integration Server:

If you are planning to install these products:		Select the migration option while installing this product:	
Network Deployment	Integration Server	Network Deployment	Integration Server
X		X	
X	X		X

The exception to the rule is to migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before extending either product.

- **5.1+** Migrating Integration Server also migrates the product that Integration Server extends.

You can perform a silent migration or configure for coexistence during a silent installation. Refer to “Installing silently” on page 174 for a description of performing a silent installation, including the options that you can specify.

The migration prompt appears only when the Installation wizard detects a previous version. The coexistence prompt appears when the Installation wizard detects any other installation, including another Version 5 installation.

If you choose to coexist, the wizard displays a Port selection panel, where you can specify port assignments that do not conflict with existing ports. For example, you can change the HTTP transport

port for coexistence, from 9081 (one more than the default Version 5 port number) to 9085 or higher, to avoid potential conflicts with port numbers that previous versions of WebSphere Application Server commonly use.

Use the **netstat -a** command to display all ports in use.

In some cases, such as when installing a non-English version, the Installation wizard might not detect a previous version. You can force the Migration panel or the Coexistence panel to appear, by starting the installation with an option on the **/cdrom/aix/install** command, where `mnt/cdrom` is the mount point for the product CD-ROM and `aix` is the platform directory on the CD-ROM.

You can also force the appearance of the Coexistence panel to change conflicting port number assignments. For example, the AIX WebSM system management server listens on port 9090 by default. To avoid a conflict with the administrative console port (`HTTP_TRANSPORT_ADMIN`) assignment, which is also 9090 by default, force the Coexistence panel to appear using this command:

```
/cdrom/aix/install
-W showCoexistencePanel.active="true"
```

The AIX WebSM system management server listens on port 9090 by default. If you suspect you have a port conflict, verify it by shutting down WebSphere Application Server and issuing the following command:

```
netstat -an |grep 9090
```

If you get a match, another process is already listening on port 9090. If you want the WebSM server and WebSphere Application Server to coexist, change the WebSphere Application Server administrative console port when installing WebSphere Application Server, or after installation.

Although not recommended, you can also disable the WebSM server. To disable the WebSM server, issue the following command to permanently disable WebSM server startup:

```
/usr/websm/bin/wmsserver -disable
```

5.1 +

If you choose neither the migration option nor the coexistence option, you can run Version 5.1.x and the previous version, but not at the same time. Although it is possible that both versions might coexist without port conflicts, you can ensure that both versions run together by selecting the coexistence option and checking for conflicting port assignments.

The Migration panel lists all previous releases that it can identify. If you highlight a release, the text boxes labeled, "select previous version," show the location of the previous product. Select the product to migrate. If you do not see the previous version that you intend to migrate, click **Select previous version** to enter a location and configuration file name if you are migrating a WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x installation.

The field labeled "Configuration file" is valid only for WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x. For the other versions of WebSphere Application Server that are supported by migration (Version 3.5 Standard Edition, Version 3.5 Extended Edition, and Version 4.0 Advanced Edition), the `admin.config` file provides the host and port values for the administrative server. If you use a file name other than `admin.config`, issue the commands that call the migration tools instead of migrating while installing. Issuing the commands that call the migration tools is described in "Migrating and coexisting" on page 311.

Migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before installing the Integration Server product.

You must start the administrative server of some previous versions so that the Installation wizard can export the configuration from the `admin.config` file.

Although you might select migration at this point in the installation process, the actual migration does not begin until after the Version 5 installation is complete. At that time, if the `WASPreUpgrade` tool fails, the Installation wizard does not call the `WASPostUpgrade` tool to complete the migration, but instead displays the `WASPreUpgrade.log` and `WASPostUpgrade.log` log files for you to diagnose the problem. After fixing the problem, such as starting the administrative server of a previous release, you can start the migration again, as described in "Migrating and coexisting" on page 311.

27. Select features to install and click **Next** to continue. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Choose from these features:

Deployment manager

Installs the product run time. It provides high performance and scalability across your deployment environment. It includes multiserver administration, server clustering, load balancing and workload management for hosting highly available e-business applications.

Web services

The UDDI registry and the IBM Web Services Gateway are enterprise applications that you can deploy to:

- A base WebSphere Application Server product node federated within a Network Deployment cell
- A stand-alone base WebSphere Application Server node

The Network Deployment product is not a stand-alone product for running enterprise applications. To deploy UDDI or the gateway, install the base WebSphere Application Server product. Although it is not installed by default, a copy of the base WebSphere Application Server product is packaged with the Network Deployment product.

5.1 + See Developing Web services for more information.

UDDI Registry

Installs a Version 2 compliant universal description, discovery, and identification (UDDI) registry, accessible from the UDDI registry user console application, or from SOAP or EJB interfaces.

5.1 + See IBM WebSphere UDDI Registry for more information.

Web Services Gateway

Includes a gateway between Internet and intranet environments so that clients can invoke Web services safely from outside a firewall. The gateway uses automatic protocol conversion for externalizing Web services.

5.1 + See Enabling Web services through the IBM Web Services Gateway for more information.

Embedded messaging client

Includes the client necessary for the administration of WebSphere MQ Queues and the mapping of JMS resources into the deployment manager JNDI namespace. It is the same client that you can install as part of the base product embedded messaging feature.

You can run the uninstaller program to remove all installed features.

28. Specify a destination directory and click **Next** to continue. Deleting the default target location and leaving an installation directory field empty stops you from continuing the installation process. The Installation wizard does not proceed when you click **Next**. Enter the required target directory to proceed to the next panel. Non-ASCII special characters are not supported in the name of the installation directory. Spaces are also not supported in the name of the installation directory.

The installer program checks for required space at the beginning of the installation. If you do not have enough space, stop the installation program, free space by deleting unused files and emptying the recycle bin, and restart the installation.

If not enough space is available, cancel the installation, allocate the 150 MB of temporary space that is required, and reinstall. The actual space required depends on the features that you are installing.

The installer program does not let you install the base product during an umbrella installation if sufficient space is not available to apply Cumulative Fix 2 and Cumulative Fix 1 for SDKs.

Suppose the installer program installs the base product and then runs into an error during the installation of the cumulative fix because of an insufficient amount of temporary space. In such a case, cancel the installation, allocate enough temporary space to install the cumulative fix, and restart the installation. The installer program picks up the installation where it left off, installing the cumulative fix to the base product, which you can select at the beginning of the installation.

If you have problems accessing the administrative console after installation, check the `installAdminConsole.log` file for a failure indication. Clean up the `/tmp` space and reinstall the administrative console using the `wsadmin` scripting facility.

If you must increase the `/tmp` allocation, stop the installation program, increase the allocation, and restart the installation.

29. Specify node information and click **Next**. Specify the node name and host name. Although the wizard inserts the machine name (of the installation platform) as the node name, you can specify any unique name. The node name is an arbitrary WebSphere Application Server-specific name that must be unique within a cell.

The host name is the network name for the physical machine on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote WebSphere Application Server nodes use the host name to connect to and to communicate with this node. Selecting a host name that other machines can reach within your network is extremely important. Do not use the generic `localhost` identifier for this value.

If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. WebSphere Application Server configuration files do not provide domain name resolution for multiple IP addresses on a machine with a single network address.

The value that you specify for the host name is used as the value of the `hostName` property in WebSphere Application Server configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as `xmachine.manhattan.ibm.com`
- The default short DNS host name string, such as `xmachine`
- Numeric IP address, such as `127.1.255.3`

The fully qualified DNS host name has the advantage of being totally unambiguous and also flexible. You have the flexibility of changing the actual IP address for the host system without having to change the WebSphere Application Server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A format disadvantage is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run WebSphere Application Server even when disconnected from the network. Define the short name to `127.0.0.1` (local loopback) in the hosts file to run disconnected. A format disadvantage is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A format disadvantage is that the numeric IP address is fixed. You must change the setting of the `hostName` property in WebSphere Application Server configuration documents whenever you change the machine IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another format disadvantage is that you cannot use the node if the host is disconnected from the network.

30. Review the summary information and click **Next** to install the product code or **Back** to change your specifications. The Summary panel displays the directory for the embedded messaging feature incorrectly on all Linux and UNIX-based platforms, as `/opt/IBM/WebSphere MQ`. Actual installation locations are `/usr/mqm` on AIX systems, and `/opt/mqm` on Linux and all UNIX-based platforms except AIX. When the installation is complete, the wizard displays the `install_root/logs/mq_install.log` installation log if you selected the embedded messaging feature and errors occur with its installation.
31. Review the `install_root/logs/mq_install.log` installation log if it displays. Click **Next** to continue. The wizard displays the Registration panel.

32. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Installation wizard starts the First Steps tool. See “firststeps command” on page 182.
33. Verify the success of the installer program by examining the Exit summary panel and the *install_root/logs/log.txt* for installation status. ISMP records a success message in the *install_root/logs/log.txt* file: "INSTFIN: The installation is complete." The log is the only source of status information for a silent installation.

Look for severe errors that the installer records in the *install_root/logs/log.txt* file in the installation root directory to verify that no file system or other unusual errors occurred during installation.

If the *install_root/logs/log.txt* file does not contain a record of any problems but problems exist, verify or troubleshoot the installation, as described in “Troubleshooting the installation” on page 185 and in Installation component troubleshooting tips .

If problems exist, correct them, uninstall the product, as described in “Uninstalling the product” on page 273, log off as root and back on, and reinstall.

When installing WebSphere Application Server with embedded messaging from the CD-ROM, the following warning messages are issued despite successful installation:

```
MQSERIES.MQM-CL-HPUX: A test command parameter is not valid." and "*
"hostname:/cdrom/hpux/messaging/images": Cannot open the
logfile on this target or source. Possibly the media is
read-only or there is a permission problem." on mq_install.log.
```

You can ignore the warning messages, because the problem does not affect the successful installation of WebSphere Application Server with embedded messaging.

34. Click **Finish** to close the Installation wizard.
35. Restrict access to the */var/mqm/errors* directory and messaging logging files. After installing the embedded messaging feature, you must restrict access to the */var/mqm* directory and to log files that are needed for embedded messaging, such that only the mqm user or members of the mqm user group have write access. For detailed information, see Installing WebSphere embedded messaging as the JMS provider and Securing messaging directories and log files .
36. Install the most current fix pack and cumulative fix for the Network Deployment product. See Recommended updates for WebSphere Application Server for information about downloading and installing the upgrades.
37. Tune your system for performance. For best performance on any platform, see Tuning performance parameter index .
38. Optional: Create a monitored process for the deployment manager process, as described in “Automatically restarting server processes” on page 176.

You can create monitored processes after the installation is complete.

Processes started by a **startManager.sh** command are not running as monitored processes, regardless of how you have configured them. You must start the deployment manager process with a shell script based on the example *rc.was* file, to have the deployment manager running as a monitored process.

The Installation wizard configures the product. It is not necessary to perform further configuration at this time.

You have now registered and successfully installed WebSphere Application Server Network Deployment and the features that you selected.

Uninstalling and reinstalling

See “Uninstalling the product” on page 273 for information about uninstalling any WebSphere Application Server product.

5.1 + If you uninstalled the product but left the embedded messaging feature installed, and now you want to reinstall the V5.1 product, see “Reinstalling V5.1 after uninstalling and leaving the embedded messaging feature installed” on page 310.

After uninstalling a WebSphere Application Server product, reinstalling into the same directory without first deleting all directory contents results in invalid XML configurations because of the retention of old files. To delete all of the files so that you can reinstall with a clean system, uninstall manually, as described in “Uninstalling the product” on page 273.

Miscellaneous tips for AIX platforms

Table 11. Installation and migration tips

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
AIX platforms	AIX platforms <ul style="list-style-type: none"> • Avoiding a segmentation fault error when installing or uninstalling • Ignoring DBCS messages when you do not require DBCS support. Otherwise, install the necessary patches • Navigating if the scroll bar disappears on the installation feature panel • Avoiding a DSAPI filter-loading error when the Lotus Domino Server starts
All platforms	All platforms
All Linux and UNIX platforms	All Linux and UNIX-based platforms

Mounting a CD-ROM drive on AIX

This topic describes how to mount a CD-ROM drive as the root user on an AIX operating system platform.

This topic describes a procedure for using SMIT to mount a CD-ROM drive. Before using SMIT to mount a CD-ROM drive, try the following command:

```
# mount -rv cdrfs /dev/cd0 /cdrom
```

The cd0 device is the name assigned to your CD-ROM. Your system might use another device name. You can use smit to see the device name.

The /cdrom directory is the mount directory. If a CD-ROM drive has ever been mounted on your system, there is a mount directory created with smit. It might be a different directory than the /cdrom directory.

If the command was successful, you can unmount the CD-ROM drive with the following command:

```
# umount /cdrom
```

If you cannot determine the device name and the mount directory, use the procedure in this topic to mount the CD-ROM drive.

This topic describes mounting a local CD-ROM drive. This procedure assumes you are already logged on as root.

1. Insert the CD-ROM labeled, *Integration Server Disk 1 of 2* into the CD-ROM drive.
2. If the /cdrom directory does not already exist, enter the following command to create it:


```
# mkdir /cdrom
```
3. Start SMIT by entering the following command:


```
# smit
```
4. From the SMIT System Management menu, select **System Storage Management (Physical & Logical Storage)**.
5. Click **File Systems**.
6. Click **Add/Change/Show/Delete File Systems**.

7. Click **CDROM File Systems**.
8. Click **Add a CDROM File System**.
9. Click **List** and select **CD0** as the device name.
10. Enter `/cdrom` in the Mount Point field.
11. Click **OK**. The Output: field contains messages about any errors that can occur at this point in the procedure. If another device is already mounted on the mount point, the process fails.
12. Click **Done**.
13. Click **Cancel** to clear the Add CDROM File System dialog box.
14. Press **F12** to close SMIT or click **Exit > Exit SMIT**.
15. Enter the following command:


```
# smit mountfs
```

 The Mount a File System dialog box appears.
16. Click **List** and select the **CD0** CD-ROM device.
17. Type `/cdrom` in the DIRECTORY field.
18. Click **List** in the Type of File System field.
19. Click **cdrfs** as the file system type.
20. Click **yes** for the Set the Mount as Read-Only system field.
21. Click **OK**. A successful mount generates no errors in the Output: field.
22. Click **Done**.
23. Click **Cancel** to exit SMIT.

For more detailed instructions, consult your AIX documentation.

Installing Network Deployment on HP-UX platforms

This topic describes how to install the WebSphere Application Server Network Deployment product as the root user on an HP-UX operating system platform.

You cannot install the product correctly as a non-root user. If you back up the product CD-ROM, do so as root. Backup copies made from non-root users do not preserve the correct file attributes and do not work.

If you encounter a problem such as not having enough temporary space or not having the right packages on your system, cancel the installation, make the required changes, and restart the installation.

The installation uses InstallShield for Multiplatforms (ISMP) to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

Installation programs in WebSphere Application Server products

5.1 +

Product	Link to installation procedure	Label on product CD
(Base) WebSphere Application Server	Base product installation	Application Server, IBM HTTP Server
Network Deployment	Network Deployment installation (which links to this procedure!)	Deployment Manager
WebSphere Business Integration Server Foundation	Integration Server installation	Integration Server Disk 1 and Integration Server Disk 2

Use the Network Deployment installation image to manage a multimachine environment, where you have installed the base product on different machines and want to manage the Application Servers in a group, or *cell*. If you buy the Network Deployment product, you also get the base product in the package.

WebSphere Business Integration Server Foundation is the V5.1 level of the Enterprise product. WebSphere Business Integration Server Foundation, Version 5.1 extends the base WebSphere Application Server V5.1 product. After migrating the underlying product to V5.1, do not reinstall the V5.0.x Enterprise product. Version 5.0.x of the Enterprise product does not extend Version 5.1 of the base WebSphere Application Server product nor does it extend the V5.1 Network Deployment product.

5.1 + If you buy the WebSphere Business Integration Server Foundation product, you also get the Network Deployment product and the base product in the package.

Creating multiple Application Servers on a single machine

Although you can create multiple servers on a base WebSphere Application Server node, the servers all share one set of configuration files. Changes that you make to one server affect the others. Configuration documents might become corrupted. The **wsinstance** command can create multiple configuration instances. Each instance is a stand-alone Application Server with its own set of configuration files. Or install the Network Deployment product to create and manage multiple base Application Servers.

Order of installation

5.1 + Install the base product before installing the Network Deployment product when installing both products on the same machine. Install the Network Deployment product before the Integration Server product that extends the Network Deployment product. You can install the Integration Server product before the base product. The Integration Server product can install the base product in what is known as an *umbrella* installation but Integration Server cannot install the Network Deployment product. Some features of the base product cannot be installed by Integration Server.

The embedded messaging feature that is included in the default installation requires that you install base before Network Deployment when installing both on the same machine. Otherwise, the order does not matter. See the following installation tip for more information:

Table 12. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component

This topic is available in Adobe PDF format on the product CD-ROM and online in an information center. The information center always has the most current information. The information center displays in the language of your machine locale if possible.

The installation guide PDF is updated occasionally. Download a new installation guide from:

5.1 + ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf

The Launchpad tool lets you access the product overview, the `readme.html` file, and the installation guide. See “Using the Launchpad to start the installation” on page 173.

You also use the Launchpad during the installation procedure to install the product. The installation program performs the following actions:

- Checks prerequisites automatically

- Looks for a previous WebSphere Application Server installation, to determine whether to display the Migration panel or the Coexistence panel during the installation

A known problem with the Launchpad can prevent it from using Netscape to open the documentation links.

If you use the Mozilla browser, some Launchpad links do not work. The Launchpad attempts to call the Netscape browser in the `/usr/bin/netscape` directory. Try a symbolic link to the Mozilla browser to fix the problem as shown in the following example:

```
ln -sf /usr/bin/mozilla /usr/bin/netscape
```

You can also launch the following links directly from your browser:

Button Link

Product Overview

<http://www.ibm.com/software/webservers/appserv/>

It is possible that your security options might prevent redirection, such as to the <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/appserv> Web site. If so, you can identify the IBM Web site as a trusted site to allow the redirection. For example, on Internet Explorer, add <http://www-306.ibm.com> to the list of Trusted Sites by selecting **Tools > Internet Options > Security (tab) > Trusted Sites**.

Readme File

CD-ROM mount point/readme/readme_en.html

Installation Guide

CD-ROM mount point/docs/InstallGuide_en.PDF

The *Installation Guide* is in Adobe PDF format. Viewing this document requires you to have Adobe Acrobat Reader, which is a free download from the Adobe Acrobat Web site.

After using the Launchpad to view product documentation, perform the following procedure. The procedure includes several preliminary steps before actually launching the installer program.

1. Stop all WebSphere Application Server-related Java processes on the machine where you are installing the product.
2. Stop any Web server process such as the IBM HTTP Server, if you are extending the base product.
3. Provide adequate disk space.

5.1 + The Network Deployment product requires the following disk space:

550 MB for the `/opt/WebSphere/DeploymentManager` directory

The installation root directory includes the base product code.

100 MB for the `/tmp` directory

The temporary directory is the working directory for the installation program.

Verify that you have at least 100 MB of free space to avoid problems.

A message about free space occurs when less than 4 MB of free space remains after starting the installation. The InstallShield for MultiPlatforms (ISMP) program displays a message about using the `-is:tempdir` parameter to identify an alternate temporary space directory.

Ignore any `-is:tempdir` message. The `-is:tempdir` parameter is not supported. Cancel the installation, allocate a total of 100 MB of free space in the `/tmp` directory, and start the installation again.

650 MB total requirement without the embedded messaging feature

The total amount of space required includes the `/tmp` space, which is released after installation.

The Installation wizard displays required space for individual features on the Feature selection panel. The Installation wizard also warns you if you do not have enough space to install the product.

If you plan to migrate applications and the configuration from a previous version, verify that application objects have available disk space. As a rough guideline, plan for space equal to 110 percent of the size of the application objects:

- **For Version 3.5.x:** The size of application Java archive (JAR) files, Web archive (WAR) files, and servlet files
- **For Version 4.0.x:** The size of enterprise archive (EAR) files
- **5.1+ For Version 5.0.x:** The size of EAR files

4. Define the user groups and the user needed for the embedded messaging feature.
 - a. Use the System Administration Manager (SAM) to create the **mqm** and **mqbrkrs** user groups.
 - b. Use SAM to create the **mqm** user.
 - c. Use SAM to add the **mqm** and **root** users to the **mqm** group.
 - d. Use SAM to add **root** to the **mqbrkrs** group.

The recommended user ID for running the JMS server process is root. If you do run the JMS server process under another user ID, add that user ID to the **mqm** and **mqbrkrs** groups. User IDs longer than 12 characters cannot be used for authentication with the embedded WebSphere JMS provider.

The mqm user starts the JMS server for general JMS support and the WebSphere embedded broker for WebSphere Application Server topic connections.

5. Log out and back in to pick up the secondary user groups, mqm and mqbrkrs, for root. Use **ssh** instead of **telnet** to log in or run the following command after logging on:

```
su -
```

Use the **id** command or the **groups** command to see defined groups for root. If mqm and mqbrkrs are not in the list that is returned, you cannot install the embedded messaging feature:

```
[root@was2 root]# id
uid=0(root) gid=0(root)
groups=0(root),1(bin),2(daemon),
3(sys),4(adm),6(disk),10(wheel),500(mqm),501(mqbrkrs)
```

```
[root@was2 root]# groups
root bin daemon sys adm disk wheel mqm mqbrkrs
```

6. Allocate adequate disk space for the embedded messaging feature if you are planning to install the feature. The installation locations for the embedded messaging feature are fixed as shown in the following table, which lists the locations for the base messaging functions and the messaging broker functions for publish/subscribe messaging.

Table 13. Space requirements for the embedded messaging feature on HP-UX

Component	Base code	Broker code	Base data	Broker data
Path	/usr/mqm	/usr/opt/wemps	/var/mqm	/var/wemps
Client subfeature	15 MB	15 MB	5 MB	N/A

7. Prepare to install the embedded messaging feature with WebSphere MQ. The embedded messaging feature is based on the IBM WebSphere MQ product. The feature and the product each provide a Java message service (JMS) function that supports queues for point-to-point messaging and topics for publish and subscribe messaging.

You can install the embedded messaging feature with or without the WebSphere MQ product on the same machine. To support both the embedded messaging feature and the WebSphere MQ product on the same machine, the WebSphere MQ product must be at a certain fix level and must have several of its features installed.

If you already have WebSphere MQ installed, you can configure it as the JMS provider. Otherwise, you can install the embedded messaging feature during the installation or install the WebSphere MQ product or another JMS provider after you install.

Even though you might decide now to install only the embedded messaging feature, you can install the WebSphere MQ product later and use the IBM WebSphere MQ product as the JMS provider instead.

8. Verify that you have upgraded to WebSphere MQ 5.3 with the CSD04 update to install embedded messaging on a machine where you already have WebSphere MQ installed.

Determine if your WebSphere MQ 5.3 installation is at the required level by running the **mqver** utility provided by WebSphere MQ.

The required level as indicated by mqver is shown below:

```
Name:      WebSphere MQ
Version:   530.4 CSD04
...
```

9. Verify that you have installed the required WebSphere MQ 5.3 features to install embedded messaging on a machine where you already have WebSphere MQ installed. Verify that you have installed the *Java messaging* feature.

If you attempt to install the embedded messaging feature when WebSphere MQ is already installed, the level of WebSphere MQ must be Version 5.3 with the required MQ feature. Otherwise, the installation of the embedded messaging feature fails with prerequisite check errors.

10. Create and mount a journalized file system called `/var/mqm` for your messaging working data. Use a partitioning strategy with a separate volume for embedded messaging or WebSphere MQ data to isolate system activity from the potentially high volume of messaging work that can build up in the `/var/mqm` directory.
11. Create separate file systems for log data in the `var/mqm/log` directory and error files in the `var/mqm/errors` directory. Store log files on a different physical volume from the embedded messaging queues, which are in the `var/mqm` directory. This ensures data integrity in the case of a hardware failure. If you are creating separate file systems, allow the following minimum free space:

30 MB `/var/mqm`

20 MB `/var/mqm/log`

4 MB `/var/mqm/errors`

The `/var` file system stores all the security logging information for the system and stores the temporary files for e-mail and printing. Therefore, it is critical that you maintain free space in `/var` for these operations. If you do not create a separate file system for messaging data, and `/var` fills up, all security logging stops on the system until free space is available in `/var`. Also, e-mail and printing do not work without some available free space in `/var`.

You have the same options for creating file systems for the embedded messaging feature as you do for WebSphere MQ. For example, if you cannot install the embedded messaging options in the required file system (for example, if it is too small), you can do one of the following *before* installing the Embedded Messaging options:

- Create and mount a new file system for the installation directory.
- Create a new directory anywhere on your machine, and create a symbolic link from the required installation directory to the new directory. For example:

```
mkdir /bigdisk/mqm
ln -s /bigdisk/mqm /usr/mqm
```

12. Set kernel values to support Network Deployment.

Several HP-UX kernel values are typically too small for the product and the embedded messaging feature. Starting the internal JMS server or client with insufficient kernel resources produces a First Failure Support Technology (FFST) file in the `/var/mqm/errors` directory.

Before installing embedded messaging, review the machine configuration and, if needed, set appropriate HP-UX kernel settings.

Queue managers are generally independent of each other. Therefore system kernel parameters, for example `shmmni`, `semmni`, `semmns`, and `semmnu` need to allow for the number of queue managers in the system.

To set kernel parameters, perform the following steps:

- a. Log into the host machine as root.
- b. Determine the physical memory, which you must know to avoid setting certain kernel parameters above the physical capacity:
 - 1) Start the HP-UX System Administration Manager (SAM) utility.
 - 2) Select **Performance Monitors > System Properties > Memory**.
 - 3) Note the value for Physical Memory and click **OK**.
 - 4) Exit from the SAM utility.
- c. Set the `maxfiles` and `maxfiles_lim` parameters to at least 4096. (The following table recommends 8000 and 8196, respectively. You must first edit the `/usr/conf/master.d/core-hpux` file, to allow the SAM utility to set values greater than 2048:
 - 1) Open the file `/usr/conf/master.d/core-hpux` in a text editor.
 - 2) Change the line, "`*range maxfiles<=2048`" to "`*range maxfiles<=60000`"
 - 3) Change the line, "`*range maxfiles_lim<=2048`" to "`*range maxfiles_lim<=60000`"
 - 4) Save and close the file. Old values might be stored in the `/var/sam/boot.config` file. Force the SAM utility to create a new `boot.config` file:
 - a) Move the existing version of the `/var/sam/boot.config` file to another location, such as the `/tmp` directory.
 - b) Start the SAM utility.
 - c) Select **Kernel Configuration > Configurable Parameters**. When the Kernel Configuration window opens, a new `boot.config` file exists.
Alternatively, rebuild the `boot.config` file with the following command:

```
# /usr/sam/sbin/getkinfo -b
```
- d. Set new kernel parameter values:
 - 1) Start the SAM utility.
 - 2) Select **Kernel Configuration > Configurable Parameters**.
 - 3) For each of the parameters in the following table, perform this procedure:
 - a) Highlight the parameter to change.
 - b) Select **Actions > Modify Configurable Parameter**.
 - c) Type the new value in the **Formula/Value** field.
 - d) Click **OK**.

Typical kernel settings for running WebSphere Application Server appear in the following table:

Parameter	Value
<code>dbc_max_pct</code>	25
<code>maxdsiz</code>	805306358
<code>maxdsiz</code>	2048000000 (when the base and Network Deployment products are on one machine)
<code>maxfiles_lim</code>	8196 (Change this one before <code>maxfiles</code> .)
<code>maxfiles</code>	8000
<code>maxssiz</code>	8388608
<code>maxswapchunks</code>	8192
<code>max_thread_proc</code>	3000
<code>maxuprc</code>	512
<code>maxusers</code>	512
<code>msgmap</code>	2048
<code>msgmax</code>	65535
<code>msgmax</code>	131070 (when the base and Network Deployment products are on one machine)
<code>msgmnb</code>	65535

Parameter	Value
msgmnb	131070 (when the base and Network Deployment products are on one machine)
msgmni	50
msgseg	32767
msgssz	32
msgtql	2046
nfile	58145
nflocks	3000
ninode	60000
nkthread	7219
nproc	4116
npty	2024
nstrpty	1024
nstrtel	60
sema	1 (with embedded messaging)
semaem	16384 (with embedded messaging)
semmap	514
semmni	2048
semmns	16384 (with embedded messaging)
semmnu	1024
semume	200
semvmx	32767 (with embedded messaging)
shmmax	2147483647
shmem	1 (with embedded messaging)
shmmni	1024
shmseg	1024
STRMSGSZ	65535

When Network Deployment and IBM DB2 are on the same machine, some kernel values are higher than those shown in the preceding table.

See the following Web links for more information:

- Recommended HP-UX kernel configuration parameters for DB2 V8
- (V5.0.x only) Recommended HP-UX kernel configuration parameters for DB2 V7

e. Select **Actions > Process New Kernel**.

f. Click **Yes** on the information window to confirm your decision to restart the machine.

Follow on-screen instructions to restart your machine and to enable the new settings.

g. If you plan to redirect displays to non-HP machines, do the following before running the installation wizard:

1) Issue the following command to obtain information on all public locales accessible to your application:

```
# locale -a
```

2) Choose a value for your system from the output that is displayed and set the LANG environment variable to this value. Here is an example command that sets the value of LANG to en_US.iso88591

```
# export LANG=en_US.iso8859
```

13. Verify that prerequisites and corequisites are at the required release levels. Although the Installation wizard checks for prerequisite operating system patches with the prereqChecker application, review the prerequisites on the IBM WebSphere Application Server supported hardware, software, and APIs Web site if you have not already done so.

Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Some operating systems that were not supported at the time that this product was shipped on CD-ROM might now be supported. You might receive a message from the prereqChecker program that an operating system is not supported when, in fact, the operating system is supported.

Always consult the IBM WebSphere Application Server supported hardware, software, and APIs Web site to determine whether your operating system is supported when you receive a message from the prereqChecker program. The Web site lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system.

After confirming that your operating system is supported and that you have installed all necessary patches, you can click **Next** to continue an installation when you receive an error message from the prereqChecker program.

Ignoring a prerequisite checker error for a patch when a superset of the patch has been installed.

If you apply a superset patch that includes a specific patch that the prerequisite checker program requires, you can ignore the message from the installer program and continue the installation.

You can always continue the installation whenever you see an error message about a missing prerequisite. It is your option to ignore the messages and continue the installation. Although the list of prerequisite patches that the installer program checks is extensive, the list does not include all superset patches.

For example, the installer program checks for patch PHCO_29109. The HP-UX Web site recommends that you install a superset patch, PHCO_29960, instead of PHCO_29109.

After installing the patch and rebooting the machine, reinstalling the base WebSphere Application Server results in the same message about the missing PHCO_29109 prerequisite. You can ignore the error because you know that you have installed a superset patch.

14. If you have uninstalled WebSphere Application Server Version 4.1, remove any entries from the `/usr/bin/jitk.db` file. Remove any remaining artifacts from an uninstalled V4.1 system to prevent the display of the Coexistence panel or the Migration panel during installation.

The Installation wizard might display the Migration panel or the Coexistence panel even though you have uninstalled WebSphere Application Server Enterprise Edition Version 4.1. You can prevent the Installation wizard from recognizing a previously deleted version of V4.1 Application Server by removing the following entry from the `/usr/bin/jitk.db` file:

```
WebSphere Application Server 4.1
```

Remove other Version 4.x entries for WebSphere Application Server products that are no longer on your system.

15. Verify the system **cp** command when using emacs or other freeware.

If you have emacs or other freeware installed on your HP-UX system, verify that the system **cp** command is being used.

 - a. Type which cp at the command prompt before running the installation program for the WebSphere Business Integration Server Foundation product.
 - b. Remove the freeware directory from your PATH if the resulting directory output includes freeware. For example, if the output is similar to this `/opt/freeware/bin/cp` message, remove the directory from the PATH.
 - c. Install the WebSphere Application Server Network Deployment product.
 - d. Add the freeware directory back to the PATH.

If you install with a **cp** command that is part of a freeware package, the installation might appear to complete successfully, but the Java 2 SDK that the product installs might have missing files in the *install_root/java* directory.

Missing files can destroy required symbolic links. If you remove the freeware **cp** command from the PATH, you can install the Network Deployment product successfully.

Perform the following step to verify that the Java 2 SDK is working correctly.

16. Verify the Java 2 SDK on the WebSphere Application Server Network Deployment CD. If you copied the product CD to back it up and are using a backup version, perform the following steps to verify that the Java 2 SDK on the product CD-ROM is working correctly.
 - a. Change directories to the */hpux/WAS/jdk/java/bin* directory on the product CD-ROM.
For example:

```
cd /mnt/hpux/WAS/jdk/java/bin
```
 - b. Verify the Java 2 SDK version.
Type the following command:

```
./java -version
```


The command completes successfully with no errors when the Java 2 SDK is intact.
17. Select the Installation wizard method or the silent installation method but do not start the installation yet. The installer program has two interfaces, the Installation wizard and a silent command-line installation.

Performing the installation with the wizard

You can start the Installation wizard using the Launchpad or directly using the **install** command.

The default installation method is to open a command window and issue the command to start the Launchpad tool. Click the **Install the product** option on the Launchpad. (See “Using the Launchpad to start the installation” on page 173.)

This option launches the Installation wizard in the language of your machine locale unless there is no translation for your locale, in which case you receive the English version.

A short delay occurs before the ISMP wizard displays. You do not need to click the Install the product option more than once to cause the wizard to display. The delay is particularly noticeable on x-windows platforms.

You can also start the Installation wizard using the **/cdrom/hpux/install** command, where */cdrom* is the mount point for the product CD-ROM and *hpux* is the platform directory.

Performing a silent installation

You can perform a silent installation using the **-options responsefile** parameter with the command method:

```
fully_qualified_CD_pathname/install  
-options fully_qualified_hard_disk_drive_path_name/responsefile
```

```
# /cdrom/hpux/install -options /tmp/my_responsefile
```

Start the silent installation with a fully qualified path to the options response file. Otherwise, the Installation wizard starts.

A silent installation causes the installation program to read your responses from the options response file, instead of from the wizard interface. You must customize the *responsefile* before installing silently. See *ae/tins_silentND.dita*.

After customizing the file, you can issue the command to silently install. See “Installing silently” on page 174.

After issuing the command, the following text displays:

```
# .....  
.InstallShield Wizard
```

Initializing InstallShield Wizard...

Searching for Java(tm) Virtual Machine...

The silent installation runs without displaying status to the window:

- Press **Enter** when the installation is complete to return to the command prompt.
- Press **Enter** at any time to return from a background job to the command prompt.

You can change the `-W launchPRTBean.active` option to display the Registration panel to indicate the completion of a silent installation on a local system with a graphical user interface.

To determine the status of the silent installation, review the installation logs in the `install_root/logs` directory or in the `/tmp` directory. See “Troubleshooting the installation” on page 185 for more information about log files.

Silent installation is particularly useful if you install the product often.

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard.

Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize their options.

Asynchronous and synchronous command lines

After running the **install** command, the command line returns synchronously. A synchronous **install** command returns the command line after the installation is complete.

You can start the installation asynchronously with the installation process and its children processes all running as background processes. Consult your operating system documentation to learn how to issue asynchronous commands. After running the **install** command, the command line returns immediately.

Do not misinterpret an asynchronous command line to mean that the installation has finished when the command prompt returns. Although the command line returns, either the Installation wizard or a silent installation might still be in progress.

Installing with a network file system mount

If you must use an NFS mount, see the following installation tip:

Table 14. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Platform-specific issues when dealing with network file systems

18. Insert the product CD-ROM labeled, *Deployment Manager* into a CD drive.
19. Open a shell window and mount the CD drive, as described in “Mounting a CD drive on HP-UX” on page 77. Use the same shell window throughout the installation procedure.
20. Start the installation with the `/cdrom/hpux/launchpad.sh` command, where `/cdrom` is the mount point for the product CD-ROM and `hpux` is the platform directory on the CD-ROM. You can also start the installation with the `/cdrom/hpux/install` command:

```
# /cdrom/hpux/install
```

The readme link in the Launchpad is to the `readme.html` file in the CD root directory. The *Getting Started* document that contains installation information is in the `docs` directory on the CD.

5.1+ Download a new installation guide at

ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf.

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize the options.

21. Click **Next** to continue. The license agreement displays.
The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.
22. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist.

As the Network Deployment product version changes, its prerequisites and corequisites change.

The Network Deployment product simplifies migrating product prerequisites, by providing the option to install a complimentary Java 2 SDK on your supported operating system. You can uninstall back-level prerequisites and let the Installation wizard install current versions.

If the wizard finds a previous version of WebSphere Application Server, it prompts you to migrate applications and the configuration from the previous version, or to coexist with it. If it finds more than one previous version, the Installation wizard lists them for you to select which one to migrate. As of V5.1, the Installation wizard also lists WebSphere Application Server - Express as a candidate for automatic migration, when detected.

Use the same cell name when migrating Network Deployment from V5.0.x to V5.1.x. If you use a different cell name, federated nodes cannot successfully migrate to the Network Deployment V5.1.x cell.

You can use a different node name, if you prefer, when migrating Network Deployment from V5.0.x to V5.1.x. However, you must use the administrative console after installation, to change the Secure Sockets Layer (SSL) settings that mention the node name.

Migrating or coexisting with an existing WebSphere Application Server node that HP-UX does not recognize.

In some cases, the InstallShield for MultiPlatforms (ISMP) program does not detect a previously installed version of WebSphere Application Server because of a failure to read the registry keys on HP-UX. You can force the migration and coexistence panel to display, by starting the installation with an option on the **/cdrom/hpux/install** command.

For example, use this command:

```
./install -W previousVersionDetectedBean.previousVersionDetected="true"
```

You can also force the appearance of the coexistence panel to change conflicting port number assignments. For example, force the coexistence panel to appear using this command:

```
./install -W coexistenceOptionsBean.showCoexistence="true"
```

On either panel, identify the location of the existing product instance to cause it to be recognized.

Use the same cell name when migrating Network Deployment from V5.0.x to V5.1.x. If you use a different cell name, federated nodes cannot successfully migrate to the Network Deployment V5.1.x cell.

You can use a different node name, if you prefer, when migrating Network Deployment from V5.0.x to V5.1.x. However, you must use the administrative console after installation, to change the Secure Sockets Layer (SSL) settings that mention the node name.

23. Choose whether to install additional features or to install the product again, when there is a previous installation of the same level product. You can add features at any time, by running the installation wizard again.

This installation wizard panel appears when the installer program detects a previous installation at the same product level. The panel lets you select whether to add features to the existing installation, or perform a new installation to another directory.

If you intend to install additional features, follow this procedure to avoid component regression problems:

- a. Uninstall any interim fixes.
- b. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.

- c. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
 - d. Log off as root and back on.
 - e. Install new features.
 - f. Install the most current fix pack.
 - g. Install the most current cumulative fix.
 - h. Install any interim fixes to bring the node back to its previous fix level.
 - i. Use the administrative console on the Network Deployment node to synchronize all node agents.
24. Choose to migrate applications and the configuration from a previous version, to coexist with another version, or to neither coexist or migrate. Click **Next** to continue. See “Migrating and coexisting” on page 311 for more information.

All WebSphere Application Server products on a single machine share some of the code in the embedded messaging feature, if installed. The required level of the embedded messaging feature for V5.1 (CSD04) is not the same as for V5.0.0 or V5.0.1. The required level of the embedded messaging feature for V5.1 is the same as for V5.0.2.

If you attempt to install V5.1 on a machine where a version of the embedded messaging feature is at a release level earlier than CSD04, the installer program displays the message log in a panel. The message that you see is similar to one of the messages in the following example:

```
MQSeries or WebSphere MQ server at an earlier release than required to support
embedded messaging is already installed on the system.
Unsupported earlier maintenance level of MQSeries or WebSphere MQ detected.
Unsupported earlier release of MQSeries client or WebSphere MQ client detected.
Unsupported maintenance level of MQSeries client or WebSphere MQ client detected.
Software conflict with MQSeries JMS SupportPac MA88 detected.
```

To correct the problem, perform one of the following actions:

- Upgrade the full MQSeries or WebSphere MQ product to WebSphere MQ at the latest level that supports embedded messaging (CSD04).
- 5.1+** See Installing WebSphere embedded messaging as the JMS provider for more information.
- Uninstall the existing MQSeries or WebSphere MQ product if MQSeries or WebSphere MQ is not required on this system and reinstall the WebSphere Application Server product. Select the embedded messaging feature.

The MQSeries JMS SupportPac MA88 problem is slightly different. Uninstall the MQSeries JMS SupportPac MA88 and reinstall the WebSphere Application Server product, selecting the embedded messaging feature. The function provided by SupportPac MA88 is included in the embedded messaging feature.

You can upgrade the WebSphere Application Server product to V5.0.2 before migrating it to V5.1 to avoid any problem with an incorrect level of the embedded messaging feature. See “Upgrading a Version 5.0.0 or Version 5.0.1 product to Version 5.0.2” on page 362.

You can also perform the procedure for migrating V5.0.0 or V5.0.1 with embedded messaging to V5.1. See “Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1” on page 338.

To share embedded messaging in a coexistence environment, the node names for each installation must be unique, so that each installation has a message queue manager that is named uniquely. To migrate V5.0.2 to V5.1, the node names must be identical. Therefore, the queue manager names are also identical, if you are migrating from V5.0.2 to V5.1.

To prevent losing the queue manager when you uninstall V5.0.2 (or V5.1), you must create a dummy queue manager before uninstalling one of the WebSphere Application Server versions. A series of migration topics in “Migrating and coexisting” on page 311 describe how to migrate after the installation.

5.1+ The first rule of migration is to migrate after you install WebSphere Business Integration Server Foundation, if you are planning to install the Integration Server:

If you are planning to install these products:		Select the migration option while installing this product:	
Network Deployment	Integration Server	Network Deployment	Integration Server
X		X	
X	X		X

The exception to the rule is to migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before extending either product.

5.1 + Migrating Integration Server also migrates the product that Integration Server extends.

You can perform a silent migration or configure for coexistence during a silent installation. Refer to “Installing silently” on page 174 for a description of performing a silent installation, including the options that you can specify.

The migration prompt appears only when the Installation wizard detects a previous version. The coexistence prompt appears when the Installation wizard detects any other installation, including another Version 5 installation.

If you choose to coexist, the wizard displays a Port selection panel, where you can specify port assignments that do not conflict with existing ports. For example, you can change the HTTP transport port for coexistence, from 9081 (one more than the default Version 5 port number) to 9085 or higher, to avoid potential conflicts with port numbers that previous versions of WebSphere Application Server commonly use.

Use the **netstat -a** command to display all ports in use.

5.1 + If you choose neither the migration option nor the coexistence option, you can run Version 5.1.x and the previous version, but not at the same time. Although it is possible that both versions might coexist without port conflicts, you can ensure that both versions run together by selecting the coexistence option and checking for conflicting port assignments.

The Migration panel lists all previous releases that it can identify. If you highlight a release, the text boxes labeled, “select previous version,” show the location of the previous product. Select the product to migrate. If you do not see the previous version that you intend to migrate, click **Select previous version** to enter a location and configuration file name if you are migrating a WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x installation.

The field labeled “Configuration file” is valid only for WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x. For the other versions of WebSphere Application Server that are supported by migration (Version 3.5 Standard Edition, Version 3.5 Extended Edition, and Version 4.0 Advanced Edition), the `admin.config` file provides the host and port values for the administrative server. If you use a file name other than `admin.config`, issue the commands that call the migration tools instead of migrating while installing. Issuing the commands that call the migration tools is described in “Migrating and coexisting” on page 311.

Migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before installing the Integration Server product.

You must start the administrative server of some previous versions so that the Installation wizard can export the configuration from the `admin.config` file.

Although you might select migration at this point in the installation process, the actual migration does not begin until after the Version 5 installation is complete. At that time, if the WASPreUpgrade tool fails, the Installation wizard does not call the WASPostUpgrade tool to complete the migration, but instead displays the `WASPreUpgrade.log` and `WASPostUpgrade.log` log files for you to diagnose the problem. After fixing the problem, such as starting the administrative server of a previous release, you can start the migration again, as described in “Migrating and coexisting” on page 311.

25. Select features to install and click **Next** to continue. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Choose from these features:

Deployment manager

Installs the product run time. It provides high performance and scalability across your deployment environment. It includes multiserver administration, server clustering, load balancing and workload management for hosting highly available e-business applications.

Web services

The UDDI registry and the IBM Web Services Gateway are enterprise applications that you can deploy to:

- A base WebSphere Application Server product node federated within a Network Deployment cell
- A stand-alone base WebSphere Application Server node

The Network Deployment product is not a stand-alone product for running enterprise applications. To deploy UDDI or the gateway, install the base WebSphere Application Server product. Although it is not installed by default, a copy of the base WebSphere Application Server product is packaged with the Network Deployment product.

5.1+ See Developing Web services for more information.

UDDI Registry

Installs a Version 2 compliant universal description, discovery, and identification (UDDI) registry, accessible from the UDDI registry user console application, or from SOAP or EJB interfaces.

5.1+ See IBM WebSphere UDDI Registry for more information.

Web Services Gateway

Includes a gateway between Internet and intranet environments so that clients can invoke Web services safely from outside a firewall. The gateway uses automatic protocol conversion for externalizing Web services.

5.1+ See Enabling Web services through the IBM Web Services Gateway for more information.

Embedded messaging client

Includes the client necessary for the administration of WebSphere MQ Queues and the mapping of JMS resources into the deployment manager JNDI namespace. It is the same client that you can install as part of the base product embedded messaging feature.

You can run the uninstaller program to remove all installed features.

26. Specify a destination directory and click **Next** to continue. Deleting the default target location and leaving an installation directory field empty stops you from continuing the installation process. The Installation wizard does not proceed when you click **Next**. Enter the required target directory to proceed to the next panel. Non-ASCII special characters are not supported in the name of the installation directory. Spaces are also not supported in the name of the installation directory.

The installer program checks for required space at the beginning of the installation. If you do not have enough space, stop the installation program, free space by deleting unused files and emptying the recycle bin, and restart the installation.

If not enough space is available, cancel the installation, allocate the 150 MB of temporary space that is required, and reinstall. The actual space required depends on the features that you are installing.

If you have problems accessing the administrative console after installation, check the `installAdminConsole.log` file for a failure indication. Clean up the `/tmp` space and reinstall the administrative console using the `wsadmin` scripting facility.

If you must increase the `/tmp` allocation, stop the installation program, increase the allocation, and restart the installation.

27. Specify node information and click **Next**. Specify the node name and host name. Although the wizard inserts the machine name (of the installation platform) as the node name, you can specify any unique name. The node name is an arbitrary WebSphere Application Server-specific name that must be unique within a cell.

The host name is the network name for the physical machine on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote WebSphere Application Server nodes use the host name to connect to and to communicate with this node. Selecting a host name that other machines can reach within your network is extremely important. Do not use the generic localhost identifier for this value.

If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. WebSphere Application Server configuration files do not provide domain name resolution for multiple IP addresses on a machine with a single network address.

The value that you specify for the host name is used as the value of the `hostName` property in WebSphere Application Server configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as `xmachine.manhattan.ibm.com`
- The default short DNS host name string, such as `xmachine`
- Numeric IP address, such as `127.1.255.3`

The fully qualified DNS host name has the advantage of being totally unambiguous and also flexible. You have the flexibility of changing the actual IP address for the host system without having to change the WebSphere Application Server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A format disadvantage is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run WebSphere Application Server even when disconnected from the network. Define the short name to `127.0.0.1` (local loopback) in the hosts file to run disconnected. A format disadvantage is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A format disadvantage is that the numeric IP address is fixed. You must change the setting of the `hostName` property in WebSphere Application Server configuration documents whenever you change the machine IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another format disadvantage is that you cannot use the node if the host is disconnected from the network.

28. Review the summary information and click **Next** to install the product code or **Back** to change your specifications. The Summary panel displays the directory for the embedded messaging feature incorrectly on all Linux and UNIX-based platforms, as `/opt/IBM/WebSphere MQ`. Actual installation locations are `/usr/mqm` on AIX systems, and `/opt/mqm` on Linux and all UNIX-based platforms except AIX. When the installation is complete, the wizard displays the `install_root/logs/mq_install.log` installation log if you selected the embedded messaging feature and errors occur with its installation.
29. Review the `install_root/logs/mq_install.log` installation log if it displays. Click **Next** to continue. The wizard displays the Registration panel.
30. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Installation wizard starts the First Steps tool. See “firststeps command” on page 182.
31. Verify the success of the installer program by examining the Exit summary panel and the `install_root/logs/log.txt` for installation status. ISMP records a success message in the `install_root/logs/log.txt` file: “INSTFIN: The installation is complete.” The log is the only source of status information for a silent installation.

Look for severe errors that the installer records in the `install_root/logs/log.txt` file in the installation root directory to verify that no file system or other unusual errors occurred during installation.

If the `install_root/logs/log.txt` file does not contain a record of any problems but problems exist, verify or troubleshoot the installation, as described in “Troubleshooting the installation” on page 185 and in Installation component troubleshooting tips .

If problems exist, correct them, uninstall the product, as described in “Uninstalling the product” on page 273, log off as root and back on, and reinstall.

When installing WebSphere Application Server with embedded messaging from the CD-ROM, the following warning messages are issued despite successful installation:

```
MQSERIES.MQM-CL-HPUX: A test command parameter is not valid." and "*
hostname:/cdrom/hpux/messaging/images": Cannot open the
  logfile on this target or source. Possibly the media is
  read-only or there is a permission problem." on mq_install.log.
```

You can ignore the warning messages, because the problem does not affect the successful installation of WebSphere Application Server with embedded messaging.

32. Click **Finish** to close the Installation wizard.
33. Restrict access to the `/var/mqm/errors` directory and messaging logging files. After installing the embedded messaging feature, you must restrict access to the `/var/mqm` directory and to log files that are needed for embedded messaging, such that only the `mqm` user or members of the `mqm` user group have write access. For detailed information, see Installing WebSphere embedded messaging as the JMS provider and Securing messaging directories and log files .
34. Install the most current fix pack and cumulative fix for the Network Deployment product. See Recommended updates for WebSphere Application Server for information about downloading and installing the upgrades.
35. Tune your system for performance. For best performance on any platform, see Tuning performance parameter index .
36. Optional: Create a monitored process for the deployment manager process, as described in “Automatically restarting server processes” on page 176.

You can create monitored processes after the installation is complete.

Processes started by a `startManager.sh` command are not running as monitored processes, regardless of how you have configured them. You must start the deployment manager process with a shell script based on the example `rc.was` file, to have the deployment manager running as a monitored process.

The Installation wizard configures the product. It is not necessary to perform further configuration at this time.

You have now registered and successfully installed WebSphere Application Server Network Deployment and the features that you selected.

Uninstalling and reinstalling

See “Uninstalling the product” on page 273 for information about uninstalling any WebSphere Application Server product.

5.1+ Reinstalling a V5.1 instance after uninstalling and leaving the embedded messaging feature installed

5.1+ If you uninstall a V5.1 instance but leave the embedded messaging feature installed because of other WebSphere Application Server instances on the machine, rename the `/opt/WebSphere/AppServer51` directory to cause HP-UX to be unable to locate the previous installation that is still registered. Reinstall into a directory other than the `/opt/WebSphere/AppServer51` directory or the renamed directory. Use a new directory name for reinstalling the V5.1 instance.

5.1 + If you uninstall all WebSphere Application Server products on the machine and are now reinstalling, follow the procedure in “Manually uninstalling on HP-UX platforms” on page 296 instead.

5.1 + For example, assume that the V5.0.2 base product exists in /opt/WebSphere/AppServer502 with the embedded messaging server and client feature. You then install V5.1 base to the following directory, /opt/WebSphere/AppServer51, but later you uninstall the base product with the **uninstall** command and select to leave the embedded messaging feature installed. And now you decide to reinstall the V5.1 base product.

5.1 + Rename /opt/WebSphere/AppServer51 to /opt/WebSphere/AppServer51_bk before reinstalling V5.1 into the /opt/WebSphere/AppServer51New directory.

5.1 + This step is required before you can reinstall the product onto the same system. Reinstalling into the same directory without first deleting all directory contents results in invalid XML configurations because of the retention of old files. If you want to correct the registry entries and uninstall all WebSphere Application Server products on the machine, uninstall manually to delete all of the files so that you can reinstall with a clean system, as described in “Manually uninstalling on HP-UX platforms” on page 296.

Miscellaneous tips for HP-UX platforms

Table 15. Installation and migration tips

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
HP-UX platforms	HP-UX platforms <ul style="list-style-type: none">• Migrating or coexisting with an existing WebSphere Application Server node that HP-UX does not recognize• Avoiding using Netscape 4.79 on HP-UX 11 in Japanese, to avoid problems in viewing the administrative console• Configuring the converter.properties file to use EUC-JP (Japanese) encoding on HP-UX
All platforms	All platforms
All Linux and UNIX platforms	All Linux and UNIX-based platforms

Mounting a CD drive on HP-UX

This topic describes how to mount a CD drive as the root user on a Linux platform or as the root user on a UNIX-based operating system platform.

Product CDs for WebSphere Application Server products for HP-UX platforms are formatted for the Rock-Ridge file system type, which is an extension to the International Standards Organization ISO-9660 file system type.

HP-UX uses a mechanism called portable file system (PFS) on PA-RISC machines to mount Rock-Ridge CDs. This procedure describes how to use the PFS method.

The following procedure guides you through using the portable file system (PFS) to mount a CD drive.

1. Log on as root.
2. Determine the CD-ROM device name by using the **ioscan** command or **sam**. Run the **/usr/sbin/ioscan -funC disk** command to return a table of disk devices. If you cannot determine the name of the CD-ROM device, run the sam system maintenance interface to find more information about disk devices.
3. Start the pfsd daemon. PFS mounting is not active on HP-UX systems by default. The pfsd daemon must be running to mount a CD-ROM device. To start the daemon, run the following commands:

```
# /usr/sbin/pfs_mountd &
# /usr/sbin/pfsd &
```

4. Insert the product CD-ROM labeled, *Integration Server Disk 1 of 2* into the CD-ROM.
5. Mount the CD-ROM with the following command:

```
# /usr/sbin/pfs_mount -t rrip -o xlat=unix /dev/dsk/c0t0d0 /cdrom
```

The CD-ROM drive is the `/dev/dsk/c0t0d0` device. The `/cdrom` directory is where the CD-ROM drive mounts. The `rrip` parameter forces the CD-ROM drive to mount in the Rock Ridge Interchange Format. The `xlat=unix` parameter translates file names properly for HP-UX.

6. Use the following procedures if you have a problem with mounting the CD-ROM because it contains files with long names. The **mount** command can fail on any CD-ROM that contains files with long file names. Two methods exist for working around the problem and mounting the product CD-ROM:
 - Edit the `pfs_fstab` file.
 - Use `pfs_mount` to read the Rock Ridge system type.

Editing the `pfs_fstab` file

To avoid the mount problem, perform the following steps:

- a. Log in as a user with root authority.
- b. In the `/etc` directory, add the following line to the `pfs_fstab` file:

```
/dev/dsk/c0t0d0 mount_point pfs-rrip ro,hard
```

The `mount_point` variable represents the mount point of the CD-ROM.

- c. Start the `pfs` daemon by entering the following commands (if they are not already running):

```
/usr/sbin/pfs_mountd &
/usr/sbin/pfsd 4 &
```

- d. Insert the CD-ROM in the drive and enter the following commands:

```
mkdir /cdrom
/usr/sbin/pfs_mount /cdrom
```

Where `/cdrom` represents the mount point of the CD-ROM.

Using `pfs_mount` to read the Rock Ridge system type

- a. The standard HP-UX mount procedure does not read the Rock Ridge file system type correctly. You must mount the CD using the **pfs_mount** command. Start `pfs` daemons by entering these commands:

```
pfs_mountd&
pfsd&
```

- b. Mount the CD as shown in the following example, where `device` represents the raw device (for example, `/dev/rdisk/c0t0d0`), and `mntpnt` represents the mount point (for example, `/cdrom`):

```
pfs_mount -x unix device /mntpnt
```

- c. Add the following line to `/etc/pfs_exports`, where `remotemnt` represents the mount point, and `server` represents the name of the HP-UX system where you are installing the product:

```
/remotemnt -access=server
```

- d. Enter the following command:

```
pfs_exportfs -a
```

Run the remaining steps in this procedure on the local HP-UX system:

- e. Log in as root.
- f. Start `pfs` daemons by entering the following commands:

```
pfs_mountd&
pfsd&
```

- g. Enter the following command, where *remote_server* represents your remote system, *remotemnt* represents the CD-ROM drive on the remote HP-UX system, and *localmnt* represents the mount point on your local HP-UX system:

```
pfs_mount -x unix remote_server:/remotemnt /localmnt
```

- h. Change the directory to the package location, where *localmnt* represents the directory mount point.
- ```
cd /localmnt/tas
```

After installation, you can unmount the CD-ROM using the following command:

```
/usr/sbin/pfs_umount /cdrom
```

For more detailed instructions, consult your HP-UX documentation.

## Installing Network Deployment on Linux platforms

This topic describes how to install the WebSphere Application Server Network Deployment product as the root user on a Linux operating system platform.

You cannot install the product correctly as a non-root user. If you back up the product CD-ROM, do so as root. Backup copies made from non-root users do not preserve the correct file attributes and do not work.

If you encounter a problem such as not having enough temporary space or not having the right packages on your system, cancel the installation, make the required changes, and restart the installation.

The installation uses InstallShield for Multiplatforms (ISMP) to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

### Installation programs in WebSphere Application Server products

#### 5.1 +

| Product                                          | Link to installation procedure                                   | Label on product CD                                            |
|--------------------------------------------------|------------------------------------------------------------------|----------------------------------------------------------------|
| (Base) WebSphere Application Server              | Base product installation                                        | <b>Application Server, IBM HTTP Server</b>                     |
| Network Deployment                               | Network Deployment installation (which links to this procedure!) | <b>Deployment Manager</b>                                      |
| WebSphere Business Integration Server Foundation | Integration Server installation                                  | <b>Integration Server Disk 1 and Integration Server Disk 2</b> |

Use the Network Deployment installation image to manage a multimachine environment, where you have installed the base product on different machines and want to manage the Application Servers in a group, or *cell*. If you buy the Network Deployment product, you also get the base product in the package.

WebSphere Business Integration Server Foundation is the V5.1 level of the Enterprise product. WebSphere Business Integration Server Foundation, Version 5.1 extends the base WebSphere Application Server V5.1 product. After migrating the underlying product to V5.1, do not reinstall the V5.0.x Enterprise product. Version 5.0.x of the Enterprise product does not extend Version 5.1 of the base WebSphere Application Server product nor does it extend the V5.1 Network Deployment product.

**5.1 +** If you buy the WebSphere Business Integration Server Foundation product, you also get the Network Deployment product and the base product in the package.

### Creating multiple Application Servers on a single machine

Although you can create multiple servers on a base WebSphere Application Server node, the servers all share one set of configuration files. Changes that you make to one server affect the others. Configuration documents might become corrupted. The **wsinstance** command can create multiple configuration instances. Each instance is a stand-alone Application Server with its own set of configuration files. Or install the Network Deployment product to create and manage multiple base Application Servers.

## Order of installation

**5.1 +** Install the base product before installing the Network Deployment product when installing both products on the same machine. Install the Network Deployment product before the Integration Server product that extends the Network Deployment product. You can install the Integration Server product before the base product. The Integration Server product can install the base product in what is known as an *umbrella* installation but Integration Server cannot install the Network Deployment product. Some features of the base product cannot be installed by Integration Server.

The embedded messaging feature that is included in the default installation requires that you install base before Network Deployment when installing both on the same machine. Otherwise, the order does not matter. See the following installation tip for more information:

Table 16. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131                                                        |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------|
| All platforms      | Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component |

This topic is available in Adobe PDF format on the product CD-ROM and online in an information center. The information center always has the most current information. The information center displays in the language of your machine locale if possible.

The installation guide PDF is updated occasionally. Download a new installation guide from:

**5.1 +** [ftp://ftp.software.ibm.com/software/websphere/appserv/library/wasv51nd\\_gs.pdf](ftp://ftp.software.ibm.com/software/websphere/appserv/library/wasv51nd_gs.pdf)

The Launchpad tool lets you access the product overview, the `readme.html` file, and the installation guide. See “Using the Launchpad to start the installation” on page 173.

You also use the Launchpad during the installation procedure to install the product. The installation program performs the following actions:

- Checks prerequisites automatically
- Looks for a previous WebSphere Application Server installation, to determine whether to display the Migration panel or the Coexistence panel during the installation

A known problem with the Launchpad can prevent it from using Netscape to open the documentation links.

If you use the Mozilla browser, some Launchpad links do not work. The Launchpad attempts to call the Netscape browser in the `/usr/bin/netscape` directory. Try a symbolic link to the Mozilla browser to fix the problem as shown in the following example:

```
ln -sf /usr/bin/mozilla /usr/bin/netscape
```

You can also launch the following links directly from your browser:

### Button Link

#### Product Overview

<http://www.ibm.com/software/webservers/appserv/>



It is possible that your security options might prevent redirection, such as to the <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/appserv> Web site. If so, you can identify the IBM Web site as a trusted site to allow the redirection. For example, on Internet Explorer, add <http://www-306.ibm.com> to the list of Trusted Sites by selecting **Tools > Internet Options > Security (tab) > Trusted Sites**.

### Readme File

*CD-ROM mount point/readme/readme\_en.html*

### Installation Guide

*CD-ROM mount point/docs/InstallGuide\_en.PDF*

The *Installation Guide* is in Adobe PDF format. Viewing this document requires you to have Adobe Acrobat Reader, which is a free download from the Adobe Acrobat Web site.

After using the Launchpad to view product documentation, perform the following procedure. The procedure includes several preliminary steps before actually launching the installer program.

1. Stop all WebSphere Application Server-related Java processes on the machine where you are installing the product.
2. Stop any Web server process such as the IBM HTTP Server, if you are extending the base product.
3. Provide adequate disk space.

**5.1+** The base Application Server requires the following disk space:

#### **390 MB for the /opt/WebSphere/DeploymentManager directory**

The installation root directory includes the base product code.

#### **150 MB for the /tmp directory**

The temporary directory is the working directory for the installation program.

A message about free space occurs when less than 4 MB of free space remains after starting the installation. The InstallShield for MultiPlatforms (ISMP) program displays a message about using the `-is:tempdir` parameter to identify an alternate temporary space directory.

Ignore any `-is:tempdir` message. The `-is:tempdir` parameter is not supported. Cancel the installation, allocate a total of 150 MB of free space in the `/tmp` directory, and start the installation again.

#### **540 MB total requirement without the embedded messaging feature**

The total amount of space required includes the `/tmp` space, which is released after installation. Space requirements for the embedded messaging feature are described after the next few steps that describe setting up required users and user groups for the feature.

The Installation wizard displays required space for individual features on the Feature selection panel. The Installation wizard also warns you if you do not have enough space to install the product.

If you plan to migrate applications and the configuration from a previous version, verify that application objects have available disk space. As a rough guideline, plan for space equal to 110 percent of the size of the application objects:

- **For Version 3.5.x:** The size of application Java archive (JAR) files, Web archive (WAR) files, and servlet files
  - **For Version 4.0.x:** The size of enterprise archive (EAR) files
  - **5.1+ For Version 5.0.x:** The size of EAR files
4. Define the user groups and the user needed for the embedded messaging feature.
    - a. If you have not already done so, create the `mqm` and `mqbrkrs` user groups.
    - b. Create the `mqm` user.
    - c. Add the `mqm` and `root` users to the `mqm` group.
    - d. Add the user `root` to the `mqbrkrs` group.

The recommended user ID for running the JMS server process is root. If you do run the JMS server process under another user ID, add that user ID to the **mqm** and **mqbrkrs** groups. User IDs longer than 12 characters cannot be used for authentication with the embedded WebSphere JMS provider. The mqm user starts the JMS server for general JMS support and the WebSphere embedded broker for WebSphere Application Server topic connections.

- Log out and back in to pick up the secondary user groups, mqm and mqbrkrs, for root. Use the **ssh** command instead of the **telnet** command to log in. Or run the following command after logging on:

```
su -
```

Use the **id -a** command or the **groups** command to see defined groups for root. If mqm and mqbrkrs are not in the list that is returned, you cannot install the embedded messaging feature:

```
[root@wasdoc2 root]# id -a
uid=0(root) gid=0(root)
groups=0(root),1(bin),2(daemon),
3(sys),4(adm),6(disk),10(wheel),500(mqm),501(mqbrkrs)
```

```
[root@wasdoc2 root]# groups
root bin daemon sys adm disk wheel mqm mqbrkrs
```

- Allocate adequate disk space for the embedded messaging feature if you are planning to install the feature. The installation locations for the embedded messaging feature are fixed as shown in the following table, which lists the locations for the base messaging functions and the messaging broker functions for publish/subscribe messaging.

Table 17. Space requirements for the embedded messaging feature on Linux

| Component                           | Base code | Broker code | Base data | Broker data |
|-------------------------------------|-----------|-------------|-----------|-------------|
| <b>Path</b>                         | /opt/mqm  | /opt/wemps  | /var/mqm  | /var/wemps  |
| <b>Server and client subfeature</b> | 40 MB     | 100 MB      | 8 MB      | 5 MB        |
| <b>Client subfeature</b>            | 15 MB     | 15 MB       | 5 MB      | N/A         |

- Prepare to install the embedded messaging feature with WebSphere MQ. The embedded messaging feature is based on the IBM WebSphere MQ product. The feature and the product each provide a Java message service (JMS) function that supports queues for point-to-point messaging and topics for publish and subscribe messaging.

You can install the embedded messaging feature with or without the WebSphere MQ product on the same machine. To support both the embedded messaging feature and the WebSphere MQ product on the same machine, the WebSphere MQ product must be at a certain fix level and must have several of its features installed.

If you already have WebSphere MQ installed, you can configure it as the JMS provider. Otherwise, you can install the embedded messaging feature during the installation or install the WebSphere MQ product or another JMS provider after you install.

Even though you might decide now to install only the embedded messaging feature, you can install the WebSphere MQ product later and use the IBM WebSphere MQ product as the JMS provider instead.

- Verify that you have upgraded to WebSphere MQ 5.3 with the CSD04 update to install embedded messaging on a machine where you already have WebSphere MQ installed.

Determine if your WebSphere MQ 5.3 installation is at the required level by running the **mqver** utility provided by WebSphere MQ.

The required level as indicated by the mqver command:

```
Name: WebSphere MQ
Version: 530.4 CSD04
...
```

9. Verify that you have installed the required WebSphere MQ 5.3 features to install embedded messaging on a machine where you already have WebSphere MQ installed. Verify that you have installed the following features:
  - When installing the embedded messaging server and client feature, the required MQ features are *Server* and *Java messaging*.
  - When installing the embedded messaging client feature, the required MQ feature is *Java messaging*.

If you attempt to install the embedded messaging feature when WebSphere MQ is already installed, the level of WebSphere MQ must be Version 5.3 with the required MQ features. Otherwise, the installation of the embedded messaging feature fails with prerequisite check errors.

10. Create and mount a journalized file system called `/var/mqm` for your messaging working data. Use a partitioning strategy with a separate volume for embedded messaging or WebSphere MQ data to isolate system activity from the potentially high volume of messaging work that can build up in the `/var/mqm` directory.
11. Create separate file systems for log data in the `var/mqm/log` directory and error files in the `var/mqm/errors` directory. Store log files on a different physical volume from the embedded messaging queues, which are in the `var/mqm` directory. This ensures data integrity in the case of a hardware failure. If you are creating separate file systems, allow the following minimum free space:

**30 MB** `/var/mqm`

**20 MB** `/var/mqm/log`

**4 MB** `/var/mqm/errors`

The `/var` file system stores all the security logging information for the system and stores the temporary files for e-mail and printing. Therefore, it is critical that you maintain free space in `/var` for these operations. If you do not create a separate file system for messaging data, and `/var` fills up, all security logging stops on the system until free space is available in `/var`. Also, e-mail and printing do not work without some available free space in `/var`.

You have the same options for creating file systems for the embedded messaging feature as you do for WebSphere MQ. For example, if you cannot install the embedded messaging options in the required file system (for example, if it is too small), you can do one of the following *before* installing the Embedded Messaging options:

- Create and mount a new file system for the installation directory.
- Create a new directory anywhere on your machine, and create a symbolic link from the required installation directory to the new directory. For example:

```
mkdir /bigdisk/mqm
ln -s /bigdisk/mqm /usr/mqm
```

12. Verify that prerequisites and corequisites are at the required release levels. Although the Installation wizard checks for prerequisite operating system patches with the `prereqChecker` application, review the prerequisites on the IBM WebSphere Application Server supported hardware, software, and APIs Web site if you have not already done so.

Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Some operating systems that were not supported at the time that this product was shipped on CD-ROM might now be supported. You might receive a message from the `prereqChecker` program that an operating system is not supported when, in fact, the operating system is supported.

Always consult the IBM WebSphere Application Server supported hardware, software, and APIs Web site to determine whether your operating system is supported when you receive a message from the `prereqChecker` program. The Web site lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system.

After confirming that your operating system is supported and that you have installed all necessary patches, you can click **Next** to continue an installation when you receive an error message from the `prereqChecker` program.

13. Provide necessary prerequisites for Red Hat Enterprise Linux V3.0. A known limitation exists in the prerequisites checker program when examining prerequisite packages on Linux systems.

You must install the following packages on Red Hat Enterprise Linux V3.0 for xSeries platforms:

- `compat-gcc-7.3-2.96.122`
- `compat-libstdc++-7.3-2.96.122`
- `compat-libstdc++-devel-7.3-2.96.122`
- `compat-glibc-7.x-2.2.4.32.5`
- `compat-gcc-c++-7.3-2.96.122`
- `compat-db-4.0.14-5`
- `rpm-build-4.2.1-4.2`

The following package is required for Red Hat Enterprise Linux V3.0 for pSeries platforms:  
`rpm-build-4.2.1-4.2`.

You must install the following packages on Red Hat Enterprise Linux V3.0 for S/390 (z/VM and VM/ESA) platforms:

- `compat-db-4.0.14-5`
- `compat-pwdb-0.62-3`
- `compat-libstdc++-7.2-2.95.3.77`
- `rpm-build-4.2.1-4.2`

You can install a later release of any of these packages.

14. Upgrade Red Hat Enterprise Linux V3.0 to service level 1 (update 1).

Upgrade RHEL 3.0 to service level 1 by downloading and installing the service updates from Red Hat.

If you do not upgrade to service level 1, certain national language issues in the released version of GLIBC included in Red Hat Enterprise Linux V3 might cause Java virtual machine (JVM) failures due to segmentation faults. These failures can occur during installation when the locale is set to anything other than an English locale.

The prerequisites checker program for the base V5.1.0.0 product does not recognize Red Hat Enterprise Linux 3.0 as a supported operating system. If you receive a message from the prerequisites checker program that RHEL 3.0 is an unsupported operating system, ignore the message and continue the installation if you installed all prerequisite packages from the previous step and downloaded and installed the service level 1 update.

Also, do not install, log off, and log back on when you are installing from the operator console attached to the machine. This can produce segmentation faults that cause the installation to fail, as described in `tins_trouble.dita#tins_trouble/192656.RN`.

15. Prepare the SuSE Linux Enterprise Server 8.0 - Powered by UnitedLinux 1.0 operating platform for WebSphere Application Server installation.

- a. Install SP2 for the United Linux 1.0 operating platform to let you use the LaunchPad.

It is your responsibility to install this service pack. The `prereqChecker` function of the installer cannot detect service pack versions definitively on United Linux. Kernel unames and versions between 8.0 and 8.0.2 are identical. No signature RPM denotes a service pack install.

- b. Use the IBM Developer Kit that WebSphere Application Server provides to support the Java 2 SDK on the SuSE SLES 8.0 operating system to avoid potential problems when uninstalling an interim fix or a fix pack. To use the IBM Developer Kit, remove the `java2-jre-1.3.1-524` and `java2-1.3.1-524` RPMs from the machine before installing WebSphere Application Server.

16. Correct font problems on SuSE Linux Enterprise Server 8.0 in Simplified Chinese and Traditional Chinese locales.

On the Linux for Power platform that SuSE Linux Enterprise Server 8.0 provides, a missing package causes a font problem. The `ttf-hanyi` package is not installed during the normal product installation

of the SuSE 8.0 operating system. The missing package causes the Installation wizard for WebSphere Application Server products to display garbled characters in the Simplified Chinese locale and in the Traditional Chinese locale.

Copy the `ttf-hanyi-2021016-0.noarch.rpm` package on the SuSE 8.0 for i386 CD to the Power PC system. Install the package on the Power PC machine and reboot the machine to solve the problem.

17. Remove any entries from the `/usr/bin/jitk.db` file if you have uninstalled WebSphere Application Server Enterprise Edition Version 4.1. Remove any remaining artifacts from an uninstalled V4.1 system to prevent the display of the Coexistence panel or the Migration panel during installation.

The Installation wizard might display the Migration panel or the Coexistence panel even though you have uninstalled WebSphere Application Server Version 4.1. You can prevent the Installation wizard from recognizing a previously deleted V4.1 Application Server by removing the following entry from the `/usr/bin/jitk.db` file:

```
WebSphere Application Server 4.1
```

Remove other Version 4.x entries for WebSphere Application Server products that are no longer on your system.

18. Verify the system `cp` command when using emacs or other freeware.

If you have emacs or other freeware installed on your Linux operating platform, verify that the system `cp` command is being used.

- a. Type `which cp` at the command prompt before running the installation program for the WebSphere Application Server product.
- b. Remove the freeware directory from your `PATH` if the resulting directory output includes freeware. For example, if the output is similar to this `/opt/freeware/bin/cp` message, remove the directory from the `PATH`.
- c. Install the WebSphere Application Server product.
- d. Add the freeware directory back to the `PATH`.

If you install with a `cp` command that is part of a freeware package, the installation might appear to complete successfully, but the Java 2 SDK that the product installs might have missing files in the `install_root/java` directory.

Missing files can destroy required symbolic links. If you remove the freeware `cp` command from the `PATH`, you can install the Application Server product successfully.

Perform the following step to verify that the Java 2 SDK is working correctly.

19. Verify the Java 2 SDK on the WebSphere Application Server CD. If you copied the product CD to back it up and are using a backup version, perform the following steps to verify that the Java 2 SDK on the product CD-ROM is working correctly.
  - a. Change directories to the `/sun/WAS/jdk/java/bin` directory on the product CD-ROM. For example:

```
cd /mnt/linuxi386/WAS/jdk/java/bin
```
  - b. Verify the Java 2 SDK version. Type the following command:

```
./java -version
```

The command completes successfully with no errors when the Java 2 SDK is intact.

20. Select the Installation wizard method or the silent installation method but do not start the installation yet. The installer program has two interfaces, the Installation wizard and a silent command-line installation.

### Performing the installation with the wizard

You can start the Installation wizard using the Launchpad or directly using the `install` command.

The default installation method is to open a command window and issue the command to start the Launchpad tool. Click the **Install the product** option on the Launchpad. (See “Using the Launchpad to start the installation” on page 173.)

This option launches the Installation wizard in the language of your machine locale unless there is no translation for your locale, in which case you receive the English version.

A short delay occurs before the ISMP wizard displays. You do not need to click the Install the product option more than once to cause the wizard to display. The delay is particularly noticeable on x-windows platforms.

You can also start the Installation wizard using the `/mnt/cdrom/linuxi386/install` command, where `/mnt/cdrom` is the mount point for the product CD-ROM and `linuxi386` is the platform directory.

### Performing a silent installation

You can perform a silent installation using the `-options responsefile` parameter with the command method:

```
fully_qualified_CD_pathname/install
 -options fully_qualified_hard_disk_drive_path_name/responsefile
```

```
/mnt/cdrom/linuxi386/install -options /tmp/my_responsefile
```

Start the silent installation with a fully qualified path to the options response file. Otherwise, the Installation wizard starts.

A silent installation causes the installation program to read your responses from the options response file, instead of from the wizard interface. You must customize the *responsefile* before installing silently. See Customizing the Network Deployment options response file.

After customizing the file, you can issue the command to silently install. See “Installing silently” on page 174.

After issuing the command, the following text displays:

```
.....
.InstallShield Wizard
```

```
Initializing InstallShield Wizard...
```

```
Searching for Java(tm) Virtual Machine...
```

The silent installation runs without displaying status to the window:

- Press **Enter** when the installation is complete to return to the command prompt.
- Press **Enter** at any time to return from a background job to the command prompt.

You can change the `-W launchPRTBean.active` option in the response file to display the Registration panel to indicate the completion of a silent installation on a local system with a graphical user interface.

To determine the status of the silent installation, review the installation logs in the `install_root/logs` directory or in the `/tmp` directory. See “Troubleshooting the installation” on page 185 for more information about log files.

Silent installation is particularly useful if you install the product often.

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard.

Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize their options.

### Asynchronous and synchronous command lines

After running the **install** command, the command line returns synchronously. A synchronous **install** command returns the command line after the installation is complete.

You can start the installation asynchronously with the installation process and its children processes all running as background processes. Consult your operating system documentation to learn how to issue asynchronous commands. After running the **install** command, the command line returns immediately.

Do not misinterpret an asynchronous command line to mean that the installation has finished when the command prompt returns. Although the command line returns, either the Installation wizard or a silent installation might still be in progress.

### Installing with a network file system mount

If you must use an NFS mount, see the following installation tip:

Table 18. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131 |
|--------------------|--------------------------------------------------------------------------|
| All platforms      | Platform-specific issues when dealing with network file systems          |

21. Insert the product CD-ROM labeled, *Deployment Manager* into a CD drive. Most Linux systems are configured to automatically mount CD-ROM drives. Mount your CD-ROM drive if necessary:

```
mount /mnt/cdrom
```

If a file explorer window opens, close it.

22. Open a shell window and mount the CD-ROM drive if necessary. Mount the CD drive with the following command:

```
mount /mnt/cdrom
```

If a file explorer window opens, close it.

Use the same shell window throughout the installation procedure. Verify that you are in a read/write directory and not the CD-ROM directory or another read-only directory before you start the installation.

23. Start the installation with the `/mnt/cdrom/linuxi386/launchpad.sh` command, where `mnt/cdrom` is the mount point for the product CD-ROM and `linuxi386` is the platform directory for the xSeries CD-ROM. You can also start the installation directly using the `/mnt/cdrom/linuxi386/install` command. The following examples show how to issue the command:

**Syntax:**

```
fully_qualified_CD_pathname/install
```

**Examples:**

```
/mnt/cdrom/linuxi386/install (For xSeries platforms and clones)
/mnt/cdrom/linuxs390/install (For Linux guests on S/390 VM platforms)
/mnt/cdrom/linuxppc/install (For POWERPC platforms)
```

The readme link in the Launchpad is to the `readme.html` file in the CD root directory. The *Getting Started* document that contains installation information is in the `docs` directory on the CD.

Download the current version of the *Getting Started* document from:

**5.1+** [ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd\\_gs.pdf](ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf)

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize the options.

24. Click **Next** to continue. The license agreement displays.

The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.

25. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist.

As the base WebSphere Application Server product version changes, its prerequisites and corequisites change. Updating your database, Web server, Software Development Kit (SDK), and other software is probably necessary.

The base WebSphere Application Server product simplifies migrating product prerequisites, by providing the option to install a complimentary Web server and SDK on your supported operating system. You can uninstall back-level prerequisites and let the Installation wizard install current versions.

If the wizard finds a previous version of WebSphere Application Server, it prompts you to migrate applications and the configuration from the previous version, or to coexist with it. If it finds more than one previous version, the Installation wizard lists them for you to select which one to migrate.

**Migrating or coexisting with an existing WebSphere Application Server node that Linux does not recognize.**

In some cases, the InstallShield for MultiPlatforms (ISMP) program might not detect a previously installed version of WebSphere Application Server because of a failure to read the registry keys on Linux. You can force the migration and coexistence panel to display, by starting the installation with an option on the `/cdrom/cdrom0/linuxi386/install` command.

For example, use this command:

```
./install -W previousVersionDetectedBean.previousVersionDetected="true"
```

You can also force the appearance of the coexistence panel to change conflicting port number assignments. For example, force the coexistence panel to appear using this command:

```
./install -W coexistenceOptionsBean.showCoexistence="true"
```

On either panel, identify the location of the existing product instance to cause it to be recognized.

26. Choose whether to install additional features or to install the product again, when there is a previous installation of the same level product. You can add features at any time, by running the installation wizard again.

This installation wizard panel appears when the installer program detects a previous installation at the same product level. The panel lets you select whether to add features to the existing installation, or perform a new installation to another directory.

If you intend to install additional features, follow this procedure to avoid component regression problems:

- a. Uninstall any interim fixes.
  - b. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.
  - c. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
  - d. Log off as root and back on.
  - e. Install new features.
  - f. Install the most current fix pack.
  - g. Install the most current cumulative fix.
  - h. Install any interim fixes to bring the node back to its previous fix level.
  - i. Use the administrative console on the Network Deployment node to synchronize all node agents.
27. Choose to migrate applications and the configuration from a previous version, to coexist with another version, or to neither coexist or migrate. Click **Next** to continue. See “Migrating and coexisting” on page 311 for more information.

All WebSphere Application Server products on a single machine share some of the code in the embedded messaging feature, if installed. The required level of the embedded messaging feature for V5.1 (CSD04) is not the same as for V5.0.0 or V5.0.1. The required level of the embedded messaging feature for V5.1 is the same as for V5.0.2.

If you attempt to install V5.1 on a machine where a version of the embedded messaging feature is at a release level earlier than CSD04, the installer program displays the message log in a panel. The message that you see is similar to one of the messages in the following example:

```
MQSeries or WebSphere MQ server at an earlier release than required to support
embedded messaging is already installed on the system.
Unsupported earlier maintenance level of MQSeries or WebSphere MQ detected.
Unsupported earlier release of MQSeries client or WebSphere MQ client detected.
Unsupported maintenance level of MQSeries client or WebSphere MQ client detected.
Software conflict with MQSeries JMS SupportPac MA88 detected.
```

To correct the problem, perform one of the following actions:



- Upgrade the full MQSeries or WebSphere MQ product to WebSphere MQ at the latest level that supports embedded messaging (CSD04).

**5.1+** See Installing WebSphere embedded messaging as the JMS provider for more information.

- Uninstall the existing MQSeries or WebSphere MQ product if MQSeries or WebSphere MQ is not required on this system and reinstall the WebSphere Application Server product. Select the embedded messaging feature.

The MQSeries JMS SupportPac MA88 problem is slightly different. Uninstall the MQSeries JMS SupportPac MA88 and reinstall the WebSphere Application Server product, selecting the embedded messaging feature. The function provided by SupportPac MA88 is included in the embedded messaging feature.

You can upgrade the WebSphere Application Server product to V5.0.2 before migrating it to V5.1 to avoid any problem with an incorrect level of the embedded messaging feature. See “Upgrading a Version 5.0.0 or Version 5.0.1 product to Version 5.0.2” on page 362.

You can also perform the procedure for migrating V5.0.0 or V5.0.1 with embedded messaging to V5.1. See “Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1” on page 338.

To share embedded messaging in a coexistence environment, the node names for each installation must be unique, so that each installation has a message queue manager that is named uniquely. To migrate V5.0.2 to V5.1, the node names must be identical. Therefore, the queue manager names are also identical, if you are migrating from V5.0.2 to V5.1.

To prevent losing the queue manager when you uninstall V5.0.2 (or V5.1), you must create a dummy queue manager before uninstalling one of the WebSphere Application Server versions. A series of migration topics in “Migrating and coexisting” on page 311 describe how to migrate after the installation.

**5.1+** The first rule of migration is to migrate after you install WebSphere Business Integration Server Foundation, if you are planning to install the Integration Server:

| If you are planning to install these products: |                    | Select the migration option while installing this product: |                    |
|------------------------------------------------|--------------------|------------------------------------------------------------|--------------------|
| Network Deployment                             | Integration Server | Network Deployment                                         | Integration Server |
| X                                              |                    | X                                                          |                    |
| X                                              | X                  |                                                            | X                  |

The exception to the rule is to migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before extending either product.

**5.1+** Migrating Integration Server also migrates the product that Integration Server extends.

You can perform a silent migration or configure for coexistence during a silent installation. Refer to “Installing silently” on page 174 for a description of performing a silent installation, including the options that you can specify.

The migration prompt appears only when the Installation wizard detects a previous version. The coexistence prompt appears when the Installation wizard detects any other installation, including another Version 5 installation.

If you choose to coexist, the wizard displays a Port selection panel, where you can specify port assignments that do not conflict with existing ports. For example, you can change the HTTP transport port for coexistence, from 9081 (one more than the default Version 5 port number) to 9085 or higher, to avoid potential conflicts with port numbers that previous versions of WebSphere Application Server commonly use.

Use the **netstat -a** command to display all ports in use.

**5.1+** If you choose neither the migration option nor the coexistence option, you can run Version 5.1.x and the previous version, but not at the same time. Although it is possible that both versions

might coexist without port conflicts, you can ensure that both versions run together by selecting the coexistence option and checking for conflicting port assignments.

The Migration panel lists all previous releases that it can identify. If you highlight a release, the text boxes labeled, "select previous version," show the location of the previous product. Select the product to migrate. If you do not see the previous version that you intend to migrate, click **Select previous version** to enter a location and configuration file name if you are migrating a WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x installation.

The field labeled "Configuration file" is valid only for WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x. For the other versions of WebSphere Application Server that are supported by migration (Version 3.5 Standard Edition, Version 3.5 Extended Edition, and Version 4.0 Advanced Edition), the `admin.config` file provides the host and port values for the administrative server. If you use a file name other than `admin.config`, issue the commands that call the migration tools instead of migrating while installing. Issuing the commands that call the migration tools is described in "Migrating and coexisting" on page 311.

Migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before installing the Integration Server product.

You must start the administrative server of some previous versions so that the Installation wizard can export the configuration from the `admin.config` file.

Although you might select migration at this point in the installation process, the actual migration does not begin until after the Version 5 installation is complete. At that time, if the `WASPreUpgrade` tool fails, the Installation wizard does not call the `WASPostUpgrade` tool to complete the migration, but instead displays the `WASPreUpgrade.log` and `WASPostUpgrade.log` log files for you to diagnose the problem. After fixing the problem, such as starting the administrative server of a previous release, you can start the migration again, as described in "Migrating and coexisting" on page 311.

28. Select features to install and click **Next** to continue. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Choose from these features:

#### **Deployment manager**

Installs the product run time. It provides high performance and scalability across your deployment environment. It includes multiserver administration, server clustering, load balancing and workload management for hosting highly available e-business applications.

#### **Web services**

The UDDI registry and the IBM Web Services Gateway are enterprise applications that you can deploy to:

- A base WebSphere Application Server product node federated within a Network Deployment cell
- A stand-alone base WebSphere Application Server node

The Network Deployment product is not a stand-alone product for running enterprise applications. To deploy UDDI or the gateway, install the base WebSphere Application Server product. Although it is not installed by default, a copy of the base WebSphere Application Server product is packaged with the Network Deployment product.

**5.1 +** See Developing Web services for more information.

#### **UDDI Registry**

Installs a Version 2 compliant universal description, discovery, and identification (UDDI) registry, accessible from the UDDI registry user console application, or from SOAP or EJB interfaces.

**5.1 +** See IBM WebSphere UDDI Registry for more information.

#### **Web Services Gateway**

Includes a gateway between Internet and intranet environments so that clients can invoke Web services safely from outside a firewall. The gateway uses automatic protocol conversion for externalizing Web services.

**5.1+** See Enabling Web services through the IBM Web Services Gateway for more information.

### **Embedded messaging client**

Includes the client necessary for the administration of WebSphere MQ Queues and the mapping of JMS resources into the deployment manager JNDI namespace. It is the same client that you can install as part of the base product embedded messaging feature.

You can run the uninstaller program to remove all installed features.

29. Specify a destination directory and click **Next** to continue. Deleting the default target location and leaving an installation directory field empty stops you from continuing the installation process. The Installation wizard does not proceed when you click **Next**. Enter the required target directory to proceed to the next panel. Non-ASCII special characters are not supported in the name of the installation directory. Spaces are also not supported in the name of the installation directory.

The installer program checks for required space at the beginning of the installation. If you do not have enough space, stop the installation program, free space by deleting unused files and emptying the recycle bin, and restart the installation.

If not enough space is available, cancel the installation, allocate the 150 MB of temporary space that is required, and reinstall. The actual space required depends on the features that you are installing.

If you have problems accessing the administrative console after installation, check the `installAdminConsole.log` file for a failure indication. Clean up the `/tmp` space and reinstall the administrative console using the `wsadmin` scripting facility.

If you must increase the `/tmp` allocation, stop the installation program, increase the allocation, and restart the installation.

30. Specify node information and click **Next**. Specify the node name and host name. Although the wizard inserts the machine name (of the installation platform) as the node name, you can specify any unique name. The node name is an arbitrary WebSphere Application Server-specific name that must be unique within a cell.

The host name is the network name for the physical machine on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote WebSphere Application Server nodes use the host name to connect to and to communicate with this node. Selecting a host name that other machines can reach within your network is extremely important. Do not use the generic `localhost` identifier for this value.

If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. WebSphere Application Server configuration files do not provide domain name resolution for multiple IP addresses on a machine with a single network address.

The value that you specify for the host name is used as the value of the `hostName` property in WebSphere Application Server configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as `xmachine.manhattan.ibm.com`
- The default short DNS host name string, such as `xmachine`
- Numeric IP address, such as `127.1.255.3`

The fully qualified DNS host name has the advantage of being totally unambiguous and also flexible. You have the flexibility of changing the actual IP address for the host system without having to change the WebSphere Application Server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A format disadvantage is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run WebSphere Application Server even when disconnected from the network. Define the short name to `127.0.0.1` (local loopback) in the hosts

file to run disconnected. A format disadvantage is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A format disadvantage is that the numeric IP address is fixed. You must change the setting of the `hostName` property in WebSphere Application Server configuration documents whenever you change the machine IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another format disadvantage is that you cannot use the node if the host is disconnected from the network.

31. Review the summary information and click **Next** to install the product code or **Back** to change your specifications. The Summary panel displays the directory for the embedded messaging feature incorrectly on all Linux and UNIX-based platforms, as `/opt/IBM/WebSphere MQ`. Actual installation locations are `/usr/mqm` on AIX systems, and `/opt/mqm` on Linux and all UNIX-based platforms except AIX. When the installation is complete, the wizard displays the `install_root/logs/mq_install.log` installation log if you selected the embedded messaging feature and errors occur with its installation.
32. Review the `install_root/logs/mq_install.log` installation log if it displays. Click **Next** to continue. The wizard displays the Registration panel.
33. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Installation wizard starts the First Steps tool. See “firststeps command” on page 182.
34. Verify the success of the installer program by examining the Exit summary panel and the `install_root/logs/log.txt` for installation status. ISMP records a success message in the `install_root/logs/log.txt` file: “INSTFIN: The installation is complete.” The log is the only source of status information for a silent installation.

Look for severe errors that the installer records in the `install_root/logs/log.txt` file in the installation root directory to verify that no file system or other unusual errors occurred during installation.

If the `install_root/logs/log.txt` file does not contain a record of any problems but problems exist, verify or troubleshoot the installation, as described in “Troubleshooting the installation” on page 185 and in Installation component troubleshooting tips .

If problems exist, correct them, uninstall the product, as described in “Uninstalling the product” on page 273, log off as root and back on, and reinstall.

35. Click **Finish** to close the Installation wizard.
36. Restrict access to the `/var/mqm/errors` directory and messaging logging files. After installing the embedded messaging feature, you must restrict access to the `/var/mqm` directory and to log files that are needed for embedded messaging, such that only the `mqm` user or members of the `mqm` user group have write access. For detailed information, see Installing WebSphere embedded messaging as the JMS provider and Securing messaging directories and log files .
37. Install the most current fix pack and cumulative fix for the Network Deployment product. See Recommended updates for WebSphere Application Server for information about downloading and installing the upgrades.
38. Tune your system for performance. For best performance on any platform, see Tuning performance parameter index .

### Preparing Linux for S/390 for better performance

Linux for S/390 (which refers to the Linux distributions available from Linux distributors that run on IBM eServer zSeries and S/390 systems in 31-bit mode) provides a configuration technique that affects the installation and run time performance of WebSphere Application Server. The technique configures the environment where the Linux image runs to use swap space efficiently. Some performance guidelines recommend running Linux with the VM/ESA or z/VM swap turned off because of VM/ESA or z/VM virtualization of hardware. Virtualization can produce double-swapping situations where VM/ESA or z/VM swaps storage and Linux also swaps storage, which degrades performance.

Excessive swapping affects the performance of the base WebSphere Application Server, which might require 200 MB when all of the Sample applications are loaded. On a system without swap space configured for use, and with a relatively small amount of memory (such as 256 MB), WebSphere

Application Server might encounter problems obtaining enough free memory to work properly, particularly when competing for resources against other applications and products that run in the Linux environment.

The solution is to disable swapping in Linux, but to enable swapping in VM/ESA or z/VM. You can increase performance by letting VM/ESA or z/VM handle the swapping. Double or triple the specification for physical memory for the Linux image. For example, if the physical memory allocation as seen by the Linux image is 256 MB, disable swap in Linux, enable swap in VM/ESA or z/VM, and increase the physical memory specification as seen by Linux to 512 MB or 768 MB. This amount of memory handles any large spikes in application memory usage that might occur.

You can fine tune the amount of physical memory to allocate to each Linux guest operating system. Size the JVM heap size for the application running in the Application Server, add 90 MB to that amount for the Application Server, 20 MB for Linux, and another 10-20% to handle peak usage. This measurement provides better memory management from a VM/ESA or z/VM perspective.

Avoid socket timeout exceptions (SocketTimeoutExceptions) when running WebSphere Business Integration Server Foundation on a Network Deployment node on a Linux for S/390 guest operating system on z/VM or VM/ESA. The exceptions are an indicator that too many processes are running and that the Linux system is being overloaded. If the deployment manager is under a heavy load, 1 GB of memory is required. In addition, move all base nodes to other Linux for S/390 guests to allow the deployment manager to run by itself on the Linux for S/390 system.

See the Linux on IBM eserver zSeries and S/390: ISP/ASP Solutions IBM Redbook and the Performance Analysis for Java Web sites book for more information.

39. Optional: Create a monitored process for the deployment manager process, as described in “Automatically restarting server processes” on page 176.

You can create monitored processes after the installation is complete.

Processes started by a **startManager.sh** command are not running as monitored processes, regardless of how you have configured them. You must start the deployment manager process with a shell script based on the example `rc.was` file, to have the deployment manager running as a monitored process.

The Installation wizard configures the product. It is not necessary to perform further configuration at this time.

You have now registered and successfully installed WebSphere Application Server Network Deployment and the features that you selected.

## Uninstalling and reinstalling

See “Uninstalling the product” on page 273 for information about uninstalling any WebSphere Application Server product.

**5.1 +** If you uninstalled the V5.1 base product but left the embedded messaging feature installed, and now you want to reinstall the V5.1 product, see “Reinstalling V5.1 after uninstalling and leaving the embedded messaging feature installed” on page 310.

After uninstalling a WebSphere Application Server product, reinstalling into the same directory without first deleting all directory contents results in invalid XML configurations because of the retention of old files. Uninstall manually to delete all of the files so that you can reinstall with a clean system, as described in Manually uninstalling on Linux.

Symptoms that you might experience if you reinstall without uninstalling manually include:

1. The **versionInfo.sh** command states that a product is installed but the product is not installed.
2. Specific directories might be missing in the installation root directory.

If you experience symptoms such as these, uninstall everything manually and reinstall. Reinstallation is successful if you manually uninstall before reinstalling.

## Miscellaneous tips for Linux platforms

Table 19. Installation and migration tips

| Operating platform           | Tip in “Platform-specific tips for installing and migrating” on page 131                                                                                                                                                                                                                     |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Linux platforms              | Linux platforms <ul style="list-style-type: none"> <li>• <b>5.1+</b> Ignoring YAST2 dependency conflicts for embedded messaging packages</li> <li>• Avoiding utility hangs and accessing the deployment manager</li> <li>• Accessing First Steps items on Linux for S/390 systems</li> </ul> |
| All platforms                | All platforms                                                                                                                                                                                                                                                                                |
| All Linux and UNIX platforms | All Linux and UNIX-based platforms                                                                                                                                                                                                                                                           |

## Installing Network Deployment on Solaris platforms

This topic describes how to install the WebSphere Application Server Network Deployment product as the root user on a Solaris operating environment platform.

You cannot install the product correctly as a non-root user. If you back up the product CD-ROM, do so as root. Backup copies made from non-root users do not preserve the correct file attributes and do not work.

If you encounter a problem such as not having enough temporary space or not having the right packages on your system, cancel the installation, make the required changes, and restart the installation.

The installation uses InstallShield for Multiplatforms (ISMP) to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

## Installation programs in WebSphere Application Server products

### 5.1+

| Product                                          | Link to installation procedure                                   | Label on product CD                                            |
|--------------------------------------------------|------------------------------------------------------------------|----------------------------------------------------------------|
| (Base) WebSphere Application Server              | Base product installation                                        | <b>Application Server, IBM HTTP Server</b>                     |
| Network Deployment                               | Network Deployment installation (which links to this procedure!) | <b>Deployment Manager</b>                                      |
| WebSphere Business Integration Server Foundation | Integration Server installation                                  | <b>Integration Server Disk 1 and Integration Server Disk 2</b> |

Use the Network Deployment installation image to manage a multimachine environment, where you have installed the base product on different machines and want to manage the Application Servers in a group, or *cell*. If you buy the Network Deployment product, you also get the base product in the package.

WebSphere Business Integration Server Foundation is the V5.1 level of the Enterprise product. WebSphere Business Integration Server Foundation, Version 5.1 extends the base WebSphere Application Server V5.1 product. After migrating the underlying product to V5.1, do not reinstall the V5.0.x Enterprise product. Version 5.0.x of the Enterprise product does not extend Version 5.1 of the base WebSphere Application Server product nor does it extend the V5.1 Network Deployment product.

**5.1+** If you buy the WebSphere Business Integration Server Foundation product, you also get the Network Deployment product and the base product in the package.

## Creating multiple Application Servers on a single machine

Although you can create multiple servers on a base WebSphere Application Server node, the servers all share one set of configuration files. Changes that you make to one server affect the others. Configuration documents might become corrupted. The **wsinstance** command can create multiple configuration instances. Each instance is a stand-alone Application Server with its own set of configuration files. Or install the Network Deployment product to create and manage multiple base Application Servers.

### Order of installation

**5.1 +** Install the base product before installing the Network Deployment product when installing both products on the same machine. Install the Network Deployment product before the Integration Server product that extends the Network Deployment product. You can install the Integration Server product before the base product. The Integration Server product can install the base product in what is known as an *umbrella* installation but Integration Server cannot install the Network Deployment product. Some features of the base product cannot be installed by Integration Server.

The embedded messaging feature that is included in the default installation requires that you install base before Network Deployment when installing both on the same machine. Otherwise, the order does not matter. See the following installation tip for more information:

Table 20. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131                                                        |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------|
| All platforms      | Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component |

This topic is available in Adobe PDF format on the product CD-ROM and online in an information center. The information center always has the most current information. The information center displays in the language of your machine locale if possible.

The installation guide PDF is updated occasionally. Download a new installation guide from:

**5.1 +** [ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd\\_gs.pdf](ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf)

The Launchpad tool lets you access the product overview, the `readme.html` file, and the installation guide. See “Using the Launchpad to start the installation” on page 173.

You also use the Launchpad during the installation procedure to install the product. The installation program performs the following actions:

- Checks prerequisites automatically
- Looks for a previous WebSphere Application Server installation, to determine whether to display the Migration panel or the Coexistence panel during the installation

A known problem with the Launchpad can prevent it from using Netscape to open the documentation links.

If you use the Mozilla browser, some Launchpad links do not work. The Launchpad attempts to call the Netscape browser in the `/usr/bin/netscape` directory. Try a symbolic link to the Mozilla browser to fix the problem as shown in the following example:

```
ln -sf /usr/bin/mozilla /usr/bin/netscape
```

You can also launch the following links directly from your browser:

### Button Link

## Product Overview

<http://www.ibm.com/software/webservers/appserv/>

It is possible that your security options might prevent redirection, such as to the <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/appserv> Web site. If so, you can identify the IBM Web site as a trusted site to allow the redirection. For example, on Internet Explorer, add <http://www-306.ibm.com> to the list of Trusted Sites by selecting **Tools > Internet Options > Security (tab) > Trusted Sites**.

## Readme File

*CD-ROM mount point/readme/readme\_en.html*

## Installation Guide

*CD-ROM\_mount\_point/docs/InstallGuide\_en.PDF*

The *Installation Guide* is in Adobe PDF format. Viewing this document requires you to have Adobe Acrobat Reader, which is a free download from the Adobe Acrobat Web site.

After using the Launchpad to view product documentation, perform the following procedure. The procedure includes several preliminary steps before actually launching the installer program.

1. Stop all WebSphere Application Server-related Java processes on the machine where you are installing the product.
2. Stop any Web server process such as the IBM HTTP Server, if you are extending the base product.
3. Provide adequate disk space.

**5.1+** The Network Deployment product requires the following disk space:

### **460 MB for the /opt/WebSphere/DeploymentManager directory**

The installation root directory includes the base product code.

### **150 MB for the /tmp directory**

The temporary directory is the working directory for the installation program.

Verify that you have at least 100 MB of free space to avoid problems.

A message about free space occurs when less than 4 MB of free space remains after starting the installation. The InstallShield for MultiPlatforms (ISMP) program displays a message about using the `-is:tempdir` parameter to identify an alternate temporary space directory.

Ignore any `-is:tempdir` message. The `-is:tempdir` parameter is not supported. Cancel the installation, allocate a total of 100 MB of free space in the `/tmp` directory, and start the installation again.

### **610 MB total requirement without the embedded messaging feature**

The total amount of space required includes the `/tmp` space, which is released after installation. Space requirements for the embedded messaging feature are described after the next few steps that describe setting up required users and user groups for the feature.

The Installation wizard displays required space for individual features on the Feature selection panel. The Installation wizard also warns you if you do not have enough space to install the product.

If you plan to migrate applications and the configuration from a previous version, verify that application objects have available disk space. As a rough guideline, plan for space equal to 110 percent of the size of the application objects:

- **For Version 3.5.x:** The size of application Java archive (JAR) files, Web archive (WAR) files, and servlet files
  - **For Version 4.0.x:** The size of enterprise archive (EAR) files
  - **5.1+ For Version 5.0.x:** The size of EAR files
4. Define the user groups and the user needed for the embedded messaging feature.
    - a. If you have not already done so, create the `mqm` and `mqbrkr` user groups.
    - b. Create the `mqm` user.



- c. Add the **mqm** and **root** users to the **mqm** group.
- d. Add the user **root** to the **mqbrkrs** group.

The recommended user ID for running the JMS server process is **root**. If you do run the JMS server process under another user ID, add that user ID to the **mqm** and **mqbrkrs** groups. User IDs longer than 12 characters cannot be used for authentication with the embedded WebSphere JMS provider.

The **mqm** user starts the JMS server for general JMS support and the WebSphere embedded broker for WebSphere Application Server topic connections.

Put the user groups into effect when you next start WebSphere Business Integration Server Foundation. For example, either log off and on again with the process user ID or open a new shell in which to start WebSphere Business Integration Server Foundation.

5. Log off and back in as **root** to pick up the secondary user groups, **mqm** and **mqbrkrs**. Use the **ssh** command instead of the **telnet** command to log in. Or run the following command after logging on:

```
su -
```

Use the **id -a** command or the **groups** command to see defined groups for **root**. If **mqm** and **mqbrkrs** are not in the list that is returned, you cannot install the embedded messaging feature:

```
[root@wasdoc2 root]# id -a
uid=0(root) gid=0(root)
groups=0(root),1(bin),2(daemon),
3(sys),4(adm),6(disk),10(wheel),500(mqm),501(mqbrkrs)
```

```
[root@wasdoc2 root]# groups
root bin daemon sys adm disk wheel mqm mqbrkrs
```

6. Allocate adequate disk space for the embedded messaging feature if you are planning to install the feature. The installation locations for the embedded messaging feature are fixed as shown in the following table, which lists the locations for the base messaging functions and the messaging broker functions for publish/subscribe messaging.

Table 21. Space requirements for the embedded messaging feature on Solaris

| Component         | Base code | Broker code      | Base data |
|-------------------|-----------|------------------|-----------|
| Path              | /usr/ mqm | /usr/ opt/ wemps | /var/ mqm |
| Client subfeature | 15 MB     | 15 MB            | 15 MB     |

7. Prepare to install the embedded messaging feature with WebSphere MQ. The embedded messaging feature is based on the IBM WebSphere MQ product. The feature and the product each provide a Java message service (JMS) function that supports queues for point-to-point messaging and topics for publish and subscribe messaging.

You can install the embedded messaging feature with or without the WebSphere MQ product on the same machine. To support both the embedded messaging feature and the WebSphere MQ product on the same machine, the WebSphere MQ product must be at a certain fix level and must have several of its features installed.

If you already have WebSphere MQ installed, you can configure it as the JMS provider. Otherwise, you can install the embedded messaging feature during the installation or install the WebSphere MQ product or another JMS provider after you install.

Even though you might decide now to install only the embedded messaging feature, you can install the WebSphere MQ product later and use the IBM WebSphere MQ product as the JMS provider instead.

8. **5.1+** Verify that you have upgraded to WebSphere MQ 5.3 with the CSD04 update to install embedded messaging on a machine where you already have WebSphere MQ installed. Determine if your WebSphere MQ 5.3 installation is at the required level by running the **mqver** utility provided by WebSphere MQ.

The required level as indicated by **mqver** is shown below:

Name: WebSphere MQ  
Version: 530.4 CSD04  
...

9. **5.1+** Verify that you have installed the required WebSphere MQ 5.3 features to install embedded messaging on a machine where you already have WebSphere MQ installed. When installing the embedded messaging client feature, the required MQ feature is *Java messaging*.
10. **5.1+** Create and mount a journalized file system called `/var/mqm` for your messaging working data. Use a partition strategy with a separate volume for embedded messaging or WebSphere MQ data to isolate system activity from the potentially high volume of messaging work that can build up in the `/var/mqm` directory.
11. **5.1+** Create separate file systems for log data in the `var/mqm/log` directory and error files in the `var/mqm/errors` directory. Store log files on a different physical volume from the embedded messaging queues, which are in the `var/mqm` directory. This ensures data integrity in the case of a hardware failure. If you are creating separate file systems, allow the following minimum free space:

**30 MB** `/var/mqm`

**20 MB** `/var/mqm/log`

**4 MB** `/var/mqm/errors`

The `/var` file system stores all the security logging information for the system and stores the temporary files for e-mail and printing. Therefore, it is critical that you maintain free space in `/var` for these operations. If you do not create a separate file system for messaging data, and `/var` fills up, all security logging stops on the system until free space is available in `/var`. Also, e-mail and printing do not work without some available free space in `/var`.

You have the same options for creating file systems for the embedded messaging feature as you do for WebSphere MQ. For example, if you cannot install the embedded messaging options in the required file system (for example, if it is too small), you can do one of the following *before* installing the Embedded Messaging options:

- Create and mount a new file system for the installation directory.
- Create a new directory anywhere on your machine, and create a symbolic link from the required installation directory to the new directory. For example:

```
mkdir /bigdisk/mqm
ln -s /bigdisk/mqm /usr/mqm
```

12. Set kernel values to support the embedded messaging feature. Several Solaris kernel values are typically too small for the embedded messaging options. Starting the internal JMS server or client with insufficient kernel resources produces a First Failure Support Technology (FFST) file in the `/var/mqm/errors` directory.

Before installing embedded messaging, review the machine configuration:

```
sysdef -i
```

The kernel values are set in the `/etc/system` file, as shown in the following example.

```
set shmsys:shminfo_shmmax = 4294967295
set shmsys:shminfo_shmseg = 1024
set shmsys:shminfo_shmmni = 1024
set semsys:seminfo_semaem = 16384
set semsys:seminfo_semni = 1024
set semsys:seminfo_semmap = 1026
set semsys:seminfo_semmns = 16384
set semsys:seminfo_semmsl = 100
set semsys:seminfo_semopm = 100
set semsys:seminfo_semmnu = 2048
set semsys:seminfo_semume = 256
set msgsys:msginfo_msgmap = 1026
set msgsys:msginfo_msgmax = 65535
set rlim_fd_cur=1024
```

You can change kernel values by editing the `/etc/system` file then rebooting the operating system. For more information about setting up the Solaris system, see the Solaris System Administration documentation. For example, the *Solaris Tunable Parameters Reference Manual*.

Queue managers are generally independent of each other. Therefore system kernel parameters, for example `shmmni`, `semmni`, `semmns`, and `semmnu` need to allow for the number of queue managers in the system.

13. Verify that prerequisites and corequisites are at the required release levels. Although the Installation wizard checks for prerequisite operating system patches with the `prereqChecker` application, review the prerequisites on the IBM WebSphere Application Server supported hardware, software, and APIs Web site if you have not already done so. Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Some operating systems that were not supported at the time that this product was shipped on CD-ROM might now be supported. You might receive a message from the `prereqChecker` program that an operating system is not supported when, in fact, the operating system is supported.

Always consult the IBM WebSphere Application Server supported hardware, software, and APIs Web site to determine whether your operating system is supported when you receive a message from the `prereqChecker` program. The Web site lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. After confirming that your operating system is supported and that you have installed all necessary patches, you can click **Next** to continue an installation when you receive an error message from the `prereqChecker` program.

14. Verify the system `cp` command when using emacs or other freeware.

If you have emacs or other freeware installed on your HP-UX system, verify that the system `cp` command is being used.

- a. Type `which cp` at the command prompt before running the installation program for the WebSphere Business Integration Server Foundation product.
- b. Remove the freeware directory from your `PATH` if the resulting directory output includes freeware. For example, if the output is similar to this `/opt/freeware/bin/cp` message, remove the directory from the `PATH`.
- c. Install the WebSphere Application Server Network Deployment product.
- d. Add the freeware directory back to the `PATH`.

If you install with a `cp` command that is part of a freeware package, the installation might appear to complete successfully, but the Java 2 SDK that the product installs might have missing files in the `install_root/java` directory.

Missing files can destroy required symbolic links. If you remove the freeware `cp` command from the `PATH`, you can install the Network Deployment product successfully.

Perform the following step to verify that the Java 2 SDK is working correctly.

15. Verify the Java 2 SDK on the WebSphere Application Server Network Deployment CD. If you copied the product CD to back it up and are using a backup version, perform the following steps to verify that the Java 2 SDK on the product CD-ROM is working correctly.

- a. Change directories to the `/sun/WAS/jdk/java/bin` directory on the product CD-ROM.

For example:

```
cd /mnt/sun/WAS/jdk/java/bin
```

- b. Verify the Java 2 SDK version.

Type the following command:

```
./java -version
```

The command completes successfully with no errors when the Java 2 SDK is intact.

16. Select the Installation wizard method or the silent installation method but do not start the installation yet. The installer program has two interfaces, the Installation wizard and a silent command-line installation.

### Performing the installation with the wizard

You can start the Installation wizard using the Launchpad or directly using the **install** command. The default installation method is to open a command window and issue the command to start the Launchpad tool. Click the **Install the product** option on the Launchpad. (See “Using the Launchpad to start the installation” on page 173.)

This option launches the Installation wizard in the language of your machine locale unless there is no translation for your locale, in which case you receive the English version.

A short delay occurs before the ISMP wizard displays. You do not need to click the Install the product option more than once to cause the wizard to display. The delay is particularly noticeable on x-windows platforms.

You can also start the Installation wizard using the `/cdrom/cdrom0/sun/install` command, where `/cdrom/cdrom0` is the mount point for the product CD-ROM and `sun` is the platform directory.

### Performing a silent installation

You can perform a silent installation using the `-options responsefile` parameter with the command method:

```
fully_qualified_CD_pathname/install
 -options fully_qualified_hard_disk_drive_path_name/responsefile
```

```
/cdrom/sun/install -options /tmp/my_responsefile
```

Start the silent installation with a fully qualified path to the options response file. Otherwise, the Installation wizard starts.

A silent installation causes the installation program to read your responses from the options response file, instead of from the wizard interface. You must customize the *responsefile* before installing silently. See Customizing the Network Deployment options response file.

After customizing the file, you can issue the command to silently install. See “Installing silently” on page 174.

After issuing the command, the following text displays:

```
.....
.InstallShield Wizard
```

```
Initializing InstallShield Wizard...
```

```
Searching for Java(tm) Virtual Machine...
```

The silent installation runs without displaying status to the window:

- Press **Enter** when the installation is complete to return to the command prompt.
- Press **Enter** at any time to return from a background job to the command prompt.

You can change the `-W launchPRTBean.active` option in the response file to display the Registration panel to indicate the completion of a silent installation on a local system with a graphical user interface.

To determine the status of the silent installation, review the installation logs in the *install\_root/logs* directory or in the `/tmp` directory. See “Troubleshooting the installation” on page 185 for more information about log files.

Silent installation is particularly useful if you install the product often.

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard.

Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize their options.

### Asynchronous and synchronous command lines

After running the **install** command, the command line returns synchronously. A synchronous **install** command returns the command line after the installation is complete.

You can start the installation asynchronously with the installation process and its children processes all running as background processes. Consult your operating system documentation to learn how to issue asynchronous commands. After running the **install** command, the command line returns immediately.

Do not misinterpret an asynchronous command line to mean that the installation has finished when the command prompt returns. Although the command line returns, either the Installation wizard or a silent installation might still be in progress.

### Installing with a network file system mount

If you must use an NFS mount, see the following installation tip:

Table 22. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131 |
|--------------------|--------------------------------------------------------------------------|
| All platforms      | Platform-specific issues when dealing with network file systems          |

17. If you have uninstalled WebSphere Application Server Enterprise Edition Version 4.1, remove any entries from the `/usr/bin/jitk.db` file. Remove any remaining artifacts from an uninstalled Enterprise Edition V4.1 system to prevent the display of the Coexistence panel or the Migration panel during installation.

The Installation wizard for WebSphere Business Integration Server Foundation might display the Migration panel or the Coexistence panel even though you have uninstalled WebSphere Application Server Enterprise Edition Version 4.1. You can prevent the Installation wizard from recognizing a previously deleted version of Enterprise Edition by removing the following entry from the `/usr/bin/jitk.db` file:

```
WebSphere Enterprise Edition Server 4.1
```

Remove other Version 4.x entries for WebSphere Application Server products that are no longer on your system.

18. Insert the product CD-ROM labeled, *Deployment Manager* into a CD drive to automatically mount the drive.  
If the Solaris machine is configured to auto mount CDs, Solaris automatically mounts the product CD.
19. Open a shell window if Solaris did not automatically mount the CD. Use the same shell window throughout the installation procedure.
20. Start the installation with the `/cdrom/cdrom0/sun/launchpad.sh` command, where `/cdrom/cdrom0` is the mount point for the product CD-ROM and `sun` is the platform directory on the CD. You can also start the installation with the `/cdrom/cdrom0/sun/install` command:

```
/cdrom/cdrom0/sun/install
```

The readme link in the Launchpad is to the `readme.html` file in the CD root directory. The *Getting Started* document that contains installation information is in the `docs` directory on the CD.

#### 5.1+ Download a new installation guide at

[ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd\\_gs.pdf](ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf).

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize the options.

21. Click **Next** to continue. The license agreement displays.  
The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.
22. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist.

As the Network Deployment product version changes, its prerequisites and corequisites change.

The Network Deployment product simplifies migrating product prerequisites, by providing the option to install a complimentary Java 2 SDK on your supported operating system. You can uninstall back-level prerequisites and let the Installation wizard install current versions.

If the wizard finds a previous version of WebSphere Application Server, it prompts you to migrate applications and the configuration from the previous version, or to coexist with it. If it finds more than one previous version, the Installation wizard lists them for you to select which one to migrate. As of V5.1, the Installation wizard also lists WebSphere Application Server - Express as a candidate for automatic migration, when detected.

Use the same cell name when migrating Network Deployment from V5.0.x to V5.1.x. If you use a different cell name, federated nodes cannot successfully migrate to the Network Deployment V5.1.x cell.

You can use a different node name, if you prefer, when migrating Network Deployment from V5.0.x to V5.1.x. However, you must use the administrative console after installation, to change the Secure Sockets Layer (SSL) settings that mention the node name.

**Migrating or coexisting with an existing WebSphere Application Server node that Solaris does not recognize.**

In some cases, the InstallShield for MultiPlatforms (ISMP) program does not detect a previously installed version of WebSphere Application Server because of a failure to read the registry keys on Solaris. You can force the migration and coexistence panel to display, by starting the installation with an option on the **/cdrom/sun/install** command.

For example, use this command:

```
./install -W previousVersionDetectedBean.previousVersionDetected="true"
```

You can also force the appearance of the coexistence panel to change conflicting port number assignments. For example, force the coexistence panel to appear using this command:

```
./install -W coexistenceOptionsBean.showCoexistence="true"
```

On either panel, identify the location of the existing product instance to cause it to be recognized.

Use the same cell name when migrating Network Deployment from V5.0.x to V5.1.x. If you use a different cell name, federated nodes cannot successfully migrate to the Network Deployment V5.1.x cell.

You can use a different node name, if you prefer, when migrating Network Deployment from V5.0.x to V5.1.x. However, you must use the administrative console after installation, to change the Secure Sockets Layer (SSL) settings that mention the node name.

23. Choose whether to install additional features or to install the product again, when there is a previous installation of the same level product. You can add features at any time, by running the installation wizard again.

This installation wizard panel appears when the installer program detects a previous installation at the same product level. The panel lets you select whether to add features to the existing installation, or perform a new installation to another directory.

If you intend to install additional features, follow this procedure to avoid component regression problems:

- a. Uninstall any interim fixes.
- b. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.
- c. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
- d. Log off as root and back on.
- e. Install new features.
- f. Install the most current fix pack.
- g. Install the most current cumulative fix.
- h. Install any interim fixes to bring the node back to its previous fix level.
- i. Use the administrative console on the Network Deployment node to synchronize all node agents.

24. Choose to migrate applications and the configuration from a previous version, to coexist with another version, or to neither coexist or migrate. Click **Next** to continue. See “Migrating and coexisting” on page 311 for more information.

All WebSphere Application Server products on a single machine share some of the code in the embedded messaging feature, if installed. The required level of the embedded messaging feature for V5.1 (CSD04) is not the same as for V5.0.0 or V5.0.1. The required level of the embedded messaging feature for V5.1 is the same as for V5.0.2.

If you attempt to install V5.1 on a machine where a version of the embedded messaging feature is at a release level earlier than CSD04, the installer program displays the message log in a panel. The message that you see is similar to one of the messages in the following example:

```
MQSeries or WebSphere MQ server at an earlier release than required to support
embedded messaging is already installed on the system.
Unsupported earlier maintenance level of MQSeries or WebSphere MQ detected.
Unsupported earlier release of MQSeries client or WebSphere MQ client detected.
Unsupported maintenance level of MQSeries client or WebSphere MQ client detected.
Software conflict with MQSeries JMS SupportPac MA88 detected.
```

To correct the problem, perform one of the following actions:

- Upgrade the full MQSeries or WebSphere MQ product to WebSphere MQ at the latest level that supports embedded messaging (CSD04).

**5.1+** See Installing WebSphere embedded messaging as the JMS provider for more information.

- Uninstall the existing MQSeries or WebSphere MQ product if MQSeries or WebSphere MQ is not required on this system and reinstall the WebSphere Application Server product. Select the embedded messaging feature.

The MQSeries JMS SupportPac MA88 problem is slightly different. Uninstall the MQSeries JMS SupportPac MA88 and reinstall the WebSphere Application Server product, selecting the embedded messaging feature. The function provided by SupportPac MA88 is included in the embedded messaging feature.

You can upgrade the WebSphere Application Server product to V5.0.2 before migrating it to V5.1 to avoid any problem with an incorrect level of the embedded messaging feature. See “Upgrading a Version 5.0.0 or Version 5.0.1 product to Version 5.0.2” on page 362.

You can also perform the procedure for migrating V5.0.0 or V5.0.1 with embedded messaging to V5.1. See “Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1” on page 338.

To share embedded messaging in a coexistence environment, the node names for each installation must be unique, so that each installation has a message queue manager that is named uniquely. To migrate V5.0.2 to V5.1, the node names must be identical. Therefore, the queue manager names are also identical, if you are migrating from V5.0.2 to V5.1.

To prevent losing the queue manager when you uninstall V5.0.2 (or V5.1), you must create a dummy queue manager before uninstalling one of the WebSphere Application Server versions. A series of migration topics in “Migrating and coexisting” on page 311 describe how to migrate after the installation.

**5.1+** The first rule of migration is to migrate after you install WebSphere Business Integration Server Foundation, if you are planning to install the Integration Server:

| If you are planning to install these products: |                    | Select the migration option while installing this product: |                    |
|------------------------------------------------|--------------------|------------------------------------------------------------|--------------------|
| Network Deployment                             | Integration Server | Network Deployment                                         | Integration Server |
| X                                              |                    | X                                                          |                    |
| X                                              | X                  |                                                            | X                  |

The exception to the rule is to migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before extending either product.

**5.1+** Migrating Integration Server also migrates the product that Integration Server extends.

You can perform a silent migration or configure for coexistence during a silent installation. Refer to “Installing silently” on page 174 for a description of performing a silent installation, including the options that you can specify.

The migration prompt appears only when the Installation wizard detects a previous version. The coexistence prompt appears when the Installation wizard detects any other installation, including another Version 5 installation.

If you choose to coexist, the wizard displays a Port selection panel, where you can specify port assignments that do not conflict with existing ports. For example, you can change the HTTP transport port for coexistence, from 9081 (one more than the default Version 5 port number) to 9085 or higher, to avoid potential conflicts with port numbers that previous versions of WebSphere Application Server commonly use.

Use the **netstat -a** command to display all ports in use.

**5.1+** If you choose neither the migration option nor the coexistence option, you can run Version 5.1.x and the previous version, but not at the same time. Although it is possible that both versions might coexist without port conflicts, you can ensure that both versions run together by selecting the coexistence option and checking for conflicting port assignments.

The Migration panel lists all previous releases that it can identify. If you highlight a release, the text boxes labeled, “select previous version,” show the location of the previous product. Select the product to migrate. If you do not see the previous version that you intend to migrate, click **Select previous version** to enter a location and configuration file name if you are migrating a WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x installation.

The field labeled “Configuration file” is valid only for WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x. For the other versions of WebSphere Application Server that are supported by migration (Version 3.5 Standard Edition, Version 3.5 Extended Edition, and Version 4.0 Advanced Edition), the `admin.config` file provides the host and port values for the administrative server. If you use a file name other than `admin.config`, issue the commands that call the migration tools instead of migrating while installing. Issuing the commands that call the migration tools is described in “Migrating and coexisting” on page 311.

Migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before installing the Integration Server product.

You must start the administrative server of some previous versions so that the Installation wizard can export the configuration from the `admin.config` file.

Although you might select migration at this point in the installation process, the actual migration does not begin until after the Version 5 installation is complete. At that time, if the WASPreUpgrade tool fails, the Installation wizard does not call the WASPostUpgrade tool to complete the migration, but instead displays the `WASPreUpgrade.log` and `WASPostUpgrade.log` log files for you to diagnose the problem. After fixing the problem, such as starting the administrative server of a previous release, you can start the migration again, as described in “Migrating and coexisting” on page 311.

25. Select features to install and click **Next** to continue. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Choose from these features:

#### **Deployment manager**

Installs the product run time. It provides high performance and scalability across your deployment environment. It includes multiserver administration, server clustering, load balancing and workload management for hosting highly available e-business applications.

#### **Web services**

The UDDI registry and the IBM Web Services Gateway are enterprise applications that you can deploy to:

- A base WebSphere Application Server product node federated within a Network Deployment cell
- A stand-alone base WebSphere Application Server node



The Network Deployment product is not a stand-alone product for running enterprise applications. To deploy UDDI or the gateway, install the base WebSphere Application Server product. Although it is not installed by default, a copy of the base WebSphere Application Server product is packaged with the Network Deployment product.

**5.1+** See Developing Web services for more information.

#### **UDDI Registry**

Installs a Version 2 compliant universal description, discovery, and identification (UDDI) registry, accessible from the UDDI registry user console application, or from SOAP or EJB interfaces.

**5.1+** See IBM WebSphere UDDI Registry for more information.

#### **Web Services Gateway**

Includes a gateway between Internet and intranet environments so that clients can invoke Web services safely from outside a firewall. The gateway uses automatic protocol conversion for externalizing Web services.

**5.1+** See Enabling Web services through the IBM Web Services Gateway for more information.

#### **Embedded messaging client**

Includes the client necessary for the administration of WebSphere MQ Queues and the mapping of JMS resources into the deployment manager JNDI namespace. It is the same client that you can install as part of the base product embedded messaging feature.

You can run the uninstaller program to remove all installed features.

26. Specify a destination directory and click **Next** to continue. Deleting the default target location and leaving an installation directory field empty stops you from continuing the installation process. The Installation wizard does not proceed when you click **Next**. Enter the required target directory to proceed to the next panel. Non-ASCII special characters are not supported in the name of the installation directory. Spaces are also not supported in the name of the installation directory.  
The installer program checks for required space at the beginning of the installation. If you do not have enough space, stop the installation program, free space by deleting unused files and emptying the recycle bin, and restart the installation.  
If not enough space is available, cancel the installation, allocate the 150 MB of temporary space that is required, and reinstall. The actual space required depends on the features that you are installing.  
If you have problems accessing the administrative console after installation, check the `installAdminConsole.log` file for a failure indication. Clean up the `/tmp` space and reinstall the administrative console using the `wsadmin` scripting facility.  
If you must increase the `/tmp` allocation, stop the installation program, increase the allocation, and restart the installation.
27. Specify node information and click **Next**. Specify the node name and host name. Although the wizard inserts the machine name (of the installation platform) as the node name, you can specify any unique name. The node name is an arbitrary WebSphere Application Server-specific name that must be unique within a cell.  
The host name is the network name for the physical machine on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote WebSphere Application Server nodes use the host name to connect to and to communicate with this node. Selecting a host name that other machines can reach within your network is extremely important. Do not use the generic `localhost` identifier for this value.  
If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. WebSphere Application Server configuration files do not provide domain name resolution for multiple IP addresses on a machine with a single network address.

The value that you specify for the host name is used as the value of the `hostName` property in WebSphere Application Server configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as `xmachine.manhattan.ibm.com`
- The default short DNS host name string, such as `xmachine`
- Numeric IP address, such as `127.1.255.3`

The fully qualified DNS host name has the advantage of being totally unambiguous and also flexible. You have the flexibility of changing the actual IP address for the host system without having to change the WebSphere Application Server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A format disadvantage is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run WebSphere Application Server even when disconnected from the network. Define the short name to `127.0.0.1` (local loopback) in the hosts file to run disconnected. A format disadvantage is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A format disadvantage is that the numeric IP address is fixed. You must change the setting of the `hostName` property in WebSphere Application Server configuration documents whenever you change the machine IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another format disadvantage is that you cannot use the node if the host is disconnected from the network.

28. Review the summary information and click **Next** to install the product code or **Back** to change your specifications. The Summary panel displays the directory for the embedded messaging feature incorrectly on all Linux and UNIX-based platforms, as `/opt/IBM/WebSphere MQ`. Actual installation locations are `/usr/mqm` on AIX systems, and `/opt/mqm` on Linux and all UNIX-based platforms except AIX. When the installation is complete, the wizard displays the `install_root/logs/mq_install.log` installation log if you selected the embedded messaging feature and errors occur with its installation.
29. Review the `mq_install.log` installation log if it displays. Click **Next** to continue. The wizard displays the Registration panel.
30. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Installation wizard starts the First Steps tool. See “firststeps command” on page 182.
31. Verify the success of the installer program by examining the Exit summary panel and the `install_root/logs/log.txt` for installation status. ISMP records a success message in the `install_root/logs/log.txt` file: “INSTFIN: The installation is complete.” The log is the only source of status information for a silent installation.

Look for severe errors that the installer records in the `install_root/logs/log.txt` file in the installation root directory to verify that no file system or other unusual errors occurred during installation.

If the `install_root/logs/log.txt` file does not contain a record of any problems but problems exist, verify or troubleshoot the installation, as described in “Troubleshooting the installation” on page 185 and in Installation component troubleshooting tips .

If problems exist, correct them, uninstall the product, as described in “Uninstalling the product” on page 273, log off as root and back on, and reinstall.

When installing WebSphere Application Server with embedded messaging from the CD-ROM, the following warning messages are issued despite successful installation:

```
MQSERIES.MQM-CL-HPUX: A test command parameter is not valid." and "*"
"hostname:/cdrom/hpux/messaging/images": Cannot open the
logfile on this target or source. Possibly the media is
read-only or there is a permission problem." on mq_install.log.
```

You can ignore the warning messages, because the problem does not affect the successful installation of WebSphere Application Server with embedded messaging.

32. Click **Finish** to close the Installation wizard.
33. Close the terminal window that remains open after the installation finishes. When installing WebSphere Application Server from the product CD onto a Solaris system, the ISMP installation wizard launches a terminal window, which remains open after the installation is complete. This window contains the following text:

```
InstallShield Wizard
Initializing InstallShield Wizard...

Searching for Java(tm) Virtual Machine...
.....
```

Close the window after the installation completes.

34. Restrict access to the `/var/mqm/errors` directory and messaging logging files. After installing the embedded messaging feature, you must restrict access to the `/var/mqm` directory and to log files that are needed for embedded messaging, such that only the `mqm` user or members of the `mqm` user group have write access. For detailed information, see *Installing WebSphere embedded messaging as the JMS provider and Securing messaging directories and log files* .
35. Install the most current fix pack and cumulative fix for the Network Deployment product. See *Recommended updates for WebSphere Application Server* for information about downloading and installing the upgrades.
36. Tune your system for performance. For best performance on any platform, see *Tuning performance parameter index* .
37. Optional: Create a monitored process for one or more Application Server processes or for the HTTP Server, as described in “Automatically restarting server processes” on page 176.

You can create monitored processes after the installation is complete.

Processes started by a **startManager.sh** command are not running as monitored processes, regardless of how you have configured them. You must start the deployment manager process with a shell script based on the example `rc.was` file, to have the deployment manager running as a monitored process.

The Installation wizard configures the product. It is not necessary to perform further configuration at this time.

You have now registered and successfully installed WebSphere Application Server Network Deployment and the features that you selected.

## Uninstalling and reinstalling

See *Uninstalling the product* for information about uninstalling any WebSphere Application Server product.

### **5.1 +** Reinstalling a V5.1 instance on a Solaris platform after uninstalling and leaving the embedded messaging feature installed

**5.1 +** If you uninstall a V5.1 instance but leave the embedded messaging feature because of other WebSphere Application Server instances, clean up the `/var/sadm/pkg` directory before reinstalling V5.1.

**5.1 +** If you uninstall all WebSphere Application Server products on the machine and are now reinstalling, follow the procedure in “Manually uninstalling on Solaris platforms” on page 302 instead of the following procedure.

**5.1+** You must perform this procedure before you can reinstall the product into the same installation root directory on Solaris platforms. Otherwise you must reinstall the product into another directory and you might have a problem reinstalling the embedded messaging feature.

**5.1+** Assume that V5.0.x base exists in /opt/WebSphere/AppServer50 with the embedded messaging server and client feature. You then install V5.1 base to the following directory, /opt/WebSphere/AppServer51, but later you uninstall the V5.1 base product and leave the embedded messaging feature installed. And now you decide to reinstall the V5.1 base product.

1. Make the /var/backupForWAS51Reinstall backup directory.
2. Change directories to /var/sadm/pkg, and type `ls -lt | grep WSBAA51` to find packages that contain the base WebSphere Application Server.  
(The directory for Network Deployment is WSNAA51.)
3. If the WSBAA51 directory exists, change directories to the WSBAA51 directory and read the pkginfo file in the /var/sadm/pkg/WSBAA51 directory.

Find the installation location for V5.1. If the location is the same installation location as the V5.1 product you uninstalled, move the /var/sadm/pkg/WSBAA51 folder to /var/backupForWAS51Reinstall.

The following example shows how to display the pkginfo file:

```
pwd
/var/sadm/pkg/WSBAA51
ls -lt
total 6
-rw-r--r-- 1 root other 591 Oct 27 15:46 pkginfo
drwxr-xr-x 2 root other 512 Oct 27 15:46 install
drwxr-xr-x 2 root other 512 Oct 27 15:46 save
cat pkginfo
VPD created by InstallShield MultiPlatform Edition
Mon Oct 27 15:46:34 EST 2003
ARCH=sparc
BASEDIR=/opt/WebSphere/AppServer51
CATEGORY=application
CLASSES=none
DESC=
ISJE_INSTALL_STATUS=3
ISJE_IS_ACTIVE=true
ISJE_IS_PUBLICLY_SHAREABLE=false
ISJE_NAME=IBM WebSphere Application Server
ISJE_PARENTS=WSBAA51:5.1.0.0.DSP=:1
ISJE_PRODUCT_NUMBER=5.1.0.0
ISJE_REQUIRED_COUNT=0
ISJE_SOURCES=
ISJE_UID=WSBAA51
ISJE_UNINSTALLER="_uninst" "uninstall.jar" "uninstall.dat" ""
ISJE_VPD_VERSION=4.5
ISJE_WEB=
NAME=WebSphere Application Server
PKG=WSBAA51
PKGINST=WSBAA51
VENDOR=
VERSION=5.1.0.0.DSP=
```

When two instances of V5.1 WebSphere Application Server exist, for example, with one installed and one uninstalled, two files exist in the /var/sadm/pkg directory. One is the /var/sadm/pkg/WSBAA51 file; the other is the /var/sadm/pkg/WSBAA51.2 file. For example:

```
ls -lt | grep WSBAA
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBAA51.2
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBAA51
drwxr-xr-x 3 root other 512 Oct 21 13:32 WSBAA
```

4. Verify that you are moving the correct file to the backup directory.

You can do this by examining the `pkginfo` files. Do not touch entries for other WebSphere Application Server products that might be present on the platform, such as WS entries for existing WebSphere Application Server V5.0.x.

- Run the `ls -lt | grep WSBM` command to get all of the embedded messaging packages.

For example:

```
#
ls -lt | grep WSBM
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMQ1AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMQ2AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMS4AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMS6AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMQAA.2
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMS4AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMS6AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQ1AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQ2AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQ3AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQAA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMSAA
```

- Identify entries that are related to the V5.1 product that you uninstalled.

Examine the `pkginfo` file for each entry. In the previous example, “.2” entries relate to the product you uninstalled, which was originally in the `/opt/WebSphere/AppServer51` directory. Use a command to move all the “.2” entries into the backup directory. For example, run the following command from the `/var/sadm/pkg/` directory:

```
mv WSBM*.2 /var/backupForWAS51Reinstall
```

For V5.1 of Network Deployment, all related embedded messaging entries start with `WSNM`. Run the following command to show the Network Deployment entries:

```
ls -lt | grep WSNM
```

For V5.1, all related embedded messaging entries are related to the base product or to the Network Deployment product. No embedded messaging entries start with `WSEM`.

- Rename `/opt/WebSphere/AppServer51` to `/opt/WebSphere/AppServer51_bk` before reinstalling V5.1.

## Miscellaneous tips for Solaris platforms

Table 23. Installation and migration tips

| Operating platform           | Tip in “Platform-specific tips for installing and migrating” on page 131                                                                                                                                                                                                                                                                                                                                                                  |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Solaris platforms            | <ul style="list-style-type: none"> <li>Solaris operating environment</li> <li>Using the <b>unzip</b> function, not the <b>jar</b> command, to decompress downloaded files</li> <li>Configuring the <code>converter.properties</code> file to use EUC-JP (Japanese) encoding</li> <li>Configuring the Domino Server plug-in</li> <li>Avoiding problems starting application servers in <code>zh_TW.EUC</code> locale on Solaris</li> </ul> |
| All platforms                | All platforms                                                                                                                                                                                                                                                                                                                                                                                                                             |
| All Linux and UNIX platforms | All Linux and UNIX-based platforms                                                                                                                                                                                                                                                                                                                                                                                                        |

## Installing Network Deployment on Windows platforms

This topic describes how to install the WebSphere Application Server Network Deployment product from a user ID that belongs to the administrator group.

You cannot install the product correctly from a user ID that does not belong to the administrator group. Some steps of the installation procedure require the administrator group ID to have the following advanced user rights, *Act as part of the operating system* and *Log on as a service*.

If you encounter a problem such as not having enough space or not having the right prerequisite updates on your system, cancel the installation, make the required changes, and restart the installation.

The installation uses InstallShield for Multiplatforms (ISMP) to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

## Installation programs in WebSphere Application Server products

### 5.1 +

| Product                                          | Link to installation procedure                                   | Label on product CD                                            |
|--------------------------------------------------|------------------------------------------------------------------|----------------------------------------------------------------|
| (Base) WebSphere Application Server              | Base product installation                                        | <b>Application Server, IBM HTTP Server</b>                     |
| Network Deployment                               | Network Deployment installation (which links to this procedure!) | <b>Deployment Manager</b>                                      |
| WebSphere Business Integration Server Foundation | Integration Server installation                                  | <b>Integration Server Disk 1 and Integration Server Disk 2</b> |

Use the Network Deployment installation image to manage a multimachine environment, where you have installed the base product on different machines and want to manage the Application Servers in a group, or *cell*. If you buy the Network Deployment product, you also get the base product in the package.

WebSphere Business Integration Server Foundation is the V5.1 level of the Enterprise product. WebSphere Business Integration Server Foundation, Version 5.1 extends the base WebSphere Application Server V5.1 product. After migrating the underlying product to V5.1, do not reinstall the V5.0.x Enterprise product. Version 5.0.x of the Enterprise product does not extend Version 5.1 of the base WebSphere Application Server product nor does it extend the V5.1 Network Deployment product.

**5.1 +** If you buy the WebSphere Business Integration Server Foundation product, you also get the Network Deployment product and the base product in the package.

## Creating multiple Application Servers on a single machine

Although you can create multiple servers on a base WebSphere Application Server node, the servers all share one set of configuration files. Changes that you make to one server affect the others. Configuration documents might become corrupted. The **wsinstance** command can create multiple configuration instances. Each instance is a stand-alone Application Server with its own set of configuration files. Or install the Network Deployment product to create and manage multiple base Application Servers.

## Order of installation

**5.1 +** Install the base product before installing the Network Deployment product when installing both products on the same machine. Install the Network Deployment product before the Integration Server product that extends the Network Deployment product. You can install the Integration Server product before the base product. The Integration Server product can install the base product in what is known as an *umbrella* installation but Integration Server cannot install the Network Deployment product. Some features of the base product cannot be installed by Integration Server.

The embedded messaging feature that is included in the default installation requires that you install base before Network Deployment when installing both on the same machine. Otherwise, the order does not matter. See the following installation tip for more information:

Table 24. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131                                                        |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------|
| All platforms      | Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component |

This topic is available in Adobe PDF format on the product CD-ROM and online in an information center. The information center always has the most current information. The information center displays in the language of your machine locale if possible.

The installation guide PDF is updated occasionally. Download a new installation guide from:

**5.1 +** [ftp://ftp.software.ibm.com/software/websphere/appserv/library/wasv51nd\\_gs.pdf](ftp://ftp.software.ibm.com/software/websphere/appserv/library/wasv51nd_gs.pdf)

The Launchpad tool lets you access the product overview, the readme.html file, and the installation guide. See “Using the Launchpad to start the installation” on page 173.

You also use the Launchpad during the installation procedure to install the product. The installation program performs the following actions:

- Checks prerequisites automatically
- Looks for a previous WebSphere Application Server installation, to determine whether to display the Migration panel or the Coexistence panel during the installation
- Installs the IBM HTTP Server and other features, if you select them

After using the Launchpad to view product documentation, perform the following procedure. The procedure includes several preliminary steps before actually launching the installer program.

1. Stop all WebSphere Application Server-related Java processes on the machine where you are installing the product.
2. Stop any Web server process such as the IBM HTTP Server, if you are extending the base product.
3. **5.1 +** Stop all instances of the process\_spawner.exe program on a Windows system before installing.
4. Provide adequate disk space.

**5.1 +** The Network Deployment product requires the following disk space:

**520 MB for the C:\Program Files\WebSphere\AppServer directory**

The installation root directory includes the base product code.

**100 MB for the C:\temp directory**

The temporary directory is the working directory for the installation program.

Verify that you have at least 100 MB of free space to avoid problems.

A message about free space occurs when less than 4 MB of free space remains on the disk that contains the system temp directory after starting the installation.

**620 MB total requirement without the embedded messaging feature**

Space requirements for the embedded messaging feature are described in the next step.

The Installation wizard displays required space for individual features on the Feature selection panel. The Installation wizard also warns you if you do not have enough space to install the product and the cumulative fixes.

If you plan to migrate applications and the configuration from a previous version, verify that application objects have available disk space. As a rough guideline, plan for space equal to 110 percent of the size of the application objects:

- **For Version 3.5.x:** The size of application Java archive (JAR) files, Web archive (WAR) files, and servlet files
- **For Version 4.0.x:** The size of enterprise archive (EAR) files

- **5.1+** For Version 5.0.x: The size of EAR files
5. Allocate adequate disk space for the embedded messaging feature if you are planning to install the feature. You can specify the file system into which the embedded messaging feature is installed. The following table shows default locations and free space requirements for the base messaging functions and the messaging broker functions for publish/subscribe messaging.

Table 25. Space requirements for the embedded messaging feature on Windows platforms

| Component              | Base messaging                    |
|------------------------|-----------------------------------|
| Installation directory | C:\Program Files\IBM\WebSphere MQ |
| Client subfeature      | 15 MB                             |

If you are using the Installation wizard to install IBM WebSphere Application Server, you can specify an installation location for the embedded messaging client feature. You can also specify an installation location for the embedded messaging feature when installing silently.

6. Prepare to install the embedded messaging feature. The embedded messaging feature is based on the IBM WebSphere MQ product. The feature and the product each provide a Java message service (JMS) function that supports queues for point-to-point messaging and topics for publish and subscribe messaging. You can install the deployment manager with the embedded messaging client feature with or without the WebSphere MQ product on the same machine. To support both the embedded messaging feature and the WebSphere MQ product on the same machine, the WebSphere MQ product must be at a certain fix level and must have several of its features installed.

If you already have WebSphere MQ installed, you can configure it as the JMS provider. Otherwise, you can install the embedded messaging feature during the installation of the base Application Server. You can also install the WebSphere MQ product or another JMS provider after you install the WebSphere Application Server products.

On a machine where you want application servers to run messaging applications that use a JMS provider on another host, install IBM WebSphere Application Server or the deployment manager with the embedded messaging client feature.

Even though you might decide now to install only the embedded messaging feature, you can install the WebSphere MQ product later and use the IBM WebSphere MQ product as the JMS provider instead.

7. **5.1+** Verify that you have upgraded to WebSphere MQ 5.3 with the CSD04 update to install embedded messaging on a machine where you already have WebSphere MQ installed. The WebSphere Business Integration Server Foundation package includes copies of the WebSphere MQ 5.3 and Event Broker installation images, with restricted licensing. Use the MQ CD-ROMs to install the required MQ features or WebSphere MQ 5.3 for use with Integration Server.

Determine if your WebSphere MQ 5.3 installation is at the required level by running the **mqver** utility provided by WebSphere MQ.

The required level as indicated by mqver is shown below:

```
Name: WebSphere MQ
Version: 530.4 CSD04
...
```

8. **5.1+** Verify that you have installed the required WebSphere MQ 5.3 features to install embedded messaging on a machine where you already have WebSphere MQ installed. When installing the embedded messaging client feature, the required MQ feature is *Java messaging*.

If you attempt to install the embedded messaging feature when WebSphere MQ is already installed, the level of WebSphere MQ must be Version 5.3 with the required MQ features. Otherwise, the installation of the embedded messaging feature fails with prerequisite check errors.

9. Verify that prerequisites and corequisites are at the required release levels. Although the Installation wizard checks for prerequisite operating system patches with the prereqChecker application, review the prerequisites on the IBM WebSphere Application Server supported hardware, software, and APIs



Web site if you have not already done so. Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Some operating systems that were not supported at the time that this product was shipped on CD-ROM might now be supported. You might receive a message from the prereqChecker program that an operating system is not supported when, in fact, the operating system is supported.

Always consult the IBM WebSphere Application Server supported hardware, software, and APIs Web site to determine whether your operating system is supported when you receive a message from the prereqChecker program. The Web site lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. After confirming that your operating system is supported and that you have installed all necessary patches, you can click **Next** to continue an installation when you receive an error message from the prereqChecker program.

10. Install the required msvcp60.dll file for Windows 2000 Server platforms. The embedded messaging feature requires a DLL file on Windows 2000 Server platforms. The DLL file is already installed on Windows 2000 Advanced Server and other Windows platforms that are supported. If you see a message similar to the following example, download the msvcp60.dll file:

```
wempsdeletebroker.exe - Unable To Locate DLL
The dynamic link library MSVCP60.dll could
not be found in the specified path...
```

The prereq checker in the installer program does not check for this DLL file on the Windows 2000 Server platform. If you select the Windows 2000 Support Tools during Windows 2000 Server installation, the installation program for Windows 2000 Server installs the DLL file in the C:\Program Files\Support Tools directory. The DLL file is installed during the installation of Windows 2000 Advanced Server in the C:\WINNT\system32 directory.

You can download the DLL file from the Microsoft Web site.

11. Lower the Internet Explorer security level from high to medium to view the Product overviews option from the Launchpad on Windows 2003 machines.

You might see a blank page when trying to view Product overviews from the Launchpad on Windows 2003 machines. Internet Explorer is set with high security level on the internet zone. Use the following procedure to lower the Internet zone security level to medium:

- a. Right-click the Internet Explorer icon on the desktop.
- b. Click **Properties > Security > Internet Zone**.
- c. In the **Security level for this zone** area, move the slider from **High** to **Medium**.
- d. Click **OK** to save the change.

12. Verify the SDK on the product CD. Perform the following steps to verify that the SDK on the product CD-ROM is working correctly.

- a. Change directories to the \win\WAS\jdk\java\bin directory on the product CD-ROM. Type the following command:

```
cd CD_Drive_D:\win\WAS\jdk\java\bin
```

- b. Verify the SDK version. Type the following command:

```
.\java -version
```

The command completes successfully with no errors when the SDK is intact.

13. Select the Installation wizard method or the silent installation method but do not start the installation yet. The installer program has two interfaces, the Installation wizard and a silent command-line installation.

#### **Performing the installation with the wizard**

You can start the Installation wizard using the Launchpad or directly using the **install** command.

The default installation method is to open a command window and issue the command to start the Launchpad tool. Click the **Install the product** option on the Launchpad. (See “Using the Launchpad to start the installation” on page 173.)

This option launches the Installation wizard in the language of your machine locale unless there is no translation for your locale, in which case you receive the English version. A short delay occurs before the ISMP wizard displays. You do not need to click the Install the product option more than once to cause the wizard to display.

You can also start the Installation wizard using the **C:\> D:\win\install.bat** command, where the D: drive is your CD-ROM drive and win is the platform directory.

The following example shows a fully qualified call to the installer program:

```
> fully_qualified_CD_pathname\install.bat
```

```
C:\> D:\win\install
```

### Performing a silent installation

You can also perform a silent installation using the `-options C:\fully_qualified_path\responsefile` parameter with the command method:

```
C:\> fully_qualified_CD_pathname\install
 -options fully_qualified_HDD_pathname\responsefile
```

```
C:\> D:\win\install -options C:\temp\my_responsefile
```

Start the silent installation with a fully qualified path to the options response file. Otherwise, the Installation wizard starts.

A silent installation causes the installation program to read your responses from the options response file, instead of from the wizard interface. You must customize the *responsefile* before installing silently. See Customizing the Network Deployment options response file.

After customizing the file, you can issue the command to silently install. See “Installing silently” on page 174.

After issuing the command, the following text displays:

```
.....
.InstallShield Wizard
```

```
Initializing InstallShield Wizard...
```

```
Searching for Java(tm) Virtual Machine...
```

The silent installation runs without displaying status to the window:

- Press **Enter** when the installation is complete to return to the command prompt.
- Press **Enter** at any time to return from a background job to the command prompt.

You can change the `-W launchPRTBean.active` option in the response file to display the Registration panel to indicate the completion of a silent installation on a local system with a graphical user interface.

To determine the status of the silent installation, review the installation logs in the *install\_root\logs* directory or in the system `%TEMP%` directory. See “Troubleshooting the installation” on page 185 for more information about log files.

Silent installation is particularly useful if you install the product often.

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard.

Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize their options.

### Installing with a network file system mount

If you must use an NFS mount, see the following installation tip:

Table 26. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131 |
|--------------------|--------------------------------------------------------------------------|
| All platforms      | Platform-specific issues when dealing with network file systems          |

14. Insert the product CD-ROM labeled, *Deployment Manager* into the CD-ROM drive.
15. Open a DOS window. Use the same DOS window throughout the installation procedure. Verify that you are in a read/write directory and not the CD-ROM directory or another read-only directory before you start the installation.
16. Start the installation with the **D:\win\launchpad.bat** command, where the D: drive is your CD-ROM drive and win is the platform directory on the CD-ROM. You can also start the installation from a read/write directory using the **D:\win\install.bat** command:

```
C:\> D:\win\install
```

The readme link in the Launchpad is to the readme.html file in the CD root directory. The *Getting Started* document that contains installation information is in the docs directory on the CD.

Download the current version of the *Getting Started* document from:

**5.1+** [ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd\\_gs.pdf](ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51nd_gs.pdf)

The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Review the description of the “responsefile” on page 175 for more information. Comments in the file describe how to customize the options.

17. Click **Next** to continue. The license agreement displays.  
The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.
18. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist.

As the WebSphere Application Server Network Deployment product version changes, its prerequisites and corequisites change.

The Network Deployment product simplifies migrating product prerequisites, by providing the option to install a complimentary Java 2 SDK on your supported operating system. You can uninstall back-level prerequisites and let the installation wizard install current versions.

If the wizard finds a previous version of WebSphere Application Server, it prompts you to migrate applications and the configuration from the previous version, or to coexist with it. If it finds more than one previous version, the installation wizard lists them for you to select which one to migrate.

Use the same cell name when migrating Network Deployment from V5.0.x to V5.1.x. If you use a different cell name, federated nodes cannot successfully migrate to the Network Deployment V5.1.x cell.

You can use a different node name, if you prefer, when migrating Network Deployment from V5.0.x to V5.1.x. However, you must use the administrative console after installation, to change the secure sockets layer (SSL) settings that mention the node name.

19. Choose whether to install additional features or to install the product again, when there is a previous installation of the same level product. You can add features at any time, by running the installation wizard again. Reboot the Windows platform after installing and before installing again.  
This installation wizard panel appears when the installer program detects a previous installation at the same product level. The panel lets you select whether to add features to the existing installation, or perform a new installation to another directory.  
If you intend to install additional features, follow this procedure to avoid component regression problems:
  - a. Uninstall any interim fixes.
  - b. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.
  - c. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
  - d. Reboot the Windows system.

- e. Install new features.
  - f. Install the most current fix pack.
  - g. Install the most current cumulative fix.
  - h. Install any interim fixes to bring the node back to its previous fix level.
  - i. Use the administrative console on the Network Deployment node to synchronize all node agents.
20. Choose to migrate applications and the configuration from a previous version, to coexist with another version, or to neither coexist or migrate. Click **Next** to continue.

All WebSphere Application Server products on a single machine share some of the code in the embedded messaging feature, if installed. The required level of the embedded messaging feature for V5.1 (CSD04) is not the same as for V5.0.0 or V5.0.1. The required level of the embedded messaging feature for V5.1 is the same as for V5.0.2.

If you attempt to install V5.0.2 V5.1 on a machine where a version of the embedded messaging feature is at a release level earlier than CSD04, the installer program displays the message log in a panel. The message that you see is similar to one of the messages in the following example:

```
MQSeries or WebSphere MQ server at an earlier release than required to support
embedded messaging is already installed on the system.
```

```
Unsupported earlier maintenance level of MQSeries or WebSphere MQ detected.
```

```
Unsupported earlier release of MQSeries client or WebSphere MQ client detected.
```

```
Unsupported maintenance level of MQSeries client or WebSphere MQ client detected.
```

```
Software conflict with MQSeries JMS SupportPac MA88 detected.
```

To correct the problem, perform one of the following actions:

- Upgrade the full MQSeries or WebSphere MQ product to WebSphere MQ at the latest level that supports embedded messaging (CSD04).

**5.1+** See Installing WebSphere embedded messaging as the JMS provider for more information.

- Uninstall the existing MQSeries or WebSphere MQ product if MQSeries or WebSphere MQ is not required on this system and reinstall the WebSphere Application Server product. Select the embedded messaging feature.

The MQSeries JMS SupportPac MA88 problem is slightly different. Uninstall the MQSeries JMS SupportPac MA88 and reinstall the WebSphere Application Server product, selecting the embedded messaging feature. The function provided by SupportPac MA88 is included in the embedded messaging feature.

After installing the product, reboot the system before installing again.

You can upgrade the WebSphere Application Server product to V5.0.2 before migrating it to V5.1 to avoid any problem with an incorrect level of the embedded messaging feature. See “Upgrading a Version 5.0.0 or Version 5.0.1 product to Version 5.0.2” on page 362.

You can also perform the procedure for migrating V5.0.0 or V5.0.1 with embedded messaging to V5.1. See “Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1” on page 338.

To share embedded messaging in a coexistence environment, the node names for each installation must be unique, so that each installation has a message queue manager that is named uniquely. To migrate V5.0.2 to V5.1, the node names must be identical. Therefore, the queue manager names are also identical, if you are migrating from V5.0.2 to V5.1. To prevent losing the queue manager when you uninstall V5.0.2 (or V5.1), you must create a dummy queue manager before uninstalling one of the WebSphere Application Server versions.

**5.1+** The first rule of migration is to migrate when you install WebSphere Business Integration Server Foundation, if you are planning to install the Integration Server:

## 5.1+

| If you are planning to install these products: |                    | Select the migration option while installing this product: |                    |
|------------------------------------------------|--------------------|------------------------------------------------------------|--------------------|
| Network Deployment                             | Integration Server | Network Deployment                                         | Integration Server |
| X                                              |                    | X                                                          |                    |
| X                                              | X                  |                                                            | X                  |

The exception to the rule is to migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before installing the Integration Server product.

**5.1+** Migrating Integration Server also migrates the product that Integration Server extends.

You can also perform a silent migration or configure for coexistence during a silent installation. Refer to “Installing silently” on page 174 for a description of performing a silent installation, including the options that you can specify.

The migration prompt appears only when the Installation wizard detects a previous version. The coexistence prompt appears when the Installation wizard detects any other installation, including another Version 5 installation.

If you choose to coexist, the wizard displays a Port selection panel, where you can specify port assignments that do not conflict with existing ports. For example, you can change the HTTP transport port for coexistence, from 9081 (one more than the default Version 5 port number) to 9085 or higher, to avoid potential conflicts with port numbers that previous versions of WebSphere Application Server commonly use.

Use the **netstat -a** command to display all ports in use.

In some cases, such as when installing a non-English version, the Installation wizard might not detect a previous version. You can force the Migration panel or the Coexistence panel to appear, by starting the installation with an option on the **D:\win\install.bat** command, where the D: drive is your CD-ROM drive and win is the platform directory on the CD-ROM.

For example, use this command:

```
D:\win\install
-W showPreviousVersionDetectedPanel.active="true"
```

You can also force the appearance of the Coexistence panel to change conflicting port number assignments using this command:

```
D:\win\install
-W showCoexistencePanel.active="true"
```

**5.1+** If you choose neither the migration option nor the coexistence option, you can run Version 5.1.x and the previous version, but not at the same time. Although it is possible that both versions might coexist without port conflicts, you can ensure that both versions run together by selecting the coexistence option and checking for conflicting port assignments.

The Migration panel lists all previous releases that it can identify. If you highlight a release, the text boxes labeled, “Select previous version,” show the location of the previous product. Select the product to migrate. If you do not see the previous version that you intend to migrate, click **Select previous version** to enter a location and configuration file name if you are migrating a WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x installation.

The field labeled “Configuration file” is valid only for WebSphere Application Server Advanced Edition Single Server Edition, Version 4.0.x. For the other versions of WebSphere Application Server that are supported by migration (Version 3.5 Standard Edition, Version 3.5 Extended Edition, and Version 4.0 Advanced Edition), the `admin.config` file provides the host and port values for the administrative server. If you use a file name other than `admin.config`, issue the commands that call the migration tools instead of migrating while installing. Issuing the commands that call the migration tools is described in “Migrating and coexisting” on page 311.

Migrate V3.5.x to V5.1 during the installation of the base product or the Network Deployment product, before installing the Integration Server product.

You must start the administrative server of some previous versions so that the Installation wizard can export the configuration from the `admin.config` file.

Although you might select migration at this point in the installation process, the actual migration does not begin until after the Version 5 installation is complete. At that time, if the WASPreUpgrade tool fails, the Installation wizard does not call the WASPostUpgrade tool to complete the migration, but instead displays the `WASPreUpgrade.log` and `WASPostUpgrade.log` log files for you to diagnose the problem. After fixing the problem, such as starting the administrative server of a previous release, you can start the migration again, as described in “Migrating and coexisting” on page 311.

21. Select features to install and click **Next** to continue. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Choose from these features:

#### **Deployment manager**

Installs the product run time. It provides high performance and scalability across your deployment environment. It includes multiserver administration, server clustering, load balancing and workload management for hosting highly available e-business applications.

#### **Web services**

The UDDI registry and the IBM Web Services Gateway are enterprise applications that you can deploy to:

- A base WebSphere Application Server product node federated within a Network Deployment cell
- A stand-alone base WebSphere Application Server node

The Network Deployment product is not a stand-alone product for running enterprise applications. To deploy UDDI or the gateway, install the base WebSphere Application Server product. Although it is not installed by default, a copy of the base WebSphere Application Server product is packaged with the Network Deployment product.

**5.1+** See Developing Web services for more information.

#### **UDDI Registry**

Installs a Version 2 compliant universal description, discovery, and identification (UDDI) registry, accessible from the UDDI registry user console application, or from SOAP or EJB interfaces.

**5.1+** See IBM WebSphere UDDI Registry for more information.

#### **Web Services Gateway**

Includes a gateway between Internet and intranet environments so that clients can invoke Web services safely from outside a firewall. The gateway uses automatic protocol conversion for externalizing Web services.

**5.1+** See Enabling Web services through the IBM Web Services Gateway for more information.

#### **Embedded messaging client**

Includes the client necessary for the administration of WebSphere MQ Queues and the mapping of JMS resources into the deployment manager JNDI namespace. It is the same client that you can install as part of the base product embedded messaging feature.

You can run the uninstall program to remove all installed features.

22. Specify a destination directory. Click **Next** to continue.

Deleting the default target location and leaving an installation directory field empty stops you from continuing the installation process. The Installation wizard does not proceed when you click **Next**. Enter the required target directory to proceed to the next panel. Non-ASCII special characters are not supported in the name of the installation directory. Spaces are also not supported in the name of the installation directory.

The installer program checks for required space at the beginning of the installation. If you do not have enough space, stop the installation program, free space by deleting unused files and emptying the recycle bin, and restart the installation.

If you have problems accessing the administrative console after installation, check the `installAdminConsole.log` file for a failure indication. Clean up the system `%TEMP%` directory and reinstall the administrative console using the `wsadmin` scripting facility.

23. Specify node information and click **Next**.

Specify the node name, cell name, and host name or internet protocol (IP) address for the deployment manager. Although the wizard inserts the machine name (of the installation platform) as the node name, you can specify any unique name. The node name is an arbitrary WebSphere Application Server-specific name that must be unique within a cell.

The host name is the network name for the physical machine on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote WebSphere Application Server nodes use the host name to connect to and to communicate with this node. Selecting a host name that other machines can reach within your network is extremely important. Do not use the generic `localhost` identifier for this value.

If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. WebSphere Application Server configuration files do not provide domain name resolution for multiple IP addresses on a machine with a single network address.

The value that you specify for the host name is used as the value of the `hostName` property in WebSphere Application Server configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as `xmachine.manhattan.ibm.com`
- The default short DNS host name string, such as `xmachine`
- Numeric IP address, such as `127.1.255.3`

The fully qualified DNS host name has the advantage of being totally unambiguous and also flexible. You have the flexibility of changing the actual IP address for the host system without having to change the WebSphere Application Server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A format disadvantage is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run WebSphere Application Server even when disconnected from the network. Define the short name to `127.0.0.1` (local loopback) in the hosts file to run disconnected. A format disadvantage is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A format disadvantage is that the numeric IP address is fixed. You must change the setting of the `hostName` property in all WebSphere Application Server configuration documents whenever you change the machine IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another format disadvantage is that you cannot use the node if the host is disconnected from the network.

24. Create a Windows service and click **Next**. Run WebSphere Application Server Network Deployment as a service on Windows systems only by clicking the check box. Clicking the check box on this panel configures a manually started service. Processes started by a **startManager** command are not running as monitored processes, regardless of how you have configured them.

For example, you can configure a deployment manager as a WebSphere Application Server Windows service. However, if you start the deployment manager using the **startManager** command, Windows does not monitor or restart the deployment manager because it was not started as a Windows service.

To perform this installation task, the user ID must not have spaces in its name. The user ID must also belong to the administrator group and must have the advanced user rights *Act as part of the operating system* and *Log on as a service*. The Installation wizard grants the user ID the advanced user rights if it does not already have them, if the user ID belongs to the administrator group.

You can also create other Windows services after the installation is complete, to start other server processes.

See “Automatically restarting server processes” on page 176.

25. Review the summary information and click **Next** to install the product code or **Back** to change your specifications. When the installation is complete, the wizard displays the `install_root\logs\mq_install.log` installation log if you selected the embedded messaging client feature and errors occur during its installation.
26. Review the `mq_install.log` installation log if it appears. Click **Next** to continue. The wizard displays the Registration panel.
27. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Installation wizard starts the First Steps tool. See “firststeps command” on page 182.
28. Examine the `install_root\logs\log.txt` file for errors. Look for severe errors that the installer records in the `install_root\logs\log.txt` file in the installation root directory. ISMP also records a success message at the completion of product installation.

Examine the `install_root\logs\log.txt` file to verify that no file system or other unusual errors occurred during installation.

If the `install_root\logs\log.txt` file does not contain a record of any problems but problems exist, verify or troubleshoot the installation as described in “Troubleshooting the installation” on page 185 and Installation component troubleshooting tips .

Examine the Start menu entry for **Programs > IBM WebSphere > Application Server 5.1 > WAS Help**. The WAS Help entry is new for Version 5.1.1. For example, the following graphic shows the Start menu for an English locale:

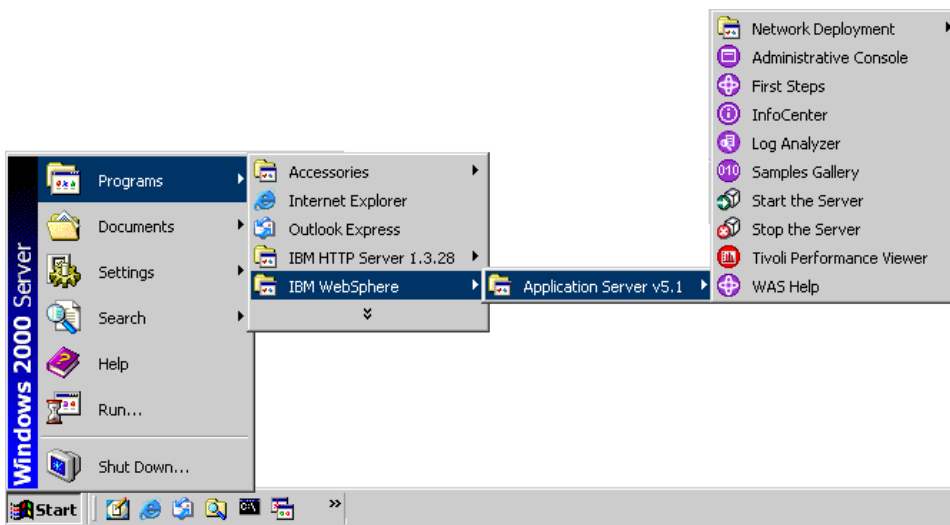


Figure 2. Start menu options for Version 5.1.1

If problems exist, correct them, uninstall the product, reboot the system, and reinstall. See “Uninstalling the product” on page 273.

29. Click **Finish** to close the Installation wizard.



30. Add the CORBA local host property to Network Deployment on Windows 2003 platforms.
  - a. Open a command window.
  - b. Change directories to the `install_root\java` directory.
  - c. Issue the following command:
 

```
attrib -r /S *
```
  - d. Edit the `install_root\java\jre\lib\orb.properties` file.
  - e. Add the following line to the file:
 

```
com.ibm.CORBA.LocalHost=hostname
```

The `hostname` parameter is your local host name. Do not use the generic `localhost` identifier for this value.

31. Install the most current fix pack and cumulative fix for the Network Deployment product. See Recommended updates for WebSphere Application Server for information about downloading and installing the upgrades.
32. Tune your system for performance. For best performance on any platform, see Tuning performance parameter index

The Installation wizard configures the product. It is not necessary to perform further configuration at this time.

You have now registered and successfully installed WebSphere Application Server Network Deployment and the features that you selected.

### Uninstalling and reinstalling:

See “Uninstalling the product” on page 273 for information about uninstalling any WebSphere Application Server product.

**5.1 +** If you uninstalled the product but left the embedded messaging feature installed, and now you want to reinstall the V5.1 product, see “Reinstalling V5.1 after uninstalling and leaving the embedded messaging feature installed” on page 310.

After uninstalling a WebSphere Application Server product, reinstalling into the same directory without first deleting all directory contents results in invalid XML configurations because of the retention of old files. To delete all of the files so that you can reinstall with a clean system, uninstall manually, as described in “Uninstalling the product” on page 273.

### Miscellaneous tips for Windows platforms

Table 27. Installation and migration tips

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Windows platforms  | Windows platforms <ul style="list-style-type: none"> <li>• Viewing accurate migration messages</li> <li>• Decompressing a WebSphere Application Server elmage that you download</li> <li>• Editing or removing the <code>vpd.properties</code> file</li> <li>• Preparing for Adaptive Fast Path Architecture (AFPA) driver availability when migrating from IBM HTTP Server Version 1.3.19.x, or earlier</li> <li>• Ignoring an incorrect ImagePath entry in the registry after changing the document root</li> <li>• Avoiding a user ID with spaces when installing the WebSphere Application Server as a Windows service</li> <li>• Changing the display settings from high-contrast black to high-contrast white to view panels with hypertext links</li> </ul> |

Table 27. Installation and migration tips (continued)

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131 |
|--------------------|--------------------------------------------------------------------------|
| All platforms      | All platforms                                                            |

## Installing additional features on a base Application Server

This topic describes how to install additional features to an existing base WebSphere Application Server product.

If you encounter a problem such as not having enough temporary space or not having the right packages on your system, cancel the installation, make the required changes, and restart the installation.

The installation of new features uses the InstallShield for Multiplatforms (ISMP) wizard to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

### Installation programs in WebSphere Application Server products

#### 5.1 +

| Product                                          | Link to installation procedure                             | Label on product CD                                            |
|--------------------------------------------------|------------------------------------------------------------|----------------------------------------------------------------|
| (Base) WebSphere Application Server              | Base product installation (which links to this procedure!) | <b>Application Server, IBM HTTP Server</b>                     |
| Network Deployment                               | Network Deployment installation                            | <b>Deployment Manager</b>                                      |
| WebSphere Business Integration Server Foundation | Integration Server installation                            | <b>Integration Server Disk 1 and Integration Server Disk 2</b> |

You can add features to an unfederated base node or to a deployment manager node at any time, by running the Installation wizard again. Use the following procedure to avoid component regression problems when installing additional features.

If the node is a base product node in a deployment manager cell, it is important that you understand that there is no way to add features to a base node without unfederating it from its cell and losing its configuration. The configuration for the base node is restored back to its original configuration, before it was federated and before you changed it with the administrative console of the deployment manager.

If the node is a base product node in a deployment manager cell, restoring an original configuration means that you lose the configuration for any additional servers that you created on the base node from the deployment manager. (The additional servers will not exist any more.) You will lose any other configuration you performed from the deployment manager administrative console.

1. Log on as the root user on a Linux or UNIX system or as a user in the administrator group on a Windows system.

You cannot install features correctly as a non-root user on a Linux or UNIX system, or as a non-administrator on a Windows system.

2. Use the **backupConfig** command to save the configuration.

The **backupConfig** command works well in an unfederated environment, to save the configuration of the base node. Issue the command from the *install\_root/bin* directory of the base node.

Although you can use the **backupConfig** command to save the configuration on a federated node, you cannot restore it with the **restoreConfig** command because the configuration you restore does not include any features that you add to the base node.

You can use the **backupConfig** command to save the Network Deployment configuration when adding features to a federated node. Then you can restore it if necessary. You can also refer to the configuration as you use the administrative console of the deployment manager to reconfigure the base node. Issue the command from the *install\_root/bin* directory of the deployment manager node. For example, use the following command on a Linux system:

```
#!/backupConfig.sh /tmp/my_backup_2004-06-03.zip -user user_ID -password password
```

See backupconfig command for more information.

3. If the node is a base product node in a deployment manager cell, unfederate the node with the **removeNode** command. Removing the base node from the cell also removes the configuration for the base node, which resides at the cell level. Issue the removeNode command from the *install\_root/bin* directory of the base Application Server. For example, on a Linux server, issue the following command:

```
#!/removeNode.sh -user user_ID -password password
```

See removeNode command for more information.

4. Uninstall any interim fixes, starting with the last one and finishing with the first one. See “Uninstalling interim fixes, cumulative fixes, and fix packs” on page 239 for more information about how to remove an interim fix, a cumulative fix, or a fix pack.
5. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.
6. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
7. Log off and back on or reboot a Windows system.
8. Insert the product CD labeled, *Application Server, IBM HTTP Server*. Windows systems and most Linux systems are configured to automatically mount CD-ROM drives.
9. Open a shell window and mount the CD-ROM drive if necessary.
10. Start the installation with the **./launchpad.sh** command or the **launchpad** command on a Windows system. See the platform-specific installation instructions if you need more information about restarting the installation from the product CD-ROM, such as instructions for mounting the CD drive. The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Comments in the response file describe how to customize the options. Review the description of the “responsefile” on page 175 for more information.
11. Click **Next** to continue. The license agreement displays. The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.
12. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist. If the wizard finds a previous version of WebSphere Application Server, it prompts you to migrate applications and the configuration from the previous version, or to coexist with it. If it finds more than one previous version, the Installation wizard lists them for you to select which one to migrate.
13. Choose to install additional features. The Installation wizard panel that appears when the installer program detects a previous installation at the same product level has an option for installing additional features. Select the check box labeled, Add features to the existing copy. Select the check box labeled, Add features to the existing copy. If you did not unfederate a managed node, you see a message that the installation wizard is unable to add components to a federated node. Exit the installation, start from the beginning of this procedure, and unfederate the node this time.
14. Skip the coexistence panel that lets you select new ports, if it appears.

15. Select features to install and click **Next** to continue when performing a custom installation. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Previously installed features are checked and grayed out with the term **(Installed)** at the end of the feature name. The IBM HTTP Server feature is an exception. You can install more than one instance of the IBM HTTP Server product. It is not grayed out or labeled **(Installed)**. You can install the plug-in feature each time you install IBM HTTP Server, too. Select a new directory for each instance of IBM HTTP Server that you install.

**5.1+** Selecting certain features causes the installation of other prerequisite features. The following table shows this feature relationship.

Table 28. Features and feature dependencies

| If you select this feature:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | This feature is also installed:                                                    | Feature description                                                                                                               |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| <p><b>Application Server</b></p> <p>Sub-features: Application Server Samples</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Admin Scripting                                                                    | These features are the Application Server run time and its Samples.                                                               |
| <p><b>Administration</b></p> <p>Sub-features: Admin Scripting</p> <p>Administrative Console</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <p>Application Server</p> <p>Admin Scripting</p>                                   | Getting started with System Administration                                                                                        |
| <p><b>Deployment Tools</b></p> <p>Sub-features: Deploy Tool</p> <p>ANT Utilities</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | <p>Admin Scripting</p> <p>Application Server</p> <p>Admin Scripting</p>            | <ul style="list-style-type: none"> <li>Deploying and managing applications</li> <li>Overview of the ANT Utilities</li> </ul>      |
| <p><b>Embedded Messaging</b></p> <p>Sub-features: Server and Client</p> <p>Client Only</p> <p>Message-driven Bean Samples (These samples are the MDBSamples application, which is for the server and not for the client.)</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | <p>Application Server</p> <p>Application Server samples</p> <p>Admin Scripting</p> | <ul style="list-style-type: none"> <li>Installing and configuring a JMS provider</li> <li>Message-driven beans samples</li> </ul> |
| <p>You can install the MDBSamples application without installing the embedded messaging server and client feature. Perhaps you have already installed the server and client feature. Or perhaps you intend to reconfigure the MDBSamples to use messaging resources that WebSphere MQ software provides.</p> <p>In any case, if you install the MDBSamples application and do not have the server and client feature installed, errors occur. The Application Server starts the MDBSamples application at start up. The MDBSamples application cannot start the required messaging resources in the Java Naming and Directory Interface (JNDI).</p> <p>Because of the lack of required resources, the message-driven beans fail to start, which causes an exception that the Application Server logs. The Application Server is not affected other than being unable to start the MDBSamples application.</p> |                                                                                    |                                                                                                                                   |

Table 28. Features and feature dependencies (continued)

| If you select this feature:           | This feature is also installed:                | Feature description                                                                                                                                                                                             |
|---------------------------------------|------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>IBM HTTP Server</b>                |                                                |                                                                                                                                                                                                                 |
| <b>Web Server Plug-ins</b>            |                                                |                                                                                                                                                                                                                 |
| Sub-features:                         | IBM HTTP Server                                |                                                                                                                                                                                                                 |
|                                       | Apache Web Server                              |                                                                                                                                                                                                                 |
|                                       | Microsoft Internet Information Services (IIS)  | Preparing to install and configure a Web server                                                                                                                                                                 |
|                                       | Sun ONE (formerly known as iPlanet) Web Server |                                                                                                                                                                                                                 |
|                                       | Lotus Domino Web Server                        |                                                                                                                                                                                                                 |
| <b>Performance And Analysis Tools</b> |                                                |                                                                                                                                                                                                                 |
| Sub-features:                         | Tivoli Performance Viewer                      | <ul style="list-style-type: none"> <li>• Tivoli Performance Viewer features</li> <li>• Displaying cache information</li> <li>• Performance Monitoring Infrastructure servlet</li> <li>• Log Analyzer</li> </ul> |
|                                       | Dynamic Cache Monitor                          |                                                                                                                                                                                                                 |
|                                       | Performance Servlet                            |                                                                                                                                                                                                                 |
|                                       | Log Analyzer                                   |                                                                                                                                                                                                                 |
| <b>Javadocs</b>                       |                                                |                                                                                                                                                                                                                 |
|                                       |                                                | IBM WebSphere Application Server, Release 5 API Specification in Javadoc format                                                                                                                                 |

### Performance considerations

For better performance in either a development or production environment, do not install the Samples. By omitting the Samples, you can improve Application Server startup time by 60 percent and save 15 percent of disk space. You can save up to 30 percent of process footprint (based on a maximum heap size of 256 MB).

If your applications do not use Java Message Service (JMS) messaging, do not install embedded messaging. This recommendation is especially true if your system has 256 MB or less of physical memory.

Not installing the embedded messaging feature saves about 36 MB of message queuing process memory and can improve application startup time up to 5 percent and save an additional 72 MB of disk space. (This was tested on Windows machines. Results on a Linux platform might differ slightly.)

In addition, if you install the embedded messaging client feature you do not need to install the message-driven Bean Samples. The message-driven Bean Samples are for the embedded messaging server in the server and client feature.

### Web Server plug-in feature considerations

Select the IBM HTTP Server feature to install and configure IBM HTTP Server on the same, or on a different machine than the WebSphere Application Server product. You can migrate plug-ins from an earlier version of WebSphere Application Server to access the current WebSphere Application Server product.

Install IBM HTTP Server on a separate machine using the CD-ROM labeled, **Application Server, IBM HTTP Server**. After installing IBM HTTP Server, you can install the plug-in by installing the base WebSphere Application Server product and clearing all features except the plug-in for IBM HTTP server.

See Preparing to install and configure a Web server.

The Installation wizard automatically installs and configures IBM HTTP Server and the IBM HTTP Server plug-in on the same machine as the WebSphere Application Server if you choose the full installation. No further configuration is necessary. Installing the IBM HTTP Server product with the

WebSphere Application Server product also includes it in the WebSphere Application Server uninstaller program that the Installation wizard creates. Choose the custom type of installation to clear the selection of the IBM HTTP Server feature, if you want to install and uninstall it separately.

You can run the uninstaller program to remove all installed features.

16. Specify a destination directory for the IBM HTTP Server feature, if selected. Click **Next** to continue. The destination directory for the base product is grayed out. You cannot change it.

Specify a target directory for the IBM HTTP Server feature, if you are installing the feature. You cannot change the default installation directory for the embedded messaging feature, if you are installing the feature.

Deleting the default target location and leaving an installation directory field empty stops you from continuing the installation process. The Installation wizard does not proceed when you click **Next**. Enter the required target directory to proceed to the next panel. Non-ASCII special characters are not supported in the name of the installation directory. Spaces are also not supported in the name of the installation directory.

If you have problems accessing the administrative console after installation, check the `installAdminConsole.log` file for a failure indication. Clean up the `/tmp` space and reinstall the administrative console using the `wsadmin` scripting facility.

**5.1+** Version 5.1.x does not allow you to install the IBM HTTP Server feature in the same directory as an earlier version.

If you select the embedded messaging feature and prerequisites are missing, the Installation wizard displays the `mq_prereq.log` error log and takes you back to the Installation type panel. Choose **Custom** installation and clear the checkbox for the embedded messaging feature. The `mq_prereq.log` file is in the system temp directory.

17. Specify target directories for configuration files for any selected Web server plug-ins. Click **Next** to continue.

If you are installing the IBM HTTP Server, you do not need to specify a location for its plug-in configuration file. The wizard uses the installation path you specified for the Web server to derive the location.

If you have previously installed the IBM HTTP Server product on the same machine as the WebSphere Application Server, and are now installing just the plug-in, enter a configuration file location of `IHS_DIR/conf/httpd.conf`, where `IHS_DIR` is the directory where the IBM HTTP Server product is installed.

18. Review the features summary information and click **Next** to install the product code for the additional features or click **Back** to change your specifications. When the installation is complete, the wizard displays the `install_root\logs\mq_install.log` installation log if you selected the embedded messaging feature and errors occur with its installation.
19. Review the `mq_install.log` installation log if it displays. Click **Next** to continue. The wizard displays the Registration panel.
20. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Registration panel is in English when the machine locale uses a double byte character set (DBCS) language.

You can register the product now by using the link on this panel or later, by using the Web address in the `install_root/prt/readme_prt.txt` file. The Web address in the file is encoded for your locale. A known problem for some locales corrupts the Web address so that it is similar to the following example. (The Web address is all one line in the file. The following example shows the address on two lines for clarity.)

```
http%3A%2F%2Fwww.ibm.com%2Fsoftware%2Fwebprt%2Fregister%2Fprodreg.nsf%
2Fcontact%3F0penForm%26lang%3DENUS%26pid%3D45P8842%26q2%3DWindows+2000
```

Use one of the following Web addresses if your `install_root/prt/readme_prt.txt` file is corrupt:

- <http://www.ibm.com/software/webprt/register/prodreg.nsf/contact?OpenForm&>
- <https://www6.software.ibm.com/reg/prt/prt-r>

The Installation wizard starts the First Steps tool. See “firststeps command” on page 182 for more information.

21. Click **Finish** to close the Installation wizard.
22. Verify the success of the installer program by examining the Exit summary panel and the *install\_root/logs/log.txt* for installation status. ISMP records a success message in the *install\_root/logs/log.txt* file: "INSTFIN: The installation is complete." The log is the only source of status information for a silent installation.

Look for severe errors that the installer records in the *install\_root/logs/log.txt* file in the installation root directory to verify that no file system or other unusual errors occurred during installation.

If the *install\_root/logs/log.txt* file does not contain a record of any problems but problems exist, verify or troubleshoot the installation, as described in “Troubleshooting the installation” on page 185 and in Installation component troubleshooting tips .

If problems exist, correct them, uninstall the product, as described in “Uninstalling the product” on page 273, log off as root and back on, and reinstall.

When installing WebSphere Application Server with embedded messaging from the CD-ROM, the following warning messages are issued on some platforms despite a successful installation:

```
MQSERIES.MQM-CL-HPUX: A test command parameter is not valid." and "*
"hostname:/cdrom/hpux/messaging/images": Cannot open the
 logfile on this target or source. Possibly the media is
 read-only or there is a permission problem." on mq_install.log.
```

You can ignore the warning messages, because the problem does not affect the successful installation of WebSphere Application Server with embedded messaging.

23. Install the most current fix pack for the WebSphere Application Server product. See Recommended updates for WebSphere Application Server for information about downloading and installing the upgrades. See “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for information about using the update installer program.
24. Install the most current cumulative fix for the WebSphere Application Server product.
25. Install any interim fixes to bring the node back to its previous fix level.
26. If the node was part of a cell, rejoin the node to the cell with the **addNode** command or the deployment manager administrative console. This action synchronizes the master cell configuration with changes you make to the Application Server configuration during the installation of the new features.
27. Reconfigure the base node with the Network Deployment administrative console.

You have now successfully installed additional features for the WebSphere Application Server product.

## Installing additional features on the deployment manager

This topic describes how to install additional features onto an existing Network Deployment product to update the deployment manager node.

You can add features to a deployment manager node at any time, by running the Installation wizard again. If you encounter a problem such as not having enough temporary space or not having the right packages on your system, cancel the installation, make the required changes, and restart the installation.

The installation of new features uses the InstallShield for Multiplatforms (ISMP) wizard to perform the installation. You can use the Installation wizard or perform the installation from a command line, using the silent installation method.

### Installation programs in WebSphere Application Server products

## 5.1 +

| Product                                          | Link to installation procedure                                   | Label on product CD                                            |
|--------------------------------------------------|------------------------------------------------------------------|----------------------------------------------------------------|
| (Base) WebSphere Application Server              | Base product installation                                        | <b>Application Server, IBM HTTP Server</b>                     |
| Network Deployment                               | Network Deployment installation (which links to this procedure!) | <b>Deployment Manager</b>                                      |
| WebSphere Business Integration Server Foundation | Integration Server installation                                  | <b>Integration Server Disk 1 and Integration Server Disk 2</b> |

Use the following procedure to avoid component regression problems when installing additional features.

1. Log on as the root user on a Linux or UNIX system or as a user in the administrator group on a Windows system.

You cannot install features correctly as a non-root user on a Linux or UNIX system, or as a non-administrator on a Windows system.

2. Use the **backupConfig** command to save the configuration. You can restore it with the **restoreConfig** command although the configuration that you restore does not include any features that you add after backing up the configuration.

For example, issue the following command from the *install\_root/bin* directory of the deployment manager node on a Linux system:

```
#!/backupConfig.sh /tmp/my_backup_2004-06-03.zip -user user_ID -password password
```

See backupconfig command for more information.

3. Stop the deployment manager with the **stopManager** command. Issue the **stopManager** command from the *install\_root/bin* directory of the Network Deployment product. For example, use the following command on a Linux system:

```
#!/stopManager.sh -user user_ID -password password
```

See stopManager command for more information.

4. Uninstall any interim fixes, starting with the last one and finishing with the first one. See “Uninstalling interim fixes, cumulative fixes, and fix packs” on page 239 for more information about how to remove an interim fix, a cumulative fix, or a fix pack.
5. Uninstall any cumulative fixes you installed, starting with the last one and finishing with the first one.
6. Uninstall any fix packs you installed, starting with the last one and finishing with the first one.
7. Log off and back on or reboot a Windows system.
8. Insert the product CD labeled, *Deployment Manager*. Windows systems and most Linux systems are configured to automatically mount CD-ROM drives.
9. Open a shell window and mount the CD-ROM drive if necessary.
10. Start the installation with the **./launchpad.sh** command or the **launchpad** command on a Windows system. See the platform-specific installation instructions if you need more information about restarting the installation from the product CD-ROM, such as instructions for mounting the CD drive.  
The rest of this procedure assumes that you are using the Installation wizard. Corresponding entries in the response file exist for every prompt that is described as part of the wizard. Comments in the response file describe how to customize the options.  
Review the description of the “responsefile” on page 175 for more information.
11. Click **Next** to continue. The license agreement displays.  
The Installation wizard does not support hot keys, such as **Alt-N**. You can tab to **Next** and press **Enter** to select it, for example.



12. Click the radio button beside the **I accept the terms in the license agreement** message if you agree to the license agreement and click **Next** to continue. After you accept the licensing terms, the Installation wizard checks for prerequisites and for previous versions, with which it can either migrate or coexist.
13. Choose to install additional features. The Installation wizard panel that appears when the installer program detects a previous installation at the same product level has an option for installing additional features. Select the check box labeled, Add features to the existing copy.  
Previously installed features are checked and grayed out with the term **(Installed)** at the end of the feature name. The IBM HTTP Server feature is an exception. You can install more than one instance of the IBM HTTP Server product. It is not grayed out or labeled **(Installed)**. You can install the plug-in feature each time you install IBM HTTP Server, too. Select a new directory for each instance of IBM HTTP Server that you install.
14. Skip the coexistence panel that lets you select new ports, if it appears.
15. Select features to install and click **Next** to continue. A description of each feature appears at the bottom of the panel when you roll the cursor over the feature.

Choose from these features:

#### **Deployment manager**

Installs the product run time. It provides high performance and scalability across your deployment environment. It includes multiserver administration, server clustering, load balancing and workload management for hosting highly available e-business applications.

#### **Web services**

The UDDI registry and the IBM Web Services Gateway are enterprise applications that you can deploy to:

- A base WebSphere Application Server product node federated within a Network Deployment cell
- A stand-alone base WebSphere Application Server node

The Network Deployment product is not a stand-alone product for running enterprise applications. To deploy UDDI or the gateway, install the base WebSphere Application Server product. Although it is not installed by default, a copy of the base WebSphere Application Server product is packaged with the Network Deployment product.

**5.1 +** See Developing Web services for more information.

#### **UDDI Registry**

Installs a Version 2 compliant universal description, discovery, and identification (UDDI) registry, accessible from the UDDI registry user console application, or from SOAP or EJB interfaces.

**5.1 +** See IBM WebSphere UDDI Registry for more information.

#### **Web Services Gateway**

Includes a gateway between Internet and intranet environments so that clients can invoke Web services safely from outside a firewall. The gateway uses automatic protocol conversion for externalizing Web services.

**5.1 +** See Enabling Web services through the IBM Web Services Gateway for more information.

#### **Embedded messaging client**

Includes the client necessary for the administration of WebSphere MQ Queues and the mapping of JMS resources into the deployment manager JNDI namespace. It is the same client that you can install as part of the base product embedded messaging feature.

You can run the uninstaller program to remove all installed features.

16. Review the features summary information and click **Next** to install the product code or **Back** to change your specifications. The Summary panel displays the directory for the embedded messaging feature incorrectly on all Linux and UNIX-based platforms, as `/opt/IBM/WebSphere MQ`. Actual installation locations are `/usr/mqm` on AIX systems, and `/opt/mqm` on Linux and all UNIX-based

platforms except AIX. When the installation is complete, the wizard displays the *install\_root/logs/mq\_install.log* installation log if you selected the embedded messaging feature and errors occur with its installation.

17. Review the *install\_root/logs/mq\_install.log* installation log if it displays. Click **Next** to continue.
18. Click **Next** to register the product, or clear the check box and click **Next** to register at a later time. The Registration panel is in English when the machine locale uses a double byte character set (DBCS) language.

You can register the product now by using the link on this panel or later, by using the Web address in the *install\_root/prt/readme\_prt.txt* file. The Web address in the file is encoded for your locale. A known problem for some locales corrupts the Web address so that it is similar to the following example. (The Web address is all one line in the file. The following example shows the address on two lines for clarity.)

```
http%3A%2F%2Fwww.ibm.com%2Fsoftware%2Fwebprt%2Fregister%2Fprodreg.nsf%
2Fcontact%3F0penForm%261ang%3DENU%26pid%3D45P8842%26q2%3DWindows+2000
```

Use one of the following Web addresses if your *install\_root/prt/readme\_prt.txt* file is corrupt:

- <http://www.ibm.com/software/webprt/register/prodreg.nsf/contact?OpenForm&>
- <https://www6.software.ibm.com/reg/prt/prt-r>

The Installation wizard starts the First Steps tool. See “firststeps command” on page 182 for more information.

19. Verify the success of the installer program by examining the Exit summary panel and the *install\_root/logs/1og.txt* for installation status. ISMP records a success message in the *install\_root/logs/1og.txt* file: “INSTFIN: The installation is complete.” The log is the only source of status information for a silent installation.

Look for severe errors that the installer records in the *install\_root/logs/1og.txt* file in the installation root directory to verify that no file system or other unusual errors occurred during installation.

If the *install\_root/logs/1og.txt* file does not contain a record of any problems but problems exist, verify or troubleshoot the installation, as described in “Troubleshooting the installation” on page 185 and in Installation component troubleshooting tips .

If problems exist, correct them, uninstall the product, as described in “Uninstalling the product” on page 273, log off as root and back on, and reinstall.

20. Click **Finish** to close the Installation wizard.
21. Install the most current fix pack for the WebSphere Application Server Network Deployment product. See Recommended updates for WebSphere Application Server for information about downloading and installing the upgrades. See “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for information about using the update installer program.
22. Install the most current cumulative fix for the WebSphere Application Server Network Deployment product.
23. Install any interim fixes to bring the node back to its previous fix level.
24. Start the deployment manager with the **startManager** command. For example, start the deployment manager from the *install\_root/bin* directory of the Network Deployment product on a Linux system with the following command:  

```
./startManager.sh
```

See **startManager** command for more information.
25. Use the administrative console on the deployment manager node to synchronize all node agents. This action synchronizes the master cell configuration with all of the Application Server nodes.

You have now successfully installed additional features for the WebSphere Application Server Network Deployment product.

## Platform-specific tips for installing and migrating

This topic is a collection of platform-specific tips that can help you install and migrate the base WebSphere Application Server product and the Network Deployment product.

If you encounter a problem such as not having enough disk space or temporary space or not having the right packages on your Linux or UNIX-based system, cancel the installation, make the required changes, and restart the installation.

For information related to installing the embedded messaging feature, refer to Tips for installing the embedded messaging feature.

The following sections contain applicable tips:

- All platforms
- All Linux and UNIX-based platforms
- AIX platforms
- HP-UX platforms
- Linux platforms
- Solaris Operating Environment
- Windows platforms

### All platforms

#### Summary of tips that apply to all platforms

- Running servers in different code pages within a single cell is not supported
- Updating all of the cluster members to the same service level
- Avoiding non-ASCII special characters or double-byte character set (DBCS) characters in the name of the installation directory
- Verifying the SDK that WebSphere Application Server products install
- **5.1 +** Avoiding the use of a V5.0.x deployment manager after migrating to V5.1
- **5.1 +** Migrating from embedded messaging to WebSphere MQ requires setting the MQ\_INSTALL\_ROOT variable to the location of the installation root of WebSphere MQ
- **5.1 +** Ignoring the Registration panel when installing only IBM HTTP Server
- Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component
- **5.1 +** Recovering from configuration errors when the deployment manager was not running during migration.
- **5.1 +** Installing all of the Web server plug-ins during the initial installation
- Recovering from an InvalidExecutableException error while starting the jmsserver process
- 134Restarting the server after a configuration change
- 135Updating ports for coexistence requires a WebSphere Application Server installation
- Manually uninstalling all of the beta products before installing WebSphere Application Server
- Installing from a directory with a name beginning with the word disk fails
- Accessing migration tools in the migration subdirectory on the WebSphere Application Server product CD-ROM
- Avoiding license files with bad characters in certain languages
- Updating the XMLConfig utility on WebSphere Version 4.0 Advanced Edition before migration
- Planning to not use the launchClient command on the WebSphere Application Server Network Deployment product
- Installing interim fixes for the IBM HTTP Server feature and the embedded messaging feature
- Uninstalling interim fixes for the IBM HTTP Server feature and the embedded messaging (WebSphere MQ) feature before installing fix pack updates to the features
- Applying interim fixes and fix packs to the embedded messaging feature
- Installing the WebSphere Application Server product on top of WebSphere MQ

- Editing the port number when choosing the coexistence option during installation
- Avoiding using the installation verification test on node names that contain double-byte characters
- Avoiding the use of hot keys in the Installation wizard
- Downloading a refreshed PDF from the WebSphere Application Server Library page, or refer to the information in the information center
- Logging in appropriately to pick up secondary user groups for root
- Retaining the embedded messaging feature when uninstalling an instance that does not have the feature installed
- Installing WebSphere Application Server 5.1 Cumulative Fix 3
- Verifying that no files exist in the `install_root/classes` directory during installation
- Platform-specific issues when dealing with network file systems
- Avoiding the underscore (`_`) character in machine names
- Avoiding the installation of the server and client feature or the client feature more than once
- Locating more information about the embedded messaging feature or WebSphere MQ
- Installing the embedded messaging server feature if WebSphere MQ Version 5.3 is already installed
- Logging off and back on, or rebooting a Windows machine, after uninstalling the embedded messaging feature
- Planning to not use terminal services with the embedded messaging feature
- Avoiding a coexistence problem between embedded messaging, IBM WebSphere Studio Application Developer Integration Edition, and IBM WebSphere Application Server
- Using a version of Internet Explorer other than Version 6.0 to view the IBM HTTP Administration server
- Avoiding a core that is caused by an incompatibility between functions in various C or C++ run-time libraries
- Hiding one copy of the OpenSSL module so that the Apache Web server can start
- Defining the name of the WebSEAL HTTP server in lower case
- Bringing up the IKEYMAN administrative console with the Conversational Monitoring System option for IBM HTTP server 1.3.28 and IBM HTTP Server 2.0.47
- Enabling cryptographic hardware
- Using the WebSphere MQ client instead of the embedded messaging client feature

### Tips that apply to all platforms

- **Running servers in different code pages within a single cell is not supported.**

Both the IBM WebSphere Application Server Network Deployment node and the IBM WebSphere Application Server nodes must be on the same code page.

- **Updating all of the cluster members to the same service level**

Apply the same interim fixes and fix packs to all of the WebSphere Application Server nodes in a cluster. When all of the cluster members are not at the same service level, the following exception can occur:

```
DRSCacheApp E DRSW0008E:
 Exception is: com.ibm.disthub.impl.jms.JMSWrappedException:
 {-1361012295|unknown|java.io.OptionalDataException|}
```

This error can cause memory replication to not function properly.

- **Avoiding special non-ASCII characters or double byte character set (DBCS) characters in the name of the installation directory.**

Do not use non-ASCII special characters or double-byte characters in the directory name. This problem is a known limitation.

- **Verifying the Java 2 SDK that WebSphere Application Server products install.**

Verify that the CD-ROM that you are using is a good copy, by verifying the Java 2 SDK on the CD-ROM.

To verify that the Java 2 SDK on the product CD-ROM is working correctly:

1. Change directories to the `/WAS/jdk/java/bin` directory in the operating system root directory on the product CD-ROM.

For example, on an IBM AIX platform:

```
cd /mnt/aix/WAS/jdk/java/bin
```

On a Windows platform, type the following command:

```
cd CD-ROMDrive:\win\WAS\jdk\java\bin
```

2. Verify the Java 2 SDK version.

On a Linux or UNIX-based platform, type the following command:

```
./java -version
```

On a Windows platform, type the following command:

```
.\java -version
```

The command completes successfully with no errors when the Java 2 SDK is intact.

- **5.1+** **Avoiding the use of a V5.0.x deployment manager after migrating to V5.1.**

After a deployment manager configuration has been migrated to Version 5.1.x, avoid using the Version 5.0.x deployment manager node. Using the V5.0.x node results in an inconsistent configuration and creates an unsupported environment. Use the Version 5.1.x deployment manager node to manage all of the the federated nodes in that cell.

If you inadvertently start the Version 5.0.x deployment manager node, the deployment manager attempts to manage the same cell as the Version 5.1.x deployment manager, but with an older version of the configuration. Any changes made through the Version 5.1.x deployment manager can disappear after the nodes synchronize with the Version 5.0.x deployment manager. This problem can cause major disruptions to the operation of your cell.

It is advisable to disable the Version 5.0.x deployment manager node. Be sure to perform a backup of the configuration using the **backupConfig** command before proceeding. You can disable the Version 5.0.x deployment manager node in a number of ways. The recommended approach is renaming the config directory to something else.

- **5.1+** **Migrating from embedded messaging to WebSphere MQ requires setting the MQ\_INSTALL\_ROOT variable to the location of the installation root of WebSphere MQ.**

When installing V5.1.x with the WebSphere MQ product as the JMS provider, you must use the administrative console to set the *MQ\_INSTALL\_ROOT* variable to the proper path. The installation program does not set the path when you use the WebSphere MQ product. Although WebSphere MQ is a supported JMS provider, migrating from V5.0.x with embedded messaging does not reset the variable, which still points to the installation root of the V5.0.x embedded messaging code.

Use the administrative console to set the *MQ\_INSTALL\_ROOT* to the installation root of the WebSphere MQ product.

1. Click **Environment > Manage WebSphere Variables > MQ\_INSTALL\_ROOT**.
2. Type the fully qualified file path for the installation root of the WebSphere MQ product in the Value field.
3. Click **Apply > Save > Save** to apply and save your changes.

- **5.1+** **Ignoring the Registration panel when installing only IBM HTTP Server.**

If you install IBM HTTP Server on another machine by installing WebSphere Application Server, Version 5.1 and clearing all of the features but the one for IBM HTTP Server and its plug-in, the installer program displays the Registration panel. Select the check box to clear it and click **Next** to avoid registration, which is not required when installing only the Web server. If you do attempt to register the feature when you have not installed WebSphere Application Server, Version 5.1, you can ignore the following error:

```
java.io.FileNotFoundException:
C:\Program Files\WebSphere\AppServer\prt\PRT_Welcome.html
(the system cannot find the path specified)
```

- **Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component.**

When installing the embedded messaging feature of the WebSphere Application Server product and the Network Deployment product on the same machine, install the WebSphere Application Server product first. Then install the Network Deployment product. Otherwise, embedded messaging installation can fail, or can install with errors.

- **5.1+ Recovering from configuration errors when the deployment manager was not running during migration.**

During migration of federated nodes, the V5.1 deployment manager must be running for the migration tools to:

- Update the configuration for each federated node
- Request full synchronization

If the V5.1 deployment manager is not running, failures can occur. You can correct the failures by starting the V5.1 deployment manager, running the **updateVariables.jacl** command in the */migration-specific-backup/variables\_files* directory that the migration tools create, and performing a full synchronization for the cell.

Use the following procedure to correct the problem:

1. Start the deployment manager with the **startManager** command.
2. Update the variables.xml files:

```
>cd /websphere/appserver/bin
>wsadmin -f /.../websphereBackup/variables_files/updateVariables.jacl
```
3. Use the administrative console or the scripting facilities of the deployment manager to perform a full synchronization for the node.

For example, you can use the **syncNode** command to synchronize the node.

- **Installing all of the Web server plug-ins during the initial installation.**

**5.1+** You must select all of the plug-ins that you require during the initial installation of the WebSphere Application Server product to automatically control the configuration of the Web servers. If the installer program configures the Web servers, the uninstaller program can remove the configuration from each Web server when you uninstall WebSphere Application Server. If you install the product again to add features, any Web server plug-ins that you select get configured but the uninstaller program is already configured. The uninstaller program does not remove the configuration from each additional Web server whose plug-in that you selected during the second installation.

**5.1+** Install a new instance of the WebSphere Application Server to create an uninstaller program that can remove the configuration from all of the Web servers whose plug-ins that you select. Or you can remove the configuration manually.

- **Recovering from an InvalidExecutableException error while starting the jmsserver process**

You might get an exception while starting the jmsserver process when you install Network Deployment first and then install the base WebSphere Application Server product and its embedded messaging feature on the same node. The error message is recorded in the *install\_root/logs/jmsserver/SystemOut.log* file:

```
[9/5/02 14:35:37:818 EDT] 36349b90 JMSService
E MSGS0001E: Starting the Server failed with exception: com.ibm.ws.process.exception.
InvalidExecutableException: Error creating new process.
 002: No such file or directory
```

In addition, although the *mq\_install.log* file might appear to contain no errors, the *createMQ.nodeName\_jmsserver.log* file can contain I/O exceptions. These exceptions result from a corrupted installation of the embedded messaging feature caused by installing the Network Deployment product before the base WebSphere Application Server product. The workaround is to uninstall both products, reinstall the base WebSphere Application Server product, and then reinstall the Network Deployment product.

- **Restarting the server after a configuration change.**

If you make any changes to the configuration, restart the server as noted in the messages section of the administrative console.

- **Updating ports for coexistence requires installing WebSphere Application Server.**

Port updates for coexistence require the installation of WebSphere Application Server. This requirement affects port updates for IBM HTTP Server coexistence. Port updates do not occur if only the IBM HTTP Server is installed. In this case, manually update the `httpd.conf` file. Verify that the ports that you use are available. For example, use the `netstat -a` command to see ports that are being used.

- **Manually uninstalling all of the beta products before installing WebSphere Application Server.**

You might experience problems if portions of the beta product remain.

To make sure that you get a clean uninstall, follow these steps:

1. Uninstall the beta version of WebSphere Application Server.

Refer to Uninstalling WebSphere Application Server for more information.

2. Confirm that WebSphere MQ and WebSphere embedded messaging Publish and Subscribe (WEMPS) are uninstalled.

If not, uninstall them by using the Windows Add/Remove programs tool or Smitty/installp, whichever is appropriate. Refer to Uninstalling WebSphere Application Server for more information about manually uninstalling on specific platforms.

3. Remove all of the directory trees from the earlier installations of WebSphere Application Server products.

If the stand-alone WebSphere MQ Version 5.3 is installed, remove the WEMPS directory only. Do not uninstall or remove other WebSphere MQ Version 5.3 items.

- **Installing from a directory with a name beginning with the word disk fails.**

Installing WebSphere Application Server Version 5 from a folder that begins with the word `disk` results in an error. Provide another name for the folder.

- **Accessing migration tools in the migration subdirectory on the WebSphere Application Server product CD-ROM.**

**5.1+** A migration subdirectory on the installation image on the CD-ROM contains the WASPreUpgrade migration tool. The WASPreUpgrade migration tool is intended for scenarios where you might save the currently installed configuration before installing the Version 5.1 product. One example of this situation is where you must upgrade the operating system as part of the Version 5.1 installation. You can migrate the earlier version, copy the migrated files in the backup directory to another system, update the operating system, restore the migrated files in their backup directory, install Version 5.1, and complete the migration.

**5.1+** You can also use the migration directory on the CD-ROM to back up a Version 5.1 configuration in the event of an operating system upgrade. After the upgrade, you can restore the Version 5.1 configuration using the WASPostUpgrade tool.

- **Avoiding license files with bad characters in certain languages.**

Use the graphical installation interface to avoid this problem.

- **Updating the XMLConfig utility on WebSphere Application Server Advanced Edition, Version 4.0 before migration.**

The migration tools use the XMLConfig utility to export the configuration of V4.0 Advanced Edition. You can install any of the following fixes:

- PQ52555 - XMLConfig does not export clone property configuration
- PQ55064 - XMLConfig does not export Enter JavaBeans (EJB) to data source level mappings
- PQ58038 - Performing an XMLConfig export produces an extra CRLF
- PQ62103 - XMLConfig full export fails with a NullPointerException error in a multinode environment
- PQ62471 - Security AdminRoles are not getting exported during the XML export
- PQ63815 - The "=" symbol is not a valid character for the value string in the XMLConfig file

- **5.1+ Planning to not use the launchClient command on the WebSphere Application Server Network Deployment product.**

The **launchClient** command works on a base WebSphere Application Server node and on a WebSphere Business Integration Server Foundation node, where the Integration Server product extends the base product. The WebSphere Application Server client program does not work on a Network Deployment node.

- **Installing interim fixes for the IBM HTTP Server feature and the embedded messaging feature.**

Three different sites contain service for WebSphere Application Server products and features:

- Download interim fixes for the IBM HTTP Server feature from the IBM Support site for IBM HTTP Server. The readme file that is bundled with the interim fix describes how to install the interim fix. You do not use the update installer program for the WebSphere Application Server (also known as the updateInstaller program or the Update Installation wizard) to install interim fixes for the IBM HTTP Server feature.
- Download interim fixes for the embedded messaging feature from the WebSphere MQ Service download site. Download the readme file for a description of how to install the interim fix. You do not use the update installer for the WebSphere Application Server to install interim fixes for the embedded messaging feature.
- Download any interim fixes, cumulative fixes, and fix packs for the WebSphere Application Server family of products from the IBM Support site for WebSphere Application Server. Cumulative fixes and fix packs do include service for the IBM HTTP Server feature and for the embedded messaging feature. Interim fixes for a WebSphere Application Server product do not include service for the IBM HTTP Server feature or the embedded messaging feature. The readme file that is bundled with the WebSphere Application Server interim fix, the cumulative fix, or the fix pack describes how to use the update installer program for WebSphere Application Server products to install the interim fix, the cumulative fix, or the fix pack.

Installing a cumulative fix or a fix pack removes all of the interim fixes for serviceable components, except for interim fixes for the IBM HTTP Server feature or the embedded messaging feature. You must remove those manually as described in the following tip.

- **Uninstalling interim fixes for the IBM HTTP Server feature and the embedded messaging (WebSphere MQ) feature before installing cumulative fixes and fix packs to the features**

If you have installed interim fixes for the IBM HTTP Server feature from the IBM Support site for IBM HTTP Server, or if you have installed interim fixes for the embedded messaging feature from the WebSphere MQ Service download site, the update installer program cannot uninstall interim fixes for these feature components before installing a cumulative fix or a fix pack that might include service for the features. The update installer program does uninstall interim fixes for all of the other components. If the interim fixes for the IBM HTTP Server feature and the embedded messaging feature are not uninstalled for some reason, installing a cumulative fix or a fix pack to the IBM HTTP Server feature or to the embedded messaging feature might fail, or the updated features might fail when you begin using them.

If you reinstall all of the interim fixes for either feature that are more current than the cumulative fix or the fix pack, there is no problem.

You can also choose to have the update installer skip applying cumulative fix or fix pack updates to IBM HTTP Server or embedded messaging if you do not require the updates. You can skip these updates and still apply updates to the rest of the product.

- **Applying interim fixes and fix packs to the embedded messaging feature.**

The IBM WebSphere MQ product can coexist with the embedded messaging feature, if they are at compatible service levels. Always apply any outstanding corrective service to the stand-alone WebSphere MQ product, before installing an interim fix (from the WebSphere MQ Support site) or installing a cumulative fix or a fix pack (from the WebSphere Application Server Support site) that might include a service update for the embedded messaging feature.

In each of the following cases, use the readme file associated with the interim fix or with the cumulative fix or the fix pack to install the service. Do not use the update installer program (also known as the updateInstaller program or the Update installation wizard) to install interim fixes for the embedded messaging feature. Download interim fixes for the embedded messaging feature from the WebSphere MQ Service download site.



For example:

- If you have the embedded messaging server and client feature and do not have WebSphere MQ, do not attempt to apply a corrective service diskette (CSD) for WebSphere MQ. Instead, go to the WebSphere MQ Service download site and download the *WebSphere Embedded Messaging interim fixes for WebSphere Application Server V5.1*.

Upgrading the embedded messaging CSD level by applying a WebSphere MQ CSD is supported only for customers with a full WebSphere MQ license.

- If you have the embedded messaging client feature and a stand-alone WebSphere MQ product, apply any outstanding corrective service to the WebSphere MQ product before installing the interim fix (from the WebSphere MQ Support site) for the embedded messaging client feature or installing a fix pack (from the WebSphere Application Server Support site) for the product that includes service for the embedded messaging client feature.
  - If you have the embedded messaging client feature and the stand-alone WebSphere MQ product is already at the latest corrective service level, install the interim fix (from the WebSphere MQ Support site) for the embedded messaging client feature or install a fix pack (from the WebSphere Application Server Support site) for the product that includes service for the embedded messaging client feature.
- **Installing the WebSphere Application Server product on top of WebSphere MQ.**

When planning to install a WebSphere Application Server, Version 5 release on a machine that already has a copy of WebSphere MQ V5.3 installed on it, you are recommended to complete the following steps:

1. Ensure that all the WebSphere MQ features required by the WebSphere Application Server embedded messaging are installed:

- on UNIX:
  - for WebSphere Application Server Network Deployment or Client: Java Messaging
  - for WebSphere Application Server (base): Runtime, Base (or SDK), Server, Client, Java Messaging and All Message Catalogs
- on Windows:
  - for WebSphere Application Server ND or Client: Java Messaging
  - for WebSphere Application Server (base): Server and Java Messaging

2. Ensure that WebSphere MQ v5.3 is upgraded to the appropriate prerequisite CSD level:

- CSD1 for WebSphere Application Server V5.0
- CSD3 for WebSphere Application Server V5.0.1
- CSD4 for WebSphere Application Server V5.0.2 and later, including V5.1

3. Install the WebSphere Application Server V5 GM release.

4. Install any appropriate WebSphere Application Server fix packs.

After the initial installation of WebSphere MQ and WebSphere Application Server, you should service WebSphere MQ independently of the WebSphere Application Server fix packs. For example, download and apply the WebSphere MQ V5.3 CSD4 before upgrading your WebSphere Application Server to V5.0.2 or before installing V5.1. This approach helps to avoid failures in uninstalling WebSphere Application Server fix packs.

- **Editing port numbers that First Steps uses after selecting the coexistence option during installation.**

Change the First Steps port number for the administrative console to match the port number in a coexistence scenario.

Edit the administrative console address when First Steps attempts to open the administrative console Web address. Change the port to match the coexistence port number for the administrative console port. The following example shows how to change the address to the default coexistence port number for the administrative console port:

```
http://localhost:9091/admin
```

- **Avoiding using the installation verification test on node names that contain double-byte characters.**

The installation verification test is not supported on node names that contain double-byte characters. See “Troubleshooting the installation” on page 185 for more information.

- **Avoiding the use of hot keys while using the Installation wizard.**

The Installation wizard does not support hot keys.

Hot keys do not work for any languages on any operating system platforms. For example, on the **Cancel Confirmation** panel, you cannot type **Alt - Y** or **Alt - N** to say Yes or No to cancel the installation.

- **Downloading a refreshed PDF from the WebSphere Application Server Library page, or refer to the information in the information center.**

The Getting Started PDF on the product CD-ROM is back level. The information in the information center is more current. This is particularly true for the information related to migrating or uninstalling a migrated or coexisting node that uses the embedded messaging feature.

Download a refreshed PDF from the WebSphere Application Server Library page, or refer to the information in the information center before migrating, uninstalling a node after migration, or uninstalling a coexisting node that has the embedded messaging feature installed.

- **Logging in appropriately to pick up secondary user groups for root**

When the root user on all platforms (except Windows platforms) does not belong to the mqbrkrs and mqm user groups, errors occur when installing the embedded messaging feature.

On many systems, such as SuSe Linux, if you telnet and issue the **id** command or the **groups** command, you cannot see the groups mqm or mqbrkrs even though they might exist. Solve this problem in one of two ways:

- Use the **ssh** command to log in
- Issue the **su -** command

After using one of the commands, verify the required groups with the **id** command or the **groups** command.

In a normal root login, issue the **su** command. For a real root login, issue the **su -** command.

Display settings for a normal root login are automatic. For a real root login, you must set your display environment properly to successfully view the GUI installation wizard. Otherwise, you see a message about Preparing Java(tm) Virtual Machine... and seven rows of dots, but no installation GUI and no further messages. Refer to the documentation for your platform to determine proper display settings.

See “Troubleshooting the installation” on page 185 for a description of the error messages that you might receive if you have not picked up the required secondary groups for root.

- **Retaining the embedded messaging feature when uninstalling an instance that does not have the feature installed.**

The wizard prompts you to uninstall the embedded messaging feature, if the feature is installed. To obey licensing rules, always remove the embedded messaging feature unless other product instances share the code.

This panel can appear whenever the embedded messaging feature is installed on the node and not just when you are uninstalling an instance that has the feature installed. For example, assume that you install the embedded messaging feature when you install the base product. Assume that you install a second instance of the base product, but do not install the embedded messaging feature with the product. If you uninstall the second instance, the Installation wizard displays the prompt to uninstall the embedded messaging feature.

- **Installing WebSphere Application Server 5.1 Cumulative Fix 3.**

This collection of fixes for Version 5.1 updates the WebSphere Application Server base product or the Network Deployment product to Version 5.1.0.3. Cumulative Fix 3 includes all of the interim fixes in Cumulative Fix 2. Cumulative Fix 3 is available at the <http://www-1.ibm.com/support/docview.wss?uid=swg24006675> Web address.

Always install the latest cumulative fixes as they are released. See Cumulative Fix Strategy for WebSphere Application Server V5.0 and V5.1 for more information.

- **Verifying that no files exist in the `install_root/classes` directory during installation.**

The *install\_root/classes* directory is reserved for testing and debugging fixes when customers call IBM Support. Having files in the directory during installation can cause various problems depending on the fix left in the directory. Verify that no files exist in the *install\_root/classes* directory.

When IBM Support queues work for you and provides you test or debug fixes, you put the fixes in the *install\_root/classes* directory. By default, the *install\_root/classes* directory is picked up first in the WebSphere Application Server class path to let it override other classes.

This directory lets you verify or debug a fix. After accepting the test fix or finishing with the debugging of the debug fix, delete the fix from the *install\_root/classes* directory to return the system to a working state. If you do not remove such fixes from the *install\_root/classes* directory, you can experience errors.

- **Platform-specific issues when dealing with network file systems.**

The installation CDs are platform specific. Each CD label identifies the platform for which it is designed to be used. The CDs for a particular platform should only be inserted and run from a CD drive that is mounted on the platform that is identified on the CD.

**Installing with a network file system mount:**

When installing from a drive mounted on a network file system (NFS), use NFS client and server platforms with identical versions of the operating system and a well-tuned NFS, if possible. For example, if installing on an xSeries (i386) RHEL3 machine, the recommended NFS server is also an xSeries (i386) RHEL3 machine.

If you are forced to use a cross-platform NFS mount, such as when installing from CD on a Linux for S/390 system, mount the drive with the `-o tcp` option if the option is available. For example:

```
mount -o tcp mycdrom.hostname:/media/cdrom /mnt
```

Refer to the man pages for the **mount** command to see if the `tcp` option is available. Using a CD drive or a disk drive mounted with the default mount option (`udp`) is known to cause errors on some versions of Linux. Red Hat Enterprise Linux 3.0 UL1 on zLinux platforms is particularly sensitive to this problem, but the recommendation applies to all Linux platforms.

If you must copy CD images for one platform to an NFS disk on another operating system platform, use a command that preserves symbolic links. For example, on an AIX platform use the **cp -hrp** command to copy data from an HP-UX disk and preserve the symbolic links. Without the `-h` option, the `cp` command on an AIX platform attempts to resolve symbolic links when copying an HP-UX disc instead of preserving the links.

Even with the `-h` option, the **cp** command on a Solaris platform does not preserve symbolic links when copying an HP-UX disc. On a Solaris platform, use the **tar -cvf** command to copy data from an HP-UX disk and preserve the symbolic links.

Consult the man page for the copy command on the NFS system to understand how the platform supports copying symbolic links.

Because many NFS incompatibilities can exist among various platforms, do not use a drive mounted on a network file system to install from one operating system platform to another without first verifying that the NFS configuration is robust. For example, if your mounted CD is `/mnt`, issue the following command on a Linux for S/390 system to see if you have an NFS problem:

```
unzip -l /mnt/linuxs390/WAS/setup.jar
```

The proper output from the command lists all the files in the `setup.jar` file without error. If you encounter problems in running the command, verify that your `/etc/exports` file includes the `no_root_squash` option.

See <http://www.redhat.com/docs/manuals/enterprise/RHEL-3-Manual/sysadmin-guide/s1-nfs-mount.html> for more information about the `-o tcp` option.

See Troubleshooting the installation for more information about possible symptoms when installing with a faulty NFS mount. Contact your system administrator for help if you continue to have NFS problems.

- **Avoiding the underscore (\_) character in machine names.**

Internet standards dictate that domain names conform to the host name requirements described in Internet Official Protocol Standards RFC 952 and RFC 1123. Domain names must contain only letters

(upper or lower case) and digits. Domain names can also contain dash characters ( - ) as long as the dashes are not on the ends of the name. Underscore characters ( \_ ) are not supported in the host name.

If you have installed WebSphere Application Server on a machine with an underscore character in the machine name, access the machine with its IP address or the localhost designator until you rename the machine.

- **Locating more information about the embedded messaging feature or WebSphere MQ.**

For more information about the actions to take before installing the embedded messaging feature, refer to Installing WebSphere embedded messaging as the JMS provider . For more information about installing JMS providers, refer to Installing and configuring a JMS provider .

For information about installing the WebSphere MQ Version 5.3 product, or migrating to WebSphere MQ Version 5.3 from an earlier release, refer to the appropriate WebSphere MQ *Quick Beginnings* book:

- *WebSphere MQ for Windows, V5.3 Quick Beginnings*, GC34-6073
- *WebSphere MQ for AIX, V5.3 Quick Beginnings*, GC34-6076
- *WebSphere MQ for Solaris, V5.3 Quick Beginnings*, GC34-6075
- *WebSphere MQ for Linux for iSeries and Linux for zSeries, V5.3 Quick Beginnings*, GC34-6078

These books are available at the WebSphere MQ messaging platform-specific books Web page.

- **Installing the embedded messaging server feature if WebSphere MQ Version 5.3 is already installed.**

You have a choice if WebSphere MQ Version 5.3 is already installed:

- You can install only the embedded messaging client feature on a machine that already has WebSphere MQ Version 5.3.

To use WebSphere MQ Version 5.3 as the JMS provider, install the IBM WebSphere Application Server product with only the embedded messaging client feature. The Network Deployment product has only the embedded messaging client feature. Installing and using the WebSphere Application Server embedded messaging client feature is recommended with either the server feature or the full WebSphere MQ Version 5.3 product.

WebSphere Application Server messaging applications can use the WebSphere MQ Version 5.3 product as the JMS provider. Using the client feature, however, requires that you install the WebSphere MQ Version 5.3 Java messaging feature.

- You can install the embedded messaging server and client features on a machine that already has WebSphere MQ Version 5.3.

To install the embedded messaging server feature when WebSphere MQ Version 5.3 is already installed, upgrade WebSphere MQ Version 5.3:

- **5.1+** Apply the CSD04 update to the original WebSphere MQ Version 5.3 release if you are running V5.0.2 or V5.1 of WebSphere Application Server.
- Install the WebSphere MQ Version 5.3 features, server and Java messaging, which the WebSphere Application Server embedded messaging server feature requires.

If you install WebSphere MQ Version 5.3 without the required features, the installation of either embedded messaging feature of IBM WebSphere Application Server is unsuccessful because of prerequisite check errors. The Enterprise package includes installation images of the WebSphere MQ Version 5.3 product and the WebSphere MQ Event Broker product, with restricted licensing. You can use the products to install the required WebSphere MQ Version 5.3 features or to install the refresh release of WebSphere MQ Version 5.3 for use with the Enterprise product.

- **Logging off and back on, or rebooting a Windows machine, after uninstalling the embedded messaging feature.**

If you uninstall the embedded messaging feature, log off and back on, or reboot a Windows machine before reinstalling.

- **Planning to not use terminal services with the embedded messaging feature.**

Terminal services is not supported as a valid installation scenario when installing WebSphere Application Server and the embedded messaging feature.

- **Avoiding a coexistence problem between embedded messaging, IBM WebSphere Studio Application Developer Integration Edition, and IBM WebSphere Application Server.**

The IBM WebSphere Studio Application Developer Integration Edition and IBM WebSphere Application Server both include an option to install embedded messaging. The embedded messaging option in these two products is incompatible.

To avoid this problem, do not install embedded messaging for both products on the same machine.

- **Using a version of Internet Explorer other than Version 6.0 to view the IBM HTTP Administration server.**

Forms Proxy Settings and Proxy Cache do not behave correctly When using Internet Explorer Version 6.0 to view the IBM HTTP Administration server, the Forms Proxy Settings and Proxy Cache do not behave correctly. If you select the radio buttons that display editable fields, the fields do not display.

To work around this problem, use another version of the browser.

- **Avoiding a core that is caused by an incompatibility between functions in various C or C++ run-time libraries.**

Depending on the order that the libraries are loaded, a core can occur or IBM HTTP Server fails to initialize. This problem is seen on various Red Hat Intel versions, and SuSE SLES and Red Hat on S/390 platforms.

The GSKit libraries used by the IKEYMAN utility and the IBM HTTP Server SSL module require the C++ library `libstdc++-libc6.1-1.so.2` file on the Intel platform and the `libstdc++-libc6.1-2.so.3` file on the OS/390 platform. If the libraries do not exist, the IKEYMAN utility might provide only limited functionality and the IBM HTTP Server SSL fails to initialize. If another version of this library, or another library with a common function, loaded first, it might result in the use of an incompatible library routine.

If you encounter this problem, set the environment variable `LD_PRELOAD` to the following values before starting IBM HTTP Server or IKEYMAN:

- Intel: `export LD_PRELOAD=/usr/lib/libstdc++-libc6.1-1.so.2`
- OS/390: `export LD_PRELOAD=/usr/lib/libstdc++-libc6.1-2.so.3`

This change forces the library to load first when the application starts.

- **Hiding one copy of the OpenSSL module so that the Apache Web server can start.**

Apache Web server fails to start when using the `mod_ssl` plug-in module and issuing the following command:

```
apachectl startssl
```

A problem exists with a single process that has two copies of the OpenSSL modules.

Hide the Global Security Kit (GSKit) copy of the OpenSSL modules by renaming the `opt/ibm/gsk7/icc` directory to `opt/ibm/gsk7/icc_save`.

- **Defining the name of the WebSEAL HTTP server in lower case.**

When defining your WebSEAL HTTP Server in the administrative console, you must define the name of the WebSEAL HTTP Server all in lower case.

To define the name of the WebSEAL server name, click **LTPA > Trust Association > Interceptors > com.ibm.ws.security.web.WebSealTrustAssociationInterceptor > Custom Properties > com.ibm.websphere.security.webseal.hostnames**.

- **Bringing up the IKEYMAN administrative console with the Conversational Monitoring System option for IBM HTTP server 1.3.28 and IBM HTTP Server 2.0.47**

- You cannot bring up IKEYMAN.
- IKEYMAN does not have the Conversational Monitoring System (CMS) database option to choose from when creating a key database for IBM HTTP Server.

IKEYMAN is changed in Global Security Kit (GSKit) 7 to integrate IKEYMAN for Java Secure Socket Extension (JSSE) that is Java-based and GSKit that is C++-based. IKEYMAN is a tool used to manage digital certificates. You can now use the same IKEYMAN for WebSphere Application Server and IBM HTTP Server. For example, creating a Java Key Store (JKS) or a CMS key database from the same panel within IKEYMAN.

A key database enables a client application to connect to those servers that have digital certificates signed by those certificate authorities (CAs). You must perform extra steps to use the same IKEYMAN with the CMS option in addition to the JKS option. The CMS option is needed when you set up IBM HTTP Server for Secure Sockets Layer (SSL) transactions.

Set the `JAVA_HOME` variable to the location of the IBM Software Development Kit on the machine. If you are using WebSphere Application Server, set the `JAVA_HOME` variable to the Java Runtime Environment (JRE) that is shipped with WebSphere Application Server.

For example:

- On the Linux platform: Export the `JAVA_HOME=/opt/WebSphere/AppServer/java` variable.
- On the UNIX platform: Run the `<IHS install root>/bin/ikeyman`, or `<WAS_HOME>/bin/ikeyman.sh` command, from a command prompt to bring up the IKEYMAN administrative console.
- On the Windows platform: To bring up IKEYMAN, click **Start > Programs > IBM HTTP Server 1.3.28 (or IBM HTTP Server 2.0.47) > Start Key Management Utility**, or run `<WAS_HOME>/bin/ikeyman.bat` at a command prompt.

If you cannot bring up IKEYMAN after the previous steps, perform the following steps:

1. Set the `JAVA_HOME` variable to the home of the IBM Software Development Kit for the Java platform located on your machine. If you are using WebSphere Application Server, you should set the `JAVA_HOME` variable to the JRE that is shipped with WebSphere Application Server.
2. If you are not using the JRE that comes with WebSphere Application Server, and the `gskikm.jar` file exists in your `JAVA_HOME` directory, rename and move the `$JAVA_HOME/jre/lib/ext/gskikm.jar` file to a directory that is not visible to the IBM Software Development Kit class path, external directory, and the bootstrap class path. For example, on the Linux platform: `mv $JAVA_HOME/jre/lib/ext/gskikm.jar/gskfiles/gskikm.jar.org`.
3. Ensure that the `java.security` file has the following providers:

You can find the `java.security` file in the security directory path. For example, the full path to the file on the Windows platform is `C:\Program Files\IBM\Java141\jre\lib\security\java.security`.

```
security.provider.2=com.ibm.crypto.provider.IBMJCE
security.provider.3=com.ibm.spi.IBMCMSProvider
```

If you try to use cryptographic hardware with IBM HTTP Server, ensure that the file contains the following lines in the order that is shown here:

```
security.provider.1=com.ibm.spi.IBMCMSProvider
security.provider.2=com.ibm.crypto.provider.IBMJCE
security.provider.3=com.ibm.jsse.IBMJSSEProvider
security.provider.4=com.ibm.crypto.pkcs11.provider.IBMPKCS11
```

If you are not using an IBM Software Development Kit or if the IBM Software Development Kit files in the `lib\ext` directory are older than the ones that come with the GSKit, you have to copy all the Java archive (JAR) files from the `ibm\gsk7\classes\jre\lib\ext` directory to the `Java141\jre\lib\ext` directory.

- On the UNIX platform:  
To bring up IKEYMAN, run the `<IHS install root>/bin/ikeyman`, or `<WAS_HOME>/bin/ikeyman.sh` command, from a command prompt.
- On the Windows platform:  
To bring up IKEYMAN, click **Start > Programs > IBM HTTP Server 1.3.28 (or IBM HTTP Server 2.0.47) > Start Key Management Utility**, or run `WAS_HOME/bin/ikeyman.bat` at a command prompt.

#### • Enabling cryptographic hardware

The cryptographic token is no longer a separate item on the IKEYMAN GUI menu. It is treated as one of the key store types. You can specify the PKCS11 module name by specifying the property of `DEFAULT_CRYPTOGRAPHIC_MODULE` in the `ikmuser.properties` file as before. However, IKEYMAN no longer tries to load the dynamic link lists (DLL) or Library (LIB) at startup time to decide whether to support the cryptographic token. The value of the `DEFAULT_CRYPTOGRAPHIC_MODULE` property is used only for the default value shown on the GUI.

When you open the Cryptographic Token, IKEYMAN first retrieves the value of the `DEFAULT_CRYPTOGRAPHIC_MODULE` property in the `ikmuser.properties` file and pre-fill the value in the **File Name** and **Location** fields in the **Key Database File > Open dialog box of IKEYMAN** GUI. You can modify the value in the **File Name** and **Location** fields or click the **Browse** button.

Once the specified DLL or LIB is loaded successfully, you can use IKEYMAN. After opening a cryptographic token successfully, IKEYMAN displays the certificates stored in the cryptographic token.

- **Using the WebSphere MQ client instead of the embedded messaging client feature.**

Use the client from the full WebSphere MQ installation and do not use the embedded messaging client feature. It is not necessary to install the embedded messaging client feature. Both clients are the same, except that the embedded messaging client has a licensing restriction that prohibits it from communicating with queue managers outside of the WebSphere Application Server environment.

## All Linux and UNIX-based platforms

### Summary of tips that apply to all Linux and UNIX-based platforms

- Preparing a Linux or UNIX operating platform for the embedded messaging feature
- Installing to fixed locations on Linux and UNIX-based operating systems
- Defining the prerequisite Linux or UNIX operating system groups, `mqm` and `mqbrkrs`
- Creating and mounting the required journalized file system, `/var/mqm`, before installing WebSphere Application Server
- Restricting access to the `var/mqm/errors` directory and to the messaging logging files
- Verifying the system `cp` command when using emacs or other freeware
- Ignoring the desktop icon messages from InstallShield for Multiplatforms (ISMP) during installation
- Avoiding setting WebSphere Application Server environment variables in the user profile or in the system profile
- Avoiding spaces in the name of the installation directory on Linux and UNIX platforms
- Moving all of the core dump files from the `var/sadm/pkg` directory
- Starting the `jmsserver` process before running some Samples on UNIX platforms
- Using an option on the **install** or **uninstall** commands to identify a temporary directory other than the default `/tmp` directory
- Adding additional name space bindings with a new console user
- Migrating when the SDK that ships with the product is not available
- A delay occurs before the Installation wizard displays its first panel
- Manually entering Launchpad Web addresses to launch the links directly from your browser
- Using the `tcp` option to mount CD drives or disk drives in a network file system mount if the option is available
- Preserving symbolic links when copying product CDs to network file server disks

### Tips that apply to all Linux and UNIX-based platforms

- **Preparing a Linux or UNIX operating platform for the embedded messaging feature.**

Perform this step only if you are installing the Java Message Service (JMS) provider for WebSphere Application Server on a Linux or UNIX-based operating system. You must install the embedded messaging feature to use the JMS provider that WebSphere Application Server provides.

If you are installing the embedded messaging feature, you must create two operating system groups as described in *Installing WebSphere embedded messaging as the JMS provider* .

The Solaris Operating Environment and HP-UX also require that you increase kernel settings as described in *Installing WebSphere embedded messaging as the JMS provider* .

For an index of platform-specific information about using the embedded messaging feature, see “Tips for installing the embedded messaging feature” on page 171.

- **Installing to fixed locations on Linux and UNIX-based operating systems**

On Linux and UNIX-based operating systems, the embedded messaging feature installs to fixed locations that you cannot override. The default locations are:

- Linux platforms: `/var/mqm`, `/var/wemps`, `/opt/mqm`, `/opt/wemps`

- AIX platforms: /usr/opt/mqm, /usr/opt/wemps, /var/mqm
- Solaris platforms: /var/mqm, /var/wemps, and /usr

The Installing WebSphere embedded messaging as the JMS provider topic describes directory locations in detail.

- **Defining the prerequisite Linux or UNIX operating system groups, mqm and mqbrkrs**

Before you install the embedded messaging component on UNIX or Linux platforms, you must define the operating system groups **mqm** and **mqbrkrs**, and the user IDs needed for WebSphere embedded messaging. For detailed information, see Installing WebSphere embedded messaging as the JMS provider .

- **Creating and mounting the required journalized file system, /var/mqm, before installing WebSphere Application Server**

On UNIX platforms, use a partition strategy with a separate volume for the messaging data. A separate partition prevents other system activity from being affected when a large amount of messaging work builds up in the /var/mqm directory.

The /var file system is used to store all of the security logging information for the system, and is used to store the temporary files for e-mail and printing. Therefore, it is critical that you maintain free space in the /var directory for these operations. If you do not create a separate file system for messaging data, and the /var directory fills up, all of the security logging, e-mail, and printing is stopped on the system until some free space is available in the /var directory.

- **Restricting access to the /var/mqm/errors directory and messaging logging files**

After installing WebSphere embedded messaging, you must restrict access to the /var/mqm directories and to the log files needed for WebSphere embedded messaging, such that only the mqm user ID or members of the mqm user group have write access. For detailed information, see Installing WebSphere embedded messaging as the JMS provider and Securing messaging directories and log files .

- **Verifying the system cp command when using emacs or other freeware.**

If you have emacs or other freeware installed on your Linux or UNIX-based system, verify that the system **cp** command is being used.

1. Type which cp at the command prompt before running the installation program for a WebSphere Application Server product.
2. Remove the freeware directory from your PATH if the resulting directory output includes freeware. For example if the output shows /opt/freeware/bin/cp, remove the directory from the PATH.
3. Install the WebSphere Application Server product.
4. Add the freeware directory back to the PATH.

If you install with a **cp** command that is part of a freeware package, the installation completes but the Java 2 SDK that the WebSphere Application Server product installs, which is *install\_root/java*, can have missing files. Some required symbolic links can be destroyed. If you remove the freeware cp command from the PATH, you can install the WebSphere Application Server product successfully.

To verify that the Java 2 SDK that WebSphere Application Server installs is working correctly:

1. Go to *install\_root/java/bin*.
2. Type `./java -version`.

The command completes successfully with no errors when the Java 2 SDK is intact.

- **Ignoring the desktop icon messages from InstallShield for Multiplatforms (ISMP) during installation.**

You can ignore any messages in the log.txt file about not being able to create a desktop icon on Linux and UNIX-based platforms.

For instance, ignore messages that are similar to the following example when installing Network Deployment:

```
Setup.product.install,
 com.installshield.product.actions.DesktopIcon,
 msg1, Unable to find platform specific desktop icon information.
 Could not create desktop icon "Start the deployment manager"
```



```

Setup.product.install,
 com.installshield.product.service.product.PureJavaProductServiceImpl,
 msg1, installing Desktop Icon (StopCellMgrIconBean)
Setup.product.install,
 com.installshield.product.actions.DesktopIcon,
 msg1, Unable to find platform specific desktop icon information.
 Could not create desktop icon "Stop the deployment manager"
Setup.product.install,
 com.installshield.product.service.product.PureJavaProductServiceImpl,
 msg1, installing Desktop Icon (AdminConsoleIconBean)
Setup.product.install,
 com.installshield.product.actions.DesktopIcon,
 msg1, Unable to find platform specific desktop icon information.
 Could not create desktop icon "Administrative Console"
Setup.product.install,
 com.installshield.product.service.product.PureJavaProductServiceImpl,
 msg1, installing Desktop Icon (LogAnalyzerIconBean)
Setup.product.install,
 com.installshield.product.actions.DesktopIcon,
 msg1, Unable to find platform specific desktop icon information.
 Could not create desktop icon "Log Analyzer"

```

- **Avoiding setting WebSphere Application Server environment variables in the user or system profile.**

WebSphere Application Server products use environment variables to control many dynamic settings, such as the path to the product in the \$WAS\_HOME variable, the path to the /logs directory in \$LOG\_ROOT, or the path to the IBM Development Kit in \$JAVA\_HOME. Avoid setting these variables in the system profile or in user profiles. You might have a setting in a profile that overrides the proper setting. If you reset variables that the products use to function, you can produce unpredictable results, or even failure. To view WebSphere Application Server environment variables, open the administrative console and click **Environment > Manage WebSphere Variables**. Scroll down to see the variables. You can also click **Next** and **Previous** to see other pages of variables. Click a variable to see more information, including its description. Other variable settings are described on other administrative console pages.

- **Avoiding spaces in the name of the installation directory on Linux and UNIX platforms.**

Do not use spaces in the installation directory name. Spaces in directory names are not supported on Linux and UNIX.

- **Moving all of the core dump files from the var/sadm/pkg directory.**

The InstallShield for MultiPlatforms (ISMP) Installation wizard iterates through all of the directories in the /var/sadm/pkg directory, assuming that each entry that the wizard finds is a directory. The wizard tries to open the pkginfo file within the directory. The ISMP wizard fails when it cannot find an entry under the core file. Remove the core file from the directory to avoid the problem.

- **Starting the JMS server before running some Samples on UNIX platforms.**

To run Samples that use JMS APIs, you must manually start the JMS server (jmsserver) before running the samples on UNIX platforms.

To start the jmsserver process, complete the following steps:

1. Open the administrative console: <http://localhost:9090/admin>.
2. Click **Servers > Application Servers > server1 > Server Components > JMS Servers**.
3. Change the Initial State from **STOP** to **START**.
4. Save the configuration and log out of the administrative console.
5. Stop and restart the Application Server from the command line. For example,

```
stopServer.sh server1 followed by startServer.sh server1
```

- **Using an option on the install or uninstall command to identify a temporary directory other than the default /tmp directory.**

If the tmp disk does not have a large enough allocation, this message appears:

```
Error writing file = There may not be enough temporary disk space.
```

```
Try using -is:tmpdir to use a temporary directory on a partition with more disk space.
```

Use the `-is:tempdir` installation option to specify a different temporary disk to use. For example, the following command uses the `/swap` file system as a temporary disk during installation:

```
./install -is:tempdir /swap
```

Some space in the `/tmp` directory is required for a copy of the native install launcher program, the Java virtual machine (JVM), and some utility programs. ISMP uses the location you specify with the `-is:tempdir` parameter for its temporary work area.

- **Adding additional name space bindings with a new console user.**

On UNIX platforms, when you click **Name Space Bindings**, you receive an Error 500 message on the browser. When you click **Show Details**, a `NullPointerException` displays. This only occurs with the administrative ID that was configured as the server ID.

To solve the problem, add a different user to **System Management > Console Users** and login with that user to add additional name space bindings.

- **Migrating when the Java 2 SDK that ships with the product is not available.**

If you upgrade the Java 2 SDK that ships with the WebSphere Application Server product, and you intend to use the migration tools to migrate the configuration of a Version 4.0.x installation to Version 5.0.x, you can work around any Java wrapper errors that you might receive.

For example, if you upgrade to Java 2 SDK 1.3.1\_08 and attempt to migrate V4.0.4 to V5.0.2, you receive a Java wrapper not found error when you issue the following command, which is shown on more than one line for clarity:

```
./WASPreUpgrade.sh
/tmp/WAS40Backup
/opt/WebSphere/AppServer
wassun3
-traceString " *=all=enabled"
-traceFile /tmp/PreUpgradeTrace.log
```

The error that you receive is about a Java wrapper not found. Create a soft link to get around the problem:

1. `cd install_root/java/bin/aparc/native_threads`
2. `ln -s java .java_wrapper`

When you call the `.java_wrapper`, the Java 2 SDK assumes that the Java code is called and the migration can continue.

- **A delay occurs before the Installation wizard displays its first panel.**

The default installation method is to open a command window to a read/write directory, issue the fully qualified command to start the Launchpad tool, and click the **Install the product** option. This option launches the Installation wizard, which is an InstallShield for MultiPlatforms (ISMP) application. This action launches the Installation wizard in the language of your machine locale unless no translation exists for your locale, in which case you receive the English version. A short delay occurs before the ISMP wizard displays.

You do not need to click the *Install the product* option more than once to cause the wizard to display the first panel. The delay is particularly noticeable on x-windows platforms.

- **Manually entering Launchpad Web addresses to launch the links directly from your browser.**

A known problem with the Launchpad can prevent it from using Netscape to open the documentation links on some operating system platforms.

If you use the Mozilla browser, some Launchpad links do not work. The Launchpad attempts to call the Netscape browser in the `/usr/bin/netscape` directory. Try a symbolic link to the Mozilla browser to fix the problem as shown in the following example:

```
ln -sf /usr/bin/mozilla /usr/bin/netscape
```

You can also launch the following links directly from your browser:

### Product Overview

<http://www.ibm.com/software/webservers/appserv/>

Your security options might prevent redirection, such as to the <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/appserv> Web site. If so, you can identify the IBM Web site as a trusted site to allow the redirection. For example, on Internet Explorer, add <http://www-306.ibm.com> to the list of Trusted Sites by selecting **Tools > Internet Options > Security (tab) > Trusted Sites**.

### Readme File

`CD-ROM_mount_point/readme/readme_en.html`

### Installation Guide

`CD-ROM_mount_point/docs/InstallGuide_en.PDF`

The *Installation Guide* is in Adobe PDF format. Viewing this document requires you to have Adobe Acrobat Reader, which is a free download from the Adobe Acrobat Web site.

**5.1+** Download a new installation guide at [ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51base\\_gs.pdf](ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51base_gs.pdf).

See “Using the Launchpad to start the installation” on page 173 for more information about the Launchpad program.

- **Using the tcp option to mount CD drives or disk drives in a network file system mount if the option is available.**

Refer to the man pages for the **mount** command to check if the tcp option is available. Using a CD drive or a disk drive mounted with the default mount option (udp) is known to cause errors on some versions of Linux.

Red Hat Enterprise Linux 3.0 UL1 on zLinux platforms is particularly sensitive to this problem, but the recommendation applies to all Linux platforms. Installing with product CDs that are physically located on an NFS server can fail. Symptoms that occur when installing with disk 1 on an NFS server include:

- An indication that the `/tmp/_wbi_/jdk/java/bin/java` file does not exist after issuing the **launchpad.s** command or the **install** command and the installation fails.
- An indication of copy errors in the `/tmp/wbisf_launcher.log` file:

```
cp: cannot stat `~/mnt/linuxs390/WAS/jdk/java/jre/lib/images':
 No such file or directory
cp: cannot stat `~/mnt/linuxs390/WAS/jdk/java/jre/lib/security':
 No such file or directory
cp: cannot stat `~/mnt/linuxs390/WAS/jdk/java/lib':
 No such file or directory
```
- An indication of stale NFS file handle errors when trying to list the files under the NFS-mounted file system:

```
#ls /mnt/linuxs390/WAS/jdk/java/jre/lib
ls.: :stale NFS file handle
```
- An indication of server-not-responding errors in the `/var/log/messages` file on the NFS client:

```
Mar 15 16:29:59 aushat178 kernel:
 nfs: server svtnx4 not responding, still trying
```

Symptoms that occur when installing with disk 2 on the NSF server include:

- After clicking **Next** on the Prompt for disk 2 panel, the Installation wizard fails while copying the update installer application with an error that is similar to the following example:

```
Copying cumulative fix and update installer application failed.
Cumulative fix or interim fixes cannot be applied.
This installation is terminating.
```
- After the failure of the installation of the cumulative fix, there are missing directories in the `install_root/WAS_update` directory. You can verify missing directories by comparing the directory structure to the directory structure on disk 2.

Before installing with an NFS-mounted CD or hard disk, verify that the NFS is working correctly. For example, if your mounted CD is `/mnt`, issue the following command on zLinux to see if you have an NFS problem:

```
unzip -l /mnt/linuxs390/WAS/setup.jar
```

The proper output from the command lists all the files in the setup.jar file without error. If you encounter problems in running the command, verify that your /etc/exports file includes the no\_root\_squash option. Also mount the NFS drive with the -o tcp option. For example:

```
mount -o tcp mycdrom.hostname:/media/cdrom /mnt
```

See <http://www.redhat.com/docs/manuals/enterprise/RHEL-3-Manual/sysadmin-guide/s1-nfs-mount.html> for more information about the -o tcp option.

Contact your system administrator for help if you continue to have NFS problems.

- **Preserving symbolic links when copying product CDs to network file server disks**

When copying a CD for one operating system platform to a network file system (NFS) disk on another type of operating system using the **cp** command, you can encounter errors such as those in the following example:

```
a file is bad
no such file or directory exists
libCSup.2 cannot be accessed
```

The copy error occur when incorrectly copying symbolic link files.

An example of such an error occurs when copying an HP-UX CD image onto an AIX platform with the **cp -frp** command. The default **cp** command behavior on an AIX platform is to resolve the symbolic links by copying the files to which the symbolic links point. Errors occur when a symbolic link resolves to a platform-specific library or file that is not present on the NFS operating system. Use options on the copy command of the NFS system to copy symbolic links instead of resolving them. For example, the -h option on the **cp** command of the AIX platform copies symbolic links from the HP-UX CD to the NFS disk on the AIX platform.

Even with the -h option, the **cp** command on a Solaris platform does not preserve symbolic links when copying an HP-UX disk. On a Solaris platform, use the **tar -cvf** command to copy data from an HP-UX disk and preserve the symbolic links:

1. Insert disk 2 for HP-UX platforms into the drive on the Solaris system.
2. Close the file explorer window if one opens.
3. Open a command window.
4. Change directories:

```
CD /cdrom/cdrom0
```

5. Issue the following command:

```
tar cvf * /workarea/filename.tar
```

6. Change directories:

```
CD /workarea
```

7. Issue the following command:

```
tar -xvf filename.tar
```

Consult the man page for the copy command on the NFS system to understand how the platform supports copying symbolic links.

## AIX platforms

### Summary of tips that apply to AIX platforms

- Providing adequate disk space for installation
- **5.1 +** Unmounting file systems with broken links to avoid java.lang.NullPointerException errors
- Installing the prerequisite Java130.rte.lib Version 1.3.0 on AIX Version 4.3.3 or AIX Version 5
- Correcting directory permissions on AIX platforms before reinstalling WebSphere Application Server with the embedded messaging feature
- A core dump might occur when running WebSphere Application Server with DB2 V7.2 FP8 client in AIX 5.2
- Avoiding a segmentation fault problem when installing or uninstalling

- Avoiding a null pointer exception during the interactive installation of the IBM HTTP Server product on AIX platforms.
- Ignoring DBCS messages when you do not require DBCS support. Otherwise, install the necessary patches.
- Using UNIX line-end characters (0x0D0A) to terminate each line of the options response file for silent installation
- Navigating if the scroll bar disappears on the Feature panel of the Installation wizard
- Ignoring error messages that appear in the log.txt installation log
- Avoiding a potential port conflict between the administrative console and the WebSM system management console on AIX 5.1, with maintenance level 2.
- Canceling the installation and updating your operating system before restarting the installation, if you install on an AIX machine and receive a message that a fileset is missing, such as the X11.fnt.ucs.ttf fileset
- Setting up the display environment for a real root login on AIX systems
- Avoiding a DSAPI filter-loading error when the Lotus Domino Server starts
- Avoiding a null pointer exception during the interactive installation of the IBM HTTP Server on AIX platforms
- Starting IBM HTTP Server 2.0.47 on AIX 5.2 systems
- Using the AIX 5.1 and AIX 5.2 Fast Response Cache Accelerator feature with IBM HTTP Server 2.0.47

### Tips that apply to AIX platforms

- **Providing adequate disk space for AIX platforms.**

Space requirements for the embedded messaging feature are described in the Installing WebSphere embedded messaging as the JMS provider topic.

Table 29. Installed sizes of installation directories for AIX platforms

|                                            | Base product                     | Network Deployment                           | IBM HTTP Server          | Embedded messaging           | Tivoli Global Security Kit | Temp space |
|--------------------------------------------|----------------------------------|----------------------------------------------|--------------------------|------------------------------|----------------------------|------------|
| <b>Installation directory</b>              | /usr/<br>WebSphere/<br>AppServer | /usr/<br>WebSphere/<br>Deployment<br>Manager | /usr/ IBM<br>Http Server | /usr/ mqm and<br>/opt/ wemps | /usr/ ibm/<br>gsk7         | /tmp       |
| <b>Minimum free space for installation</b> | 450 MB                           | 370 MB                                       | 20 MB                    | 17 MB; 40 MB<br>on base      | 32.5 MB on<br>base         | 150 MB     |

- **Unmounting file systems with broken links to avoid java.lang.NullPointerException errors.**

Unmount file systems with broken links before installing on an AIX system. The installation of the cumulative fix pack can fail with the following error when broken links to file systems occur:

An error occurred during wizard bean change notification:

```
java.lang.NullPointerException
 at com.ibm.wizard.platform.aix.AixFileUtils.
 getFileSystemData(AixFileUtils.java:388)
 at com.ibm.wizard.platform.aix.AixFileUtils.
 getPartitionDataWithExecs(AixFileUtils.java:172)
 at com.ibm.wizard.platform.aix.AixFileUtils.
 getPartitionData(AixFileUtils.java:104)
 at com.ibm.wizard.platform.aix.AixFileServiceImpl.
 getPartitionNames(AixFileServiceImpl.java:397)
...

```

Use the **df -k** command to check for broken links to file systems. Look for file systems that list blank values in the *1024-blocks size* column. Columns with a value of "-" (dash) are not a problem. A problem with the /dev/lv00 file system is illustrated in the following example:

```
> df -k
Filesystem 1024-blocks Free %Used Iused %Iused Mounted on
/dev/hd4 1048576 447924 58% 2497 1% /
/dev/hd3 4259840 2835816 34% 484 1% /tmp

```

```

/proc - - - - - /proc
/dev/lv01 2097152 229276 90% 3982 1% /storage
/dev/lv00
/dev/hd2 2097152 458632 79% 42910 9% /usr
iw031864:/cdrom/db2_v72_eee_aix32_sbcs

```

The /proc file system is not a problem. The iw031864:/cdrom/db2\_v72\_eee\_aix32\_sbcs file system is a definite problem. The /dev/lv00 file system is also a likely problem. Use one of the following commands to solve this problem:

```

> umount /cdrom/db2_v72_eee_aix32_sbcs
> umount /cdrom

```

Start the installation again. If the problem continues, unmount any file systems that have blank values, such as the /dev/lv00 file system in the example. If you cannot solve the problem by unmounting file systems with broken links, reboot the machine and start the installation again.

- **Installing the prerequisite Java130.rte.lib Version 1.3.0 on AIX Version 4.3.3 or AIX Version 5.**

On AIX Version 4.3.3 (not supported by WebSphere Application Server V5.1) and AIX Version 5, you must install Java130.rte.lib Version 1.3.0 to verify that the embedded messaging feature installs correctly. To download a copy of Java 1.3.0:

1. Go to [www.ibm.com/java](http://www.ibm.com/java): Java Technology Zone.
2. Go to the bottom of the page: **Most popular links**.
3. Click **IBM Developer Kit for AIX**.
4. Click **Download**.
5. Click **Java 1.3.0** from the **Java Version** column in the table.
6. (Optional) Register for a user ID and password.

To correct an existing problem:

1. Uninstall the following components, if present:
  - WebSphere Application Server
  - WebSphere MQ (MQSeries)
  - WebSphere MQ (MQSeries) Java classes and classes for WebSphere MQ for the Java Message Service
  - IBM HTTP Server
  - WebSphere Embedded Messaging Publishing and Subscribe Edition
2. Edit the vpd.properties file and remove any entry related to WebSphere Application Server. WebSphere Application Server-related entries begin with:
  - **WSB** for the base WebSphere Application Server, V5.0.x product
  - **WSBAA51** for the base WebSphere Application Server, V5.1 product
  - **WSC** for the WebSphere Application Server Application Clients
  - **WSE** for the Enterprise product, V5.0.x
  - **WSN** for the Network Deployment product, V5.0.x
  - **WSNAA51** for the Network Deployment product, V5.1
  - **WSM** for the WebSphere MQ product
3. Reinstall the WebSphere Application Server product.

- **Correcting directory permissions on AIX platforms before reinstalling WebSphere Application Server with the embedded messaging feature**

After manually uninstalling the WebSphere Application Server product and before reinstalling the product with the embedded messaging component, look for the following directory: `/var/mqm/log/WAS_system_name_server_name`. If this directory exists, confirm that the directory is empty and that the user, mqm, can open and write to it. If it is not accessible, the embedded messaging installation program throws the following error:

```

AMQ7064: Log path not valid or inaccessible is written in the
 createMQ.system_name_server_name.log file.

```

If this error occurs, the installation of the embedded messaging component fails.

- **Avoiding a segmentation fault problem when installing or uninstalling.**

If a core dump causes the desktop to disappear while installing or uninstalling the product and leaves only the log in prompt, run the DBX utility on the resulting core file.

If you see the following output, apply the AIX PTF fix for PMR:82869,004 to solve the problem:

```
dbx /usr/bin/X11/X core
Type 'help' for help.
reading symbolic information ...warning: no source compiled with -g
```

```
[using memory image in core]
```

```
Segmentation fault in . at 0x10040aac
0x10040aac (???) 931d0000 stw r24,0x0(r29)
(dbx) where
SetFontPathElements(??, ??, ??, ??) at 0x10040aac
SetFontPath(??, ??, ??, ??) at 0x10042e58
SetFontPath(??, ??, ??, ??) at 0x10042e80
ProcSetFontPath(??) at 0x10015ce4
DeleteClientFromAnySelections(??) at 0x1001e8a8
```

- **Avoiding a null pointer exception during the interactive installation of the IBM HTTP Server product on AIX platforms.**

When installing IBM HTTP Server using the wizard on an AIX platform, you might receive a null pointer Java error on the installation panels when the installer begins to copy files to your machine. This error is caused by an unstable AIX ODM registry.

To work around this problem:

1. Exit the installer.
2. Issue the following command to see if any remaining registry entries for IBM HTTP Server are present:

```
ls|pp -l|grep IHS
```

3. If registry entries for IBM HTTP Server remain, install IBM HTTP Server silently by running the following command:

```
java -jar setup.jar -silent -P ihs.installLocation=the desired installation location
```

**Note:** To remove the installation silently, use the `-silent` parameter on the `uninstall` command. For example, run this command from the IBM HTTP Server installation root:

```
java -jar _uninst/uninstall.jar -silent
```

- **Ignoring DBCS messages or installing necessary patches when you do not require DBCS support.**

While installing any WebSphere Application Server product, you might see the following messages on AIX 5.1 and AIX 4.3.3 (not supported by WebSphere Application Server V5.1) until you install the missing filesets.

Operating system patches of particular concern:

```
Fileset X11.fnt.ucs.ttf_KR was not found on the system.
v5.1.0.0 - Font required for Korean character display
```

```
Fileset X11.fnt.ucs.ttf_TW was not found on the system.
v5.1.0.0 - Font required for Taiwanese character display
```

```
Fileset X11.fnt.ucs.ttf_CN was not found on the system.
v5.1.0.0 - Font required for Chinese character display
```

- **Using UNIX line-end characters (0x0D0A) to terminate each line of the options response file for silent installation.**

During a silent installation on AIX machines, the response file passed to the installer program must not contain ASCII line-end characters (0x0D0A). The response file must contain UNIX line-end characters only. When the options response file contains ASCII line-end characters, the `install` command is unsuccessful and does not log or display an error. To verify the cause of failure, use the Java argument `-Dis.debug=1` on the `install` command. The debug information describes a service exception about invalid characters in the options response file.

- **Navigating if the scroll bar disappears on the Feature panel of the Installation wizard.**

If the scroll bar disappears, use the up and down arrow keys to navigate the features in the Feature panel. Use the tab key to move the focus to the navigation. You can also use the mouse.

- **Ignoring the following error messages that appear in the log.txt installation log.**

You can ignore the following error messages that might appear during installation of the base product in an umbrella installation:

```
WASBase, com.ibm.wizard.platform.aix.AixProductServiceImpl,
 wrn, - WARNING:
Got invalid size of 0 for file:
 /usr/WebSphere/AppServer/
 config/cells/
 BaseApplicationServerCell/
 nodes/
 DefaultNode/
 spi.policy:
WASBase, com.ibm.wizard.platform.aix.AixRegistryServiceImpl,
 wrn, AixRegistryServiceImpl:
Error attempting to modify AIX VPD.
```

- **Avoiding a potential port conflict between the administrative console and the WebSM system management console on AIX 5.1, with maintenance level 2.**

The AIX WebSM system management server listens on port 9090 by default. If you suspect that you have a port conflict, verify it by shutting down WebSphere Application Server. Then issue this command:

```
netstat -an |grep 9090
```

If you get a match, another process is already listening on port 9090. If you want the WebSM server and WebSphere Application Server to coexist, change the WebSphere Application Server administrative console port when installing WebSphere Application Server, or after installation. Although not recommended, you can also disable the WebSM server. To disable the WebSM server, issue this command:

```
/usr/websm/bin/wmsserver -disable
```

The command permanently disables WebSM server startup.

- **Canceling the installation and updating your operating system before restarting the installation, if you install on an AIX machine and receive a message that a fileset is missing, such as the X11.fnt.ucs.ttf fileset.**

Update your operating system if it is missing required filesets. If you receive a message that a fileset is missing, such as the X11.fnt.ucs.ttf fileset, cancel the installation, update the operating system, and restart the installation.

- **Setting up the display environment for a real root login on AIX systems.**

In a normal root login, issue the command `su`. For a real root login, issue the command `su -`.

Display settings for a normal root login are automatic. For a real root login, you must set your display environment properly to successfully view the Installation wizard. Otherwise, you see a message about Preparing Java(tm) Virtual Machine..., and seven rows of dots, but no Installation wizard and no further messages. Refer to the documentation for your AIX platform to determine proper display settings.

- **Avoiding a DSAPI filter-loading error when the Lotus Domino Server starts.**

On a UNIX-based operating system, if you start the Lotus Domino Web server from a non-root user, you are likely to generate a DSAPI filter-loading error:

```
Error loading DSAPI filter.
```

```
Filter not loaded: /usr/WebSphere/AppServer/bin/libdomino5_http.a
```

Manually change the WebSphere Application Server bin directory permissions from 750 to 755 to run Lotus Domino Server as a non-root user and not generate the error. This change does, however, pose a security risk.

You must also change permissions on the WebSphere Application Server logs directory to 777 to let Lotus Domino Server write to the log.



If the Lotus Domino Server is started as root, the problem does not occur.

- **Avoiding a null pointer exception during the interactive installation of the IBM HTTP Server on AIX platforms.**

A null pointer exception is thrown during the interactive installation of IBM HTTP Server on AIX systems. When installing IBM HTTP Server as a standalone installation through the administrative console on the AIX platform, you can receive a null pointer Java error on the installation panels, when the installer begins to copy files to your machine.

This situation is caused by an unstable AIX Object Data Manager (ODM) registry.

To work around this problem:

1. Exit the installer.
2. Issue the `lslpp -l |grep IHS` command to check if there are remaining IBM HTTP Server registry entries that cannot exist.
3. If there are remaining IBM HTTP Server registry entries, do a silent installation of IBM HTTP Server by running the following command:

```
java -jar setup.jar -silent -P ihs.installLocation=<the desired install location>
```

**Note:** To remove the installation silently, add the `-silent` parameter to the regular **uninstall** command from the IBM HTTP Server installation location, for example:

```
java -jar _uninst/uninstall.jar -silent
```

- **Starting IBM HTTP Server 2.0.47 on AIX 5.2 systems.**

A potential IBM HTTP Server 2.0.47 startup failure exists on AIX 5.2 systems when a non-default AIX resolver configuration is used (for example, in `/etc/netsvc.conf`), and the IBM HTTP Server configuration file (for example, `httpd.conf`), uses `Listen` directives that do not specify an Internet Protocol address. The following example shows error messages issued by IBM HTTP Server:

```
[crit] (78) A remote host did not respond within the timeout period.:
 alloc_listener: failed to set up sockaddr for ::
Syntax error on line 130 of /usr/IBMIHS/conf/httpd.conf
Listen setup failed.
```

This error is caused by a defect in the AIX resolver library.

The defect is fixed by APAR IY40908 for AIX 5.2. The fix is shipped with AIX52B Gold and later.

Alternatively, you can take one of the following actions to solve the problem:

- Use the IPv4-specific `Listen` directive in the IBM HTTP Server configuration file, for example, `"Listen 0.0.0.0:80"`
- Do not override the default AIX resolver configuration.
- Change the following line in the `/etc/netsvc.conf` file from:

```
hosts = local = auth , bind
to
hosts = local = auth , bind , local
```

- **Using the AIX 5.1 and 5.2 Fast Response Cache Accelerator feature with IBM HTTP Server 2.0.47.**

IBM HTTP Server 2.0.47 supports the AIX 5.1 and 5.2 Fast Response Cache Architecture (FRCA) feature through the `mod_afpa_cache.so` plug-in module. However, you must install the interim fix IY49388 for AIX 5.2. IBM HTTP Server 1.3.28 does not support the AIX FRCA feature.

## HP-UX platforms

### Summary of tips that apply to HP-UX platforms

- Providing adequate disk space for installation
- Installing required patches for correct character display of the installation panel on Asian language platforms
- Ignoring a prerequisite checker error for a patch when a superset of the patch has been installed.

- **5.1+** Uninstalling a V5.1 instance on HP-UX platforms before reinstalling the instance
- Mounting the product CD-ROM
- Configuring HP-UX kernel settings before installing WebSphere Application Server
- Migrating or coexisting with an existing WebSphere Application Server node that HP-UX does not recognize
- Avoiding using Netscape 4.79 on HP-UX 11 in Japanese, to avoid problems in viewing the administrative console
- Configuring the converter.properties file to use EUC-JP (Japanese) encoding on HP-UX
- Using certificate revocation lists (CRLs) with IBM HTTP Server on HP-UX and Linux for S/390 platforms is not supported at this time
- Ignoring certain error messages when installing WebSphere Application Server with embedded messaging from the CD-ROM

### Tips that apply to HP-UX platforms

- **Providing adequate disk space for HP-UX platforms.**

Table 30. Installed sizes of installation directories for HP-UX platforms

|                                            | Base product                     | Network Deployment                           | IBM HTTP Server          | Tivoli Global Security Kit | Temp space      |
|--------------------------------------------|----------------------------------|----------------------------------------------|--------------------------|----------------------------|-----------------|
| <b>Installation directory</b>              | /opt/<br>WebSphere/<br>AppServer | /opt/<br>WebSphere/<br>Deployment<br>Manager | /opt/ IBM Http<br>Server | /opt/ ibm/ gsk7            | /tmp            |
| <b>Minimum free space for installation</b> | 539 MB                           | 539 MB                                       | 18.6 MB                  | 15.9 MB                    | At least 100 MB |

Space requirements for the embedded messaging feature are described in the Installing WebSphere embedded messaging as the JMS provider topic.

- **Installing required patches for correct character display of the installation panel on Asian language platforms.**

The installation panel displays garbled characters on Asian language platforms, including Korean, traditional Chinese, Japanese, and simplified Chinese, for the HP-UX 11i operating system.

Verify that the following patches are installed on your HP-UX 11i machine:

| Name                                          | Description                                                 |
|-----------------------------------------------|-------------------------------------------------------------|
| B6848BA                                       | Ximian GNOME 1.4 GTK+ Libraries for HP-UX 11.00 and 11i     |
| BUNDLE11i                                     | Required Patch Bundle for HP-UX 11i, February 2001 or later |
| PHSS_26974 (for 11.0) or PHSS_26973 (for 11i) | Korean TrueType fonts                                       |
| PHSS_24937 (for 11.0) or PHSS_26977 (for 11i) | Traditional Chinese TrueType fonts                          |
| PHSS_26972(for 11.0) or PHSS_26971 (for 11i)  | Japanese TrueType fonts                                     |
| PHSS_26976(for 11.0) or PHSS_24975 (for 11i)  | Simplified Chinese TrueType fonts                           |
| PHSS_25091(for 11.0) or PHSS_25092 (for 11i)  | Common TrueType fonts                                       |

You can find these patches at the following URL:

<http://www.hp.com/products1/unix/java/infolibrary/patches.html#1>.

- **Ignoring a prerequisite checker error for a patch when a superset of the patch has been installed.**

If you apply a superset patch that includes a specific patch that the prerequisite checker program requires, you can ignore the message from the installer program and continue the installation. You can always continue the installation whenever you see an error message about a missing prerequisite. You can ignore the messages and continue the installation. Although the list of prerequisite patches that the installer program checks is extensive, the list does not include all of the superset patches.

For example, the installer program checks for patch PHCO\_29109. The HP-UX Web site recommends that you install a superset patch, PHCO\_29960, instead of PHCO\_29109. After installing the patch and rebooting the machine, reinstalling the base WebSphere Application Server results in the same message about the missing PHCO\_29109 prerequisite. You can ignore the error because you know that you have installed a superset patch.

- **5.1+ Uninstalling a V5.1 instance on HP-UX platforms before reinstalling the instance.**

If you uninstall a V5.1 instance, but leave the embedded messaging feature installed because of other WebSphere Application Server instances on the machine, rename the `/opt/WebSphere/AppServer51` directory so that HP-UX cannot locate the earlier installation that is still registered. Reinstall into a directory other than the `/opt/WebSphere/AppServer51` directory or the renamed directory. Use a new directory name for reinstalling the V5.1 instance.

If you uninstall every WebSphere Application Server product on the machine and are now reinstalling, follow the procedure in “Manually uninstalling on HP-UX platforms” on page 296 instead.

For example, assume that V5.0.2 exists in the `/opt/WebSphere/AppServer502` directory with the embedded messaging server and client feature. You then install V5.1 to the `/opt/WebSphere/AppServer51` directory, but later you uninstall the product with the **uninstall** command and select to leave the embedded messaging feature installed. And now you decide to reinstall V5.1.

Rename the `/opt/WebSphere/AppServer51` directory to the `/opt/WebSphere/AppServer51_bk` directory before reinstalling V5.1 into the `/opt/WebSphere/AppServer51New` directory.

This step is required before you can reinstall the product onto the same system. To correct the registry entries and uninstall all of the WebSphere Application Server products on the machine, follow the procedure in “Manually uninstalling on HP-UX platforms” on page 296 instead.

- **Configuring HP-UX kernel settings before installing.**

To set kernel parameters, perform the following steps:

1. Log into the host machine with superuser (root) privileges.
2. Determine the physical memory to avoid setting certain kernel parameters above the physical capacity:
  - a. Start the HP-UX System Administration Manager (SAM) utility.
  - b. Click **Performance Monitors > System Properties > Memory**.
  - c. Note the value for Physical Memory and click **OK**.
  - d. Exit from the SAM utility.
3. Set the `maxfiles` and `maxfiles_lim` parameters to at least 4096. (The following table recommends 8000 and 8196, respectively. You must first edit the `/usr/conf/master.d/core-hpux` file, to allow the SAM utility to set values greater than 2048:
  - a. Open the file `/usr/conf/master.d/core-hpux` in a text editor.
  - b. Change the line, `*range maxfiles<=2048` to `*range maxfiles<=60000`
  - c. Change the line, `*range maxfiles_lim<=2048` to `*range maxfiles_lim<=60000`
  - d. Save and close the file. Old values might be stored in the `/var/sam/boot.config` file. Force the SAM utility to create a new `boot.config` file:
    - 1) Move the existing version of the `/var/sam/boot.config` file to another location, such as the `/tmp` directory.
    - 2) Start the SAM utility.
    - 3) Click **Kernel Configuration > Configurable Parameters**. When the Kernel Configuration window opens, a new `boot.config` file exists.

Alternatively, rebuild the `boot.config` file with the following command:

```
/usr/sam/sbin/getkinfo -b
```

4. Set new kernel parameter values:
  - a. Start the SAM utility.

- b. Click **Kernel Configuration > Configurable Parameters**.
- c. For each of the parameters in the following table, perform this procedure:
  - 1) Highlight the parameter to change.
  - 2) Click **Actions > Modify Configurable Parameter**.
  - 3) Type the new value in the **Formula/Value** field.
  - 4) Click **OK**.

Some kernel values for WebSphere Application Server with the embedded messaging feature are higher than those shown in the following table. See the Installing WebSphere embedded messaging as the JMS provider topic for more information.

Some kernel values for WebSphere Application Server IBM DB2 on the same machine, are higher than those shown in the following table.

- Recommended HP-UX kernel configuration parameters for DB2 Version 8
- Recommended HP-UX kernel configuration parameters for DB2 V7

Typical kernel settings for running WebSphere Application Server appear in the following table:

| Parameter       | Value                                                                         |
|-----------------|-------------------------------------------------------------------------------|
| dbc_max_pct     | 25                                                                            |
| maxdsiz         | 805306358                                                                     |
| maxdsiz         | 2048000000 (when the base and Network Deployment products are on one machine) |
| maxfiles_lim    | 8196 (Change this setting before changing the maxfiles setting.)              |
| maxfiles        | 8000                                                                          |
| maxssiz         | 8388608                                                                       |
| maxswapchunks   | 8192                                                                          |
| max_thread_proc | 3000                                                                          |
| maxuprc         | 512                                                                           |
| maxusers        | 512                                                                           |
| msgmap          | 2048                                                                          |
| msgmax          | 65535                                                                         |
| msgmax          | 131070 (when the base and Network Deployment products are on one machine)     |
| msgmnb          | 65535                                                                         |
| msgmnb          | 131070 (when the base and Network Deployment products are on one machine)     |
| msgmni          | 50                                                                            |
| msgseg          | 32767                                                                         |
| msgssz          | 32                                                                            |
| msgtql          | 2046                                                                          |
| nfile           | 58145                                                                         |
| nflocks         | 3000                                                                          |
| ninode          | 60000                                                                         |
| nkthread        | 7219                                                                          |
| nproc           | 4116                                                                          |
| npty            | 2024                                                                          |
| nstrpty         | 1024                                                                          |
| nstrtel         | 60                                                                            |

| Parameter | Value                           |
|-----------|---------------------------------|
| sema      | 1 (with embedded messaging)     |
| semaem    | 16384 (with embedded messaging) |
| semmap    | 514                             |
| semmni    | 2048                            |
| semmns    | 16384 (with embedded messaging) |
| semmnu    | 1024                            |
| semume    | 200                             |
| semvmx    | 32767 (with embedded messaging) |
| shmmax    | 2147483647                      |
| shmem     | 1 (with embedded messaging)     |
| shmmni    | 1024                            |
| shmseg    | 1024                            |
| STRMSGSZ  | 65535                           |

5. click **Actions > Process New Kernel**.

6. Click **Yes** on the information window to confirm your decision to restart the machine.

Follow on-screen instructions to restart your machine and to enable the new settings.

7. If you plan to redirect displays to non-HP machines, do the following steps before running the WebSphere Application Server Installation wizard:

a. Issue the following command to obtain information on all of the public locales accessible to your application:

```
locale -a
```

b. Choose a value for your system from the output that is displayed and set the LANG environment variable to this value. Here is an example command that sets the value of LANG to en\_US.iso88591

```
export LANG=en_US.iso88591
```

- **Migrating or coexisting with an existing WebSphere Application Server node that HP-UX does not recognize.**

In some cases, the InstallShield for MultiPlatforms (ISMP) program does not detect a previously installed version of WebSphere Application Serve because of a failure to read the registry keys on HP-UX.

You can force the Migration panel or the Coexistence panel to display, by starting the installation with an option on the **Install.exe** or **install** command. For example, use this command:

```
./install -W previousVersionDetectedBean.previousVersionDetected="true"
```

You can also force the display of the Coexistence panel to change conflicting port number assignments. For example, force the Coexistence panel to display using this command:

```
./install -W coexistenceOptionsBean.showCoexistence="true"
```

On either panel, identify the location of the existing product instance that is not being recognized.

- **Avoiding using Netscape 4.79 on HP-UX 11 in Japanese, to avoid problems in viewing the administrative console.**

It is possible that you cannot view the menu or areas of the console that scroll. Only certain portions of the administrative console are visible. You cannot scale the administrative console.

The workaround for the problem is to use another supported browser or another browser and operating system platform. You can also open the menu frame in a separate window, to avoid the problem.

- **Configuring the converter.properties file to use EUC-JP (Japanese) encoding on HP-UX.**

The `install_root/java/jre/lib/i18n.jar` file on HP-UX platforms does not have the converters for Cp33722C, but does have the converter for Cp33722. To use EUC-JP encoding on HP-UX platforms, change the EUC-JP=Cp33722C entry in the `converter.properties` file to EUC-JP=Cp33722 or EUC-JP=EUC\_JP.

- **Using certificate revocation lists (CRLs) with IBM HTTP Server on HP-UX and Linux for S/390 platforms is not supported at this time.**

Certificate revocation lists are not supported for IBM HTTP Server on HP-UX and Linux for S/390 platforms.

- **Ignoring certain error messages when installing WebSphere Application Server with embedded messaging from the CD-ROM.**

The following warning messages are issued despite successful installation. You might see the following messages in the `mq_install.log` file when installing WebSphere Application Server with embedded messaging from the CD-ROM:

```
MQSERIES.MQM-CL-HPUX: A test command parameter is not valid." and "*
"hostname:/cdrom/hpux/messaging/images": Cannot open the
 logfile on this target or source. Possibly the media is
 read-only or there is a permission problem."
```

You can ignore the warning messages. The problem does not affect the successful installation of WebSphere Application Server with embedded messaging.

## Linux platforms

### Summary of tips that apply to Linux platforms

- Providing adequate disk space for installation
- **5.1 +** Providing prerequisites for Red Hat Enterprise Linux V3.0
- **5.1 +** Upgrading Red Hat Enterprise Linux V3.0 to service level 1 (update 1)
- Preparing Linux for S/390 for better performance
- Changing the locale encoding to English to install WebSphere Application Server on SuSE SLES 8.0 for PowerPC platforms
- Japanese not recommended on Red Hat Enterprise Linux (RHEL) Advanced Server Version 2.1
- **5.1 +** Ignoring YAST2 dependency conflicts for embedded messaging packages
- Preparing the SuSE Linux Enterprise Server 8.0 - Powered by UnitedLinux 1.0 platform for WebSphere Application Server installation
- Configuring Apache 1.3 to run on SuSE Linux Enterprise Server 8.0
- Correct font problems on SuSE Linux Enterprise Server 8.0 in Simplified Chinese and Traditional Chinese locales
- Avoiding the certificate revocation list (CRL) function, which is not supported for IBM HTTP Server on Linux for S/390
- Using the `ikeycmd` command line interface for `ikeyman` from IBM HTTP Server on Linux for S/390
- Avoiding utility hangs and accessing the deployment manager
- Using the GNOME desktop on Red Hat Linux Advanced Server V2.1 in the Korean locale
- Upgrading Red Hat Enterprise Linux V3.0 to service level 1 (update 1)
- Changing the Chinese locale on a Red Hat Enterprise Linux 3.0 platform from `zh_CN.gb18030` to `zh_CN.gbk`
- Accessing First Steps items on Linux for S/390 systems
- Using certificate revocation lists (CRLs) with IBM HTTP Server HP-UX and Linux for S/390 platforms is not supported at this time
- Editing the `keyman` script to bring up `IKEYMAN` for IBM HTTP Server 1.3.28 shipped with WebSphere Application Server
- Ignoring messages from the prerequisites checker when your operating system is at the level specified by the WebSphere Application Server supported hardware, software, and APIs Web site
- Using the LinuxThreads threading mode instead of the Native POSIX Threading Library (NPTL)

- Avoiding an error that occurs when starting the Launchpad program for WebSphere Application Server clients, Version 5.1 using the Konqueror file in the K Desktop Environment on SuSE Linux Enterprise Server (SLES)
- Avoiding a segmentation fault on RHEL V3.0 Update 1

### Tips that apply to Linux platforms

- **Providing adequate disk space on Linux platforms.**

Table 31. Installed sizes of installation directories for Linux platforms

|                                                              | Base product               | Network Deployment                  | IBM HTTP Server      | Embedded messaging        | Tivoli Global Security Kit | Temp space      |
|--------------------------------------------------------------|----------------------------|-------------------------------------|----------------------|---------------------------|----------------------------|-----------------|
| <b>Installation directory</b>                                | /opt/ WebSphere/ AppServer | /opt/ WebSphere/ Deployment Manager | /opt/ IBM Http Sever | /opt/ wemps and /opt/ mqm | /opt/ ibm/ gsk7            | /tmp            |
| <b>Minimum free space for Linux for xSeries installation</b> | 470 MB                     | 390 MB                              | 20.4 MB on base      | 6 MB; 60 MB on base       | 11.8 MB                    | At least 150 MB |
| <b>Minimum free space for Linux for S/390 installation</b>   | 470 MB                     | 390 MB                              | 20.4 MB on base      | 6 MB; 60 MB on base       | 13.2 MB                    | At least 150 MB |

Space requirements for the embedded messaging feature are described in Installing WebSphere embedded messaging as the JMS provider .

- **5.1+ Providing prerequisites for Red Hat Enterprise Linux V3.0**

A known limitation in the prerequisites checker program prevents correctly examining prerequisite packages on Linux platforms. If you plan to install WebSphere Application Server Version 5.1.x products on a Red Hat Enterprise Linux 3.0 system, install the following prerequisite packages on the RHEL 3.0 system.

You must install the following packages on Red Hat Enterprise Linux V3.0 for xSeries platforms:

- compat-gcc-7.3-2.96.122
- compat-libstdc++-7.3-2.96.122
- compat-libstdc++-devel-7.3-2.96.122
- compat-glibc-7.x-2.2.4.32.5
- compat-gcc-c++-7.3-2.96.122
- compat-db-4.0.14-5
- rpm-build-4.2.1-4.2

The following package is required for Red Hat Enterprise Linux V3.0 for pSeries platforms:  
rpm-build-4.2.1-4.2.

You must install the following packages on Red Hat Enterprise Linux V3.0 for S/390 (z/OS and S/390 in 31-bit mode) platforms:

- compat-db-4.0.14-5
- compat-pwdb-0.62-3
- compat-libstdc++-7.2-2.95.3.77
- rpm-build-4.2.1-4.2

You can install a later release of any of these packages.

- **5.1+ Upgrading Red Hat Enterprise Linux V3.0 to service level 1 (update 1).**

Upgrade RHEL 3.0 to service level 1 by downloading and installing the service updates from Red Hat.

If you do not upgrade to service level 1, certain national language issues in the released version of GLIBC included in Red Hat Enterprise Linux V3 might cause Java virtual machine (JVM) failures due to segmentation faults. These failures can occur during installation when the locale is set to anything other than an English locale.

The prerequisites checker program for the base V5.1.0.0 product does not recognize Red Hat Enterprise Linux 3.0 as a supported operating system. If you receive a message from the prerequisites checker program that RHEL 3.0 is an unsupported operating system, ignore the message and continue the installation if you installed all prerequisite packages from the previous step and downloaded and installed the service level 1 update.

- **Preparing Linux for S/390 for better performance**

Linux for S/390 (which refers to the Linux distributions available from Linux distributors that run on IBM eServer zSeries and S/390 systems in 31-bit mode) provides a configuration technique that affects the installation and run time performance of WebSphere Application Server. The technique configures the environment where the Linux image runs to use swap space efficiently. Some performance guidelines recommend running Linux with the VM/ESA or z/VM swap turned off because of VM/ESA or z/VM virtualization of hardware. Virtualization can produce double-swapping situations where VM/ESA or z/VM swaps storage and Linux also swaps storage, which degrades performance.

Excessive swapping affects the performance of WebSphere Application Server, which might require 200 MB when all of the Sample applications are loaded. On a system without swap space configured for use, and with a relatively small amount of memory (such as 256 MB), WebSphere Application Server might encounter problems obtaining enough free memory to work properly, particularly when competing for resources against other applications and products that run in the Linux environment.

The solution is to disable swapping in Linux, but to enable swapping in VM/ESA or z/VM. You can increase performance by letting VM/ESA or z/VM handle the swapping. Double or triple the specification for physical memory for the Linux image. For example, if the physical memory allocation as seen by the Linux image is 256 MB, disable swap in Linux, enable swap in VM/ESA or z/VM, and increase the physical memory specification as seen by Linux to 512 MB or 768 MB. This amount of memory handles any large spikes in application memory usage that might occur.

You can fine tune the amount of physical memory to allocate to each Linux guest operating system. Size the JVM heap size for the application running in the Application Server, add 90 MB to that amount for the Application Server, 20 MB for Linux, and another 10-20% to handle peak usage. This measurement provides better memory management from a VM/ESA or z/VM perspective.

Avoid socket timeout exceptions (SocketTimeoutExceptions) when running WebSphere Business Integration Server Foundation on a Network Deployment node on a Linux for S/390 guest operating system on z/VM or VM/ESA. The exceptions are an indicator that too many processes are running and that the Linux system is being overloaded. If the deployment manager is under a heavy load, 1 GB of memory is required. In addition, move all base nodes to other Linux for S/390 guests to allow the deployment manager to run by itself on the Linux for S/390 system.

See the Linux on IBM eserver zSeries and S/390: ISP/ASP Solutions IBM Redbook and the Performance Analysis for Java Web sites book for more information.

- **Changing the locale encoding to English to install WebSphere Application Server on SuSE 8.0 SLES for PowerPC platforms.**

The WebSphere Application Server installation program is corrupted when you start the installation on SuSE Linux Enterprise Server (SLES) 8.0 for a PowerPC platform in a double-byte character set (DBCS) language. The double-byte character set languages display the characters incorrectly. The single-byte character set (SBCS) languages display the characters correctly.

The reason for this problem is either missing fonts on the system, incorrectly called fonts, or incorrectly displayed fonts.

Change the locale encoding to English to properly install WebSphere Application Server on SuSE SLES 8.0 for PowerPC platforms. To change the locale, open a terminal or window and complete the following steps:

1. Type `export LC_ALL=C` and press **Enter**.



2. Type `export LANG=C` and press **Enter**.
  3. To verify that the locale is properly set to English, type `locale` to see that all variables are set to C.
- **Japanese not recommended on Red Hat Enterprise Linux (RHEL) Advanced Server Version 2.1.**  
The Red Hat Advanced Server Version 2.1 GUI language selection menu displays two selections for Japanese:
    - Japanese (eucJP)
    - Japanese (SJIS)
 Click **Japanese (eucJP)** to install and run the IBM WebSphere Application Server in a Japanese environment. Do not use Japanese (SJIS).  
See the Red Hat Japanese Web site at <http://www.jp.redhat.com/support/7.2/sla/rh72faq.html> for more information.

- **5.1+ Ignoring YAST2 dependency conflicts for embedded messaging packages**

The YAST2 utility on SuSE Linux Enterprise Server 8.0 has a known problem reporting dependency conflicts. If you use the utility, you can ignore the Dependency conflict panel that reports conflicting packages for the embedded messaging feature. Two examples from such a report might appear like this:

```
MQSeriesClient
MQSeriesClient-U486878
```

```
MQSeriesJava
MQSeriesJava-U486878
```

Many packages, such as the embedded messaging packages in the WebSphere Application Server products, specify a conflict to control incompatible package levels. In this case, the conflict identifies a WebSphere MQ-related package level that is less than 5.3.0. YAST2 seems to be misinterpreting the specification and flagging a conflict for WebSphere MQ packages that are less than *or equal to* 5.3.0.

- **Preparing the SuSE Linux Enterprise Server 8.0 - Powered by UnitedLinux 1.0 operating platform for WebSphere Application Server installation.**

1. Install SP2 for the UnitedLinux 1.0 operating platform to let you use the WebSphere Application Server LaunchPad.

It is your responsibility to install this service pack. No definitive way is available for the `prereqChecker` function of the installer to detect service pack versions on UnitedLinux. Kernel `uname`/versions between 8.0 and 8.0.2 are identical. `signature RPM` exists for denoting a service pack installation.

2. Use the IBM Developer Kit that WebSphere Application Server provides to support the Java 2 SDK on the SuSE SLES 8.0 operating system to avoid potential problems when uninstalling an interim fix or a fix pack. To use the IBM Developer Kit, remove the `java2-jre-1.3.1-524` and `java2-1.3.1-524` RPMs from the machine before installing WebSphere Application Server.

- **Configuring Apache 1.3 to run on SuSE Linux Enterprise Server 8.0.**

Using the following directive in an otherwise unmodified `httpd.conf` file can result in an error:

```
LoadModule app_server_http_module /opt/WebSphere/AppServer/bin/mod_app_server_http.so
```

The error is indicated by a message similar to the following example:

```
[warn] Loaded DSO /opt/WebSphere/AppServer/bin/mod_app_server_http.so uses
plain Apache 1.3 API, this module might crash under EAPI!
(please recompile it with -DEAPI)
[notice] Apache/1.3.26 (Linux/SuSE) mod_python/2.7.8 Python/2.2.1
PHP/4.2.2 mod_perl/1.27 configured -- resuming normal operations
[notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
[notice] Accept mutex: sysvsem (Default: sysvsem)
[notice] child pid 3383 exit signal Segmentation fault (11)
```

1. Edit the `/etc/sysconfig/apache` file and change the value of the following parameters from **yes** to **no** to remove PHP and Python from the Web server:
  - `HTTPD_SEC_MOD_PYTHON`

- HTTPD\_SEC\_MOD\_PHP

2. Run the SuSEconfig utility to enable your changes.

Do not attempt to use the EAPI module with the version of Apache 1.3 that ships with SuSE Linux Enterprise Server 8.0. An error in this version of Apache prevents using the EAPI module. Obtain an updated version of Apache 1.3 from SuSE, or download an updated version of Apache 1.3 from another source to use the EAPI plug-in module.

Edit the `/etc/sysconfig/apache` file if you download a new version, remove PHP and Python from the Web server, and run the SuSEconfig utility.

- **Correct font problems on SuSE Linux Enterprise Server 8.0 in Simplified Chinese and Traditional Chinese locales.**

On the Linux for Power platform that SuSE Linux Enterprise Server 8.0 provides, there is a font problem caused by a missing package. There is no `ttf-hanyi` package installed during the normal product installation of the SuSE 8.0 operating system. This causes the Installation wizard for WebSphere Application Server products to display garbled characters on the Simplified Chinese locale and the Traditional Chinese locale.

Copy the `ttf-hanyi-2021016-0.noarch.rpm` package on the SuSE 8.0 for i386 CD-ROM to the Power PC system. Install the package on the Power PC machine and reboot the machine to solve the problem.

- **Avoiding the certificate revocation list (CRL) function, which is not supported for IBM HTTP Server on Linux for S/390.**

Do not use the certificate revocation list (CRL) function on Linux for S/390 at this time.

- **Using the `ikeycmd` command line interface for `ikeyman` from IBM HTTP Server on Linux for S/390.**

You might see a Java core dump after running an `ikeyman` command function, such as creating the stash file. This error occurs on both RedHat and SuSE releases as the result of a conflict in library routines caused by the default loading sequence.

To work around this problem, set the `LD_PRELOAD` environment variable before running the following command:

```
LD_PRELOAD=/usr/lib/libstdc++-libc6.1-2.so.3
```

This command loads the library first when the application is started. Setting this environment variable is also necessary to support Secure Sockets Layer (SSL) on Linux for S/390.

- **Migrating from WebSphere Application Server, Version 3.5 on Linux platforms.**

The error can occur when the migration tools cannot find the `JAVA_HOME`. The **WASPreUpgrade** command reports the error while backing up the WebSphere Application Server V3.5 environment. The error appears in the `WASPreUpgrade` log as:

```
MIGR0257E:Environment variable JAVA_HOME was not set is generated
```

To work around the problem:

1. Create a directory link to one of the following directories:

- `$WAS_HOME/IBMJavaLink`
- `/opt/IBMJava2-122`
- `$WAS_HOME/IBMJava2-122`

`WAS_HOME` is the WebSphere Application Server 3.5 `WAS_HOME`.

2. Run the **WASPreUpgrade** command again, followed by the **WASPostUpgrade** command, to migrate the environment correctly to WebSphere Application Server, Version 5.

- **Avoiding utility hangs and accessing the deployment manager.**

The default Red Hat installation creates an association between the host name of the machine and the loopback address, 127.0.0.1. In addition, the `/etc/nsswitch.conf` file is set up to use the `/etc/hosts` file before trying to look up the server using a name server (DNS). This loopback processing can hang utilities that start and stop a server, such as the **startServer** command, the **stopManager** command, and others, even though the server might successfully start or stop.

The stop can also cause failures when adding nodes on a Linux deployment manager. If you have trouble synchronizing new nodes with the Linux deployment manager during addNode processing, ensure that the host name is defined properly. The default configuration has local host defined in the /etc/hosts file. The default /etc/nsswitch.conf looks only at the host file and not the DNS server. To correct this problem, remove the 127.0.0.1 mapping to localhost in the /etc/hosts file or edit the name service configuration (/etc/nsswitch.conf) file to resolve the proper host name by using the name server.

For example, remove the 127.0.0.1 mapping from the /etc/hosts file, which might look like the following example:

```
IP Address name of machine
n.n.n.n hostname.domain.com hostname
127.0.0.1 localhost
```

Otherwise, change the etc/nsswitch.conf file to search DNS before searching the hosts file.

For example, hosts : dns files

- **Using the GNOME desktop on Red Hat Linux Advanced Server V2.1 for i386 Update3 in the Korean locale.**

You cannot start the deployment manager on a Network Deployment system when Korean characters are in the cell name. This is true in the Korean locale when running the KDE desktop on a Red Hat Linux Advanced Server V2.1 for i386 Update3 system.

This problem is caused by the default system desktop being set to "KDE", which means that file /etc/sysconfig/desktop has the value desktop="KDE".

Run the following command on your RedHat 2.1 system to change the default system desktop to GNOME:

```
#echo "DESKTOP=\"GNOME\"" >/etc/sysconfig/desktop
```

Restart your system to solve the problem.

- **Upgrading Red Hat Enterprise Linux V3.0 to service level 1 (update 1).**

Upgrade RHEL 3.0 to service level 1 by downloading and installing the service updates from Red Hat.

If you do not upgrade to service level 1, certain national language issues in the released version of GLIBC included in Red Hat Enterprise Linux V3 might cause Java virtual machine (JVM) failures due to segmentation faults. These failures can occur during installation when the locale is set to anything other than an English locale.

- **Changing the Chinese locale on a Red Hat Enterprise Linux 3.0 platform from zh\_CN.gb18030 to zh\_CN.gbk.**

A known problem with RHEL V3 before installing service level 1 (update 1) is that one of the Chinese locales and its codepage, zh\_CN.gb18030, causes the Installation wizard to render several input fields uneditable. Change the locale and its codepage to zh\_CN.gbk to allow the Installation wizard to work correctly.

- **Accessing First Steps items on Linux for S/390 systems.**

When you select the following features from the First Steps panel, the Web browser window does not open:

- WebSphere information center
- Register the Product
- Samples Gallery
- Administrative Console

To access these features:

1. Open a Web browser window on another machine.
2. Type the browser Web address that displays in the First Steps status window into the address field of the Web browser.
3. Press **Enter** to open the page in the Web browser.

- **Editing the keyman script to bring up IKEYMAN for IBM HTTP Server 1.3.28 shipped with WebSphere Application Server**

Editing the `ikeyman` script to bring up IKEYMAN for the IBM HTTP Server 1.3.28 shipped with WebSphere Application Server

You cannot bring up IKEYMAN by typing the `ikeyman` script under the `IHS_HOME/bin` directory for IBM HTTP Server 1.3.28 shipped with WebSphere Application Server, Version 5.1 on the Linux PowerPC 64 bit platform. The following error is thrown:

```
Line 139: /usr/local/ibm/gsk7_gcc295/bin/gsk7ibm_gcc295: No such file or directory
```

To work around this problem, perform the following steps:

1. Use WebSphere Application Server IKEYMAN to create the CMS keystore. You can invoke the WebSphere Application Server IKEYMAN from the `WAS_HOME/bin` directory.
  2. Edit the IBM HTTP Server `ikeyman` script: remove 2 references of "gcc295" from next to the last line.
- **Ignoring messages from the prerequisites checker when your operating system is at the level specified by the WebSphere Application Server supported hardware, software, and APIs Web site.**

The installation prerequisites checker checks for prerequisites that might have changed since the release of the product CD.

The prerequisites checker for the installation program is fixed at the time the product CDs are created. It is possible that more operating system patches are required since the time the product CDs were created. It is also possible that new versions of operating systems are supported even though the prerequisites checker identifies the new operating system level as unsupported.

Always consult the IBM WebSphere Application Server supported hardware, software, and APIs Web site to determine whether your operating system is supported when you receive a message from the prerequisites checker program during installation.

The Web site lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. After confirming that your operating system is supported and that you have installed all the necessary patches, you can click **Next** to continue an installation when you receive an error message from the `prereqChecker` program.

- **Using the LinuxThreads threading mode instead of the Native POSIX Threading Library (NPTL).**  
WebSphere Application Server, Version 5.1.0 products run under the LinuxThreads threading mode because of program dependencies on specific behavior within the LinuxThreads mode. The WAS 5.1.0 run time automatically sets up LinuxThreads mode for you. Do not try to force the NPTL mode because errors are thrown by the run time when the WebSphere embedded messaging broker and queue servers are started.

If you set `LD_ASSUME_KERNEL=2.4.19` or `LD_ASSUME_KERNEL=2.2.5` to switch to LinuxThreads outside the context of WebSphere Application Server, it is recommended that you also set the `RPM_FORCE_NPTL=1` environment variable.

Unusual behavior occurs with Red Hat Package Manager (RPM) when running under LinuxThreads on Red Hat Enterprise Linux, Version 3.0. Corruption of the RPM database can occur. Setting the environment variable forces the RPM program to always run under NPTL threading and protects the RPM database from corruption.

Set the value in the environment using the following command:

```
export RPM_FORCE_NPTL=1
```

- **Avoiding an error that occurs when starting the Launchpad program for WebSphere Application Server clients, Version 5.1 using the Konqueror file in the K Desktop Environment on SuSE Linux Enterprise Server (SLES).**

When starting the Launchpad program for WebSphere Application Server clients, Version 5.1 using the Konqueror file manager in the K Desktop Environment (KDE) on Linux systems, a "Couldn't find the program `launchpad.sh`" error occurs.

Because the `launchpad.sh` command uses a relative path to locate the Java program, you should run the `launchpad.sh` command from the directory where the `launchpad.sh` command is located for the

client program. When using the Konqueror file manager to issue the **launchpad.sh** command, the current directory is your home directory. Therefore, the **launchpad.sh** command cannot work.

Do not use the Konqueror file manager to issue the **launchpad.sh** command for the client. Open a Konsole session in the KDE, change the current directory to where the **launchpad.sh** command is located, and run the **launchpad.sh** command.

- **Avoiding a segmentation fault on RHEL V3.0 Update 1.**

An "A suitable JVM could not be found" message is displayed when installing WebSphere Business Integration Server Foundation on Red Hat Enterprise Linux 3 U1.

When installing from an operator console attached to the server (not telnet), the "A suitable JVM could not be found" message displays and the installation fails with a segmentation fault.

You might have previously installed, logged off, logged back on, and are running the installation again on the operator console attached to the server.

This is a known limitation of Red Hat Enterprise Linux V3.0 that causes a segmentation fault when calling the JVM.

Test if the JVM is failing by running the following command:

```
/mnt/cdrom/platform/linuxi386/jdk/java/jre/bin/java -version
```

If you receive a "Segmentation Fault" message, reboot your machine or press **Ctrl-X** to reinstall. Rebooting the machine resolves the problem.

## Solaris Operating Environment

### Summary of tips that apply to Solaris platforms

- Providing adequate space for Solaris platforms
- **5.1+** Uninstalling a V5.1 instance on a Solaris platform before reinstalling the instance
- Using the **unzip** function, not the **jar** command, to decompress downloaded files
- Closing the terminal window that remains open after the installation finishes
- Configuring the `converter.properties` file to use EUC-JP (Japanese) encoding
- Configuring the Domino Server plug-in
- Avoiding problems starting application servers in zh\_TW.EUC locale on Solaris
- Using the correct mount point for the product CD

### Tips that apply to Solaris platforms

- **Providing adequate disk space for Solaris platforms.**

Table 32. Installed sizes of installation directories for Solaris platforms

|                                            | Base product                     | Network Deployment                           | IBM HTTP Server          | Embedded messaging            | Tivoli Global Security Kit                 | Temp space         |
|--------------------------------------------|----------------------------------|----------------------------------------------|--------------------------|-------------------------------|--------------------------------------------|--------------------|
| <b>Installation directory</b>              | /opt/<br>WebSphere/<br>AppServer | /opt/<br>WebSphere/<br>Deployment<br>Manager | /opt/ IBM<br>Http Server | /var/ mqm<br>and /opt/<br>mqm | /opt/ ibm/<br>gsk7                         | /tmp               |
| <b>Minimum free space for installation</b> | 530 MB                           | 460 MB                                       | 20 MB                    | 4 MB; 40 MB<br>on base        | 23.1 MB<br>(GSKit 7); 23.1<br>MB (GSKit 5) | At least 150<br>MB |

Space requirements for the embedded messaging feature are described in Installing WebSphere embedded messaging as the JMS provider .

- **5.1+** **Uninstalling a V5.1 instance on a Solaris platform before reinstalling the instance.**

If you uninstall a V5.1 instance but leave the embedded messaging feature because of other WebSphere Application Server instances, clean up the `/var/sadm/pkg` directory before reinstalling V5.1.

If you uninstall all of the WebSphere Application Server products on the machine and are now reinstalling, follow the procedure in “Manually uninstalling on Solaris platforms” on page 302 instead of this procedure.

You must perform this procedure before you can reinstall the product into the same installation root directory on Solaris platforms. Otherwise, you must reinstall the product into another directory and you might have a problem reinstalling the embedded messaging feature.

Assume that V5.0.x exists in /opt/WebSphere/AppServer50 with the embedded messaging server and client feature. You then install V5.1 to the following directory, /opt/WebSphere/AppServer51, but later you uninstall the product and leave the embedded messaging feature installed. And now you decide to reinstall V5.1.

1. Make the /var/backupForWAS51Reinstall backup directory.
2. Change directories to /var/sadm/pkg, and type `ls -lt | grep WSBA51` to find packages that contain the base WebSphere Application Server.

The directory for Network Deployment is WSNA51.

3. If the WSBA51 directory exists, change directories to the WSBA51 directory and read the pkginfo file in the /var/sadm/pkg/WSBA51 directory.

Find the installation location for V5.1. If the location is the same installation location as the V5.1 product that you uninstalled, move the /var/sadm/pkg/WSBA51 folder to the /var/backupForWAS51Reinstall directory.

The following example shows how to display the pkginfo file:

```
pwd
/var/sadm/pkg/WSBA51
ls -lt
total 6
-rw-r--r-- 1 root other 591 Oct 27 15:46 pkginfo
drwxr-xr-x 2 root other 512 Oct 27 15:46 install
drwxr-xr-x 2 root other 512 Oct 27 15:46 save
cat pkginfo
VPD created by InstallShield MultiPlatform Edition
Mon Oct 27 15:46:34 EST 2003
ARCH=sparc
BASEDIR=/opt/WebSphere/AppServer51
CATEGORY=application
CLASSES=none
DESC=
ISJE_INSTALL_STATUS=3
ISJE_IS_ACTIVE=true
ISJE_IS_PUBLICLY_SHAREABLE=false
ISJE_NAME=IBM WebSphere Application Server
ISJE_PARENTS=WSBA51:5.1.0.0.DSP=:1
ISJE_PRODUCT_NUMBER=5.1.0.0
ISJE_REQUIRED_COUNT=0
ISJE_SOURCES=
ISJE_UID=WSBA51
ISJE_UNINSTALLER="_uninst" "uninstall.jar" "uninstall.dat" ""
ISJE_VPD_VERSION=4.5
ISJE_WEB=
NAME=WebSphere Application Server
PKG=WSBA51
PKGINST=WSBA51
VENDOR=
VERSION=5.1.0.0.DSP=
```

When two instances of V5.1 WebSphere Application Server exist, such as with one instance installed and one instance uninstalled, for example, two files exist in the /var/sadm/pkg directory. One is the /var/sadm/pkg/WSBA51 file, the other is the /var/sadm/pkg/WSBA51.2 file. For example:

```
ls -lt | grep WSBA51
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBA51.2
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBA51
drwxr-xr-x 3 root other 512 Oct 21 13:32 WSBA51
```

4. Verify that you are moving the correct file to the backup directory.

Examine the `pkginfo` files. Do not touch entries for other WebSphere Application Server products that might be present on the platform, such as `WS` entries for existing WebSphere Application Server V5.0.x.

5. Run the `ls -lt | grep WSBM` command to get all of the embedded messaging packages.

For example:

```
#
ls -lt | grep WSBM
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMQ1AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMQ2AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMS4AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMS6AA.2
drwxr-xr-x 4 root other 512 Oct 27 18:52 WSBMQAA.2
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMS4AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMS6AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQ1AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQ2AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQ3AA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMQAA
drwxr-xr-x 4 root other 512 Oct 27 15:46 WSBMSAA
```

6. Identify entries that are related to the V5.1 product that you uninstalled.

Examine the `pkginfo` file for each entry. In the previous example, `".2"` entries relate to the product that you uninstalled, which was originally in the `/opt/WebSphere/AppServer51` directory. Use a command to move all of the `".2"` entries into the backup directory. For example, run the following command from the `/var/sadm/pkg/` directory:

```
mv WSBM*.2 /var/backupForWAS51Reinstall
```

For V5.1 of Network Deployment, all of the related embedded messaging entries start with `WSNM`. Run the following command to show the Network Deployment entries:

```
ls -lt | grep WSNM
```

7. Rename `/opt/WebSphere/AppServer51` to `/opt/WebSphere/AppServer51_bk` before reinstalling V5.1.

- **Using the `unzip` function, not the `jar` command, to uncompress downloaded files.**

To uncompress downloaded installation files for the Solaris operating system, use the `unzip` function and not the `jar` command. Using the `jar` function sets file permissions incorrectly, which causes the installation to fail.

- **Closing the terminal window that remains open after the installation finishes.**

When installing WebSphere Application Server from the product CD onto a Solaris system, the ISMP Installation wizard launches a terminal window, which remains open after the installation is complete. This window contains the following text:

```
InstallShield Wizard
Initializing InstallShield Wizard...

Searching for Java(tm) Virtual Machine...
.....
```

Close the window after the installation completes.

- **Configuring the `converter.properties` file to use EUC-JP (Japanese) encoding.**

The `install_root/java/jre/lib/i18n.jar` file on Solaris platforms does not have the converters for `Cp33722C`, but does have the converter for `Cp33722`. To use EUC-JP encoding on Solaris platforms, change the `EUC-JP=Cp33722C` entry in the `converter.properties` file to `EUC-JP=Cp33722` or `EUC-JP=EUC_JP`.

- **Configuring the Domino Server plug-in.**

During installation, a `dsapi_stderr.txt` file is created in the `logs` directory and you can get the following error messages:

```

lotus.notes.NotesException: Could not load dll for system name SUNOS
 at lotus.notes.NotesThread.load(NotesThread.java:210)
 at lotus.notes.NotesThread.<clinit>(NotesThread.java:24)
java.lang.UnsatisfiedLinkError: NnotesInitThread
 at lotus.notes.NotesThread.NnotesInitThread(Native Method)
 at lotus.notes.NotesThread.initThread(NotesThread.java:99)
 at lotus.notes.NotesThread.run(NotesThread.java:133)

```

You can configure the IBM WebSphere Application Server or Domino Server plug-in manually using the Domino Server Web administration tool. The workarounds include:

1. Start the Domino Server.
2. Enter the Web address for the Domino Server Web Administration site using a browser. For example, `http://hostname/names.nsf`. Enter the administrator user name and password.
3. Double-click **Server-Servers**.
4. Double-click **WebServer** to configure.
5. Double-click **Edit Server**.
6. Double-click **Internet Protocol**.
7. Add the IBM WebSphere Application Server DSAPI plug-in to the **DSAPI** field. For example, `/opt/WebSphere/AppServer/bin/libdomino5_http.so`  
If DSAPI filter files are already specified, use a space to delimit the WebSphere Application Server plug-in file.
8. Double-click **Save and Close**.
9. Restart the Domino Server.

- **Avoiding problems starting application servers in the zh\_TW.EUC locale on Solaris platforms.**

If you use the embedded JMS provider in WebSphere Application Server products or WebSphere MQ as the JMS provider on Solaris, do not set the LANG and LC\_ALL variables to zh\_TW.EUC (Traditional Chinese locale) to avoid problems when starting application servers. Set the LANG and LC\_ALL variables to zh\_TW instead of zh\_TW.EUC.

- **Using the correct mount point for the product CD.**

If the Solaris Volume Manager (vold) is running, Solaris automatically mounts the product CD as the `/cdrom/was_wbisf_sun_cd1` device.

Do not run the **launchpad.sh** command or the **install** command from the automatically generated mount point. For example, do not run the `/cdrom/was_wbisf_sun_cd1/sun/install` command.

1. Close the File explorer window that Solaris automatically opens so that you can define a correct mount point.
2. Open a shell window and use the window throughout the installation.
3. Verify that you are in a read/write directory and not the CD-ROM directory or another read-only directory before you start the installation.
4. Start the installation using the mount point specification in the following command:

```
/cdrom/cdrom0/sun/1launchpad.sh
```

## Windows platforms

### Summary of tips that apply to Windows platforms

- Providing adequate space for installation
- Providing the required `msvcp60.dll` file for Windows 2000 Server
- Planning for the default installation path
- Viewing accurate migration messages
- Decompressing a WebSphere Application Server `elimage` that you download
- Editing or removing the `vpd.properties` file
- Preparing for Adaptive Fast Path Architecture (AFPA) driver availability when migrating from IBM HTTP Server Version 1.3.19.x, or earlier
- Rebooting the machine after uninstalling the embedded messaging feature
- Ignoring an incorrect `ImagePath` entry in the registry after changing the document root



- Avoiding a user ID with spaces when installing the WebSphere Application Server as a Windows service
- Changing the display settings from high-contrast black to high-contrast white to view panels with hypertext links
- Avoiding using the default service names created by the installer because they are common
- Uninstalling WebSphere Embedded Messaging Publish and Subscribe Edition on Windows 2003 Server
- Lowering the Internet Explorer security level from High to Medium to view the Product overviews from the Launchpad on Windows 2003 machines
- Expecting a delay before the Installation wizard displays its first panel

### Tips that apply to Windows platforms

- **Providing adequate disk space for installation.**

Table 33. Installed sizes of installation directories for Windows platforms

|                                       | Base product                         | Network Deployment product                    | IBM HTTP Server                  | Embedded messaging                | Tivoli Global Security Kit | Temp space      |
|---------------------------------------|--------------------------------------|-----------------------------------------------|----------------------------------|-----------------------------------|----------------------------|-----------------|
| <b>Installation directory</b>         | C:\Program Files\WebSphere\AppServer | C:\Program Files\WebSphere\Deployment Manager | C:\Program Files\IBM Http Server | C:\Program Files\IBM\WebSphere MQ | C:\Program Files\IBM\gsK7  | C:\temp         |
| <b>Minimum space for installation</b> | 422 MB                               | 360 MB                                        | 20 MB                            | 4 MB; 40 MB on base               | 16.8 MB                    | At least 150 MB |

Space requirements for the embedded messaging feature are described in the Installing WebSphere embedded messaging as the JMS provider topic.

- **Providing the required msvc60.dll file for Windows 2000 Server platforms.**

The embedded messaging feature requires a DLL file on Windows 2000 Server platforms. The DLL file is already installed on Windows 2000 Advanced Server and other Windows platforms that are supported. If you see a message similar to the following example, download the msvc60.dll file:

```
wempsdeletebroker.exe - Unable To Locate DLL
The dynamic link library MSVCP60.dll could
not be found in the specified path...
```

The prereq checker in the installer program does not check for this DLL file on the Windows 2000 Server platform. If you select the Windows 2000 Support Tools during Windows 2000 Server installation, the installation program for Windows 2000 Server installs the DLL file in the C:\Program Files\Support Tools directory. The DLL file is installed during the installation of Windows 2000 Advanced Server in the C:\WINNT\system32 directory.

You can download the DLL file from the Microsoft Web site.

- **Planning for the default installation path**

On Windows platforms, the default installation path is *drive:\Program Files\IBM\WebSphere MQ\*. You can specify a different directory during installation. On Linux and UNIX-based platforms, the path is fixed.

- **Viewing accurate migration messages.**

When migration is running during installation, the information displayed in the pre-migration and post-migration processing might display corrupted national characters. Use the PreUpgrade and PostUpgrade logs in the *backup/logs* directory, where *backup* is the backup directory specified during installation.

- **Decompressing a WebSphere Application Server image that you download.**

If you are using the downloadable archive file to install the WebSphere Application Server product, the PKWARE pkunzip utility might not decompress the download image correctly. Use another utility (such as WinZip) to unzip the image.

- **Editing or removing the vpd.properties file.**

Installing WebSphere Application Server on some international computers might result in an installation failure or in font problems. Use the following procedure to work around this problem:

1. Locate the `vpd.properties` file in the operating system installation directory.

For example, `C:\WINDOWS` or `C:\WINNT` on a Windows system.

2. Remove all of the lines containing one of these strings:

- WSE
- WSN
- WSB
- WSM

Do not delete or rename the `vpd.properties` file because the InstallShield for MultiPlatforms (ISMP) program uses it for other products that it installs. If you are sure that no other entries exist in the file, you can delete the file.

- **Preparing for Adaptive Fast Path Architecture (AFPA) driver availability when migrating from IBM HTTP Server Version 1.3.19.x, or earlier.**

The AFPA driver controls the fast response cache accelerator function, which is also known as the *cache accelerator*. The version of IBM HTTP Server installed with WebSphere Application Server shares the AFPA driver with any coexisting IBM HTTP Server. Uninstalling a coexisting Version 1.3.19.x (or earlier) IBM HTTP Server also uninstalls the common AFPA driver.

When configured to use an AFPA driver that is no longer present, the cache accelerator does not provide a response improvement. IBM HTTP Server generates errors as it starts. For example:

```
[error] (9) Bad file descriptor: Afpa Device Driver open failed.
```

You can either restore the driver or disable the cache accelerator configuration. Restore the driver by reinstalling the later version of IBM HTTP Server after you uninstall the earlier version. To verify that the AFPA driver is installed and working, see if it is listed in device manager under Non-Plug and Play Drivers. Select the Show hidden devices option in the device manager view on Windows 2000 or later platforms.

You can disable AFPA by commenting out the following directives in the `httpd.conf` configuration file:

```
AfpaEnable
AfpaCache on
AfpaLogFile "C:\Program Files\IBM HTTP Server\log\afpalog" V-ECLF
```

- **Rebooting the machine after uninstalling the embedded messaging feature.**

If you uninstall the embedded messaging feature on a Windows machine, you must reboot the machine before reinstalling.

- **Avoiding a user ID with spaces when installing the WebSphere Application Server as a Windows service.**

When installing the WebSphere Application Server as a Windows service, do not use a user ID that contains spaces. A user ID with spaces cannot be validated. The user is not allowed to proceed with the installation. To work around this problem, install with a user ID that does not contain spaces, or do not choose to install services.

- **Changing the display settings from high-contrast black to high-contrast white to view panels with hypertext links.**

For Installation wizard panels that contain hypertext links, a display setting of "high-contrast black" causes some of the text to display in the same color (black) as the background color. This makes the text impossible to display. Change your settings to "high-contrast white" to see all of the text in every Installation wizard panel.

- **Avoiding using the default service names created by the installer because they are common**

Unable to use the default service names created by the installer because they are common

In a same version coexistence scenario for IBM HTTP Server Version 1.3.28 or Version 2.0.47 on a Windows platform, you are unable to use the default service names created by the installer because they are common.

To work around this problem, perform the following steps:

1. Install the first copy of IBM HTTP Server, either by itself or with IBM WebSphere Application Server.
2. Customize the service names for the first install by running the following commands from the first install location:

```
apache -k install -n "IHS 1.3.28(1)"
 apache -k install -f conf\admin.conf -n "IHS 1.3.28 Administration (1)"
```

3. Edit the `AdminAlias` directive in the `<installLocation 1>\conf\admin.conf` file to point to the new service name, for example "IHS 1.3.28(1)".
4. Remove the default service names installed by the first installation by running:

```
apache -k uninstall -n "IBM HTTP Server 1.3.28"
 apache -k uninstall -n "IBM HTTP Administration 1.3.28"
```

5. Install the second copy of IBM HTTP Server, either by itself or with IBM WebSphere Application Server. The default service names correspond to the second installation.

The custom service names previously mentioned must be unique on your system and are not required to be those exact strings.

- **Uninstalling WebSphere Embedded Messaging Publish and Subscribe Edition on Windows 2003 Server.**

After using the Add or Remove Programs utility on Windows 2003 Server to uninstall WebSphere Application Server, the entry for WebSphere Embedded Messaging Publish and Subscribe Edition persists. Using the Add or Remove Programs utility to remove WebSphere Embedded Messaging Publish and Subscribe Edition generates a Fatal Error during installation message.

As a work around, manually remove the WEMPS entry in the Windows 2003 Server registry. See "Manually uninstalling on Windows platforms" on page 307 for more information about manually uninstalling messaging components.

- **Lowering the Internet Explorer security level from High to Medium to view the Product overviews from the Launchpad on Windows 2003 machines.**

You might see a blank page when trying to view Product overviews from the WebSphere Launchpad on Windows 2003 machines. The default security level setting for the Internet Zone on Internet Explorer is High. Use the following techniques to lower the Internet zone security level to Medium:

1. Right-click the Internet Explorer icon on the desktop.
2. Click **Properties > Security > Internet Zone**.
3. In the **Security level for this zone** area, move the slider from **High** to **Medium**.
4. Click **OK** to save the change.

- **Expecting a delay before the Installation wizard displays its first panel**

The default installation method is to open a command window to a read/write directory, issue the fully qualified command to start the Launchpad tool, and click the **Install the product** option. This option launches the Installation wizard, which is an InstallShield for MultiPlatforms (ISMP) application. This action launches the Installation wizard in the language of your machine locale unless no translation exists for your locale, in which case you receive the English version. A short delay occurs before the ISMP wizard displays.

You do not need to click the Install the product option more than once to cause the wizard to display.

## Tips for installing the embedded messaging feature

When installing IBM WebSphere Application Server, you can install the embedded messaging feature (which is selected by default) to use as the Java Message Service (JMS) provider. To install the embedded messaging feature successfully, you must complete several other actions first as described in the installation procedure. You must consider whether you also want to use WebSphere MQ on the same host. On Linux and UNIX-based platforms, you must define the groups and the user required to support the embedded messaging feature.

You can either complete the required actions before installing the WebSphere Application Server product, or deselect the embedded messaging feature, which installs the WebSphere Application Server product without embedded messaging.

If you already have WebSphere MQ installed, you can configure it as the JMS provider. Otherwise, you can install the embedded messaging feature, the WebSphere MQ product, or another JMS provider later, after you install the WebSphere Application Server product.

The WebSphere Application Server embedded messaging feature provides a JMS provider that supports both queues (for point-to-point messaging) and topics (for publish and subscribe messaging). You can install the WebSphere Application Server with the embedded messaging feature on the same host with the WebSphere MQ product. To support this combination, several WebSphere MQ product features must be at a certain level.

After installing the WebSphere Application Server product with the embedded messaging feature, you can install the WebSphere MQ product and use it as the JMS provider instead.

This topic provides an index to tips for installing and migrating the embedded messaging feature that are described in “Platform-specific tips for installing and migrating” on page 131.

### Tips for all platforms

- **5.1+** Migrating from embedded messaging to WebSphere MQ requires setting the MQ\_INSTALL\_ROOT variable to the location of the installation root of WebSphere MQ
- Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component
- Recovering from an InvalidExecutableException while starting jmsserver
- Applying fixes and fix packs to the embedded messaging feature
- Providing adequate space for installation
- Avoiding the installation of the server and client feature or the client feature more than once
- Installing the embedded messaging server feature if WebSphere MQ Version 5.3 is already installed
- Logging off and back on, or rebooting a Windows machine, after uninstalling the embedded messaging feature
- Planning not to use terminal services with the embedded messaging feature
- Avoiding a coexistence problem between embedded messaging, IBM WebSphere Studio Application Developer Integration Edition, and IBM WebSphere Application Server
- Locating more information about the embedded messaging feature or WebSphere MQ

### Tips for all Linux and UNIX platforms

- Preparing a Linux or UNIX operating platform for the embedded messaging feature
- Installing to fixed locations on UNIX-based operating systems
- Restricting access to the /var/mqm/errors directory and messaging logging files

### Tips for AIX platforms

- Installing the prerequisite Java130.rte.lib Version 1.3.0 on AIX Version 4.3.3 or AIX Version 5
- Correcting directory permissions on AIX platforms before reinstalling WebSphere Application Server with the embedded messaging feature

### Tips for Linux platforms

- **5.1+** Ignoring YAST2 dependency conflicts for embedded messaging packages

## Tips for Windows platforms

- Providing the required `msvc60.d11` file for Windows 2000 Server
- Planning for the default installation path

## Using the Launchpad to start the installation

Use the Launchpad to access a product overview, to access the readme file, to access the installation guide, and to install the product.

1. Start the Launchpad.

The Launchpad program is in the *operating\_system\_platform* directory on the product CD.

You can start the Launchpad manually using a fully qualified command instead of changing directories to the CD-ROM and running the command locally from the platform directory:

- Mount the CD-ROM drive on Linux or UNIX-based platforms, if necessary. This procedure varies per platform. Consult the installation procedure for your platform for instructions on mounting CD-ROM drives. Open a shell window and issue a fully qualified **launchpad.sh** command to start the Launchpad. On Windows platforms, issue a fully qualified **launchpad.bat** command to start the Launchpad. Issue a fully qualified **LaunchPad.sh** command to start the Launchpad for the **Application Clients** CD-ROM.

The Launchpad opens in the language of the machine locale setting. The appearance of the Launchpad is the same for the base product and the Network Deployment product. You can access the Installation Guide to see which product it describes installing if you lose track of the product to which the panel applies.

2. Use the Launchpad to access the product overview, the readme file, and installation guides.
3. Click **Install the product** to launch the installation wizard.

You can use the Launchpad to start the installation and to access information through a browser.

If you use the Mozilla browser on a Linux or UNIX system and you do not have the Netscape browser, some links might not work because the Launchpad attempts to call the Netscape browser in the `/usr/bin/netscape` directory. Try a symbolic link to the Mozilla browser to fix the problem as shown in the following example:

```
ln -sf /usr/bin/mozilla /usr/bin/netscape
```

If you still cannot use the Launchpad panel, you can launch the following links directly from your browser:

### Button Link

#### Product Overview

<http://www.ibm.com/software/webservers/appserv/>

It is possible that your security options might prevent redirection, such as to the <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/appserv> Web site. If so, you can identify the IBM Web site as a trusted site to allow the redirection. For example, on Internet Explorer, add <http://www-306.ibm.com> to the list of Trusted Sites by selecting **Tools > Internet Options > Security (tab) > Trusted Sites**.

#### Readme File

*CD-ROM mount point/readme/readme\_en.html*

#### Installation Guide

*CD-ROM mount point/docs/InstallGuide\_en.PDF*

The *Installation Guide* is in Adobe PDF format. Viewing this document requires you to have Adobe Acrobat Reader, which is a free download from the Adobe Acrobat Web site.

**5.1+** Download a new installation guide at [ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51base\\_gs.pdf](ftp://ftp.software.ibm.com/software/webserver/appserv/library/wasv51base_gs.pdf).

## Install the product

`CDROM_mount_point/platform_directory/install` on Linux and UNIX-based platforms.

`CDROM_drive:\platform_directory\Install.exe` on Windows platforms.

## Installing silently

This topic describes how to perform a silent installation of the product as the root user on a Linux or UNIX-based operating system, or from a user ID that belongs to the Administrator group on a Windows platform.

Use this procedure to perform a silent installation of the WebSphere Application Server Network Deployment product.

A silent installation uses the installation program to install the product. However, instead of displaying a wizard interface, the silent installation causes the installation program to read all of your responses from a file that you must customize.

1. Verify that the user ID has sufficient authority to perform the task.

Install silently as root on a Linux or UNIX-based platform, or as a user that belongs to the administrator group on a Windows platform. Some steps for installing silently on a Windows platform require the administrator group user to have the following advanced user rights:

- *Act as part of the operating system*
- *Log on as a service*

The installation wizard grants your Windows user ID the advanced user rights, if the user ID belongs to the administrator group. The silent installation does not grant these rights. If you create a new user ID on a Windows platform to perform the silent installation, you must restart the system to activate the proper authorizations for the user ID, before you can perform a successful silent installation.

Table 34. Installation tip

| Operating platform | Tip                                                                        |
|--------------------|----------------------------------------------------------------------------|
| Windows platforms  | A Windows user ID with spaces cannot be validated for installing services. |

If you attempt a silent installation of services or the embedded messaging feature without the required authorizations, the installation might fail while attempting to write to a `log.txt` file that it has not opened yet. Or, you might see messages in the `log.txt` file that are similar to the following messages:

```
(Jan 30, 2003 12:04:32 PM), Setup.product.install,
com.ibm.ws.install.conditions.PCWebSphereCheckCondition, msg1,
higherVersionFoundPasses: false
(Jan 30, 2003 12:04:32 PM), Setup.product.install,
com.ibm.ws.install.conditions.PCWebSphereCheckCondition, msg1,
defaultEvalResult: false
(Jan 30, 2003 12:04:32 PM), Setup.product.install,
com.ibm.ws.install.conditions.PCWebSphereCheckCondition, msg1,
checkOnlyCurrentVersion: true
(Jan 30, 2003 12:04:34 PM), Setup.product.install,
com.ibm.ws.install.conditions.PCWebSphereCheckCondition, msg1,
higherVersionFoundPasses: false
(Jan 30, 2003 12:04:34 PM), Setup.product.install,
com.ibm.ws.install.conditions.PCWebSphereCheckCondition, msg1,
defaultEvalResult: false
(Jan 30, 2003 12:04:34 PM), Setup.product.install,
com.ibm.ws.install.conditions.PCWebSphereCheckCondition, msg1,
checkOnlyCurrentVersion: false
```

(Jan 30, 2003 12:04:41 PM), Setup.product.install,  
com.ibm.ws.install.actions.LogMessageAction, msg1, INST0058E:  
The install failed or did not complete due to one or more errors.

2. Copy the response file as myoptionsfile to your disk drive and customize it.  
The name of the original file is responsefile.
3. Issue one of these commands to use your custom response file:

#### Windows platforms

```
"fully_qualified_CD_drive_D:\win\install"
-options"fully_qualified_disk_drive_C:_WAS\myoptionsfile
```

The log.txt file is in the *install\_root\logs* directory.

#### Linux and UNIX-based platforms

```
fully_qualified_mnt_cdrom/install -options
fully_qualified_disk_drive_/tmp/WAS/myoptionsfile
```

The log.txt file is in the *install\_root/logs* directory.

You can find the sample options response file in the *operating-system platform* directory on the product CD.

4. Reboot your machine in response to the prompt that might appear on Windows platforms.  
If you install the embedded messaging feature, `-P mqSeriesBean.active="true"`, certain conditions, such as a locked file, might require you to reboot. You have the option of rebooting immediately, after which the installation program resumes the installation at the point it left off. You can also defer rebooting to a convenient time, such as after the overall installation is complete.

You can install silently using the response file. Examine the log.txt file for lines similar to these:

```
com.ibm.ws.install.actions.WSAdminInstallEarsAction, msg1,
Installing ear with:
util\launcher.exe
"C:\WebSphere\DeploymentManager\bin\wsadmin.bat"
-conntype NONE
-c "$AdminApp install \\\"C:/WebSphere/DeploymentManager/
installableApps/adminconsole.ear\
"{
-node NodeName
-cell CellName
-server dmgr
-copy.sessionmgr.servername dmgr
-appname adminconsole
}"
```

This is an indicator that you have successfully installed the product.

## responsefile

This topic describes the response file for installing the WebSphere Application Server Network Deployment product, so that the root user on a Linux or UNIX-based platform, or a user that belongs to the Administrator group on a Windows platform, can install the product without the installation wizard. Some steps of the silent installation procedure on a Windows platform require the user to belong to the Administrator group and have the advanced user rights *Act as part of the operating system* and *Log on as a service*.

Silent installation is an installation method that reads all choices from the options response file, responsefile, which you must tailor before performing a silent installation. This file does not affect the installation wizard in any way.

## Location

The sample options response file, `responsefile`, is on the product CD in the *operating-system platform* directory.

## Usage notes

- The file is not a read-only file.
- Edit this file directly with your flat file editor of choice, such as WordPad on a Windows platform.
- The file is not updated by any product components.
- The file must exist to perform a silent installation. The installation program reads this file to determine installation option values when you install silently.
- Save the file in a location you can identify when you specify the fully qualified path as part of the installation command.
- If you are installing on a Windows platform, your user ID must belong to the Administrator group and have the advanced rights previously described. If you select the following option when your user ID does not comply, the installation fails:

```
Install the WebSphere Network Deployment service

-W serviceSettingsWizardBean.wasChoice="true"
```

## Extra information displays in the Network Deployment response file

Extra information displays in the IBM WebSphere Application Server Network Deployment response file. Specifically, the following line appears and always must be commented:

```
#-W coexistencePanelBean.useIhs="true"
```

## Automatically restarting server processes

There are several server processes related to WebSphere Application Server products that the operating system can monitor and automatically restart when the server processes stop abnormally. This task describes how to set up monitored processes.

To set up this function on a Linux and UNIX-based operating system, you must have root authority to edit the `inittab`.

On a Windows operating system, you must belong to the Administrator group and have the user rights *Act as part of the operating system* and *Log on as a service* to define Windows services. The installation wizard grants you the user rights, if your user ID is part of the Administrator group. It displays a message that states that although the advanced user rights are now effective, they do not display as effective until the next time you log on.

You can also add the user rights manually if you are performing a silent installation on a Windows platform. For example, to grant the user rights to your administrator group user ID on a Windows 2000 Server platform:

1. Click **Administrative Tools** in the Control Panel.
2. Click **Local Security Policy**.
3. Click **Local Policies**.
4. Click **User Rights Assignments**.
5. Right click **Act as part of the operating system**.
6. Click **Security**.
7. Click **Add**.
8. Click your user ID.
9. Click **Add**.



10. Click **OK**.
11. Click **OK**.
12. Right click **Log on as a service**.
13. Click **Security**.
14. Click **Add**.
15. Click **OK**.
16. Click **OK**.
17. Reboot your machine to make the settings effective.

Consult your operating system help for more information.

There are several environments where you might use this function of automatically restarting servers. You can restart the server1 process in a stand-alone WebSphere Application Server environment, for example. Here is a list of processes you might consider restarting:

- The server1 process on a stand-alone Application Server
- The dmgr process on a deployment manager node
- The nodeagent server process on any Application Server node in a deployment manager cell
- The jmsserver process on the Application Server node, if the Application Server node has the embedded messaging feature
- The IBM HTTP Server process
- The IBM HTTP Administration process
- The "WebSphere Embedded Messaging Publish And SubscribeWAS\_*node\_name*\_jmsserver" process, if the Application Server node has the embedded messaging feature
- The "WebSphere Embedded Messaging Publish And SubscribeWAS\_*node\_name*\_server1" process, if the Application Server node has the embedded messaging feature

You can create Windows services during installation, using the installation wizard. The wizard lets you create services for these servers:

- The server1 process in a stand-alone base product environment, defined as a manually started (versus automatic) service
- The IBM HTTP Server process and the IBM HTTP Administration process, defined as automatically started services when you choose to install the IBM HTTP Server feature during the base product installation
- The dmgr process on a deployment manager node, defined as a manually started (versus automatic) service

The installation wizard does not provide a way to create a service for a node agent because the deployment manager instantiates each node agent after installation when you add an Application Server node to the deployment manager cell. For this reason, you must manually create a function that automatically starts a failed nodeagent server process.

You must manually create a shell script that automatically starts any of the processes previously mentioned, on a Linux and UNIX-based operating system. Each Windows service or UNIX shell script controls a single process, such as a stand-alone WebSphere Application Server instance. Multiple stand-alone Application Server processes require multiple Windows service or UNIX scripts, which you can define.

In a Network Deployment environment, the **addNode** or **startNode** command starts a single unmonitored node agent only, the nodeagent process, and does not start all of the processes that you might define on the node. While running, the node agent monitors and restarts Application Server processes on that node, on either a Windows or a Linux and UNIX-based platform. Each Application Server process has MonitoringPolicy configuration settings that the node agent uses when monitoring and restarting the process.

It is recommended that you set up a monitored node agent process manually, either through a Windows service, or through the `rc.was` example shell script on Linux and UNIX-based platforms. The operating system monitors and automatically restarts the node agent process, `nodeagent`, if the process terminates abnormally, which means if the process stops without going through a normal shutdown. It is also recommended that you set up the deployment manager server, `dmgr`, as a monitored process. As mentioned, you can do this during installation on a Windows platform. On a Linux and UNIX-based platform, use the `rc.was` example shell script to set up the deployment manager `dmgr` server as a monitored process.

If you do not install the WebSphere Application Server base product or Network Deployment as a Windows service during installation, you can use the **WASService** command in the `install_root/bin` directory to do so at a later time. You can use the tool to add any WebSphere Application Server process as a Windows service. The operating system can then monitor each server process, and restart the server if it stops.

1. **Use the installation wizard** to set up a Windows service to automatically monitor and restart processes related to the WebSphere Application Server product.

- Perform the following procedure from the installation wizard for the Network Deployment product:
  - a. Click **Run WebSphere Application Server Network Deployment as a service**.
  - b. Enter your user ID and password and click **Next**.

The installation wizard creates the following service during the installation:

**IBM WebSphere Application Server V5 - dmgr**

The IBM WebSphere Application Server V5 - `dmgr` service controls the **dmgr** server process, which is the deployment manager server that administers the cell.

2. **After installing**, use the `WASService.exe` tool to manually define the `nodeagent` server process as a Windows service.

You can use the same tool to manually define a Windows service for another installation or configuration instance of either the base WebSphere Application Server product or the Network Deployment product.

3. **After installing**, set up a Linux and UNIX-based shell script to automatically monitor and restart the `nodeagent` server or any other related server process.
  - a. Locate the `rc.was` example shell script, which is in the `install_root/bin` directory.
  - b. Create a new shell script for each process that the operating system is to monitor and restart.
  - c. Edit each shell script according to comments in its header, which provide instructions for identifying a WebSphere Application Server process.
  - d. Edit the `inittab` table of the operating system, to add an entry for each shell script you have created. Comments in the header of the `rc.was` file show a sample `inittab` entry line for adding the script. This `inittab` entry causes the Linux and UNIX-based system to call each shell script whenever the system initializes. As it runs, each shell script monitors and starts the server process you specified.

Each shell script monitors and restarts an individual WebSphere Application Server process in a stand-alone environment, or a node agent or deployment manager process in an WebSphere Application Server Network Deployment environment.

You can use the **net start** and **net stop** commands to control the IBM HTTP Server services on a Windows system. For more information about these commands, see the Windows help file. Access these commands from the Start menu, clicking **Start > Programs > IBM HTTP Server**.

**5.1+** You can also use the **Start the Server** and **Stop the Server** commands to control the IBM WebSphere Application Server on a Windows system. Access these commands from the Start menu, clicking **Start > Programs > IBM WebSphere > Application Server V5.1**.

**5.1 +** You can also use the **Start the Manager** and **Stop the Manager** commands to control the Network Deployment dmgr server on a Windows system. Access these commands from the Start menu, clicking **Start > Programs > IBM WebSphere > Application Server V5.1 > Deployment Manager**.

Processes started by a **startServer** or **startNode** or **startManager** command are not running as monitored processes, regardless of how you have configured them.

For example, you can configure a base Application Server as a WebSphere Application Server monitored process. However, if you start the Application Server instance using the **startServer** command, the operating system does not monitor or restart the Application Server because the operating system did not start the process as a monitored process.

## WASService command

The **WASService** command line tool lets you add any WebSphere Application Server process as a Windows service.

WebSphere Application Server processes that you could add as Windows services include:

- Any base Application Server, such as the default server1 process
- The jmsserver process that exists on a base Application Server node when you have installed the embedded messaging server feature and the node is part of a deployment manager cell
- The nodeagent process on a base Application Server node that is part of a deployment manager cell
- The deployment manager process, dmgr, on the Network Deployment node

If the installation wizard creates a Windows service during installation, the uninstaller program removes the Windows service. If you use the **WASService** command to create a service yourself, it is your responsibility to remove the service when it is no longer valid. The uninstaller program does not remove Windows services that you create with the **WASService** command.

## User ID prerequisites

You must run the **WASService.exe** command from a user ID that belongs to the administrator group. The user ID must also have the following advanced user rights:

- Act as part of the operating system
- Log on a service

## Location of command file

The **WASService.exe** command file is located in the *install\_root*\bin directory.

## WASService.exe command syntax for starting an existing service

The command syntax is as follows:

```
WASService.exe [-start] service_name
```

## WASService.exe command syntax for creating a service

The command syntax is as follows:

```
WASService.exe -add service_name
 -serverName Server
 [-wasHome install_root]
 [-configRoot configuration_repository_directory]
 [-startArgs additional_start_arguments]
 [-stopArgs additional_stop_arguments]
 [-userid execution_id -password password]
```

```
[-logFile service_log_file]
[-logRoot server_log_directory]
[-userScript path\setupCmdLine.bat]
[-restart true | -restart false]
```

### **WASService.exe command syntax for deleting a service**

The command syntax is as follows:

```
WASService.exe -remove service_name
```

### **WASService.exe command syntax for stopping a running service**

The command syntax is as follows:

```
WASService.exe -stop service_name
```

### **WASService.exe command syntax for retrieving service status**

The command syntax is as follows:

```
WASService.exe -status service_name
```

### **Parameters**

Supported arguments include:

**-add** *service\_name*

Creates a service named *service\_name*.

**-configRoot** *configuration\_repository\_directory*

Optional parameter that identifies the configuration directory of the installation root directory of a WebSphere Application Server product.

**-logFile** *service\_log\_file*

Optional parameter that identifies a log file that the **WASService** command uses to record its activity.

**-logRoot** *server\_log\_directory*

Optional parameter that identifies a wsinstance server log directory that the **WASService** command uses to determine if the server is running.

**-remove** *service\_name*

Deletes the specified service.

**-restart true | false**

Restarts the existing service automatically if the service fails when set to true.

**-server** *Server\_name*

Identifies the server that the service controls.

**-start** *service\_name*

Starts the existing service. The **-start** parameter is the default parameter; you can omit the **-start** parameter and still start the service.

**-startArgs** *additional\_start\_arguments*

Optional parameter that identifies additional parameters.

**-status** *service\_name*

Returns the current status of the service, which includes whether the service is running or stopped.

**-stop** *service\_name*

Stops the specified service.

**-stopArgs** *additional\_stop\_arguments*

Optional parameter that identifies additional parameters.

**-userid** *execution\_ID* **-password** *password*

Optional parameters that identify a privileged user ID and password.

**-userScript** *path\setupCmdLine.bat*

Optional parameter that identifies the **setupCmdLine.bat** command file that the **WASService** command uses to understand wsinstance nodes.

**-wasHome** *install\_root*

Optional parameter that identifies the installation root directory of the WebSphere Application Server product.

### Creating a deployment manager service

This example creates a service called *IBMWAS5Service - dmgr* that starts the dmgr process:

```
WASService -add dmgr
 -servername dmgr
 -wasHome "D:\Program Files\WebSphere\DeploymentManager"
 -logfile "D:\Program Files\WebSphere\DeploymentManager\logs\dmgr\startServer.log"
 -logRoot "D:\Program Files\WebSphere\DeploymentManager\logs\dmgr"
 -restart true
```

### Creating a node agent service

This example creates a service called *IBMWAS5Service - nodeagent* that starts the nodeagent process:

```
WASService -add nodeagent
 -servername nodeagent
 -wasHome "D:\Program Files\WebSphere\AppServer"
 -logfile "D:\Program Files\WebSphere\AppServer\logs\nodeagent\startServer.log"
 -logRoot "D:\Program Files\WebSphere\AppServer\logs\nodeagent"
 -restart true
```

### Creating an Application Server service

This example creates a service called *IBMWAS5Service - server2* that starts an Application Server process:

```
WASService -add server2
 -servername server2
 -wasHome "D:\Program Files\WebSphere\AppServer"
 -logfile "D:\Program Files\WebSphere\AppServer\logs\server2\startServer.log"
 -logRoot "D:\Program Files\WebSphere\AppServer\logs\server2"
 -restart true
```

### Example of successful result

```
C:\install_root\bin> WASService -add testserver1 -serverName server1
```

```
Adding Service: testserver1
Config Root: C:\WebSphere\AppServer\config
Server Name: server1
Was Home: C:\WebSphere\AppServer\
Start Args:
Restart: 1
IBM WebSphere Application Server V5 - testserver1 service successfully added.
```

### Example result when the user ID does not have the proper authorization

```
WASService -add "NodeAgent" -serverName nodeAgent -username websphereadm -password pass1
```

```
Adding Service: NodeAgent
Config Root: C:\WebSphere\AppServer\config
Server Name: nodeAgent
```

```
Was Home: C:\WebSphere\AppServer\
Start Args: -username websphereadm -password pass1
Restart: 1
Failed to add service: 1057
```

To correct this problem, grant the user ID the proper authorization and advanced user rights.

---

## firststeps command

The **firststeps** command starts the First Steps tool.

First Steps is a post-installation ease-of-use tool for directing WebSphere Application Server elements from one place. Options dynamically appear on the First Steps panel, depending on features you install. With all options present, you can use First Steps to start or stop the Application Server, verify the installation, access the information center, access the administrative console, access the Samples Gallery, or launch the product registration.

First Steps starts automatically at the end of the installation. If it is not running, start First Steps from the command line:

### Location of the command file

- 5.1 +** Beginning with V5.1, the location of the `firststeps.sh` or `firststeps.bat` script is:
- On Linux and UNIX-based server platforms: `install_root/firststeps/firststeps.sh`
  - On Windows platforms: `install_root\firststeps\firststeps.bat`

### Parameters

No parameters are associated with this command.

### Syntax for the firststeps command on a Linux or UNIX-based platform

The command syntax is as follows:

```
./firststeps.sh
```

### Syntax for the firststeps command on a Windows platform

The command syntax is as follows:

```
firststeps.bat
```

### Usage tips

If this is the first time you have used the First Steps panel since installing the product, click **Verify installation** to make sure that all is well with your installation. The verification process starts the server for you. Do not select the **Start the server** option at the same time that the installation verification test (IVT) is running. There is a known problem if you select the **Start the server** option at the same time that the IVT is running.

After selecting the **Start the server** option, an output screen displays with status messages. The success message informs you that the server is open for business. Then the menu item changes to **Stop the server**.

Do not select any other options from the First Steps panel after selecting the **Start the server** option. There is a known problem that prevents the proper operation of the installation verification test when you select the **Verify Installation** option after selecting the **Start the server** option. Stop the server before verifying the installation.

The installation verification test is not supported on node names that contain double-byte characters. See “Troubleshooting the installation” on page 185 for more information.

The Web administrative console is a configuration editor that runs in a Web browser. The administrative console lets you work with Application Server configuration files encoded in XML. Changes made using the administrative console take effect the next time you start the Application Server. To launch the administrative console, click **Administrative console**. You can also point your browser to `http://your_host_name:9090/admin` to start the administrative console. Substitute your own host name in the address. As the administrative console opens, it prompts you for a login name. This is not a security item, but merely a tag that allows you to identify configuration changes you make during the session.

From the First Steps menu, click on **Samples gallery** to explore the sample applications. Alternatively you can point your browser directly to `http://DNS_host_name:9090/WSsamples`. This command is case sensitive. Substitute your own host name in the address.

**5.1 +** If the Mozilla browser is your default browser, some of the links on the First Steps panel do not work. If you cannot use the First Steps panel, launch the links directly from your browser.

You can also try a symbolic link to the Mozilla browser to fix the problem as shown in the following example:

```
ln -sf /usr/bin/mozilla /usr/bin/netscape
```

**5.1 +** Links on the First Steps panel for the Network Deployment product:

#### Button Link

#### Product Overview

<http://www.ibm.com/software/webservers/appserv/>

It is possible that your security options might prevent redirection, such as to the <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=products/appserv> Web site. If so, you can identify the IBM Web site as a trusted site to allow the redirection. For example, on Internet Explorer, add <http://www-306.ibm.com> to the list of Trusted Sites by selecting **Tools > Internet Options > Security (tab) > Trusted Sites**.

#### WebSphere information center

<http://publib.boulder.ibm.com/infocenter/ws51help/index.jsp>

#### Administrative Console

<http://localhost:9090/admin>

Change the First Steps port number for the administrative console option to match the port number in a coexistence scenario. See “Default coexistence settings for port numbers” on page 366.

Edit the administrative console address when First Steps attempts to open the administrative console Web address. Change the port to match the coexistence port number for the administrative console port. The following example shows how to change the address to the default coexistence port number for the administrative console port:

```
http://localhost:9091/admin
```

#### Viewing the registration page in other languages

The **Register the product** entry on the First Steps menu displays the default welcome page in English.

#### Register the product

```
install_root/prt/PRT>Welcome.html
```

To view other language versions of the page, browse the PRT directory to select a version. Use the last three letters before the .html extension of the file name to identify the language. For example, the following versions are included:

- German: PRT\_Welcome\_deu.html
- Spanish: PRT\_Welcome\_esp.html
- French: PRT\_Welcome\_fra.html
- Italian: PRT\_Welcome\_ita.html
- Portuguese: PRT\_Welcome\_ptb.html

---

## Using the installation verification test

After installing the product, you are ready to use the installation verification test (IVT).

After installing the base product during an umbrella installation, you are ready to use the installation verification test (IVT).

The IVT program scans product log files for errors and verifies core functionality of the product installation.

1. If you started the administrative server of a previous WebSphere Application Server product from V3.5.x and later, or V4.0.x or later, to let the installation wizard export its configuration and applications to Version 5, stop the server before running the IVT. Otherwise, the Version 5 server might not start due to port conflicts between the two servers.
2. Select **Verify Installation** from the First Steps panel and observe the results in the First Steps status window. A log file for the installation is created and stored in the logs directory with the name of `ivt.log`. Do not select any other options from the First Steps panel after selecting **Verify Installation** and while the IVT is running. There is a known problem if you select the **Start the server** option at the same time that the IVT is running.
3. If the First Steps tool is not running, start it from the `/bin/firststeps` directory.
  - On Linux and UNIX-based server platforms: `install_root/firststeps/firststeps.sh`
  - On Windows platforms: `install_root\firststeps\firststeps.bat`

You can also run First Steps from the Start Menu on Windows platforms.

4. You can also start the IVT tool from the `/bin` directory.
  - On Linux and UNIX-based platforms: `./ivt.sh`
  - On Windows platforms: `ivt`

The IVT program starts the server automatically if the server is not running. Once the server initializes, the IVT runs a series of verification tests and reports pass or fail status in a console window. It also logs results to the `install_root/logs/ivt.log` file. As the IVT verifies your system, it reports any errors it detects.

### ivt command

The `ivt` command starts the installation verification test (IVT) program.

The IVT program starts the deployment manager process automatically if the process is not already running. Once the process initializes, the IVT runs a series of verification tests and reports pass or fail status in a console window.

The IVT program scans product log files for errors and verifies core functionality of the product installation.

You can start the IVT program from the command line or from the First Steps menu.

The installation verification test is not supported on node names that contain double-byte characters.



## Location of the command file

The command file is a script. On Linux and UNIX platforms, the command file is named `ivt.sh` in the `install_root/bin` directory. On Windows platforms, the command file is named `ivt.bat` in the `install_root\bin` directory.

## Parameters

No parameters are associated with this command.

## Syntax for the `ivt` command on a Linux or UNIX-based platform

The command syntax is:

```
ivt.sh
```

Issue the command from the `install_root/bin` directory.

## Syntax for the `ivt` command on a Windows platform

The command syntax is:

```
ivt.bat
```

Issue the command from the `install_root\bin` directory.

## Logging

The `ivt` command logs results to the `install_root/logs/ivt.log` file.

---

## Troubleshooting the installation

If you did not receive the *Successful installation* message or the *INSTFIN: The installation is complete.* line in the installation log, use this information to troubleshoot the installation.

1. Install the base product first before installing the Network Deployment product when installing both products on the same machine. Install the base product before installing the Network Deployment product when installing both products on the same machine. The embedded messaging feature is included in the default installation.

Table 35. Installation tip

| Operating platform | Tip                                                                                                                             |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------|
| All platforms      | Installing WebSphere Application Server products in order on the same machine, when installing the embedded messaging component |

2. To manage multiple base product Application Servers, install the Network Deployment product. The WebSphere Application Server, Version 5.x product does not provide centralized management of multiple servers. The WebSphere Application Server Network Deployment, Version 5.x product provides this function.  
The deployment manager process (`dmgr`) in the Network Deployment product manages a cell of Application Servers. You can federate a base Application Server into the cell or unfederate it. While federated, the configuration of the base Application Server is managed from the deployment manager.
3. Use the First Steps tool to run the installation verification test (IVT). Select **Verify installation**. Check the `install_root\logs\ivt.log` file for a summary of test results. Correct any errors and retry. The installation wizard automatically starts the First Steps tool at the end of installation.
4. Check the installation log files for errors after installing:

During installation, a single entry in the `install_root/logs/log.txt` file points to the temporary log file, either `%TEMP%\log.txt` on Windows platforms, or `/tmp/log.txt` on Linux and UNIX-based platforms. The installation program copies the file from the temporary directory to the `install_dir/logs/log.txt` location at the end of the installation. In cases where the installation fails and the `install_dir/logs/log.txt` has only this one entry to the temporary directory, open the `logs.txt` file in the temporary directory for clues to the installation failure. Uninstalling creates the `install_root/logs/uninstlog.txt` file.

The following table shows the installation log locations when installing the application clients for Version 5.1. The location is fixed and might not agree with the installation root location if you specified a location during installation.

Table 36. Installation log locations when installing the WebSphere Application Server Application Clients

| Windows system log path name                                     | Linux and UNIX-based operating system log path name  |
|------------------------------------------------------------------|------------------------------------------------------|
| C:\Program Files\WebSphere\AppClient\logs\WAS.Client.install.log | /opt/WebSphere/AppClient/logs/WAS.Client.install.log |

Table 37. Installation log locations when installing WebSphere Application Server

| Component                                                                  | Installation log path name                   |                                                                  |
|----------------------------------------------------------------------------|----------------------------------------------|------------------------------------------------------------------|
|                                                                            | Windows system temp directory                | Linux and UNIX-based operating system: /tmp/                     |
| Embedded messaging feature (based on WebSphere MQ) prerequisites error log |                                              | mq_prereq.log                                                    |
| Component                                                                  | Windows platforms: <i>install_root\logs\</i> | Linux and UNIX-based operating system: <i>install_root/logs/</i> |
| WebSphere Application Server                                               | log.txt<br>uninstlog.txt                     |                                                                  |
| IBM HTTP Server                                                            |                                              | ihs_log.txt                                                      |
| Embedded messaging feature (based on WebSphere MQ) installation log        |                                              | mq_install.log                                                   |
| Embedded messaging feature (based on WebSphere MQ) configuration log       |                                              | createMQ.node.server1.log                                        |
| Default application                                                        |                                              | installDefaultApplication.log                                    |
| Samples Gallery                                                            |                                              | installSamples.log                                               |
| Administrative console                                                     |                                              | installAdminConsole.log                                          |
| Migration tools                                                            |                                              | WASPostUpgrade.log                                               |

Table 38. Installation log locations when installing Network Deployment

| Component                                                                  | Installation log path name                   |                                                                  |
|----------------------------------------------------------------------------|----------------------------------------------|------------------------------------------------------------------|
|                                                                            | Windows system temp directory                | Linux and UNIX-based operating system: /tmp/                     |
| Embedded messaging feature (based on WebSphere MQ) prerequisites error log |                                              | mq_prereq.log                                                    |
| Component                                                                  | Windows platforms: <i>install_root\logs\</i> | Linux and UNIX-based operating system: <i>install_root/logs/</i> |

Table 38. Installation log locations when installing Network Deployment (continued)

| Component                                                           | Installation log path name    |                                              |
|---------------------------------------------------------------------|-------------------------------|----------------------------------------------|
|                                                                     | Windows system temp directory | Linux and UNIX-based operating system: /tmp/ |
| Network Deployment                                                  |                               | log.txt<br>uninstlog.txt                     |
| Embedded messaging feature (based on WebSphere MQ) installation log |                               | mq_install.log                               |
| Administrative console                                              |                               | installAdminConsole.log                      |
| File transfer                                                       |                               | installFiletransfer.log                      |
| Migration tools                                                     |                               | WASPostUpgrade.log                           |

**5.1+** If the installation program cannot locate the `WAS_update` directory from disk 2, the installation program fails immediately. The immediate failure leaves the `WAS.WBISF.install.log` file only in the `/tmp` directory (or in the system `%TEMP%` directory on Windows platforms). Normally, the installation program creates installation logs in the `install_root/logs` directory. In some instances, such as the one where the `WAS_update` directory does not exist, the installation program fails before it can copy the log files from the system temporary directory to the `/logs` directory. If you cannot find the installation log files in the `install_root/logs` directory, look in the system temporary directory.

- See if there are files in the `install_root/classes` directory. When IBM Support queues work for you and provides you test or debugging fixes, you are supposed to place the fixes in the `install_root/classes` directory. By default, the `install_root/classes` directory is picked up first in the WebSphere Application Server class path to let it override other classes.

This directory lets you verify or debug a fix. After accepting the test fix or finishing with the debugging of the debug fix, you are supposed to delete the fix from the `install_root/classes` directory to return the system to a working state. If you do not remove such fixes from the `install_root/classes` directory, you can experience errors.

- Turn on tracing if the installation logs do not contain enough information to determine the cause of the problem.
  - Report the `stdout` and `stderr` logs to the console window, by adding the `-is:javaconsole` parameter to the **install** command:

- **5.1+** On Linux and UNIX-based platforms:

```
install -is:javaconsole
```

or capture the stream to a file with the following commands:

```
install -is:javaconsole > captureFileName.txt 2>&1
```

- On Windows platforms:

```
Install.exe -is:javaconsole
```

or capture the stream to a file with the following commands:

```
Install.exe -is:javaconsole > drive:\captureFileName.txt
```

- Capture additional information to a log of your choice with the `-is:log filename` option.
- Turn on additional installation logging by passing the `-W Setup.product.install.logAllEvents="true"` parameter to the **install** command:

- **5.1+** On Linux and UNIX-based platforms:

```
install -W Setup.product.install.logAllEvents="true"
```

- On Windows platforms:

```
Install.exe -W Setup.product.install.logAllEvents="true"
```

If you install on an AIX 5.1 system, with maintenance level 2, it is possible that the Web-based system manager, a standard component of AIX systems, already uses port 9090. When starting the server you get information that port 9090 is already in use. To resolve the conflicting port use, change the port assignment for the HTTP\_TRANSPORT\_ADMIN port in the server.xml file. The file path is:

```
usr/websphere/appserver/config/cells
/cell/nodes/node/servers/server1/server.xml
```

7. Use the First Steps tool or the command line method to start the Application Server. **To start the server from the command line:**

- On Linux and UNIX-based platforms: `install_root/bin/startServer.sh server1`
- On Windows platforms: `install_root\bin\startServer server1`

If you enable security, specify the `-user` and the `-password` parameters of the command.

8. Verify whether the server starts and loads properly, by looking for a running Java process and the *Open for e-business* message in the `SystemOut.log` and `SystemErr.log` files. If no Java process exists or if the message does not appear, examine the same logs for any miscellaneous errors. Correct any errors and retry. You can find the `SystemOut.log` and `SystemErr.log` files in the `install_root/logs/server1` (Linux and UNIX-based platforms) or `install_root\logs\server1` (Windows platforms) directory.

9. Use the First Steps tool or the command line method to stop the Application Server, if it is running, and to start the deployment manager. **To stop the server from the command line:**

- On Linux and UNIX-based platforms: `install_root/bin/stopServer.sh server1`
- On Windows platforms: `install_root\bin\stopServer server1`

If you enable security, specify the `-user` and the `-password` parameters of the command.

**To start the deployment manager from the command line:**

- On Linux and UNIX-based platforms: `install_root/bin/startManager.sh`
- On Windows platforms: `install_root\bin\startManager`

10. Verify that the server starts and loads properly by looking for a running Java process and the *Server dmgr open for e-business* message in the `dmgr_stdout.log` and `dmgr_stderr.log` files. If no Java process exists or if the message does not appear, examine the same logs for any miscellaneous errors. Correct any errors and retry.

11. Refer to the plug-in configuration documentation, if you have installed plug-ins and the Web server does not come up properly.

12. Start the Snoop servlet.

In a Network Deployment environment, the Snoop servlet is available in the cell only if you included the DefaultApplication when adding the Application Server to the cell. The `-includeapps` option for the **addNode** command migrates the DefaultApplication to the cell. If the application is not present, skip this step.

- a. Start the Application Server.
- b. Start the IBM HTTP Server or the Web server that you are using. Use a command window to change the directory to the IBM HTTP Server installed image, or to the installed image of your Web server. Issue the appropriate command to start the Web server, such as these commands for IBM HTTP Server:

**To start the IBM HTTP Server from the command line:** Access the **apache** and **apachectl** commands in the `IBMHttpServer/bin` directory.

- On Linux and UNIX-based platforms: `./apachectl start`
  - On Windows platforms: `apache`
- c. Point your browser to `http://localhost:9090/snoop` to test the internal HTTP transport provided by the Application Server. Point your browser to `http://localhost/snoop` to test the Web server plug-in.
  - d. Verify that Snoop is running. Either Web address should display the Snoop Servlet - Request/Client Information page.

13. Start the WebSphere Application Server administrative console.

- a. Start the Application Server.
- b. Point your browser to `http://localhost:9090/admin`.
- c. Type any ID and click **OK** at the administrative console window.

The server starts. The administrative console starts. You can access the administrative console through the browser. The administrative console accepts your login.

14. Federate the base Application Server into the cell. **To add the base Application Server into the cell:**
  - On Linux and UNIX-based platforms: `install_root/AppServer/bin/addNode.sh localhost 8879`
  - On Windows platforms: `install_root\AppServer\bin\addNode.bat localhost 8879`

If you enable security, specify the `-user` and the `-password` parameters of the command.

15. Verify that the Application Server was incorporated into the cell. The command window displays a sequence of messages when you issue the **addNode** command:

```
Begin federation of node xxxx with deployment manager at localhost:8879.
Successfully connected to deployment manager Server: localhost:8879
Creating node agent configuration for node: xxxx
Reading configuration for node agent process: nodeagent
Adding node xxxx configuration to cell: AdvancedDeploymentCell
Performing configuration synchronization between node and cell.
Launching node agent process for node: xxxx
Node agent launched. Waiting for initialization status.
Node agent initialization completed successfully. Process ID is: 3012
Node xxxx has been successfully federated.
```

The last message is an indicator of success. A second Java process is running, which is the `nodeagent` process. The `stdout.log` file and `stderr.log` file in the `node_name` directory contains a `Server node_name open for e-business message`.

16. Resolve any IP address caching problems.

By default, the Java 2 SDK caches the IP address for the domain name service (DNS) naming lookup. After resolving the host name successfully, the IP address stays in the cache. By default, the cache entry remains forever. This default IP caching mechanism can cause problems, as described in the following problem scenarios.

#### **Problem scenario 1**

Suppose the Application Server at `host1.ibm.com` has an initial IP address of `1.2.3.4`. When a client at `host2.ibm.com` conducts a DNS lookup of `host1.ibm.com`, the client stores the `1.2.3.4` address in the cache. Subsequent DNS name lookups return the cached value, `1.2.3.4`. The cached value is not a problem until the `host1.ibm.com` IP address changes, to `5.6.7.8`, for example. The client at `host2.ibm.com` does not retrieve the current IP address, but always retrieves the previous address from the cache. If this scenario occurs, the client cannot reach `host1.ibm.com` unless you stop and restart the client process.

#### **Problem scenario 2**

Suppose the Application Server at `host1.ibm.com` has an initial IP address of `1.2.4.5`. Although the IP address of the application server does not change, a network outage can record an exception code as the IP address in the cache, where it remains until the client is restarted on a working network. For example, if the client at `host2.ibm.com` disconnects from the network because of an unplugged cable, the disconnected lookup of the Application Server at `host1.ibm.com` fails. The failure causes the IBM Developer Kit to put the special exception code entry into the IP address cache. Subsequent DNS name lookups return the exception code, which is `java.net.UnknownHostException`.

#### **IP address caching and WebSphere Application Server process discovery**

If you change the IP address of a federated WebSphere Application Server node, processes running in other nodes cannot contact the changed node until you stop and restart them.

If a deployment manager process starts on a disconnected node, it cannot communicate with cell member processes until you stop and restart the deployment manager process. For example, plugging in an unplugged network cable does not restore proper addresses in the IP cache until the deployment manager process is restarted.

### Using the IP address cache setting

You can always stop and restart a deployment manager process to refresh its IP address cache. However, this process might be expensive or inappropriate.

The `networkaddress.cache.ttl` (public, JDK1.4) and `sun.net.inetaddr.ttl` (private, JDK1.3) parameters control IP caching. The value is an integer that specifies the number of seconds to cache IP addresses. The default value, -1, specifies to cache forever. A value of 0 specifies to never cache.

Using a zero value is not recommended for normal operation. If you do not anticipate network outages or changes in IP addresses, use the cache forever setting. The never caching setting introduces the potential for DNS spoofing attacks.

### For more information about the Java 2 SDK

The Java 2 SDK, Standard Edition 1.4 Web site at

<http://java.sun.com/j2se/1.4/docs/guide/net/properties.html> describes the private `sun.net.inetaddr.ttl` property, which works in both Java 2 SDK, Standard Edition 1.3 (WebSphere Application Server V5.0.0, V5.0.1, and V5.0.2) and Java 2 SDK, Standard Edition 1.4 (WebSphere Application Server V5.1). The *Networking* section of the Java 2 SDK, Standard Edition 1.4 Web site at <http://java.sun.com/j2se/1.4/jcp/beta/> describes a change in the behavior of the `java.net.URLConnection` class.

17. Avoid segmentation faults when installing on Red Hat Enterprise Linux 3.0 U1.

If you are installing from an operator console attached to the RHEL 3 U1 machine and you receive a message that is similar to the following message, you might be experiencing a known limitation of RHEL 3 U1:

```
A suitable JVM could not be found
```

Red Hat Enterprise Linux 3.0 causes a segmentation fault when calling the JVM. The problem occurs when you run the installation program, log off of the root user, log back on to root, and run the installation again on the operator console that is attached to the machine (not a telnet session).

The installation fails with a segmentation fault.

This is a known limitation of Red Hat Enterprise Linux V3.0 that causes a segmentation fault when calling the JVM.

Test the JVM to see if it is failing by running the following command:

```
/mnt/cdrom/platform/linuxi386/jdk/java/jre/bin/java -version
```

If you receive a "segmentation fault" message, reboot your machine or press Ctrl-x to reinstall.

Rebooting the machine or pressing Ctrl-x resolves the problem.

18. Preserve symbolic links when copying product CDs to network file server (NFS) disks. When copying a CD for one operating system platform to a network file system (NFS) disk on another type of operating system using the `cp` command, you can encounter errors such as those in the following example:

```
a file is bad
no such file or directory exists
libcSup.2 cannot be accessed
```

The copy error occur when incorrectly copying symbolic link files.

An example of such an error occurs when copying an HP-UX CD image onto an AIX platform with the `cp -frp` command. The default `cp` command behavior on AIX is to resolve the symbolic links by copying the files to which the symbolic links point. Errors occur when a symbolic link resolves to a platform-specific library or file that is not present on the NFS operating system.

Use options on the copy command of the NFS system to copy symbolic links instead of resolving them. For example, the `-h` option on the `cp` command of the AIX platform copies symbolic links from the HP-UX CD to the NFS disk on the AIX platform.

Even with the `-h` option, the `cp` command on a Solaris platform does not preserve symbolic links when copying an HP-UX disk. On a Solaris platform, use the `tar -cvf` command to copy data from an HP-UX disk and preserve the symbolic links:

- a. Insert disk 2 for HP-UX platforms into the drive on the Solaris system.
- b. Close the file explorer window if one opens.
- c. Open a command window.
- d. Change directories:  
`CD /cdrom/cdrom0`
- e. Issue the following command:  
`tar cvf * /workarea/filename.tar`
- f. Change directories:  
`CD /workarea`
- g. Issue the following command:  
`tar -xvf filename.tar`

Consult the man page for the copy command on the NFS system to understand how the platform supports copying symbolic links.

19. Pick up secondary user groups for root before installing the embedded messaging feature.

On many systems, such as SuSe Linux, if you telnet and issue the `id` command or the `groups` command, you cannot see the groups `mqm` or `mqbrkr`s even though they might exist. Solve this problem in one of two ways:

- Use the `ssh` command to log in
- Issue the `su -` command

After using one of the commands, verify the required groups with the `id` command or the `groups` command.

In a normal root login, issue the `su` command. For a real root login, issue the `su -` command.

Display settings for a normal root login are automatic. For a real root login, you must set your display environment properly to successfully view the GUI installation wizard. Otherwise, you see a message about Preparing Java(tm) Virtual Machine... and seven rows of dots, but no installation GUI and no further messages. Refer to the documentation for your platform to determine proper display settings.

If you see the following messages in the `SystemOut.log` file, you have not picked up the required secondary groups for root:

```
[date time CDT] 60cf2faf JMSRegistrati
 A MSGS0601I: WebSphere Embedded Messaging has not been installed
[date time CDT] 60cf2faf JMSEmbeddedPr A MSGS0050I:
 Starting the Queue Manager
[date time CDT] 60cf2faf JMSEmbeddedPr E MSGS0058E:
 Unable to start the JMS Server as WebSphere Embedded Messaging has not been installed
[date time CDT] 60cf2faf JMSService E MSGS0001E:
 Starting the JMS Server failed with exception:
 java.lang.Exception: MSGS0058E:
 Unable to start the JMS Server as
 WebSphere Embedded Messaging has not been installed
```

Also, the following associated messages are added to the `mq_install.log` file:

```
wmsetup: date time Checking if user "root" is in group "mqm"
wmsetup: date time
wmsetup: date time ERROR: Group "mqm" exists, id "root" is
 defined to the group but does not
wmsetup: date time have the group in its current set of effective groups.
wmsetup: date time Current group membership is :
wmsetup: date time uid=0(root) gid=0(system) groups=2(bin)
wmsetup: date time You may need to login.
wmsetup: date time
wmsetup: date time ... RC 4 from Check_root
```

```

wmsetup: date time ERROR: User "root" not in group "mqm"
wmsetup: date time Check_root mqbrkrs
wmsetup: date time Checking for group "mqbrkrs" ...
wmsetup: date time lsgroup returned "mqbrkrs id=203
 admin=false users=root adms=root registry=files " RC=0
wmsetup: date time Checking if user "root" is in group "mqbrkrs"
wmsetup: date time
wmsetup: date time ERROR: Group "mqbrkrs" exists, id "root" is
 defined to the group but does not
wmsetup: date time have the group in its current set of effective groups.
wmsetup: date time Current group membership is :
wmsetup: date time uid=0(root) gid=0(system) groups=2(bin)
wmsetup: date time You may need to login.
wmsetup: date time
wmsetup: date time ... RC 4 from Check_root
wmsetup: date time ERROR: User "root" not in group "mqbrkrs"

```

20. Do not use the installation verification test (IVT) from the First Steps program or by running the **ivt.sh** command (or the **ivt.bat** command on a Windows platform) if the node name contains double-byte characters. The installation verification test is not supported on node names that contain double-byte characters. You can receive an error that is similar to the following example:

```

java.lang.reflect.InvocationTargetException
 at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
 at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
 at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:41)
 at java.lang.reflect.Method.invoke(Method.java:386)
 at com.ibm.ws.bootstrap.WSLauncher.main(WSLauncher.java:94)Caused
 by: java.lang.NumberFormatException: For input string: ""
 at java.lang.NumberFormatException.forInputString(NumberFormatException.java:62)
 at java.lang.Integer.parseInt(Integer.java:469)
 at java.lang.Integer.parseInt(Integer.java:498)
 at com.ibm.websphere.ivt.client.ivtClient.main(Unknown Source)
 ...

```

21. Starting the Launchpad program for WebSphere Application Server clients, Version 5.1 using the Konqueror file in the K Desktop Environment on SuSE Linux Enterprise Server (SLES) causes an error.

When starting the Launchpad program for WebSphere Application Server clients, Version 5.1 using the Konqueror file manager in the K Desktop Environment (KDE) on Linux systems, a "Couldn't find the program launchpad.sh" error occurs.

Because the **launchpad.sh** command uses a relative path to locate the Java program, you should run the **launchpad.sh** command from the directory where the **launchpad.sh** command is located for the client program. When using the Konqueror file manager to issue the **launchpad.sh** command, the current directory is your home directory. Therefore, the **launchpad.sh** command cannot work.

You can troubleshoot the installation.

The Troubleshooting or problem determination contains more detailed debugging and reporting instructions. See Installation component troubleshooting tips for more information about troubleshooting the installation.

For current information available from IBM Support on known problems and their resolution, see the IBM Support page.

IBM Support has documents that can save you time gathering information needed to resolve this problem. Before opening a PMR, see the IBM Support page.



---

## Creating multiple Version 5 configuration instances

*Multiple configuration instances*, as it applies to WebSphere Application Server products, is the ability to install the base WebSphere Application Server product or the Network Deployment product once, and to use the **wsinstance** command to create multiple instances of the initial installation, all running on the same machine at the same time.

In contrast to multiple configuration instances is *coexistence*, which refers to multiple *installations* of WebSphere Application Server, running on the same machine at the same time. Both coexistence and multiple configuration instances of the base WebSphere Application Server product imply various combinations of Web server interaction.

You can find a description of coexistence in “Migrating and coexisting” on page 311.

You can install the base WebSphere Application Server product or the Network Deployment product one time and use the `wsinstance` tool in the `install_root\bin\wsinstance` folder to create multiple configuration instances. Each WebSphere Application Server configuration instance is a stand-alone instance with a unique name, and its own set of configuration files and user data folders. Configuration folders include `config`, `etc` and `properties`. User data folders include `installedApps`, `installableApps`, `temp`, `logs`, `tranlog` and `wstemp`. Each instance has the administrative console application to manage the configuration instance. The configuration instance shares all run-time scripts, libraries, the Software Development Kit, and other files with the initial Application Server.

Each configuration instance of the Network Deployment product is a stand-alone deployment manager (`dmgr`) with its own set of unique configuration files and user data folders for the cell. Configuration folders include `config`, `etc` and `properties`. User data folders include `installedApps`, `installableApps`, `temp`, `logs`, `tranlog` and `wstemp`. The `dmgr` configuration instance also includes the administrative console and the file transfer applications to manage the configuration instance.

You can configure and operate each configuration instance independently of other instances.

These limitations apply to multiple configuration instances:

- You cannot federate a configuration instance of the base product into a deployment manager cell. A deployment manager cannot manage a configuration instance.
- You cannot use the **Uninstall.exe** or **uninstall** command to remove a configuration instance. You must remove configuration instances with the `-delete` option of the **wsinstance** command, before uninstalling the installation instance.
- You cannot migrate the configuration and applications from an earlier release to a Version 5 configuration instance.

Reasons to use configuration instances include:

- You install the product only once.
- You can have stand-alone environments without the space or procedural constraints of multiple, full product installations. (Configuration instances use a minimal amount of space.)
- You can have a centralized development environment that is easy to maintain, where multiple developers and testers share one machine.
- You can create identical environments, where everyone is developing, testing, or performing in an identical workspace.

One reason to not use configuration instances is that you cannot federate a configuration instance of the base product into a deployment manager cell. A deployment manager cannot manage a configuration instance.

Use the following procedure to create and configure multiple V5 instances:

1. Install the Network Deployment product.

2. Back up the `install_root/bin/wsinstance/configdefaults` directory. In certain situations, it might become necessary to restore the following files from the backup to the `install_root/bin/wsinstance/configdefaults` directory:

- `./security.xml`
- `./nodes/$NODE_NAME/resources.xml`
- `./nodes/$NODE_NAME/serverindex.xml`
- `./nodes/$NODE_NAME/servers/$SERVER_NAME/resources.xml`

See the description of a possible error that can occur for more information.

3. Use the **wsinstance** command in the `WAS_HOME\bin\wsinstance` folder, to create a configuration instance of the server you installed in step 1. Refer to the description of the **wsinstance** command to learn more about the command, and to see examples of use.

You must specify a unique directory path and a unique node name for each configuration instance.

You must also specify unique port numbers for each configuration instance. For example, on a Windows platform, specify ports beginning at 20002, for node `shasti`, in configuration instance `instance_root G:\shasti\WebSphere`, on the `planetjava` machine, by issuing this command:

```
wsinstance.bat -name shasti
 -path G:\shasti\WebSphere
 -host planetjava
 -startingPort 20002
 -create
```

This command creates a separate set of configuration and other data files.

4. Source the **setupCmdLine.sh** (or run the **setupCmdLine.bat**) script in the `bin` directory of the `instance_root` folder to set the WebSphere Application Server environment to the configuration instance.

Or you can set `WAS_USER_SCRIPT` to `instance_root\bin\setupCmdLine.bat`, which has the same effect as sourcing the `setupCmdLine.sh` script or running the `setupCmdLine.bat` script.

On Linux and UNIX-based platforms, *sourcing* the script to the parent shell allows child processes to inherit the exported variables. Source the command by using the following format:

```
./setupCmdLine.sh (there is a space between the period and the slash)
```

The format varies per platform:

- `./install_root/bin/setupCmdLine.sh` (source the command on UNIX platforms - There is a space between the period and the installation root directory.)
- `source install_root/bin/setupCmdLine.sh` (source the command on Linux platforms )
- `install_root\bin\setupCmdLine.bat` (Windows platforms only)

Or, you can set `WAS_USER_SCRIPT` to `instance_root/bin/setupCmdLine.sh`, which has the same effect as sourcing the **setupCmdLine** command. For example:

```
export WAS_USER_SCRIPT=/opt/inst1/bin/setupCmdLine.sh
```

After completing this step, you can start the application server configuration instance with the **startManager** or **startServer** command.

If you are using the embedded messaging feature, and you are logging on as a non-root user on a UNIX-based or Linux platform, you must add the `mqm` user and the `root` user to the `mqm` and `mqbrkr` operating system groups, as described in the Installing WebSphere embedded messaging as the JMS provider topic.

You must download and apply any interim fixes for the embedded messaging feature to each configuration instance:

- Download interim fixes for the embedded messaging feature from the WebSphere MQ Service download site. The readme file that is bundled with the interim fix describes how to install the interim fix.
- Download any fix packs for the product from the IBM Support site for WebSphere Application Server. The readme file that is bundled with the fix pack describes how to use the update installer

(also known as the updateInstaller program or the update installation wizard) for WebSphere Application Server products to install the fix pack. Installing a fix pack removes all interim fixes. You might have to reinstall interim fixes that are not in the fix pack to bring the embedded messaging feature back to the fix level you require.

5. Federate multiple installation instances into a deployment manager cell.
6. Create additional servers in a multiple instance or coexistence environment.
7. Change port assignments in configuration files if you have a node that you cannot start because of port conflicts.

Refer to the description of the `winstance` command to learn more about the command, and to see examples of use.

## **winstance command**

The **winstance** command creates multiple configuration instances of one initial installation of the product.

### **Overview**

The `winstance` script creates multiple configuration instances of either the base WebSphere Application Server or the deployment manager.

Each WebSphere Application Server configuration instance is a stand-alone instance with a unique name and its own set of configuration files and user data folders. Configuration folders include `config`, `etc` and `properties`. User data folders include `installedApps`, `installableApps`, `temp`, `logs`, `tranlog` and `wstemp`. Each configuration instance also has the administrative console application to manage the configuration instance. Each configuration instance shares all run-time scripts, libraries, the Software Development Kit, and other files with the initial Application Server.

Each deployment manager configuration instance is a different cell. You can federate Application Server installation instances into a deployment manager configuration instance.

Each configuration instance of the Network Deployment product is a stand-alone deployment manager (`dmgr`) with its own set of unique configuration files and user data folders for the cell. Configuration folders include `config`, `etc` and `properties`. User data folders include `installedApps`, `installableApps`, `temp`, `logs`, `tranlog` and `wstemp`. The `dmgr` configuration instance also includes the administrative console and the file transfer applications, to manage the configuration instance.

### **Location of command file**

The command file is located in the `install_root/bin/winstance` directory. The command file is a script named `winstance.sh` for Linux and UNIX-based platforms or `winstance.bat` for Windows platforms.

### **winstance.sh command syntax for Linux and UNIX-based platforms**

The command syntax is as follows:

```
winstance.sh -name instanceName
 -path instanceLocation
 -host hostName
 [-startingPort startingPort]
 [-readOnly]
 -create|-delete
 [-debug]
```

### **winstance.sh command syntax for Windows platforms**

```
winstance.bat -name instanceName
 -path instanceLocation
 -host hostName
```

```
[-startingPort startingPort]
[-readOnly]
-createl-delete
[-debug]
```

## Parameters

Supported arguments include:

### **-create|-delete**

A required parameter that specifies whether to create or delete the configuration instance.

### **-debug**

An optional parameter that turns on the debug function of the Ant utility, which the wsinstance tool uses.

### **-host** *hostname*

A required parameter that specifies the host name where you are creating the configuration instance. This should match the host name you specified during installation of the initial product.

### **-name** *instanceName*

A required parameter that specifies the name of the new configuration instance. Use a unique value. The **wsinstance** command uses this name to construct the node name, which is *instanceName\_hostname* for the new configuration instance of a deployment manager. The instance name is also the name of the cell. It is important to have a unique name for each configuration instance, to provide a unique combination of host name and instance name, which together form the WAS\_NODE name for the configuration instance.

**5.1 +** The embedded messaging feature requires unique node names. The wsinstance tool cannot prevent reusing node names for configuration instances created from read-only partitions. Embedded messaging is not supported when you reuse a node name for a configuration instance. So use a unique name.

### **-path** *instanceLocation*

A required parameter that specifies the file path of the instance. All required folders for the instance are in this directory, which is unique to the configuration instance.

### **-readOnly**

An optional parameter. The wsinstance tool can detect whether you are creating an instance from the base WebSphere Application Server or Network Deployment product in a read-only partition.

The wsinstance tool registers a new wsinstance in the `properties/wsinstance.properties` file of the original installation, if write access to the original product is available. If the original installation is in a read-only partition or if you use the `-readOnly` option, the wsinstance tool does not register the wsinstance within the `properties/wsinstance.properties` file of the original installation.

Creating configuration instances from an original installation in a read-only partition has service implications. There is no mechanism in the original installation to identify and update a configuration instance that you create from a read-only partition. Normally, the registration file in the original installation provides the mechanism for updating configuration instances. When you apply a fix pack or an interim fix to the original installation, the update installer also updates the registered configuration instances. However, there is no registration entry for a configuration instance created from a read-only partition, where no entry is made in the registration file.

When you install a fix pack or interim fix to the original installation, you must use the wsinstance tool to delete the configuration instance and recreate it. A configuration instance must be at the same fix level as the product from which you create it. Otherwise, the configuration instance is unsupported.

### **-startingPort** *startingPort*

An optional parameter that specifies the starting port number for generating all ports for the configuration instance. If not specified, the wsinstance tool uses default ports, or custom-defined ports from a file that you can create.

## Example of creating an instance on a Linux or UNIX-based platform

To create a configuration instance for user `shasti`:

```
wsinstance.sh -name shasti
 -path ~/shasti/WebSphere
 -host planetaix
 -create
```

On a base WebSphere Application Server product installation, the command creates an Application Server configuration instance named `shasti` in location `~/shasti/WebSphere` with a node name of `shasti_planetaix`. If user `shasti` is to use the embedded messaging feature, you must add the user (`shasti`) to the `mqm` and `mqbrkr`s operating system groups, as described in *Installing WebSphere embedded messaging as the JMS provider*. This is true for any non-root user on a Linux or UNIX-based platform.

On a Network Deployment product installation, the command creates a deployment manager configuration instance named `shasti` in a cell named `shasti` in location `~/shasti/WebSphere` with a node name of `shasti_planetaix`.

## Example of creating an instance on a Windows platform

To create a configuration instance for user `shasti`:

```
wsinstance.bat -name shasti
 -path G:\shasti\WebSphere
 -host planetnt
 -create
```

On a base WebSphere Application Server product installation, the command creates an Application Server configuration instance named `shasti` in location `G:\shasti\WebSphere` with a node name of `shasti_planetnt`.

On a Network Deployment product installation, the command creates a deployment manager configuration instance named `shasti` in a cell named `shasti` in location `G:\shasti\WebSphere` with a node name of `shasti_planetnt`.

## Example of a creating an instance in a multiuser environment on a Linux or UNIX platform

Follow these steps to create a configuration instance for user `shasti` in a multiuser environment:

1. Create the configuration instance:

```
wsinstance.sh -name shasti
 -path ~/shasti/WebSphere
 -host myhost
 -startingPort 12000
 -create
```

2. Change the owner of the folder:

```
chown -R shasti ~/shasti/WebSphere
```

3. As a convenience, add a call to script `~/shasti/WebSphere/bin/setupCmdLine.sh` in the profile of user `shasti` to set the environment when user `shasti` logs in.

4. Give these folder permissions to user `shasti`:

```
install_root/bin --- rx (read and execute)
install_root/java --- rx
install_root/properties ----r (read)
install_root/deploytool ----r
install_root/config ----r
install_root/lib ----r
```

```
install_root/classes ----r
install_root/null ----r
install_root/samples ----r
install_root/Web ----r
```

### Example of deleting a configuration instance

The following command is on more than one line for formatting purposes. Enter the command on one line to delete the configuration instance named shasti:

```
wsinstance.sh -name shasti
 -host planetaix
 -delete
```

### Using predefined port numbers

When you use the wsinstance tool without the `-startingPort` parameter, the tool attempts to examine a file named with the convention `instanceName_portdef.props` in the `/bin/wsinstance` directory of the installation root directory of the original product. You can create the file by copying the `portdef.props` template file and customizing the port numbers.

### Example of using default port assignments

When a port definition file is not present, the wsinstance tool generates a `hostname_instanceName_portdef.props` file in the `/bin/wsinstance` directory of the installation root directory of the original product, using the `portdef.props` template file.

**5.1 +** When using the wsinstance tool from a read-only partition, the tool cannot create the `hostname_instanceName_portdef.props` file in a directory of the original product. Therefore, the tool creates the file in the installation root directory of the new instance.

The wsinstance tool adds one (1) to the default port numbers for the base WebSphere Application Server product. If you create two configuration instances using default port assignments, both instances have the same, conflicting port numbers until you change them.

### Incrementing default port numbers from a starting point

The wsinstance tool can assign port numbers based on a starting port value that you give on the command line, using the `-startingPort` parameter. The tool assigns port numbers sequentially from the starting port number value as it creates an `hostname_instanceName_portdef.props` file in the wsinstance subdirectory of the bin directory in the `install_root` folder.

**5.1 +** In the case of creating an instance from a read-only partition, the resulting `hostname_instanceName_portdef.props` file is in the installation root of the new instance.

For example, a `hostname_instanceName_portdef.props` file created with `-startingPort 20002` would appear similar to this:

```
HTTPS_TRANSPORT_ADMIN=20002
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS=20004
HTTP_TRANSPORT_ADMIN=20003
HTTP_TRANSPORT=20005
HTTPS_TRANSPORT=20006
INTERNAL_JMS_SERVER=20007
BOOTSTRAP_ADDRESS=20008
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS=20009
DRS_CLIENT_ADDRESS=20011
```

```
SOAP_CONNECTOR_ADDRESS=20010
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS=20012
JMSSERVER_QUEUED_ADDRESS=20013
JMSSERVER_DIRECT_ADDRESS=20014
```

The order of port assignments can be arbitrary. Predicting assignments is not possible.

### Example of startingPort parameter use

The following example of using the wsinstance tool creates the `shasti_portdef.props` file with the content shown in the previous example:

```
wsinstance.bat -name shasti
 -path G:\shasti\WebSphere
 -host planetnt
 -startingPort 20002
 -create
```

### Setting up and using the configuration instance environment

Most tasks that you perform in a configuration instance are done from one shell or command window after you run the `setupCmdLine.sh` or `setupCmdLine.bat` script in the `/bin` directory of the configuration instance. Such a scenario might be:

1. Create the configuration instance using the `/bin/wsinstance/wsinstance.sh` (or `wsinstance.bat`) script from the original installation.
2. In that command window or in another, change directories to the `/bin` directory of the new configuration instance.
3. Establish a temporary override for the normal WebSphere Application Server environment by running the `setupCmdLine.sh` or `setupCmdLine.bat` script:

```
setupCmdLine.sh
```

4. In the same window, start `server1` by changing directories to the `install_root/bin` directory of the original installation and issuing the command. There is no such command in the `/bin` directory of the configuration instance:

```
startServer.sh server1
```

This works because running the `setupCmdLine` script from the configuration instance reconfigures the WebSphere Application Server environment. Reconfiguring the environment lets you issue commands from the `/bin` directory of the original installation as if you were issuing them from the configuration instance. The configuration instance creates a virtual installation that includes the original installation `/bin` directory. The virtual environment lasts until you close the Linux or UNIX shell or the command window in a Windows environment.

Even after you close the shell or command window, the server you started is still running in a spawned process that retains the environment settings.

5. See how the environment has changed. Display all the environment variables. You can see one that points to the `setupCmdLine.sh` or `setupCmdLine.bat` file in the configuration instance. The name of the environment variable is `WAS_USER_SCRIPT`. For example, in a Linux bash shell or in the command window on a Windows platform, type:

```
set
```

6. Notice the changes in the environment. Display all the ports for the machine to see the ports that you set for the configuration instance. For example, in a Linux bash shell or in the command window on a Windows platform, type:

```
netstat -a
```

7. Open a browser window and point it at the port where you defined `HTTP_TRANSPORT_ADMIN` in the `hostname_instanceName_portdef.props` file in the installation root of the new instance. For example, suppose the setting is `HTTP_TRANSPORT_ADMIN=20003`. Open the administrative console for `server1` by pointing your browser at:

`http://hostname_orIP_address:20003/admin/`

By setting the `WAS_USER_SCRIPT` environment variable to point to itself, the `setupCmdLine` script of the configuration instance overrides the `setupCmdLine` script of the original installation and redirects scripts in the `install_root/bin` directory of the original installation to work on configuration instance objects.

### Backing up required files before installing additional features or before installing the V5.0.x Enterprise product

Creating a new WebSphere Application Server V5.x instance using the `wsinstance` utility can fail with an "adminconsole" already exists error after adding features or installing the V5.0.x Enterprise product. The new instance cannot be administered until the `adminconsole.ear` is successfully deployed, but the `adminconsole.ear` cannot be deployed on the new instance. The text of the problem is similar to the following example:

```
WASX7022E: Problem running command
"$AdminApp install adminconsole.ear
 {-node CELL_INSTANCE
 -configroot /WAS_HOME/config
 -copy.sessionmgr.servername server1
 -appname adminconsole}"
-- exception information: com.ibm.ws.scripting.ScriptingException:

WASX7279E: An application with name "adminconsole" already exists.
 Select a different name.
```

The new instance of the Application Server might start successfully. However, the administrative console cannot start because of an error in the `wsinstance` template files caused by installing additional base features, additional Network Deployment features, or by installing Enterprise.

Use the following procedure to work around the problem:

1. Back up the `install_root/bin/wsinstance/configdefaults` directory.
2. Use the installation program to add the new features or to install Enterprise.
3. Delete the following directories and their contents:
  - `install_root/bin/wsinstance/configdefaults/backup`
  - `install_root/bin/wsinstance/configdefaults/temp`
  - `install_root/bin/wsinstance/configdefaults/$CELL_NAME/applications`
4. Restore the following files from the backup you made to their original locations in the `install_root/bin/wsinstance/configdefaults` directory:
  - `./security.xml`
  - `./nodes/$NODE_NAME/resources.xml`
  - `./nodes/$NODE_NAME/serverindex.xml`
  - `./nodes/$NODE_NAME/servers/$SERVER_NAME/resources.xml`

Do not restore the entire backup you made. Restore only the files mentioned in this step.

`$CELL_NAME` refers to the name of the cell defined during the initial product installation.

`$NODE_NAME` refers to the name chosen for the node.

`$SERVER_NAME` refers to each of the servers present. Perform the procedure on each server.

See the technote [Creating a new V5 instance using the `wsinstance` command fails with "adminconsole" already exists error](#) for more information.



## Correcting a WASX7022E error

If you did not back up the wsinstance template files before installing additional features or before installing the V5.0.x Enterprise product, you can recover from the resulting WASX7022E error described in the preceding section with the following procedure.

1. Uninstall the product and the Enterprise product (if necessary).
2. Reinstall the product, selecting all features that you intend to use.
3. Back up the *install\_root/bin/wsinstance/configdefaults* directory.
4. Install the Enterprise product if necessary.
5. Delete the following directories and their contents:
  - *install\_root/bin/wsinstance/configdefaults/backup*
  - *install\_root/bin/wsinstance/configdefaults/temp*
  - *install\_root/bin/wsinstance/configdefaults/\$CELL\_NAME/applications*
6. Restore the following files from the backup you made to their original locations in the *install\_root/bin/wsinstance/configdefaults* directory:
  - *./security.xml*
  - *./nodes/\$NODE\_NAME/resources.xml*
  - *./nodes/\$NODE\_NAME/serverindex.xml*
  - *./nodes/\$NODE\_NAME/servers/\$SERVER\_NAME/resources.xml*

Do not restore the entire backup you made. Restore only the files mentioned in this step.

**If you cannot afford to reinstall the product on this system:** Do not use the wsinstance utility on this system. The corrupt templates will not affect functionality of the product other than the wsinstance utility.

## Creating servers in coexistence or multiple instance environments

WebSphere Application Server lets you create multiple servers based on an existing template, or using an existing server as a template. You can generate unique ports for the new server during its creation. Always select the unique port option when creating servers in a coexistence environment of multiple versions, installations or configuration instances, due to the likelihood of conflicting port assignments. Verify port assignments for the newly created server and change them if necessary.

1. Create a server (MyServer, for example) using either the administrative console or wsadmin scripts.
2. Log on to the administrative console.
3. Go to **Servers > Application Servers > MyServer > End Points**.
4. Go into each end point and change the port numbers.
5. Change the HTTP Transport ports.
  - a. Go to **Servers > application servers > MyServer > Web Container > HTTP Transports**.
  - b. Change the HTTP transport port numbers.
  - c. Make a record of the new port numbers.
6. Change the JMS server ports, if you create servers from an application server template that has the embedded messaging feature.
  - a. Go to **Servers > application servers > MyServer > Server Components > JMS Servers > Security Port Endpoint**.
  - b. Change the port numbers.
7. Change the JMS server ports, if you create servers from an application server template that is not federated into a deployment manager cell, but does have the embedded messaging feature.
  - a. Go to **Servers > application servers > MyServer > Server Components > JMS Servers > Security Port Endpoint**.

- b. Change the port numbers.

## Changing HTTP transport ports

This topic describes how to change HTTP transports manually by editing configuration files. Use this procedure when a conflicting HTTP transport setting is preventing an application server or deployment manager instance from starting.

Edit the configuration files to resolve conflicting port assignments, as described below.

1. Look for symptoms of port number conflicts.
2. Edit an application server configuration.
  - a. Open the `server.xml` configuration file for the `server1` application server.  
The file path for the `server1` configuration, with a node name of **myNode**, is:  
`install_root/config/cells/mynode/nodes/mynode/servers/server1`
  - b. Look for transports `xmi:type="applicationserver.webcontainer:HTTPTransport"`.  
The administrative console application uses transport ports 9090 and 9043. The Sample applications use transport ports 9080 and 9443. Change the port numbers and save the file. Make a record of the new port numbers.
  - c. Open the `virtualhosts.xml` file in the `install_root/config/cells/mynode` folder.  
This file contains alias entries for transport ports defined in the `server.xml` file.
  - d. Look for aliases `xmi:id` to change port number assignments for any ports you changed.
3. Edit a deployment manager node configuration.  
The deployment manager uses HTTP transport ports for the administrative console application. The default port is 9090.
  - a. Open the `server.xml` configuration file for the `dmgr` deployment manager server.  
The file path for the `dmgr` configuration, with a cell name of **myManager**, and a node name of **myNode**, is:  
`install_root/config/cells/myManager/nodes/mynode/servers/dmgr`
  - b. Look for transports `xmi:type="applicationserver.webcontainer:HTTPTransport"`.  
The administrative console application uses transport ports 9090 and 9043. Change the port numbers and save the file. Make a record of the new port numbers.
  - c. Open the `virtualhosts.xml` file in the `install_root/config/cells/mynode` folder.  
This file contains alias entries for transport ports defined in the `server.xml` file.
  - d. Look for aliases `xmi:id` to change port number assignments for any ports you changed.

---

## Installing interim fixes, cumulative fixes, and fix packs

This topic describes how to use the update installer program to install interim fixes, cumulative fixes, and fix packs. The update installer program is also known as the `updateInstaller` program or the Update installation wizard.

IBM Support offers tutorials on various WebSphere Application Server installation topics. See WebSphere education on demand: Installation best practices tutorials for more information. One topic describes updating WebSphere Application Server products using the update installer program.

You cannot install product updates correctly without the proper authorizations. Use the update installer program as the root user on a Linux or UNIX platform, or as the administrator on a Windows platform.

Three different sites contain service for WebSphere Application Server products and features:

- Download interim fixes for the IBM HTTP Server from the IBM Support site for IBM HTTP Server. The readme file that is bundled with the interim fix describes how to install the interim fix. You do not use the update installer program for the WebSphere Application Server (also known as the updateInstaller program or the Update Installation wizard) to install interim fixes for the IBM HTTP Server feature.
- Download interim fixes for the embedded messaging feature from the WebSphere MQ Service download site. Download the readme file for a description of how to install the interim fix. You do not use the update installer for the WebSphere Application Server to install interim fixes for the embedded messaging feature.
- Download any interim fixes, cumulative fixes, and fix packs for the WebSphere Application Server family of products from the IBM Support site for WebSphere Application Server. Cumulative fixes and fix packs do include service for the IBM HTTP Server powered by Apache 1.3 feature and for the embedded messaging feature. Interim fixes for a WebSphere Application Server product do not include service for the IBM HTTP Server feature or the embedded messaging feature. The readme file that is bundled with the WebSphere Application Server interim fix, the cumulative fix, or the fix pack describes how to use the update installer program for WebSphere Application Server products to install the interim fix, the cumulative fix, or the fix pack.

**Important:** Cumulative fixes and fix packs do not include service for the IBM HTTP Server powered by Apache 2.0. If you installed IBM HTTP Server powered by Apache 2.0, skip applying service to the IBM HTTP Server during the installation of a cumulative fix or a fix pack. Do not update the 2.0 level IBM HTTP Server with service for the 1.3 level IBM HTTP Server.

Installing a cumulative fix or a fix pack removes all of the interim fixes for serviceable components, except for interim fixes for the IBM HTTP Server feature or the embedded messaging feature. You must remove those manually as described in the following tip.

You must use the update installer program to install cumulative fixes and fix packs for the two features. The relationship among interim fixes, cumulative fixes, and fix packs is shown in the Cumulative Fix Strategy for WebSphere Application Server V5.0 and V5.1 Web page.

Fix packs are also known as *fixpacks*, *FixPaks* and *program temporary fixes*, or PTFs.

There are two interfaces to the update installer application, a wizard with a graphical interface, and a command-line, silent interface:

- The wizard interface is the **updateWizard** command. See “updateWizard command” on page 230 for more information.
- The silent, command-line interface is the **updateSilent** command. See “updateSilent command” on page 218 for more information.

Both the **updateSilent** command and the **updateWizard** command call the update installer program to install and uninstall interim fixes, cumulative fixes, and fix packs for WebSphere Application Server products.

The update installer application can also uninstall interim fixes, cumulative fixes, and fix packs. See “Uninstalling interim fixes, cumulative fixes, and fix packs” on page 239 for more information.

The following descriptions contain reference information about installing interim fixes, cumulative fixes, and fix packs on WebSphere Application Server products:

#### Installation roots

The variable *install\_root* represents the root directory for WebSphere Application Server. By default, this varies per product and operating system:

- Base WebSphere Application Server product:
  - AIX platforms: /usr/WebSphere/AppServer
  - Linux platforms: /opt/WebSphere/AppServer
  - HP-UX and Solaris platforms: /opt/WebSphere/AppServer
  - Windows platforms: *drive*\Program Files\WebSphere\AppServer

- Network Deployment product:
  - AIX platforms: `/usr/WebSphere/DeploymentManager`
  - Linux platforms: `/opt/WebSphere/DeploymentManager`
  - HP-UX and Solaris platforms: `/opt/WebSphere/DeploymentManager`
  - Windows platforms: `drive\Program Files\WebSphere\DeploymentManager`
- **5.1+** WebSphere Business Integration Server Foundation product that extends the base product:
  - AIX platforms: `/usr/WebSphere/AppServer`
  - Linux platforms: `/opt/WebSphere/AppServer`
  - HP-UX and Solaris platforms: `/opt/WebSphere/AppServer`
  - Windows platforms: `drive\Program Files\WebSphere\AppServer`
- **5.1+** WebSphere Business Integration Server Foundation product that extends the Network Deployment product
  - AIX platforms: `/usr/WebSphere/DeploymentManager`
  - Linux platforms: `/opt/WebSphere/DeploymentManager`
  - HP-UX and Solaris platforms: `/opt/WebSphere/DeploymentManager`
  - Windows platforms: `drive\Program Files\WebSphere\DeploymentManager`

### Space requirements

Space requirements vary depending on what you are installing. The size of each download is available on the Support site. After unpacking the ZIP file you download, delete the ZIP file to free space.

For a fix pack, have approximately 400 MB of free space in the `/tmp` directory and another 400 MB in the file system that hosts the WebSphere Application Server image, which is typically `/opt` for a Linux platform and for all UNIX platforms except for AIX, which is typically `/usr`, or approximately 800 MB of free space on the disk drive where you are installing on a Windows platform.

Interim fixes require much less space to install.

Additional space is required for backup files in the `install_root/properties/version/backup` directory. When installing a fix pack the space required is typically about the same as the size of the fix pack, that is, between 50 MB and 300 MB, depending on the particular fix pack.

The update installer program checks for required space before it installs an interim fix, a cumulative fix, or a fix pack. Fixes require much less space to install than do cumulative fixes or fix packs. The space requirement varies per cumulative fix. For example, Cumulative Fix 3 for Version 5.1 (5.1.0.3) requires 248 MB in the `/tmp` directory and another 248 MB in the partition that contains the installation root directory. That is a total of 496 MB.

The system temporary directory is determined by the JVM and by the operating system. It is possible for the system temporary directory and the installation root directory to be on the same partition.

Older versions of the update installer, including the version current at the time that Cumulative Fix 3 was released, check for the total space requirement but report each requirement separately. Unfortunately, the messages are misleading. If you have 300 MB of free space on a partition but both the `/tmp` directory and the installation root are on the partition, you do not have enough space for both.

In this example, you have 300 MB when you need 496 MB in the partition. Each space checking message states that you do not have enough space and that 248 MB is required. 300 MB is more than 248 MB so the message is misleading. The message should state that you need 496 MB in the partition, 248 MB of which is for the `/tmp` directory. The second message should state that you need 496 MB in the partition, 248 MB of which is for the installation root requirement.

### Command name

`updateSilent.sh`, `updateSilent.bat`, `updateWizard.sh`, and `updateWizard.bat`, command-line interfaces to the `installer.jar` file.

## Prerequisite environment setting

The JAVA\_HOME environment setting must point to the IBM SDK for WebSphere Application Server products. Source the appropriate command:

- `. install_root/bin/setupCmdLine.sh` (Source the command on UNIX platforms. There is a space between the period and the installation root directory.)
- `source install_root/bin/setupCmdLine.sh` (Source the command on Linux platforms.)
- `install_root\bin\setupCmdLine.bat` (Windows platforms only)

## Download from

Download the `updateInstaller.zip` file from the WebSphere Application Server Support page. The files that comprise the ZIP file are also part of each fix pack ZIP file package. Fix packs are named according to the Application Server product, the fix pack sequence, and the operating system platform.

Interim fixes are named according to the tracking number used for the defect that the interim fix solves. For example, PQ81989 is an interim fix. See 1.3.1 Java SDK, Java Tech Edition for WebSphere Application Server V5 for an example of a download page for an interim fix.

Cumulative fixes use a naming scheme that identifies the product, the cumulative fix number, and the operating system. The following example shows names for Cumulative Fix 3 for V5.1:

Table 39. Cumulative Fix 3 names for WebSphere Application Server, Version 5.1

| Operating system platform | Cumulative Fix 3 ZIP file            | Cumulative Fix 3 ID              | Default repository in installation root directory |
|---------------------------|--------------------------------------|----------------------------------|---------------------------------------------------|
| AIX                       | <code>was510_cf3_aix.zip</code>      | <code>was510_cf3_aix</code>      |                                                   |
| Linux                     | <code>was510_cf3_linux.zip</code>    | <code>was510_cf3_linux</code>    |                                                   |
| Linux for S/390           | <code>was510_cf3_linux390.zip</code> | <code>was510_cf3_linux390</code> | <code>../update/ fixpacks</code>                  |
| Solaris                   | <code>was510_cf3_solaris.zip</code>  | <code>was510_cf3_solaris</code>  |                                                   |
| HP-UX                     | <code>was510_cf3_hpux.zip</code>     | <code>was510_cf3_hpux</code>     |                                                   |
| Windows                   | <code>was510_cf3_win.zip</code>      | <code>was510_cf3_win</code>      | <code>..\update\ fixpacks</code>                  |

Table 40. Cumulative Fix 3 names for WebSphere Application Server Network Deployment, Version 5.1

| Operating system platform | Cumulative Fix 3 ZIP file               | Cumulative Fix 3 ID                 | Default repository in installation root directory |
|---------------------------|-----------------------------------------|-------------------------------------|---------------------------------------------------|
| AIX                       | <code>was510_nd_cf3_aix.zip</code>      | <code>was510_nd_cf3_fp2_aix</code>  | <code>../update/fixpacks</code>                   |
| Linux                     | <code>was510_nd_cf3_linux.zip</code>    | <code>was510_nd_cf3_linux</code>    |                                                   |
| Linux for S/390           | <code>was510_nd_cf3_linux390.zip</code> | <code>was510_nd_cf3_linux390</code> |                                                   |
| Solaris                   | <code>was510_nd_cf3_solaris.zip</code>  | <code>was510_nd_cf3_solaris</code>  |                                                   |
| HP-UX                     | <code>was510_nd_cf3_hpux.zip</code>     | <code>was510_nd_cf3_hpux</code>     |                                                   |
| Windows platforms         | <code>was510_nd_cf3_win.zip</code>      | <code>was510_nd_cf3_win</code>      | <code>..\update\ fixpacks</code>                  |

See WebSphere Application Server 5.1 Cumulative Fix 3 for an example of a download page for a cumulative fix.

Fix packs use a naming scheme that identifies the product, the fix pack sequence number, and the operating system.

**5.1+** The names of the cumulative fixes that the WebSphere Business Integration Server Foundation product installs for the base product and for the Network Deployment products are shown in the following table:

Table 41. Cumulative Fix 2 names for WebSphere Application Server, Version 5.1

| Operating system platform | Cumulative Fix 2 ZIP file | Cumulative Fix 2 ID | Default repository in installation root directory |
|---------------------------|---------------------------|---------------------|---------------------------------------------------|
| AIX                       | was510_cf2_aix.zip        | was510_cf2_aix      |                                                   |
| Linux                     | was510_cf2_linux.zip      | was510_cf2_linux    |                                                   |
| Linux for S/390           | was510_cf2_linux390.zip   | was510_cf2_linux390 | ../update/ fixpacks                               |
| Solaris                   | was510_cf2_solaris.zip    | was510_cf2_solaris  |                                                   |
| HP-UX                     | was510_cf2_hpux.zip       | was510_cf2_hpux     |                                                   |
| Windows                   | was510_cf2_win.zip        | was510_cf2_win      | ..\update\ fixpacks                               |

Table 42. Cumulative Fix 2 names for WebSphere Application Server Network Deployment, Version 5.1

| Operating system platform | Cumulative Fix 2 ZIP file | Cumulative Fix 2 ID   | Default repository in installation root directory |
|---------------------------|---------------------------|-----------------------|---------------------------------------------------|
| AIX                       | was51_nd_cf2_aix.zip      | was51_nd_cf2_fp2_aix  |                                                   |
| Linux                     | was51_nd_cf2_linux.zip    | was51_nd_cf2_linux    |                                                   |
| Linux for S/390           | was51_nd_cf2_linux390.zip | was51_nd_cf2_linux390 | ../update/fixpacks                                |
| Solaris                   | was51_nd_cf2_solaris.zip  | was51_nd_cf2_solaris  |                                                   |
| HP-UX                     | was51_nd_cf2_hpux.zip     | was51_nd_cf2_hpux     |                                                   |
| Windows platforms         | was51_nd_cf2_win.zip      | was51_nd_cf2_win      | ..\update\ fixpacks                               |

The latest available fix pack is Fix Pack 1 for the WebSphere Application Server V5.1.0 family of products:

Table 43. Fix Pack 1 names for WebSphere Application Server, Version 5.1.0

| Operating system platform | Fix Pack 1 ZIP file    | Fix Pack 1 ID      | Default repository in installation root directory |
|---------------------------|------------------------|--------------------|---------------------------------------------------|
| AIX                       | was51_fp1_aix.zip      | was51_fp1_aix      |                                                   |
| Linux                     | was51_fp1_linux.zip    | was51_fp1_linux    |                                                   |
| Linux for S/390           | was51_fp1_linux390.zip | was51_fp1_linux390 | ../update/ fixpacks                               |
| Solaris                   | was51_fp1_solaris.zip  | was51_fp1_solaris  |                                                   |
| HP-UX                     | was51_fp1_hpux.zip     | was51_fp1_hpux     |                                                   |
| Windows                   | was51_fp1_win.zip      | was51_fp1_win      | ..\update\ fixpacks                               |

Table 44. Fix Pack 1 names for WebSphere Application Server Network Deployment, Version 5.1.0

| Operating system platform | Fix Pack 1 ZIP file       | Fix Pack 1 ID         | Default repository in installation root directory |
|---------------------------|---------------------------|-----------------------|---------------------------------------------------|
| AIX                       | was51_nd_fp1_aix.zip      | was51_nd_fp1_aix      |                                                   |
| Linux                     | was51_nd_fp1_linux.zip    | was51_nd_fp1_linux    |                                                   |
| Linux for S/390           | was51_nd_fp1_linux390.zip | was51_nd_fp1_linux390 | ../update/fixpacks                                |
| Solaris                   | was51_nd_fp1_solaris.zip  | was51_nd_fp1_solaris  |                                                   |
| HP-UX                     | was51_nd_fp1_hpux.zip     | was51_nd_fp1_hpux     |                                                   |
| Windows platforms         | was51_nd_fp1_win.zip      | was51_nd_fp1_win      | ..\update\ fixpacks                               |

Table 45. Fix Pack 1 names for WebSphere Business Integration Server Foundation, Version 5.1.0

| Operating system platform | Fix Pack 1 ZIP file      | Fix Pack 1 ID                                                                                                 | Default repository in installation root directory |
|---------------------------|--------------------------|---------------------------------------------------------------------------------------------------------------|---------------------------------------------------|
| AIX                       | wbisf51_fp1_aix.zip      | wbisf51_fp1_aix (to extend the base product)<br>wbisf51_nd_fp1_aix (to extend the Network Deployment product) |                                                   |
| Linux                     | wbisf51_fp1_linux.zip    | wbisf51_fp1_linux (base)<br>wbisf51_nd_fp1_linux (Network Deployment)                                         |                                                   |
| Linux for S/390           | wbisf51_fp1_linux390.zip | wbisf51_fp1_linux390 (base)<br>wbisf51_nd_fp1_linux390 (Network Deployment)                                   | ../update/ fixpacks                               |
| Solaris                   | wbisf51_fp1_solaris.zip  | wbisf51_fp1_solaris (base)<br>wbisf51_nd_fp1_solaris (Network Deployment)                                     |                                                   |
| HP-UX                     | wbisf51_fp1_hpux.zip     | wbisf51_fp1_hpux<br>wbisf51_nd_fp1_hpux (Network Deployment)                                                  |                                                   |
| Windows platforms         | wbisf51_fp1_win.zip      | wbisf51_fp1_win (base)<br>wbisf51_nd_fp1_win (Network Deployment)                                             | ..\update\ fixpacks                               |

Table 46. Fix Pack 1 names for WebSphere Application Server Express, Version 5.1.0

| Operating system platform | Fix Pack 1 ZIP file            | Fix Pack 1 ID              | Default repository in installation root directory |
|---------------------------|--------------------------------|----------------------------|---------------------------------------------------|
| AIX                       | was51_express_fp1_aix.zip      | was51_express_fp1_aix      |                                                   |
| Linux                     | was51_express_fp1_linux.zip    | was51_express_fp1_linux    |                                                   |
| Linux for S/390           | was51_express_fp1_linux390.zip | was51_express_fp1_linux390 | ../update/ fixpacks                               |
| Solaris                   | was51_express_fp1_solaris.zip  | was51_express_fp1_solaris  |                                                   |
| HP-UX                     | was51_express_fp1_hpux.zip     | was51_express_fp1_hpux     |                                                   |
| Windows platforms         | was51_express_fp1_win.zip      | was51_express_fp1_win      | ..\update\ fixpacks                               |

Table 47. Fix Pack 1 names for WebSphere Application Server, Version 5.1.0 Application Clients

| Operating system platform | Fix Pack 1 ZIP file           | Fix Pack 1 ID             | Default repository in installation root directory |
|---------------------------|-------------------------------|---------------------------|---------------------------------------------------|
| AIX                       | was51_client_fp1_aix.zip      | was51_client_fp1_aix      |                                                   |
| Linux                     | was51_client_fp1_linux.zip    | was51_client_fp1_linux    |                                                   |
| Linux for S/390           | was51_client_fp1_linux390.zip | was51_client_fp1_linux390 | ../update/fixpacks                                |
| Solaris                   | was51_client_fp1_solaris.zip  | was51_client_fp1_solaris  |                                                   |
| HP-UX                     | was51_client_fp1_hpux.zip     | was51_client_fp1_hpux     |                                                   |
| Windows platforms         | was51_client_fp1_win.zip      | was51_client_fp1_win      | ..\update\fixpacks                                |

### Download to

Choose a directory for unpacking the update installer program or fix pack zip file that does not

have a space in its name. For example, do not use the default installation root on a Windows platform because it is within the Program Files directory. On a Linux or UNIX-based platform, you can use the installation root directory.

Create the *install\_root*/update directory or the *no\_spaces\_path*\update directory on Windows platforms. Unpacking a fix pack creates the *../update/fixpacks* directory. Create another directory, *../update/fixes*, for a repository of fixes you download. If you create the default subdirectories, you can accept default interim fix and fix pack file locations when using the **updateWizard** interface. Otherwise, you must browse to locate the fixes or fix packs you are installing or uninstalling.

#### **Location of extfile.jar**

WebSphere Application Server product *install\_root*/update/lib (or *no\_spaces\_path*\update\lib for Windows platforms)

#### **Location of installer.jar, updateSilent.sh/bat, and updateWizard.sh/bat**

WebSphere Application Server product *install\_root*/update (or *no\_spaces\_path*\update for Windows platforms)

#### **Location of readme files**

WebSphere Application Server product *install\_root*/update/docs (or *no\_spaces\_path*\update\docs for Windows platforms)

#### **Location of interim fix Java archive (JAR) files**

*install\_root*/update/fixes (or *no\_spaces\_path*\update\fixes)

#### **Location of cumulative fix and fix pack JAR files**

*install\_root*/update/fixpacks (or *no\_spaces\_path*\update\fixpacks)

This directory is automatically created by unpacking the fix pack in the *../update* directory.

#### **Files in updateInstaller.zip**

Always use the **updateSilent** (or **updateWizard**) command file from the updateInstaller.zip or fix pack you download, to use the most recent version. A newer version can manage previously downloaded fixes and fix packs. Files in the updateInstaller.zip package (or the fix pack ZIP package) include:

- extfile.jar
- installer.jar
- readme\_updateinstaller.txt
- readme\_updateinstaller.html
- readme\_updateinstaller.pdf
- updateSilent.sh (or updateSilent.bat)
- updateWizard.sh (or updateWizard.bat)

In addition to the listed files, the cumulative fix pack zip file or the fix pack zip file also has a JAR file, such as the was510\_nd\_cf3\_win.jar file. Each JAR file includes a cumulative fix or a fix pack.

#### **Location of log and backup files**

The update installer program records processing results in log files in the *install\_root*/logs/update directory. Backup files created during the installation of fixes and fix packs are in the *install\_root*/properties/version/backup directory. The files are required to uninstall an interim fix or fix pack.

#### **Syntax and panel examples**

##### **Using the updateWizard interface to work with interim fixes**

See “updateWizard examples for interim fixes” on page 231 for more information.

##### **Using the updateWizard interface to work with cumulative fixes and fix packs**

See “updateWizard examples for cumulative fixes and fix packs” on page 236 for more information.

##### **Using the updateSilent interface to work with interim fixes**

See “updateSilent examples for interim fixes” on page 219 for more information.

##### **Using the updateSilent interface to work with cumulative fixes and fix packs**

See “updateSilent examples for cumulative fixes and fix packs” on page 224 for more information.



## Overview of the installation procedure

To install an interim fix, create an update/fixes directory on your disk drive, download the interim fix and the update installer from the Support Web site, and use the update installer to install the interim fix.

To install a cumulative fix or fix pack, create the update directory on your disk drive if it does not already exist, download and unzip the cumulative fix or the fix pack from the Support Web site, and use the update installer to install the cumulative fix or fix pack. The ZIP file for the cumulative fix or the fix pack includes the following files:

- The update installer application
- The two interface programs (updateSilent and updateWizard)
- A readme file for the cumulative fix or fix pack
- A readme file for the update installer
- The cumulative fix or fix pack file

## Special rules for applying fixes within a cell

One requirement governs applying an interim fix or fix pack to a cell, to ensure the continued, smooth interaction of the various WebSphere Application Server nodes:

**Requirement 1:** *The Network Deployment product must be at the highest fix level within the cell.*

For example, you cannot use the **addNode** command to add a V5.1 base WebSphere Application Server node to a V5.0.2 deployment manager cell.

There is no limitation on the fix level of a base Application Server V5 node within its cell, if the fix level of the base node is the same as or lower than that of the deployment manager. There is also no limit on the number of different V5.x fix levels that can coexist or interoperate within a cell, so long as the fix level for each base node is the same as or lower than that of the deployment manager. Version 5.0.x base nodes can comprise V5.1 deployment manager cells.

## Viewing the fix level of the node

You can use the “versionInfo command” on page 268 in the *install\_root/bin* directory to display the exact fix and version level of the product. However, do not use the **versionInfo** command while installing an interim fix or fix pack.

You can also use the silent update installer application to:

- View interim fix information. See Viewing information about interim fixes.
- View cumulative fix or fix pack information. See Viewing information about cumulative fixes and fix packs.

## Updating cluster members

Refer to the following tip for information about updating cluster members:

Table 48. Installation tip

| Operating platform | Tip                                                |
|--------------------|----------------------------------------------------|
| All platforms      | Updating all cluster members to the same fix level |

**Do not launch multiple copies of the update installer program at one time** The update installer program cannot be launched concurrently with itself. Performing more than one update at the same time can lead to a failed or faulty installation.

See “Uninstalling interim fixes, cumulative fixes, and fix packs” on page 239 for a description of how to remove an interim fix or fix pack from an entire cell, or from any part of the cell.

Installing a fix pack or cumulative fix uninstalls all interim fixes that were installed with the update installer. Interim fixes for the IBM HTTP Server feature and the embedded messaging feature cannot be removed. Some of the interim fixes that are uninstalled might have been released after the release of the cumulative fix or the fix pack. Reinstall such interim fixes to bring your system back to the previous interim fix level.

This procedure describes a scenario for updating an entire cell to the same fix pack level. According to the requirements, apply a fix pack to the deployment manager node first. You can then apply the fix pack to zero, one, or more of the base nodes. If you update a base node in a cluster, install the interim fix, cumulative fix pack, or fix pack to each node in the cluster.

### Recovering from a failed installation of a fix pack, cumulative fix, or interim fix

Uninstall the fix pack, the cumulative fix, or the interim fix and reinstall it to recover from a failed installation. The installation can fail if you run out of disk space or if you have a process running and the update installer cannot update the locked files.

1. Use the Windows Services panel to stop all services for WebSphere Application Server processes. This includes services for the deployment manager (dmgr) server, Application Servers (such as server1), and WebSphere MQ queue managers.
2. Stop all Java processes that use the IBM Software Developer Kit (SDK) that WebSphere Application Server provides. Before installing or uninstalling interim fixes, cumulative fixes, and fix packs on a machine, stop all Java processes on the machine that use the IBM SDK, Java Technology Edition that WebSphere Application Server provides.

WebSphere Application Server processes include:

- Application Server processes
- All servers, such as the IBMHttpServer process, that belong to serviceable features. Features with servers include the IBM HTTP Server and the embedded messaging feature.
- The nodeagent process on a base node when the node is federated into a deployment manager cell
- The dmgr process for the deployment manager server
- The jmsserver on a base node when the node is part of a cell and the embedded messaging server and client feature is installed

Stop all Java processes, if necessary, with the **killall -9 java** command or by using the task manager on a Windows platform. If you do install or uninstall an interim fix, a cumulative fix, or a fix pack while a WebSphere Application Server-related Java process runs, IBM does not guarantee that the product can continue to run successfully, or without error.

3. Stop all WebSphere MQ processes on a Windows platform.
  - a. Stop the WebSphere MQ queue manager process.

```
endmqm queue_manager_ID
```
  - b. Stop all WebSphere MQ network activity:

```
net stop "IBM MQSeries"
```
  - c. Remove the WebSphere MQ tray icon if it is present on a Windows system. On a Windows platform, remove the WebSphere MQ tray icon if it is present. The WebSphere MQ tray icon in the lower right corner indicates that a WebSphere MQ process (**amqmtbrn.exe**) is running. Right click the tray icon and click **Hide** to remove it.
  - d. Reboot to free locked GSKit-related files. If the fix that you are installing contains service for the GSKit component, it is possible that the operating system might have locked the file. You can manually stop any Windows services as described in the first step and reboot to free all files.
4. Stop all WebSphere MQ processes on a Linux or a UNIX platform.
  - a. Type **dspmq** to show the state of any queue managers.
  - b. Type **endmqm -i** for each running queue manager.
  - c. Type **\$ ipcs -a** to check for any IPCs.
  - d. Type **\$ ipcrm -[qms] [ID]** to delete the IPCs.
  - e. Type **ps -eaf | grep mqm** or **ps -eaf | grep ^MQ\*** to search for mqm processes.
  - f. Type **kill -9 amq\_pid\_1 amq\_pid\_2 ... amq\_pid\_n** to stop any MQ processes that are running.
5. Verify that the required free space is available before beginning the installation. The space required for unpacking the ZIP file is about the same as the size of the fix pack. After unpacking the ZIP file, you

can delete the ZIP file if necessary. After it is installed, the fix pack code generally increases the installation and run-time footprints by only a small amount.

6. Install the interim fix, the cumulative fix, or the fix pack on a deployment manager node, as described in “Using the update installer application to update a deployment manager node” on page 214.
7. Install the interim fix, the cumulative fix, or the fix pack on a base node, as described in “Using the update installer to update a base node.”

You can successfully install interim fixes, cumulative fixes, and fix packs on WebSphere Application Server nodes.

## Using the update installer to update a base node

This topic describes how to use the update installer program to install interim fixes, cumulative fixes, and fix packs on a base WebSphere Application Server node.

If you have not already done so, read “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for an overview of installing service to WebSphere Application Server products. You must update the deployment manager node before updating managed base nodes, which are base nodes that you have federated into a deployment manager cell.

If you extended the Application Server node by installing the WebSphere Business Integration Server Foundation product, see Using the update installer application to update an extended node.

Install the interim fix, the cumulative fix, or the fix pack on each base node to which you intend to apply the fix using the following procedure for each node.

1. Install the interim fix, the cumulative fix, or the fix pack on the deployment manager node first if the base node is part of a cell, as described in “Using the update installer application to update a deployment manager node” on page 214. The deployment manager node must have the highest fix level within the cell.

2. Stop the nodeagent process on a managed base node with the **stopNode** command if the node is part of a cell and if you have not already done so. Issue the **stopNode** command from the *install\_root/bin* directory of each managed node. For example, issue the following command on a Linux platform:

```
./stopNode.sh -user name -password password
```

See stopNode command for more information about the command.

3. Stop each server process on the base WebSphere Application Server node with the **stopServer** command. Issue the **stopServer** command from the *install\_root/bin* directory of each managed node. For example, issue the following command on a Linux platform:

```
./stopServer.sh server1 -user name -password password
```

See stopServer command for more information about the command.

Stop all WebSphere Application Server-related Java processes. On a Windows platform, you can use the task manager to stop Java processes. On a Linux or UNIX-based platform, use the **kill** command or the **killall java -9** command to stop Java processes.

Use the Windows Services panel to stop any Windows service for the nodeagent, and for any other WebSphere Application Server related services, including Application Server processes, the jmsserver process, IBM HTTP Server and WebSphere MQ queue managers.

4. Create the *install\_root/update* directory on the Application Server node, if the directory does not already exist. Later, you will also launch the update installer from this directory. Launching the update installer is not supported from a read-only directory, or from a directory with spaces in its name.

For example, create the */opt/WebSphere/AppServer/update* directory on a Linux platform.

On a Windows platform, create the update directory in a path where none of the directory names includes a space in their names. For example, do not create the C:\Program Files\WebSphere\AppServer\update directory because the Program Files directory has a space in its name.

You can install a fix from the C:\WebSphere\update directory, for example, to the Application Server product in the default installation root directory, C:\Program Files\WebSphere\AppServer. The target directory can have a space; the source directory cannot.

5. Create the update/fixes repository if you are installing an interim fix. It is not necessary to create the fixpacks repository directory. Unpacking a cumulative fix or a fix pack creates the fixpacks directory if the directory does not already exist.
6. Download the interim fix, the cumulative fix, or the fix pack. Download an interim fix from the Support page to the update/fixes directory. Download a fix pack to the update directory.
7. Unpack the interim fix, the cumulative fix, or the fix pack. Unpacking a cumulative fix or a fix pack automatically creates the fixpacks directory.

Table 49. Installation tip

| Operating platform | Tip in “Platform-specific tips for installing and migrating” on page 131                                      |
|--------------------|---------------------------------------------------------------------------------------------------------------|
| Windows platforms  | Use another unzip product such as WINZIP, instead of the PKWARE pkunzip utility to unzip the product archive. |

8. Download the ZIP file for the current update installer and extract the update installer application to the update directory. You can use the update installer application that is packaged as part of a fix pack. Or, you can download the current version of the file even though you might have an update installer from the fix pack or from a previous fix installation. The Support page links to the current installer for the version of the product that you are updating.
9. Verify that the unpacked files are owned by root on a Linux or UNIX-based platform.
  - a. List the contents of the download directory. For example:

```
ls -al
```

```
drwxr-xr-x 6 root bin 512 Jul 21 08:50 was51fp1_linux
```

The directory list in the preceding example shows a Fix Pack 1 file for V5.1 that is not owned by root.

- b. Change the ownership of any files not owned by root. You can change the ownership of all files in the download directory.

For example:

```
chmod -R root:root *
```

10. Set up the Java environment for the update installer.

#### Setting the JAVA\_HOME environment variable

**Important:** If the update installer can set the Java environment, this step is unnecessary. Otherwise, this is a required step.

It is possible that the update installer cannot set the JAVA\_HOME environment variable. If you receive a message that the update installer cannot set JAVA\_HOME, or if you have set the JAVA\_HOME variable to point to a non-WebSphere Application Server SDK, you must set the JAVA\_HOME variable to a correct value.

Set the environment variable yourself when there is a problem, or source the appropriate command script from the **bin** directory of the product installation root so that the variable points to an IBM SDK for a WebSphere Application Server product.

If you use the same command window that you use to issue the **stopNode** command or the **stopServer** command, you have already sourced the `setupCmdLine` script and set the `JAVA_HOME` environment variable.

Otherwise, use the following procedure to update the `JAVA_HOME` variable:

- a. Open a command-line window.
  - b. Change directories to the `bin` directory of the installation root.
  - c. Issue the command to set `JAVA_HOME`. Issue the appropriate command:
    - `. install_root/bin/setupCmdLine.sh` (source the command on UNIX platforms - There is a space between the period and the installation root directory.)
    - `source install_root/bin/setupCmdLine.sh` (source the command on Linux platforms )
    - `install_root\bin\setupCmdLine.bat` (Windows platforms only)
11. Install the interim fix, cumulative fix, or fix pack on the base node.
- Use the **updateWizard** command or the **updateSilent** command to install the interim fix, cumulative fix, or fix pack on the Application Server node. The choice is whether to use a wizard. For more information about using either command, see the following articles:
- “updateWizard command” on page 230
  - “updateSilent command” on page 218
- For example, assume that you are installing Fix Pack 1 on Version 5.1.0.0. To install the `was51_nd_fp1_win` fix pack, use this **updateSilent** command:
- ```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\Program Files\WebSphere\AppServer"
-skipIHS
-skipMQ
-fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
-install
-fixpackID was51_fp1_win
```
- This example skips applying any service that might be in the fix pack for the IBM HTTP Server feature or the embedded messaging feature.
- The command is shown on more than one line for clarity.
12. Restart the node agent of the base node with the **startNode** command if the node is federated. Restart the node agent on each managed node to let the node agent continue to communicate with the updated deployment manager node.
- Issue the **startNode** command from the `install_root/bin` directory of each base node. For example, issue the following command on a Linux platform:
- ```
./startNode.sh
```
- See `startNode` command for more information about the command.
13. Restart each server on the base node with the **startServer** command. Issue the **startServer** command from the `install_root/bin` directory of each managed node. For example, issue the following command on a Linux platform:
- ```
# ./startServer.sh server1
```
- See `removeNode` command for more information about the command.
14. Verify that the Application Server node is fully functional and that it has the interim fix, the cumulative fix, or the fix pack applied. There are several ways to verify the successful application of an interim fix, a cumulative fix, or a fix pack:
- Does the fix show up in the wizard panel that lists the appropriate type of installed fix?
 - Installed interim fixes. See Uninstallable fix selection panel .
 - Installed cumulative fixes or fix packs. See Installing and Uninstalling panel.If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is installed.
 - Does the fix show up in the wizard panel that the appropriate type of installable (not yet installed) fix ?

- Installable interim fixes. See Installable fix selection panel .
- Installable cumulative fixes or fix packs. See Installable fix selection panel .

If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is not installed.

- Does one of the following files exist in the *install_root/properties/version/version* directory?
 - [interim_fix_ID].efix
 - [cumulative_fix_ID].ptf
 - [fix_pack_ID].ptf

Or, does one of the following files exist in the *install_root/properties/version/history* directory?

- [interim_fix_ID].efixApplied
- [interim_fix_ID].efixDriver
- [cumulative_fix_ID].ptf
- [cumulative_fix_ID].ptf
- [fix_pack_ID].ptfApplied
- [fix_pack_ID].ptfDriver

These types of files indicate the installation of a fix. See Product information files for more information.

- Do the reports that describe the product version and its history show that the interim fix, the cumulative fix, or the fix pack is installed or that it has been removed?
See Reports for more information about the reports.
- Do the **updateSilent** command options for viewing information about interim fixes show that the interim fix is installed or removed?
See Viewing information about interim fixes for more information.
- Do the **updateSilent** command options for viewing information about cumulative fixes and fix packs show that the cumulative fix or the fix pack is installed or removed?
See Viewing information about cumulative fixes and fix packs for more information.

You can successfully install interim fixes, cumulative fixes, and fix packs on a base node.

Using the update installer application to update a deployment manager node

This topic describes how to use the update installer program to install interim fixes, cumulative fixes, and fix packs on a deployment manager node.

If you have not already done so, read “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for an overview of installing service to WebSphere Application Server products. Update the deployment manager node before updating managed base nodes, which are base nodes that you have federated into the cell.

If you extended the deployment manager by installing the WebSphere Business Integration Server Foundation product, see Using the update installer application to update an extended node.

This topic describes the proper procedure for installing an interim fix, a cumulative fix or a fix pack in a Network Deployment, V5.x environment, using the update installer application.

1. Stop the nodeagent process on each managed node in the cell with the **stopNode** command. Issue the **stopNode** command from the *install_root/bin* directory of each managed node. For example, issue the following command on a Linux platform:

```
# ./stopNode.sh -user name -password password
```

See stopNode command for more information about the command.

Use the Windows Services panel to stop any Windows service for the nodeagent, and for any other WebSphere Application Server related services, including Application Server processes, the jmsserver process, IBM HTTP Server and WebSphere MQ queue managers.

Stop all Java processes on the node, if necessary.

2. Stop the deployment manager process with the **stopManager** command. The dmgr Java process is the deployment manager process. Issue the **stopManager** command from the *install_root/bin* directory of the deployment manager node. For example, issue the following command on a Linux platform:

```
# ./stopManager.sh -user name -password password
```

See stopManager command for more information about the command.

3. Create the *install_root/update* directory on the deployment manager node, if the directory does not already exist. Later, you will also launch the update installer from this directory. Launching the update installer is not supported from a read-only directory, or from a directory with spaces in its name.

For example, create the */opt/WebSphere/DeploymentManager/update* directory on a Linux platform.

On a Windows platform, create the update directory in a path where none of the directory names includes a space in their names. For example, do not create the *C:\Program Files\WebSphere\DeploymentManager\update* directory because the Program Files directory has a space in its name.

You can install a fix from the *C:\WebSphere\update* directory, for example, to the Network Deployment product in the default installation root directory, *C:\Program Files\WebSphere\DeploymentManager*. The target directory can have a space; the source directory cannot.

4. Create the *update/fixes* repository if you are installing an interim fix. It is not necessary to create the *fixpacks* repository directory. Unpacking a cumulative fix or a fix pack creates the *fixpacks* directory if the directory does not already exist.
5. Download the interim fix, the cumulative fix, or the fix pack. Download an interim fix from the Support page to the *update/fixes* directory. Download a fix pack to the *update* directory.
6. Unpack the interim fix, the cumulative fix, or the fix pack. Unpacking a cumulative fix or a fix pack automatically creates the *fixpacks* directory.

Table 50. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
Windows platforms	Use another unzip product such as WINZIP, instead of the PKWARE pkunzip utility to unzip the product archive.

7. Download the ZIP file for the current update installer and extract the update installer application to the *update* directory. You can use the update installer application that is packaged as part of a fix pack. Or, you can download the current version of the file even though you might have an update installer from the fix pack or from a previous fix installation. The Support page links to the current installer for the version of the product that you are updating.

8. Verify that the unpacked files are owned by root on a Linux or UNIX-based platform.

- a. List the contents of the download directory. For example:

```
# ls -al
```

```
drwxr-xr-x  6 root    bin          512 Jul 21 08:50 was51fp1_linux
```

The directory list in the preceding example shows a Fix Pack 1 file for V5.1 that is not owned by root.

- b. Change the ownership of any files not owned by root. You can change the ownership of all files in the download directory.

For example:

```
# chmod -R root:root *
```

9. Set up the Java environment for the update installer.

Setting the JAVA_HOME environment variable

Important: If the update installer can set the Java environment, this step is unnecessary. Otherwise, this is a required step.

It is possible that the update installer cannot set the JAVA_HOME environment variable. If you receive a message that the update installer cannot set JAVA_HOME, or if you have set the JAVA_HOME variable to point to a non-WebSphere Application Server SDK, you must set the JAVA_HOME variable to a correct value.

Set the environment variable yourself when there is a problem, or source the appropriate command script from the **bin** directory of the product installation root so that the variable points to an IBM SDK for a WebSphere Application Server product.

If you use the same command window that you use to issue the **stopNode** command, the **stopManager** command, or the **stopServer** command, you have already sourced the `setupCmdLine` script and set the JAVA_HOME environment variable.

Otherwise, use the following procedure to update the JAVA_HOME variable:

- a. Open a command-line window.
- b. Change directories to the `bin` directory of the installation root.
- c. Issue the command to set JAVA_HOME. Issue the appropriate command:
 - `. install_root/bin/setupCmdLine.sh` (source the command on UNIX platforms - There is a space between the period and the installation root directory.)
 - `source install_root/bin/setupCmdLine.sh` (source the command on Linux platforms)
 - `install_root\bin\setupCmdLine.bat` (Windows platforms only)

10. Install the interim fix, cumulative fix, or fix pack on the deployment manager node.

Use the **updateWizard** command or the **updateSilent** command to install the interim fix, cumulative fix, or fix pack on the deployment manager node. The choice is whether to use a wizard. For more information about using either command, see the following articles:

- “updateWizard command” on page 230
- “updateSilent command” on page 218

For example, assume that you are installing Fix Pack 1 on Version 5.1.0.0. To install the `was51_nd_fp1_win` fix pack, use this **updateSilent** command:

```
C:\WebSphere\DeploymentManager\update> updateSilent
-fixpack
-installDir "C:\Program Files\WebSphere\DeploymentManager"
-skipIHS
-fixpackDir "C:\WebSphere\DeploymentManager\update\fixpacks"
-install
-fixpackID was51_nd_fp1_win
```

This example skips applying any service that might be in the fix pack for the IBM HTTP Server feature or the embedded messaging feature.

The command is shown on more than one line for clarity.

11. Bring the deployment manager node back online with the **startManager** command. Issue the **startManager** command from the `install_root/bin` directory of the deployment manager node. For example, issue the following command on a Linux platform:

```
# ./startManager.sh
```

See `startManager` command for more information about the command.

12. Verify that the deployment manager node is fully functional and that it has the interim fix, the cumulative fix, or the fix pack applied. There are several ways to verify the successful application of an interim fix, a cumulative fix, or a fix pack:
 - Does the fix show up in the wizard panel that lists the appropriate type of installed fix?

- Installed interim fixes. See Uninstallable fix selection panel .
- Installed cumulative fixes or fix packs. See Installing and Uninstalling panel.

If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is installed.

- Does the fix show up in the wizard panel that the appropriate type of installable (not yet installed) fix ?

- Installable interim fixes. See Installable fix selection panel .
- Installable cumulative fixes or fix packs. See Installable fix selection panel .

If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is not installed.

- Does one of the following files exist in the *install_root/properties/version/version* directory?

- [interim_fix_ID].efix
- [cumulative_fix_ID].ptf
- [fix_pack_ID].ptf

Or, does one of the following files exist in the *install_root/properties/version/history* directory?

- [interim_fix_ID].efixApplied
- [interim_fix_ID].efixDriver
- [cumulative_fix_ID].ptf
- [cumulative_fix_ID].ptf
- [fix_pack_ID].ptfApplied
- [fix_pack_ID].ptfDriver

These types of files indicate the installation of a fix. See Product information files for more information.

- Do the reports that describe the product version and its history show that the interim fix, the cumulative fix, or the fix pack is installed or that it has been removed?

See Reports for more information about the reports.

- Do the **updateSilent** command options for viewing information about interim fixes show that the interim fix is installed or removed?

See Viewing information about interim fixes for more information.

- Do the **updateSilent** command options for viewing information about cumulative fixes and fix packs show that the cumulative fix or the fix pack is installed or removed?

See Viewing information about cumulative fixes and fix packs for more information.

13. Restart the node agent of each base node with the **startNode** command. Restart the node agent on each managed node to let the node agent continue to communicate with the updated deployment manager node.

You can restart all node agents, but you do not need to restart node agents on managed nodes that you intend to update with the interim fix, the cumulative fix, or the fix pack. The interim fix installation, the cumulative fix installation, and the fix pack installation each require you to stop and restart the node agent. You can simply restart the node agent at the appropriate time.

Issue the **startNode** command from the *install_root/bin* directory of each base node. For example, issue the following command on a Linux platform:

```
# ./startNode.sh
```

See startNode command for more information about the command.

14. Install the interim fix, the cumulative fix, or the fix pack on each base node that is managed by the deployment manager and that is a node to which you intend to apply the fix. See “Using the update installer to update a base node” on page 211 for information about how to perform this step.
15. Restart each server on the node with the **startServer** command.
16. Restart the node agent for the base node with the **startNode** command if the node is part of the cell.

17. Verify that the base node is fully functional and that it has the interim fix, the cumulative fix, or the fix pack installed.
18. Specify that file sets on each base node match those on the deployment manager node. Ensure consistent configuration data across a cell. You can synchronize files on individual nodes or throughout your system.

To synchronize files throughout the system, use the deployment manager administrative console page, **System administration > Nodes > *check_each_node_name* > Full Resynchronization**. You can use the administrative console page, **System Administration > Node Agents > nodeagent > File Synchronization Service** to specify automatic synchronization every minute until all base node servers are brought online.

19. Verify that all nodes are online and that the cell is functioning correctly.
20. Restore your original file synchronization settings, if you changed them. If all operations are available and functioning normally, the cell is now fully functional.

You can successfully install interim fixes, cumulative fixes, and fix packs to any node in the deployment manager cell.

updateSilent command

The **updateSilent** command is the silent, command-line interface to the update installer program of the IBM WebSphere Application Server. You can also use a wizard interface to the update installer program, the **updateWizard** command. The update installer program is also known as the updateInstaller program or the Update installation wizard.

The update installer program installs interim fixes, cumulative fixes, and fix packs to WebSphere Application Server products.

The relationship among interim fixes, cumulative fixes, and fix packs is shown in the Cumulative Fix Strategy for WebSphere Application Server V5.0 and V5.1 Web page.

Overview

Both the **updateSilent** command and the “updateWizard command” on page 230 call the update installer program to install and uninstall interim fixes, cumulative fixes, and fix packs for WebSphere Application Server products. This topic describes the silent interface to the update installer program and its command-line parameters.

Stop all Java processes on the machine that use the IBM Software Developer Kit (SDK) that WebSphere Application Server provides: Before installing or uninstalling interim fixes, cumulative fixes, and fix packs on a machine, stop all Java processes on the machine that use the IBM SDK, Java Technology Edition that WebSphere Application Server provides. WebSphere Application Server processes include:

- Application Server processes
- All servers, such as the IBMHttpServer process, that belong to serviceable features. Features with servers include the IBM HTTP Server and the embedded messaging feature.
- The nodeagent process on a base node when the node is federated into a deployment manager cell
- The dmgr process for the deployment manager server
- The jmsserver on a base node when the node is part of a cell and the embedded messaging server and client feature is installed

Stop all Java processes, if necessary. If you do install or uninstall an interim fix, a cumulative fix, or a fix pack while a WebSphere Application Server-related Java process runs, IBM does not guarantee that the product can continue to run successfully, or without error.

Remove the WebSphere MQ tray icon if present On a Windows platform, remove the WebSphere MQ tray icon if it is present. The WebSphere MQ tray icon in the lower right corner indicates that a WebSphere MQ process (**amqmtbrn.exe**) is running. Right click the tray icon and click **Hide** to remove it.

Do not launch multiple copies of the update installer program at one time The update installer program cannot be launched concurrently with itself. Performing more than one update at the same time can lead to a failed or faulty installation.

The following descriptions contain reference information about the command. See “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for more information about using the command.

Command name

updateSilent.sh and updateSilent.bat, command-line interface to the installer.jar file.

Related command

updateWizard.sh and updateWizard.bat, graphical interface to the installer.jar file.

Syntax and panel examples

Using the updateSilent interface to work with interim fixes

See “updateSilent examples for interim fixes” for more information.

Using the updateSilent interface to work with cumulative fixes and fix packs

See “updateSilent examples for cumulative fixes and fix packs” on page 224 for more information.

updateSilent examples for interim fixes

The **updateSilent** command is the silent, command-line interface for the update installer program. This topic describes using the wizard to work with interim fixes.

The update installer program installs interim fixes, cumulative fixes, and fix packs to WebSphere Application Server products.

Three different Web sites contain interim fixes for WebSphere Application Server products and features. Three sites exist because two of the features of the base product have their own service and support sites.

You cannot install or uninstall interim fixes for either of the two features using the update installer program for WebSphere Application Server. See the following tips for more information:

Table 51. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	<ul style="list-style-type: none"> Installing interim fixes for the IBM HTTP Server feature and the embedded messaging feature Uninstalling interim fixes for the IBM HTTP Server feature and the embedded messaging (WebSphere MQ) feature before installing cumulative fixes and fix packs to the features

You must use the update installer program to install cumulative fixes and fix packs for the two features. See “updateSilent examples for cumulative fixes and fix packs” on page 224 for more information about using the update installer program to install cumulative fixes and fix packs.

Syntax examples

The updateSilent interface actually provides two functions. Depending upon the parameters you choose, the command performs the following functions:

- Installs and uninstalls interim fixes, cumulative fixes, and fix packs
- Provides information about the update state of fixes and fix packs that you install or uninstall

The following examples describe various usage syntaxes. In each syntax example, optional parameters are enclosed by brackets ([]). Values that you must supply appear in *italicized font*. Choices are denoted by the pipe symbol (|).

Help

```
updateSilent -help | -? | -usage
```

```
updateSilent /help | /? | -usage (Windows platforms)
```

Syntax for using a properties file to supply values

```
updateSilent myProps.properties
```

Syntax for processing interim fixes

```
updateSilent -installDir "fully qualified product install_root"  
-fix  
-fixDir "fully qualified interim fix repository root,  
         usually install_root/update/fixes"  
-install | -uninstall | -uninstallAll  
-fixes space-delimited list of fixes  
-fixJars space-delimited list of interim fix JAR files  
[-fixDetails]  
[-prereqOverride]
```

Syntax for viewing installed interim fixes

```
updateSilent  
-fix  
-installDir "fully qualified product install_root"
```

Syntax for viewing available interim fixes

```
updateSilent  
-fix  
-installDir "fully qualified product install_root"  
-fixDir "fully qualified interim fix repository root,  
         usually install_root/update/fixes"
```

Parameters

Use the following parameters for the **updateSilent** command:

-? Shows command usage.

/? Shows command usage on Windows platforms only.

-fix

Interim fix only: Identifies the update as an interim fix update.

-fixDetails

Interim fix only: Displays interim fix detail information.

-fixDir

Interim fix only: Specifies the fully qualified directory where you download fixes. This directory is usually the *install_root/update/fixes* directory.

-fixes

Interim fix only: Specifies a list of space-delimited fixes to install or uninstall. Specify either the *fixJars* parameter or this parameter.

-fixJars

Interim fix only: Specifies a list of space-delimited interim fix JAR files to install or uninstall. Each JAR file has one or more fixes. Specify either the *fixes* parameter or this parameter.

-help

Shows command usage.

/help

Shows command usage on Windows platforms only.

-install

Installs the update type.

-installDir

Specifies the fully qualified installation root of the WebSphere Application Server product.

-prereqOverride

Interim fix only: Overrides any installation and uninstallation prerequisite checking. The update installer program does not log missing prerequisites.

<propertyFile>.properties

Specifies an externally supplied parameters file.

You can supply parameters in an external .properties file, rather than directly on the command line. There are some differences in the formats of parameters:

- You must include all parameters.
- Parameters are [name]=[value] pairs.
- Lists of parameter values are comma-delimited instead of space-delimited.
- There are two slashes before directory names.

You can use the .properties file as a template.

For example, a sample.properties file for installing two fixes might look like this:

```
#Sample.properties
#Sample parameters file to install fixes with details and prerequisite override
fix=true
install=true
installDir=C:\\WebSphere\\AppServer
fixDir=C:\\WebSphere\\AppServer\\update\\fixes
fixes=Fix1,Fix2
fixDetails=true
prereqOverride=true
```

-uninstall

Uninstalls the identified interim fix, cumulative fix, or fix pack. You must uninstall all interim fixes, cumulative fixes, and fix packs before uninstalling a WebSphere Application Server product. However, if you are manually uninstalling all WebSphere Application Server products from the machine, it is not necessary to uninstall all interim fixes, cumulative fixes or fix packs. See “Uninstalling manually” on page 292 for more information.

-uninstallAll

Uninstalls all applied interim fixes. This parameter does not uninstall fix packs.

If you installed interim fixes from the WebSphere MQ Web site, you must use the uninstaller from WebSphere MQ to uninstall interim fixes for the embedded messaging feature. For more information about managing interim fixes for the embedded messaging feature, see the WebSphere MQ Service download site.

If you installed interim fixes from the IBM HTTP Server Web site, you must use the uninstaller from IBM HTTP Server to uninstall interim fixes for the IBM HTTP Server feature. For more information about managing interim fixes for IBM HTTP Server, see the IBM Support site for IBM HTTP Server.

-usage

Shows command usage.

Examples overview

The following examples assume that:

- The installation root is the C:\WebSphere\AppServer directory.
- The location of the IBM HTTP Server feature is the C:\Program Files\IBMHttpServer directory.
- The interim fix repository is the C:\WebSphere\AppServer\update\fixes directory.
- The cumulative fix and fix pack repository is the C:\WebSphere\AppServer\update\fixpacks directory.

Examples in this section include:

- “Getting help for the command”
- “Using a parameter properties file”
- “Installing interim fixes”
- “Uninstalling interim fixes” on page 223
- “Viewing information about interim fixes” on page 224

Most of the examples are split into more than one line, for ease of publication.

Getting help for the command

To get help for the **updateSilent** command:

```
C:\WebSphere\AppServer\update> updateSilent -help
```

Using a parameter properties file

To use the `myProps.properties` file to supply parameter values for the **updateSilent** command:

```
C:\WebSphere\AppServer\update> updateSilent myProps.properties
```

Installing interim fixes

To install a collection of interim fixes:

```
C:\WebSphere\AppServer\update> updateSilent
  -fix
  -installDir "C:\WebSphere\AppServer"
  -fixDir "C:\WebSphere\AppServer\update\fixes"
  -install
  -fixes Fix1 Fix2
```

It is not possible to install interim fixes for these features:

- IBM HTTP Server feature
- Embedded messaging feature

For more information about managing interim fixes for IBM HTTP Server, see the IBM Support site for IBM HTTP Server. For more information about managing interim fixes for the embedded messaging feature, see the WebSphere MQ Service download site.

To install a collection of interim fixes, and display interim fix details:

```
C:\WebSphere\AppServer\update> updateSilent
  -fix
  -installDir "C:\WebSphere\AppServer"
  -fixDir "C:\WebSphere\AppServer\update\fixes"
  -install
  -fixes Fix1 Fix2
  -fixDetails
```

To install a collection of interim fixes, and override prerequisite checking:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-install
-fixes Fix1 Fix2
-prereqOverride
```

To install interim fixes from a Java archive (JAR) file:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-install
-fixJar Fix1
```

To install interim fixes from a Java archive (JAR) file, and display interim fix details:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-install
-fixJar Fix1
-fixDetails
```

To install interim fixes from a Java archive (JAR) file:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-install
-fixJar Fix1
-fixDetails
```

Uninstalling interim fixes

To uninstall a collection of interim fixes:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-uninstall
-fixes Fix1 Fix2
```

It is not possible to uninstall interim fixes for these features:

- IBM HTTP Server feature
- Embedded messaging feature

For more information about managing interim fixes for IBM HTTP Server, see the IBM Support site for IBM HTTP Server. For more information about managing interim fixes for the embedded messaging feature, see the WebSphere MQ Service download site.

To uninstall a collection of interim fixes, and display details:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-uninstall
-fixes Fix1 Fix2
-fixDetails
```

To uninstall a collection of interim fixes, and override prerequisite checking:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-uninstall
-fixes Fix1 Fix2
-prereqOverride
```

To uninstall interim fixes in a Java archive (JAR) file:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-uninstall
-fixJar Fix1
```

To uninstall interim fixes in a Java archive (JAR) file, and display details:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-uninstall
-fixJar Fix1
-fixDetails
```

To uninstall interim fixes in a Java archive (JAR) file:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
-uninstall
-fixJar Fix1
-fixDetails
```

Viewing information about interim fixes

To view a list of installed interim fixes:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
```

To view a list of interim fixes that are available in the repository:

```
C:\WebSphere\AppServer\update> updateSilent
-fix
-installDir "C:\WebSphere\AppServer"
-fixDir "C:\WebSphere\AppServer\update\fixes"
```

It is not possible to view information about interim fixes for the IBM HTTP Server feature or the embedded messaging feature. For more information about managing interim fixes for IBM HTTP Server, see the IBM Support site for IBM HTTP Server. For more information about managing interim fixes for the embedded messaging feature, see the WebSphere MQ Service download site.

updateSilent examples for cumulative fixes and fix packs

The **updateSilent** command is the silent, command-line interface for the update installer program. This topic describes using the wizard to work with cumulative fixes and fix packs.

The update installer program installs interim fixes, cumulative fixes, and fix packs to WebSphere Application Server products. See “updateWizard examples for interim fixes” on page 231 for information about using the update silent interface to work with interim fixes.

The relationship among interim fixes, cumulative fixes, and fix packs is shown in the Cumulative Fix Strategy for WebSphere Application Server V5.0 and V5.1 Web page.

Syntax examples

The updateSilent interface actually provides two functions. Depending upon the parameters you choose, the command performs the following functions:

- Installs and uninstalls interim fixes, cumulative fixes, and fix packs
- Provides information about the update state of interim fixes, cumulative fixes, and fix packs that you install or uninstall

The following examples describe various usage syntaxes. In each syntax example, optional parameters are enclosed by brackets ([]). Values that you must supply appear in *italicized font*. Choices are denoted by the pipe symbol (|).

Help

```
updateSilent -help | -? | -usage
```

```
updateSilent /help | /? | -usage (Windows platforms)
```

Syntax for using a properties file to supply values

```
updateSilent myProps.properties
```

Syntax for processing cumulative fixes and fix packs

```
updateSilent -installDir "fully qualified product install_root"  
-fixpack  
-fixpackDir "fully qualified FixPack repository root,  
              usually install_root/update/fixpacks"  
-install | -uninstall  
-fixPackID fix pack ID  
[-skipIHS | [-ihsOnly] -ihsInstallDir fully qualified IBM HTTP Server root]  
[-skipMQ | -mqInstallDir embedded messaging feature root]  
[-includeOptional space-delimited list of components]  
[-fixpackDetails]
```

All other valid arguments are ignored, such as the prereqOverride argument, which is for interim fix processing only.

You do not need to supply the -mqInstallDir argument for AIX, Linux, and UNIX-based platforms. The install location is fixed on those operating platforms. Use the argument on Windows platforms. The default location on Windows platforms is the C:\Program Files\IBM\WebSphere MQ directory.

Syntax for viewing installed cumulative fixes and fix packs

```
updateSilent -fixpack  
-installDir "fully qualified product install_root"
```

Syntax for viewing available cumulative fixes and fix packs

```
updateSilent -fixpack  
-installDir "fully qualified product install_root"  
-fixpackDir "fully qualified fix pack repository root,  
              usually install_root/update/fixpacks"
```

Parameters

Use the following parameters for the **updateSilent** command:

-? Shows command usage.

/? Shows command usage on Windows platforms only.

-fixpack

Fix pack only: Identifies the update as a fix pack update.

-fixpackDetails

Fix pack only: Displays fix pack detail information.

-fixpackDir

Fix pack only: Specifies the fully qualified directory where you download and unpack fix packs. By default, this directory is the *install_root/update/fixpacks* directory.

-fixpackID

Fix pack only: Specifies the ID of a fix pack to install or uninstall. The value you specify does not include the *.jar* extension. The value is not the fully qualified package file name, but is the name of the individual fix pack within the JAR file.

The current Application Server strategy for fix pack JAR files uses one JAR file per fix pack. The fix pack ID is the name of the JAR file before the *.jar* extension.

5.1 + For example:

- **Fix pack ID:** was51_fp1_linux
- **Fix pack JAR file name:** was51_fp1_linux.jar
- **Fix pack ZIP file name:** was51_fp1_linux.zip

-help

Shows command usage.

/help

Shows command usage on Windows platforms only.

-ihsInstallDir

Fix pack only: Specifies the fully qualified path of the IBM HTTP Server product, and applies any service for the IBM HTTP Server product that might exist in the fix pack.

Important: Cumulative fixes and fix packs do not include service for the IBM HTTP Server powered by Apache 2.0. If you installed IBM HTTP Server powered by Apache 2.0, skip applying service to the IBM HTTP Server during the installation of a cumulative fix or a fix pack. Do not update the 2.0 level IBM HTTP Server with service for the 1.3 level IBM HTTP Server.

-ihsOnly

Fix pack only: Skips the installation of all service but that for the IBM HTTP Server product, and applies any service for the IBM HTTP Server product that might exist in the fix pack. Requires the *-ihsInstallDir* parameter.

Important: Cumulative fixes and fix packs do not include service for the IBM HTTP Server powered by Apache 2.0. If you installed IBM HTTP Server powered by Apache 2.0, skip applying service to the IBM HTTP Server during the installation of a cumulative fix or a fix pack. Do not update the 2.0 level IBM HTTP Server with service for the 1.3 level IBM HTTP Server.

-includeOptional

Fix pack only: Specifies a space-delimited list of features. The installer program applies any service for the components, if present in the fix pack. Otherwise, the installer program does not apply the service.

-install

Installs the update type.

-installDir

Specifies the fully qualified installation root of the WebSphere Application Server product.

-mqInstallDir

Fix pack only: Specifies the fully qualified installation root of the embedded messaging feature, which is based on WebSphere MQ technology.

<propertyFile>.properties

Specifies an externally supplied parameters file.

You can supply parameters in an external .properties file, rather than directly on the command line. There are some differences in the formats of parameters:

- You must include all parameters.
- Parameters are [name]=[value] pairs.
- Lists of parameter values are comma-delimited instead of space-delimited.
- There are two slashes before directory names.

A sample.properties file for installing a fix pack might look like the following example:

-skipIHS

Fix pack only: Skips any optional service for the IBM HTTP Server product that might exist in the fix pack.

If you installed the IBM HTTP Server product as a feature, use the update installer program to update it with service in an interim fix or fix pack. Otherwise, you must download an updated IBM HTTP Server product from <http://www.ibm.com/software/webervers/httpservers/> and install it into the same directory as your existing version, to update the existing installation. You can also uninstall the current version and install the downloaded version, to avoid any issues with migration.

You must update your configuration if you reinstall. The process is described in the *Manually configuring supported Web servers* (tins_manualWebServer) topic in the information center for the base Application Server product.

Important: Cumulative fixes and fix packs do not include service for the IBM HTTP Server powered by Apache 2.0. If you installed IBM HTTP Server powered by Apache 2.0, skip applying service to the IBM HTTP Server during the installation of a cumulative fix or a fix pack. Do not update the 2.0 level IBM HTTP Server with service for the 1.3 level IBM HTTP Server.

-skipMQ

Fix pack only: Specifies no application of any optional service for the embedded messaging feature (which is based on the IBM WebSphere MQ product) that might exist in the fix pack.

Always apply any outstanding corrective service to the stand-alone IBM WebSphere MQ product if you have it, before using the WebSphere Application Server update installer program to update the embedded messaging feature with service in an interim fix or fix pack. You can skip the installation of service to the embedded messaging feature if you must install corrective service to the stand-alone IBM WebSphere MQ product.

-uninstall

Specifies to uninstall the identified interim fix, cumulative fix, or fix pack. You must uninstall all interim fixes, cumulative fixes, and fix packs before uninstalling a WebSphere Application Server product. However, if you are manually uninstalling all WebSphere Application Server products from the machine, it is not necessary to uninstall all interim fixes, cumulative fixes or fix packs. See “Uninstalling manually” on page 292 for more information.

-usage

Shows command usage.

Examples overview

The following examples assume that:

- The installation root is the C:\WebSphere\AppServer directory.
- The location of the IBM HTTP Server feature is the C:\Program Files\IBMHttpServer directory.
- The interim fix repository is the C:\WebSphere\AppServer\update\fixes directory.
- The cumulative fix and fix pack repository is the C:\WebSphere\AppServer\update\fixpacks directory.

Examples in this section include:

- “Getting help for the command”
- “Using a parameter properties file”
- “Installing cumulative fixes and fix packs”
- “Uninstalling cumulative fixes and fix packs” on page 230
- “Viewing information about cumulative fixes and fix packs” on page 230

Most of the examples are split into more than one line, for ease of publication.

Getting help for the command

To get help for the **updateSilent** command:

```
C:\WebSphere\AppServer\update> updateSilent -help
```

Using a parameter properties file

To use the `myProps.properties` file to supply parameter values for the **updateSilent** command:

```
C:\WebSphere\AppServer\update> updateSilent myProps.properties
```

Installing cumulative fixes and fix packs

To install a fix pack:

```
C:\WebSphere\AppServer\update> updateSilent
  -fixpack
  -installDir "C:\WebSphere\AppServer"
  -ihsInstallDir "C:\Program Files\IBMHttpServer"
  -fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
  -install
  -fixpackID was510_cf2_win
```

To install a fix pack, and display fix pack details:

```
C:\WebSphere\AppServer\update> updateSilent
  -fixpack
  -installDir "C:\WebSphere\AppServer"
  -ihsInstallDir "C:\Program Files\IBMHttpServer"
  -fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
  -install
  -fixpackID was510_cf2_win
  -fixpackDetails
```

To perform a partial installation of a fix pack, by choosing to skip the installation of optional service to the WebSphere Application Server embedded messaging feature, which is based on WebSphere MQ:

```
C:\WebSphere\AppServer\update> updateSilent
  -fixpack
  -installDir "C:\WebSphere\AppServer"
  -fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
  -ihsInstallDir "C:\Program Files\IBMHttpServer"
  -install
  -fixpackID was510_cf2_win
  -skipMQ
```

The fix pack status shows partial installation.

About installation status

An interim fix or a fix pack is a collection of updates to one or more product components. Depending on installed product components and on update installer program selections you make, the update installer program applies either a full or partial set of interim fix or fix pack updates to product components.

Installed status implies that the interim fix or fix pack has no more updates to product components that you can install.

Partially installed status implies that you have updated one or more product components, but the interim fix or fix pack has at least one more update you can apply to an installed product component. (There might be other updates to product components you never installed. These updates do not count in the status determination.)

Uninstalled status implies that you have not updated a single product component.

Examples of partially installed states: Several scenarios can lead to a partial installation of an interim fix or fix pack:

- Installation fails, leaving some component updates applied and some unapplied. This is a partial installation accompanied by error messages that describe the problem.
- You select to skip certain optional component updates, such as might be present in a fix pack for these features: IBM HTTP Server or embedded messaging. This is a partial installation.
- You trigger a dynamic change in interim fix or fix pack status because:
 1. You apply all changes in an interim fix or fix pack that results in an installed status.
 2. You reinstall the WebSphere Application Server product to select an additional, optional feature.
 3. The interim fix or fix pack contains an update to a product component you just installed.

The status changes dynamically from installed to partially installed.

How to distinguish between a normal partially installed state and an installation failure

A partially installed state implies that some components on the system are at the level of the fix pack being applied. However, there might be some scenarios where the update installer program can report a partially installed state even if no components are at the newer level. A partially installed state does not imply a failed install. However, a failed install does result in a partially installed state.

5.1 + To check for a failed install, open the master log file and search for errors. The update installer program outputs the name of the master log file at the start of the installation process. The file name matches this pattern: *install_root/logs/update/time-stamp_was51_fpnumber_platform_selective-install.log*

To perform a partial installation of a fix pack by choosing to skip the installation of optional service to the IBM HTTP Server feature:

```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\WebSphere\AppServer"
-fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
-mqInstallDir "C:\WebSphere MQ"
-install
-fixpackID was510_cf2_win
-skipIHS
```

The fix pack status shows partial installation.

To perform a partial installation of a fix pack, by choosing to skip the installation of optional service to both the embedded messaging feature and the IBM HTTP Server feature:

```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\WebSphere\AppServer"
-fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
-install
-fixpackID was510_cf2_win
-skipIHS
-skipMQ
```

The fix pack status shows partial installation.

Uninstalling cumulative fixes and fix packs

To uninstall a fix pack:

```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\WebSphere\AppServer"
-uninstall
-fixpackID was510_cf2_win
```

To uninstall a fix pack, and display fix pack details:

```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\WebSphere\AppServer"
-uninstall
-fixpackID was510_cf2_win
-fixpackDetails
```

Viewing information about cumulative fixes and fix packs

To view a list of installed fix packs:

```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\WebSphere\AppServer"
```

To view a list of fix packs available in the repository for the base WebSphere Application Server product:

```
C:\WebSphere\AppServer\update> updateSilent
-fixpack
-installDir "C:\WebSphere\AppServer"
-fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
```

updateWizard command

The **updateWizard** command is the wizard interface to the IBM WebSphere Application Server update installer program. You can also use a silent, command-line interface to the update installer program, the **updateSilent** command. The update installer program is also known as the updateInstaller program or the Update installation wizard.

The update installer program installs interim fixes, cumulative fixes, and fix packs to WebSphere Application Server products.

The relationship among interim fixes, cumulative fixes, and fix packs is shown in the Cumulative Fix Strategy for WebSphere Application Server V5.0 and V5.1 Web page.

Overview

Both the **updateWizard** command and the “updateSilent command” on page 218 command call the update installer program to install and uninstall interim fixes, cumulative fixes, and fix packs for WebSphere Application Server products. This topic describes the wizard interface to the update installer command, and gives some information about its panels and fields.

Stop all Java processes on the machine that use the IBM Software Developer Kit (SDK) that WebSphere Application Server provides: Before installing or uninstalling interim fixes, cumulative fixes, and fix packs on a machine, stop all Java processes on the machine that use the IBM SDK, Java Technology Edition that WebSphere Application Server provides. WebSphere Application Server processes include:

- Application Server processes

- All servers, such as the IBMHttpServer process, that belong to serviceable features. Features with servers include the IBM HTTP Server and the embedded messaging feature.
- The nodeagent process on a base node when the node is federated into a deployment manager cell
- The dmgr process for the deployment manager server
- The jmsserver on a base node when the node is part of a cell and the embedded messaging server and client feature is installed

Stop all Java processes, if necessary. If you do install or uninstall an interim fix, a cumulative fix, or a fix pack while a WebSphere Application Server-related Java process runs, IBM does not guarantee that the product can continue to run successfully, or without error.

Remove the WebSphere MQ tray icon if present On a Windows platform, remove the WebSphere MQ tray icon if it is present. The WebSphere MQ tray icon in the lower right corner indicates that a WebSphere MQ process (**amqmtbrn.exe**) is running. Right click the tray icon and click **Hide** to remove it.

Do not launch multiple copies of the update installer program at one time The update installer program cannot be launched concurrently with itself. Performing more than one update at the same time can lead to a failed or faulty installation.

The following descriptions contain reference information about the command. See “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for more information about using the command.

Command name

updateWizard.sh and updateWizard.bat, graphical interface to the installer.jar file.

Related command

updateSilent.sh and updateSilent.bat, command-line interface to the installer.jar file.

See “updateSilent command” on page 218.

Location of the command

See “Installing interim fixes, cumulative fixes, and fix packs” on page 202 for a list of product installation roots, fix pack space requirements and names, and other information that also applies when using the **updateWizard** command interface.

Syntax and panel examples

Using the updateWizard interface to work with interim fixes

See “updateWizard examples for interim fixes” for more information.

Using the updateWizard interface to work with cumulative fixes and fix packs

See “updateWizard examples for cumulative fixes and fix packs” on page 236 for more information.

updateWizard examples for interim fixes

The **updateWizard** command starts the Update installer wizard. This topic describes using the wizard to work with interim fixes.

The update installer program installs interim fixes, cumulative fixes, and fix packs to WebSphere Application Server products.

Installing interim fixes for features that are products

Three different Web sites contain interim fixes for WebSphere Application Server products and features. Three sites exist because two of the features of the base product have their own service and support sites.

You cannot install or uninstall interim fixes for either of the two features using the update installer program for WebSphere Application Server. See the following tips for more information:

Table 52. Installation tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	<ul style="list-style-type: none">• Installing interim fixes for the IBM HTTP Server feature and the embedded messaging feature• Uninstalling interim fixes for the IBM HTTP Server feature and the embedded messaging (WebSphere MQ) feature before installing cumulative fixes and fix packs to the features

You must use the update installer program to install cumulative fixes and fix packs for the two features. See “updateWizard examples for cumulative fixes and fix packs” on page 236 for information about using the update installer program to install cumulative fixes and fix packs.

Parameters

Apply zero, one, or more of these optional parameters in any order, by issuing the **updateWizard** command from the command line.

-dplInstall

Lets you install an interim fix when there are prerequisite check errors.

The default behavior of the update installer application prevents further action if prerequisites are missing.

-dpUninstall

Lets you uninstall an interim fix when there are prerequisite check errors.

-fixOnly

Installs or uninstalls only interim fixes. To install and uninstall interim fixes, the **updateWizard** command does not require a local copy of the IBM Developer Kit or the Java Runtime Environment (JRE) in a client environment. This option bypasses making a local copy of the IBM Developer Kit or the JRE.

The default action for the update installer program enables both installing and uninstalling interim fixes, cumulative fixes, and fix packs. The wizard installs a temporary version of one of the IBM products that WebSphere Application Server uses to support the Java 2 SDK on your operating system platform, such as the IBM Developer Kit for AIX, Java Technology Edition.

The **updateWizard** command copies the IBM Developer Kit from the `JAVA_HOME` directory to the directory where you are running the **updateWizard** command (usually `install_root/update`). If the IBM Developer Kit is already in the directory (for example, if you used the **updateWizard** command before), the **updateWizard** command does not make a new copy. The copy of the IBM Developer Kit remains in the directory until you remove it. The IBM Developer Kit requires approximately 43 MB of free space.

If you install on a client platform, where there is a JRE instead of the IBM Developer Kit, the **updateWizard** command copies the JRE to the `install_root/update` directory. The JRE requires about 18 MB of free space.

-usage

Displays a list of parameters and how to use them in the correct command syntax.

Syntax for starting the wizard

The **updateWizard** command (`install_root/update/updateWizard.sh` and `install_root\update\updateWizard.bat`) launches the wizard interface to the update installer application.

`install_root/update/updateWizard.sh` (Linux and UNIX-based platforms)

`install_root\update\updateWizard.bat` (Windows platforms)

Syntax for displaying usage information

This command displays command syntax:

```
updateWizard -usage
```

Syntax for bypassing the local copy of the IBM Developer Kit

The following command bypasses making a local copy of the IBM Developer Kit. Installing and uninstalling interim fixes does not require the local copy.

```
updateWizard -fixOnly
```

Disabling prerequisite checking

Disabling prerequisite checking is recommended only as directed by IBM Support personnel. Disabling prerequisite checking can leave the installation in a non-valid state.

To disable prerequisite checking when installing interim fixes:

```
updateWizard -fixOnly -dpInstall
```

To disable prerequisite checking when uninstalling interim fixes:

```
updateWizard -fixOnly -dpUninstall
```

Panel descriptions

Panels in the wizard let you select installable interim fixes, view installed interim fixes, and view prerequisite interim fixes:

- **General**
 - Welcome panel
 - Product selection panel
 - Menu panel
- **Installing and uninstalling interim fixes**
 - Fix repository specifier panel
 - Installable interim fix selection panel
 - Uninstallable interim fix selection panel
 - Prerequisite check panel
 - Pre-installing and pre-uninstalling summary panels
 - Installing and uninstalling
 - Post installing and post uninstalling summary panels

Welcome panel

Use this panel to view a welcome message that contains a brief summary of the update wizard interface, or to link to the WebSphere Application Server Support page. (This link is not available on some UNIX-based platforms.) You can also view relevant legal notices.

Product selection panel

Use this panel to select an installed WebSphere Application Server product. If the wizard cannot detect an installed product, specify the product location in the **directory input** field. After selecting a product, its directory location appears in the **input field** for verification purposes. To make corrections or enter another product location, click **Specify product location**.

Menu panel

Use this panel to install or uninstall interim fixes, or to install or uninstall cumulative fixes and fix packs. If you started the wizard in `-fixOnly` mode, cumulative fix and fix pack options are disabled.

Fix repository specifier panel

Use this panel to provide the interim fix repository location in a directory input field. Specify the directory location of the downloaded interim fix JAR files. The default location for the repository is the `install_root/update/fixes` directory.

Installable fix selection panel

Use this panel to select one or more installable fixes for installing. Installed fixes do not appear in the list. Only uninstalled fixes or partially-installed fixes appear. The list includes interim fix ID name, build date, and the current applied state (uninstalled or partially-installed). Click **Details** for detailed information about selected fixes. The window that appears contains build version information, a long description, and a list of installation prerequisites.

About installation status

An interim fix, a cumulative fix, or a fix pack is a collection of updates to one or more product components. Depending on installed product components and on update installer program selections you make, the update installer program applies either a full or partial set of interim fix, cumulative fix, or fix pack updates to product components.

Installed status implies that the interim fix, cumulative fix, or fix pack has no more updates to product components that you can install.

Partially installed status implies that you have updated one or more product components, but the interim fix, cumulative fix, or fix pack has at least one more update that you can apply to an installed product component. (Updates to product components that you never installed do not count in the status determination.)

Uninstalled status implies that you have not updated a single product component.

Examples of partially installed states: Several scenarios can lead to a partial installation of an interim fix, a cumulative fix, or a fix pack:

- Installation fails, leaving some component updates applied and some unapplied. This is a partial installation accompanied by error messages that describe the problem.
- You skip optional component updates for the IBM HTTP Server or embedded messaging feature. This is a partial installation.
- You trigger a dynamic change in interim fix, cumulative fix, or fix pack status because:
 1. You apply all changes in an interim fix, a cumulative fix, or a fix pack that results in an installed status.
 2. You reinstall the WebSphere Application Server product to select an additional, optional feature.
 3. The interim fix, cumulative fix, or fix pack contains an update to a product component you just installed.

The status changes dynamically from installed to partially installed.

How to distinguish between a normal partially installed state and an installation failure

A partially installed state implies that some components on the system are at the level of the cumulative fix or fix pack being applied. However, there might be some scenarios where the update installer program can

report a partially installed state even if no components are at the newer level. A partially installed state does not imply a failed install. However, a failed install does result in a partially installed state.

To check for a failed install, open the master log file and search for errors. The update installer program outputs the name of the master log file at the start of the installation process. The file name matches this pattern: `install_root/logs/update/time-stamp_was50_fpnumber_platform_selective-install.log`

Uninstallable fix selection panel

Use this panel to select one or more installed interim fixes for uninstalling. Available fixes do not appear in the list. Only installed fixes appear. The list includes interim fix ID name, build date, and the current applied state (installed). Click **Details** to obtain more detailed information about a selected fix. The window that appears contains build version information, a long description, and a list of installation prerequisites.

You must uninstall all interim fixes before uninstalling the base WebSphere Application Server product, the Network Deployment product, or the product.

Prerequisite check panel

Use this panel to view prerequisite information when a selected interim fix has prerequisite fixes that are not installed. You cannot click **Next** until you correct the problem. The `-dpInstall` and the `-dpUninstall` parameters can override this lock, to let you continue installing or uninstalling even though a prerequisite checking failure occurs.

Pre-installing and Pre-uninstalling summary panels

Pre-installing summary panel Use this panel to display a summary of interim fixes that are selected for installation, the WebSphere Application Server product each interim fix is for, and the directory where each interim fix is located.

Pre-uninstalling Summary panel Use this panel to display a summary of interim fixes that are selected for uninstalling, and the WebSphere Application Server product to which each interim fix is currently applied.

Installing and Uninstalling panel

Installation action Use this panel to view the progress of installing selected interim fixes. Click **cancel** to revert the installation. Once cancelled, a message confirms that installed fixes are being rolled back. A similar progress panel then appears, to monitor the progress of rolling back the installation of selected fixes.

Uninstalling action Use this panel to view the progress of uninstalling selected interim fixes.

Post installation summary and Post uninstalling summary panels

Post installation summary panel Use this panel to view the results of the installation. Depending on the result, this panel can display a success message, a failure message, or a canceled message. When the success message appears, the installation process is complete. Click **Finish** to exit the panel. You can go back and install additional fixes, which takes you to the **Menu** panel.

Post uninstalling summary panel Use this panel to view the results of the uninstall procedure. Depending on the result, this panel can display a success message, a failure message, or a canceled message. When the success message appears, the uninstall process is complete. Click **Finish** to exit the panel. You can go back and uninstall additional fixes, which takes you to the **Menu** panel.

updateWizard examples for cumulative fixes and fix packs

The **updateWizard** command starts the Update installer wizard. This topic describes using the wizard to work with cumulative fixes and fix packs.

The update installer program installs interim fixes, cumulative fixes, and fix packs to WebSphere Application Server products. See “updateWizard examples for interim fixes” on page 231 for information about using the Update installer wizard to install interim fixes.

The relationship among interim fixes, cumulative fixes, and fix packs is shown in the Cumulative Fix Strategy for WebSphere Application Server V5.0 and V5.1 Web page.

Syntax for starting the wizard

The **updateWizard** command (*install_root*/update/updateWizard.sh and *install_root*\update\updateWizard.bat) launches the wizard interface to the update installer application.

install_root/update/updateWizard.sh (Linux and UNIX-based platforms)

install_root\update\updateWizard.bat (Windows platforms)

Syntax for displaying usage information

This command displays command syntax:

```
updateWizard -usage
```

Panel descriptions

Panels in the wizard let you select installable interim fixes, cumulative fixes, and fix packs, view installed interim fixes, cumulative fixes, and fix packs, and view prerequisite fixes:

- **General**
 - Welcome panel
 - Product selection panel
 - Menu panel
- **Installing and uninstalling cumulative fixes and fix packs**
 - Cumulative fix and fix pack repository specifier panel
 - Cumulative fix and fix pack selection panel
 - Cumulative fix and fix pack features selection panel
 - Pre-installing and pre-uninstalling summary panels
 - Installing and uninstalling
 - Post installing and post uninstalling summary panel

Welcome panel

Use this panel to view a welcome message that contains a brief summary of the update wizard interface, or to link to the WebSphere Application Server Support page. (This link is not available on some UNIX-based platforms.) You can also view relevant legal notices.

Product selection panel

Use this panel to select an installed WebSphere Application Server product. If the wizard cannot detect an installed product, specify the product location in the **directory input** field. After selecting a product, its directory location appears in the **input field** for verification purposes. To make corrections or enter another product location, click **Specify product location**.

Menu panel

Use this panel to install or uninstall cumulative fixes and fix packs.

Cumulative fix and fix pack repository specifier panel

Use this panel to provide the cumulative fix and fix pack repository location in a directory input field. The location should point to the directory where you unpacked downloaded fix pack JAR files. The default location for the repository is the `install_root/update/fixpacks` directory.

Cumulative fix and fix pack selection panel

Use this panel to select from a list of installable cumulative fixes and fix packs. The panel displays fix packs by ID name, with a radio button next to each for selecting a single cumulative fix or fix pack. Also displayed is the build date and current applied state (uninstalled or partially-installed) for each cumulative fix or fix pack. A cumulative fix or fix pack on this panel can be in a partially installed state. No installed cumulative fixes or fix packs appear in the list. Click **Details** for more information about a selected cumulative fix or fix pack. The window that appears contains build version information, a long description, installation prerequisites, and a list of included fixes.

About installation status

An interim fix, a cumulative fix, or a fix pack is a collection of updates to one or more product components. Depending on installed product components and on update installer program selections you make, the update installer program applies either a full or partial set of interim fix, cumulative fix, or fix pack updates to product components.

Installed status implies that the interim fix, cumulative fix, or fix pack has no more updates to product components that you can install.

Partially installed status implies that you have updated one or more product components, but the interim fix, cumulative fix, or fix pack has at least one more update that you can apply to an installed product component. (Updates to product components that you never installed do not count in the status determination.)

Uninstalled status implies that you have not updated a single product component.

Examples of partially installed states: Several scenarios can lead to a partial installation of an interim fix, a cumulative fix, or a fix pack:

- Installation fails, leaving some component updates applied and some unapplied. This is a partial installation accompanied by error messages that describe the problem.
- You skip optional component updates for the IBM HTTP Server or embedded messaging feature. This is a partial installation.
- You trigger a dynamic change in interim fix, cumulative fix, or fix pack status because:
 1. You apply all changes in an interim fix, a cumulative fix, or a fix pack that results in an installed status.
 2. You reinstall the WebSphere Application Server product to select an additional, optional feature.
 3. The interim fix, cumulative fix, or fix pack contains an update to a product component you just installed.

The status changes dynamically from installed to partially installed.

How to distinguish between a normal partially installed state and an installation failure

A partially installed state implies that some components on the system are at the level of the cumulative fix or fix pack being applied because the cumulative fix or fix pack updated them. However, there might be some scenarios where the update installer program can report a partially installed state even if no components are at the level of the cumulative fix or fix pack. A partially installed state does not imply a failed install. However, a failed install does result in a partially installed state.

To check for a failed install, open the master log file and search for errors. The update installer program outputs the name of the master log file at the start of the installation process. The file name matches this pattern: `install_root/logs/update/time-stamp_was50_fpnumber_platform_selective-install.log`

Cumulative fix or fix pack features selection panel

Use this panel to view a list of WebSphere Application Server features with optionally installable service in the selected cumulative fix or fix pack. Features that can appear include the IBM HTTP Server feature and the embedded messaging feature, which is based on IBM WebSphere MQ technology. If a feature has required service, the feature appears in the list but is grayed out. If you do not install optional service for an installed feature, the cumulative fix or fix pack installs successfully as a **partially installed fix pack**, because there is service that you did not install.

Important: Cumulative fixes and fix packs do not include service for the IBM HTTP Server powered by Apache 2.0. If you installed IBM HTTP Server powered by Apache 2.0, skip applying service to the IBM HTTP Server during the installation of a cumulative fix or a fix pack. Do not update the 2.0 level IBM HTTP Server with service for the 1.3 level IBM HTTP Server.

If you installed the IBM HTTP Server powered by Apache 1.3 product as a feature, use the update installer program to update it with service in a cumulative fix or a fix pack. You can install interim fixes from the IBM Support site for IBM HTTP Server. Otherwise, you must download an updated IBM HTTP Server product from <http://www-3.ibm.com/software/webservers/httpservers/> and install it into the same directory as your existing version to update the existing installation. You can also uninstall the current version and install the downloaded version, to avoid any issues with migration.

You must update your configuration if you reinstall. The process is described in the *Manually configuring supported Web servers* (tins_manualWebServer) topic in the base Application Server information center.

Always apply any outstanding corrective service to the stand-alone IBM WebSphere MQ product if you have it, before using the WebSphere Application Server update installer program to update the embedded messaging feature with service in an interim fix, a cumulative fix, or a fix pack. Do not select the installation of service to the embedded messaging feature if you must install corrective service to the stand-alone IBM WebSphere MQ product.

Pre-installing summary and Pre-uninstalling summary panels

Pre-installing summary panel Use this panel to display a summary of the cumulative fix or fix pack selected for installation, the WebSphere Application Server product that is the target for the cumulative fix or fix pack, and the directory where the cumulative fix or fix pack is located.

Pre-uninstalling summary panel Use this panel to display a summary of the cumulative fix or fix pack that you are uninstalling, the WebSphere Application Server product to which the cumulative fix or fix pack was applied, and the directory where the cumulative fix or fix pack is located.

Installing and Uninstalling panel

Installing action Use this panel to view the progress of installing the selected cumulative fix or fix pack. Click **cancel** to revert the installation. Once cancelled, a message confirms that the cumulative fix or fix pack is being rolled back. A similar progress panel then appears, to monitor the progress of rolling back the installation of the selected cumulative fix or fix pack.

Uninstalling action Use this panel to view the progress of uninstalling the selected cumulative fix or fix pack. There is no way to cancel the uninstall action.

You must uninstall all cumulative fix or fix pack before uninstalling the base WebSphere Application Server product, the Network Deployment product, or the WebSphere Business Integration Server Foundation product.

Post installation summary and Post uninstalling summary panels

Post installation summary panel Use this panel to view the results of the installation. Depending on the result, this panel can display a success message, a failure message, or a canceled message. When the success message appears, the installation process is complete. Click **Finish** to exit the panel. You can go back and install another cumulative fix or fix pack, which takes you to the **Menu** panel.

Post uninstalling summary panel Use this panel to view the results of the uninstall action. Depending on the result, this panel can display a success message, a failure message, or a canceled message. When the success message appears, the uninstall process is complete. Click **Finish** to exit the panel. You can go back and uninstall another cumulative fix or fix pack. Going back takes you to the **Menu** panel.

Uninstalling interim fixes, cumulative fixes, and fix packs

This topic describes the proper procedure for using the update installer application to uninstall an interim fix, a cumulative fix, or a fix pack. The update installer program is also known as the updateInstaller program or the Update installation wizard.

You cannot uninstall product updates correctly without the proper authorizations. Use the update installer program as the root user on a Linux or UNIX platform, or as the administrator on a Windows platform.

Fix packs are also known as *fixpacks*, *FixPaks* and *program temporary fixes*, or PTFs.

Removing an interim fix, a cumulative fix, or a fix pack requires setting the JAVA_HOME environment variable for the update installer. The update installer performs the task by running the **setupCmdLine** or **setupClient** command script. It is possible that the update installer cannot set the JAVA_HOME environment variable. If the update installer throws an error because it cannot set the Java environment, set the JAVA_HOME variable yourself. Then use the update installer to remove an interim fix, a cumulative fix, or a fix pack using either its wizard interface, the **updateWizard** command or its silent, command-line interface, the **updateSilent** command.

The update installer application can also install interim fixes, cumulative fixes, and fix packs.

Installation roots

The variable *install_root* represents the root directory for WebSphere Application Server. By default, this varies per product and operating system:

- Base WebSphere Application Server product:
 - AIX platforms: /usr/IBM/WebSphere/AppServer
 - Linux platforms: /opt/IBM/WebSphere/AppServer
 - HP-UX and Solaris platforms: /opt/IBM/WebSphere/AppServer
 - Windows platforms: *drive*\Program Files\WebSphere\AppServer
- Network Deployment product:
 - AIX platforms: /usr/WebSphere/DeploymentManager
 - Linux platforms: /opt/WebSphere/DeploymentManager
 - HP-UX and Solaris platforms: /opt/WebSphere/DeploymentManager
 - Windows platforms: *drive*\Program Files\WebSphere\DeploymentManager
- **5.1+** WebSphere Business Integration Server Foundation product that extends the base product:
 - AIX platforms: /usr/WebSphere/AppServer
 - Linux platforms: /opt/WebSphere/AppServer
 - HP-UX and Solaris platforms: /opt/WebSphere/AppServer
 - Windows platforms: *drive*\Program Files\WebSphere\AppServer

- **5.1+** WebSphere Business Integration Server Foundation product that extends the Network Deployment product
 - AIX platforms: `/usr/WebSphere/DeploymentManager`
 - Linux platforms: `/opt/WebSphere/DeploymentManager`
 - HP-UX and Solaris platforms: `/opt/WebSphere/DeploymentManager`
 - Windows platforms: `drive\Program Files\WebSphere\DeploymentManager`

Command name

`updateSilent.sh`, `updateSilent.bat`, `updateWixard.sh`, and `updateWizard.bat`, command-line interfaces to the `installer.jar` file.

Prerequisite environment setting

The `JAVA_HOME` environment setting must point to the IBM SDK for WebSphere Application Server products. Source the appropriate command:

- `. install_root/bin/setupCmdLine.sh` (Source the command on UNIX platforms. There is a space between the period and the installation root directory.)
- `source install_root/bin/setupCmdLine.sh` (Source the command on Linux platforms.)
- `install_root\bin\setupCmdLine.bat` (Windows platforms only)

Location of log and backup files

The update installer program records processing results in log files in the `install_root/logs/update` directory. Backup files created during the installation of fixes and fix packs are in the `install_root/properties/version/backup` directory. The files are required to uninstall an interim fix or fix pack.

Syntax and panel examples

Using the `updateWizard` interface to remove interim fixes

See “`updateWizard` examples for interim fixes” on page 231 for more information.

Using the `updateWizard` interface to remove cumulative fixes and fix packs

See “`updateWizard` examples for cumulative fixes and fix packs” on page 236 for more information.

Using the `updateSilent` interface to remove interim fixes

See “`updateSilent` examples for interim fixes” on page 219 for more information.

Using the `updateSilent` interface to remove cumulative fixes and fix packs

See “`updateSilent` examples for cumulative fixes and fix packs” on page 224 for more information.

Overview of the removal procedure

Use the update installer to uninstall a cumulative fix or a fix pack.

Use the update installer to uninstall an interim fix that you installed with the update installer.

If you have installed interim fixes for the IBM HTTP Server feature from the IBM Support site for IBM HTTP Server, or if you have installed interim fixes for the embedded messaging feature from the WebSphere MQ Service download site, the update installer program cannot uninstall interim fixes for these feature components before installing a cumulative fix or a fix pack that might include service for the features. The update installer program does uninstall interim fixes for all of the other components. If the interim fixes for the IBM HTTP Server feature and the embedded messaging feature are not uninstalled for some reason, installing a cumulative fix or a fix pack to the IBM HTTP Server feature or to the embedded messaging feature might fail, or the updated features might fail when you begin using them.

If you reinstall all of the interim fixes for either feature that are more current than the cumulative fix or the fix pack, there is no problem.

You can also choose to have the update installer skip applying cumulative fix or fix pack updates to IBM HTTP Server or embedded messaging if you do not require the updates. You can skip these updates and still apply updates to the rest of the product.

Special rules for removing fixes within a cell

One requirement governs applying an interim fix or fix pack to a cell, to ensure the continued, smooth interaction of the various WebSphere Application Server nodes:

Requirement 1: *The Network Deployment product must be at the highest fix level within the cell.*

For example, you cannot use the **addNode** command to add a V5.1 base WebSphere Application Server node to a V5.0.2 deployment manager cell.

There is no limitation on the fix level of a base Application Server V5 node within its cell, if the fix level of the base node is the same as or lower than that of the deployment manager. There is also no limit on the number of different V5.x fix levels that can coexist or interoperate within a cell, so long as the fix level for each base node is the same as or lower than that of the deployment manager. Version 5.0.x base nodes can comprise V5.1 deployment manager cells.

Viewing the fix level of the node

You can use the “versionInfo command” on page 268 in the *install_root/bin* directory to display the exact fix and version level of the product. However, do not use the **versionInfo** command while installing an interim fix or fix pack.

You can also use the silent update installer application to:

- View interim fix information. See “Viewing information about interim fixes” on page 224.
- View cumulative fix or fix pack information. See “Viewing information about cumulative fixes and fix packs” on page 230.

Updating cluster members

Refer to the following tip for information about updating cluster members:

Table 53. Installation tip

Operating platform	Tip
All platforms	Updating all cluster members to the same service level

Although the tip is about installing interim fixes and fix packs, the same principle applies to uninstalling interim fixes and fix packs. All clusters must be at the same service level.

Avoiding application errors after uninstalling an interim fix, a cumulative fix, or a fix pack

Generally speaking, if an application uses functions that are provided by a particular fix and you remove the fix, the application can throw an error. If you remove a fix, retest your applications to verify that there are no errors. Redeploy any applications that throw an error because of the missing fix.

For example, suppose that you install Fix Pack 1 on Version 5.1.0 of the base Application Server product. You then use Web services to create a stock quote service, StockQuote. The *wsdl2java* utility when run in the deployer role creates a ServiceLocator class in the emitted code that extends the AgnosticService class. The AgnosticService class is new as of V5.1.1.

If you uninstall the fix pack, the application is using a new Web services class that V5.1.0 does not support. The application throws the following error:

```
java.lang.NoClassDefFoundError:  
  Error while defining class:  
    com.ibm.ws.wsfvt.test.stockquote.StockQuoteServiceLocator  
  This error indicates that the class:  
    com.ibm.ws.webservices.multiprotocol.AgnosticService  
  could not be located while defining the class:  
    com.ibm.ws.wsfvt.test.stockquote.StockQuoteServiceLocator
```

Redeploy the application on the V5.1.0 Application Server to emit code that does not use the new V5.1.1 Web services class.

Always uninstall the highest level interim fix, cumulative fix, or fix pack before uninstalling other interim fixes or fix packs.

1. Stop all Java processes that use the IBM Software Developer Kit (SDK) that WebSphere Application Server provides. Before installing or uninstalling interim fixes, cumulative fixes, and fix packs on a machine, stop all Java processes on the machine that use the IBM SDK, Java Technology Edition that WebSphere Application Server provides.

WebSphere Application Server processes include:

- Application Server processes
- All servers, such as the IBMHttpServer process, that belong to serviceable features. Features with servers include the IBM HTTP Server and the embedded messaging feature.
- The nodeagent process on a base node when the node is federated into a deployment manager cell
- The dmgr process for the deployment manager server
- The jmsserver on a base node when the node is part of a cell and the embedded messaging server and client feature is installed

Stop all Java processes, if necessary, with the **killall -9 java** command or by using the task manager on a Windows platform. If you do install or uninstall an interim fix, a cumulative fix, or a fix pack while a WebSphere Application Server-related Java process runs, IBM does not guarantee that the product can continue to run successfully, or without error.

2. Remove the WebSphere MQ tray icon if it is present on a Windows system. On a Windows platform, remove the WebSphere MQ tray icon if it is present. The WebSphere MQ tray icon in the lower right corner indicates that a WebSphere MQ process (**amqmtbrn.exe**) is running. Right click the tray icon and click **Hide** to remove it.
3. Remove the interim fix, the cumulative fix, or the fix pack from a deployment manager node, as described in “Removing a fix from a Network Deployment node” on page 244.
4. Remove the interim fix, the cumulative fix, or the fix pack on a base node, as described in “Removing a fix from a base node.”

You can successfully remove interim fixes and fix packs from WebSphere Application Server products.

Removing a fix from a base node

This topic describes the proper procedure for using the update installer application to uninstall an interim fix, a cumulative fix, or a fix pack from an Application Server node. The update installer program is also known as the updateInstaller program or the Update installation wizard.

The update installer program removes interim fixes, cumulative fixes, and fix packs from WebSphere Application Server products, including base Application Server nodes.

5.1+ See these information center topics:

- If the base Application Server node is extended by the WebSphere Business Integration Server Foundation product, see Removing a fix from an extended node before you use this topic.
- If the base WebSphere Application Server node is within a cell, see “Removing a fix from a Network Deployment node” on page 244 before you use this topic.

This procedure describes a scenario for updating a base node by removing an interim fix, a cumulative fix, or a fix pack. See “Installing interim fixes, cumulative fixes, and fix packs” on page 202 to install an interim fix, a cumulative fix, or a fix pack.

Always uninstall the highest level interim fix, cumulative fix, or fix pack before uninstalling other interim fixes or fix packs.

1. Stop each server on the base node with the **stopServer** command.
2. Set up and configure your WebSphere Application Server environment. Set the JAVA_HOME environment variable for the update installer. If the update installer can set the Java environment, this step is unnecessary. Otherwise, this is a required step.

The location of the update, fixes, and fixpacks directories is arbitrary. You can create the directories anywhere so long as you do not use a directory name with a space in the name.

The update installer can fail to set the JAVA_HOME environment variable. If you receive a message that the update installer cannot set JAVA_HOME or if you have previously set JAVA_HOME to a Java 2 SDK that is not the one that IBM ships with WebSphere Application Server, set the environment variable yourself or source the appropriate command script from the **bin** directory of the product installation root:

- a. Open a command-line window.
 - b. Change directories to the **bin** directory of the installation root.
 - c. Run the appropriate command:
 - # ./setupCmdLine.sh (Source the command on UNIX platforms. There is a space between the period and the slash.)
 - # source setupCmdLine.sh (Source the command on Linux platforms.)
 - > setupCmdLine.bat (Windows platforms)
 - # ./setupClient.sh (Source the command for the Application Server client. There is a space between the period and the slash.)
 - # source setupClient.sh (Source the command on Linux platforms.)
 - > setupClient.bat (Windows platforms)
3. Use the appropriate command to remove the interim fix, cumulative fix, or fix pack from the base node. Depending on the interface you use to the update installer:
- Refer to “updateWizard command” on page 230 for usage information.
 - Refer to the **updateSilent** command description of the proper syntax for uninstalling the interim fix, cumulative fix, or fix pack:
 - “Uninstalling interim fixes” on page 223
 - “Uninstalling cumulative fixes and fix packs” on page 230

5.1+ For example, to uninstall the was51_fp1_win fix pack, use this **updateSilent** command:

```
C:\WebSphere\AppServer\update> updateSilent -fixpack
-installDir "C:\Program Files\WebSphere\AppServer"
-skipIHS
-fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
-uninstall
-fixpackID was51_fp1_win
```

The command is shown here on more than one line, for clarity.

4. Restart each server on the node with the **startServer** command.
5. Verify that the node is online and functioning correctly. There are several ways to verify the successful removal of an interim fix, a cumulative fix, or a fix pack:
 - Does the fix show up in the wizard panel that lists the appropriate type of installed fix?
 - Installed interim fixes. See “Uninstallable fix selection panel ” on page 235.
 - Installed cumulative fixes or fix packs. See “Installing and Uninstalling panel” on page 238.If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is installed.
 - Does the fix show up in the wizard panel that the appropriate type of installable (not yet installed) fix ?
 - Installable interim fixes. See “Installable fix selection panel ” on page 234.
 - Installable cumulative fixes or fix packs. See “Cumulative fix and fix pack selection panel” on page 237.If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is not installed.
 - Does one of the following files exist in the *install_root/properties/version/version* directory?
 - [interim_fix_ID].efix
 - [cumulative_fix_ID].ptf
 - [fix_pack_ID].ptf

Or, does one of the following files exist in the *install_root/properties/version/history* directory?

- [interim_fix_ID].efixApplied
- [interim_fix_ID].efixDriver
- [cumulative_fix_ID].ptf
- [cumulative_fix_ID].ptf
- [fix_pack_ID].ptfApplied
- [fix_pack_ID].ptfDriver

These types of files indicate the installation of a fix. See “Product information files” on page 247 for more information.

- Do the reports that describe the product version and its history show that the interim fix, the cumulative fix, or the fix pack is installed or that it has been removed?
See “Reports” on page 248 for more information about the reports.
- Do the **updateSilent** command options for viewing information about interim fixes show that the interim fix is installed or removed?
See “Viewing information about interim fixes” on page 224 for more information.
- Do the **updateSilent** command options for viewing information about cumulative fixes and fix packs show that the cumulative fix or the fix pack is installed or removed?
See “Viewing information about cumulative fixes and fix packs” on page 230 for more information.

6. Uninstall the base product manually if you cannot successfully uninstall the interim fix, cumulative fix, or fix pack.

The following platform-specific procedures for uninstalling each describe a manual process that guides you through removing every trace of the products and features you might have installed, including directories that might have changed data.

Before you begin one of the procedures, back up the *config* and *properties* directories in the installation root. Backing up the directories lets you refer to the changed data when you reinstall.

- “Manually uninstalling on AIX platforms” on page 293
- “Manually uninstalling on Linux platforms” on page 298
- “Manually uninstalling on HP-UX platforms” on page 296
- “Manually uninstalling on Solaris platforms” on page 302
- “Manually uninstalling on Windows platforms” on page 307

You can successfully remove interim fixes, cumulative fixes, and fix packs from a base WebSphere Application Server node.

Removing a fix from a Network Deployment node

This topic describes the proper procedure for using the update installer application to uninstall an interim fix, a cumulative fix, or a fix pack on a deployment manager node. The update installer program is also known as the *updateInstaller* program or the Update installation wizard.

The update installer program uninstalls interim fixes, cumulative fixes, and fix packs from WebSphere Application Server products. See *Installing interim fixes and fix packs to install an interim fix, a cumulative fix, or a fix pack*.

This procedure describes a scenario for removing an interim fix, a cumulative fix, or a fix pack from an entire Network Deployment cell, or from any part of the cell. According to the guidelines, uninstall the interim fix, cumulative fix, or fix pack from each base node in a cell before you uninstall the interim fix, cumulative fix, or fix pack from the deployment manager node.

Always uninstall the highest level interim fix, cumulative fix, or fix pack before uninstalling other interim fixes or fix packs.

1. Stop all Java processes on the machine that use the IBM SDK that WebSphere Application Server provides to support the Java 2 SDK on your operating system platform, as described in “Uninstalling interim fixes, cumulative fixes, and fix packs” on page 239.
2. Remove the highest level interim fix, cumulative fix, or fix pack from all base nodes, as described in “Removing a fix from a base node” on page 242. If you remove the interim fix, cumulative fix, or fix pack from every base node in the cell, you can also remove the interim fix, cumulative fix, or fix pack from the Network Deployment node. The interim fix level of the Network Deployment node must be equal to, or higher than the interim fix level of any base node in the cell.
3. Stop the deployment manager with the **stopManager** command.
4. Set up and configure your WebSphere Application Server environment. Set the JAVA_HOME environment variable for the update installer.

Setting the JAVA_HOME environment variable

Important: If the update installer can set the Java environment, this step is unnecessary. Otherwise, this is a required step.

The location of the update, fixes, and fixpacks directories is arbitrary. You can create the directories anywhere so long as you do not use a directory name with one or more spaces.

It is possible that the update installer cannot set the JAVA_HOME environment variable. If you receive a message that the update installer cannot set JAVA_HOME, set the environment variable yourself, or issue the appropriate command script from the **bin** directory of the product installation root:

- a. Open a command-line window.
 - b. Change directories to the bin directory of the installation root.
 - c. Run the appropriate command:
 - # . /setupCmdLine.sh (Source the command on UNIX platforms. There is a space between the period and the slash.)
 - # source setupCmdLine.sh (Source the command on Linux platforms.)
 - > setupCmdLine.bat (Windows platforms)
 - # . /setupClient.sh (Source the command for the Application Server client. There is a space between the period and the slash.)
 - # source setupClient.sh (Source the command on Linux platforms.)
 - > setupClient.bat (Windows platforms)
5. Use the appropriate command to remove the interim fix, cumulative fix, or fix pack from the base node.

Depending on the interface you use to the update installer:

- Refer to “updateWizard command” on page 230 for usage information.
- Refer to the **updateSilent** command description of the proper syntax for uninstalling the interim fix, cumulative fix, or fix pack:
 - “Uninstalling interim fixes” on page 223
 - “Uninstalling cumulative fixes and fix packs” on page 230

5.1 + For example, to uninstall the was51_nd_fp1_win fix pack, use this **updateSilent** command:

```
C:\WebSphere\AppServer\update> updateSilent -fixpack
    -installDir "C:\Program Files\WebSphere\AppServer"
    -skipIHS
    -fixpackDir "C:\WebSphere\AppServer\update\fixpacks"
    -uninstall
    -fixpackID was51_nd_fp1_win
```

The command is shown here on more than one line, for clarity.

6. Uninstall the Network Deployment product manually if you cannot successfully uninstall the interim fix, cumulative fix, or fix pack.

The following platform-specific procedures for uninstalling each describe a manual process that guides you through removing every trace of the products and features you might have installed, including directories that might have changed data.

Before you begin one of the procedures, back up the `config` and `properties` directories in the installation root. Backing up the directories lets you refer to the changed data when you reinstall.

- Manually uninstalling on AIX platforms
- Manually uninstalling on Linux platforms
- Manually uninstalling on HP-UX platforms
- Manually uninstalling on Solaris Operating Environment platforms
- Manually uninstalling on Windows platforms

7. Start the deployment manager node with the **startManager** command.

8. Verify that the deployment manager node is fully functional and that it has the interim fix, the cumulative fix, or the fix pack applied. There are several ways to verify the successful removal of an interim fix, a cumulative fix, or a fix pack:

- Does the fix show up in the wizard panel that lists the appropriate type of installed fix?
 - Installed interim fixes. See “Uninstallable fix selection panel ” on page 235.
 - Installed cumulative fixes or fix packs. See “Installing and Uninstalling panel” on page 238.If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is installed.
- Does the fix show up in the wizard panel that the appropriate type of installable (not yet installed) fix ?
 - Installable interim fixes. See “Installable fix selection panel ” on page 234.
 - Installable cumulative fixes or fix packs. See “Cumulative fix and fix pack selection panel” on page 237.

If you see the interim fix, the cumulative fix, or the fix pack in one of the panels, the fix is not installed.

- Does one of the following files exist in the `install_root/properties/version/version` directory?
 - `[interim_fix_ID].efix`
 - `[cumulative_fix_ID].ptf`
 - `[fix_pack_ID].ptf`

Or, does one of the following files exist in the `install_root/properties/version/history` directory?

- `[interim_fix_ID].efixApplied`
- `[interim_fix_ID].efixDriver`
- `[cumulative_fix_ID].ptf`
- `[cumulative_fix_ID].ptf`
- `[fix_pack_ID].ptfApplied`
- `[fix_pack_ID].ptfDriver`

These types of files indicate the installation of a fix. See “Product information files” on page 247 for more information.

- Do the reports that describe the product version and its history show that the interim fix, the cumulative fix, or the fix pack is installed or that it has been removed?
See “Reports” on page 248 for more information about the reports.
- Do the **updateSilent** command options for viewing information about interim fixes show that the interim fix is installed or removed?
See “Viewing information about interim fixes” on page 224 for more information.
- Do the **updateSilent** command options for viewing information about cumulative fixes and fix packs show that the cumulative fix or the fix pack is installed or removed?
See “Viewing information about cumulative fixes and fix packs” on page 230 for more information.

You can also run the installation verification tool on the node to verify that the node is operational. See “Using the installation verification test” on page 184 for more information.

9. Specify that file sets on one node match those on the deployment manager node.

Ensure consistent configuration data across a cell. You can synchronize files on individual nodes or throughout your system. To synchronize files throughout the system, use the deployment manager administrative console page, **System administration > Nodes > *check_each_node_name* > Full Resynchronization**. You can use the administrative console page, **System Administration > Node Agents > nodeagent > File Synchronization Service**, to specify automatic synchronization every minute until all base node servers are brought online.

10. Verify that all nodes are online and that the cell is functioning correctly.
11. Restore your original file synchronization settings. The cell is now fully functional. All operations are available and functioning normally.

You can successfully remove interim fixes, cumulative fixes, and fix packs from a deployment manager node.

Product version and history information

The WebSphere Application Server product contains structural differences from previous versions. The `/properties/version` directory in the `install_root` contains important data about the product and its installed components, such as the build version and build date. This information is included in `[product].product` and `[component].component` files. The `/properties/version/history` directory in the `install_root` contains a collection of records for installed interim fixes and fix packs. This information is included in `[interim fixID].efixApplied`, `[interim fixID].efixDriver`, `[fix packID].ptfApplied`, and `[fix packID].ptfDriver` files. A driver file has useful information about the entire contents of an interim fix or fix pack. The applied file has relevant information about the interim fixes or fix packs that are currently applied. Event.history files are also present. They contain a detailed log about updates you have applied, either successfully or unsuccessfully. Time-stamped, detailed logs record each update process in the `/properties/version/log` directory of the `install_root`.

This topic describes the XML data files that store product information for Version 5 WebSphere Application Server products. By default, the document type declarations (DTDs) for these files are in the `properties/version/dtd` folder of the `install_root`, or the server root directory. See the Storage locations section for more information.

This topic includes these sections:

- A list of product information files and file locations
- A list of reports for displaying version and history information
- A description of logs and component backups
- A list of properties you can use to change storage locations
- A description of how the update service makes operational use of the product information
- A data dictionary that describes data in the files

Product information files

XML files in the in the `/properties/version` directory that store version information: `platform.websphere`

One file whose existence indicates that a WebSphere Application Server product is installed. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE websphere PUBLIC "websphereId" "websphere.dtd">
<websphere name="IBM WebSphere Application Server" version="5.0"/>
```

The following XML files in the `/properties/version` directory represent installed items and installation events.

product-id.**product**

One file whose existence indicates the particular WebSphere Application Server product that is installed. Data in the file indicates the version, build date, and build level. For example, the file might be the *ND*.product file, which indicates that the installed product is WebSphere Application Server Network Deployment. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE product PUBLIC "productId" "product.dtd">
<product name="IBM WebSphere Application Server for Network Deployment">
  <id>ND</id>
  <version>5.0.0</version>
  <build-info date="10/5/02" level="s0239.28"/>
</product>
```

component-name.**component**

Any number of component files that each indicate the presence of an installed component, which is part of the product. Data in the file indicates the component build date, build version, component name, and product version. For example, the file might be the *activity*.component file, which indicates that the **activity** component is installed. The activity component is part of the Network Deployment product. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE component PUBLIC "componentId" "component.dtd">
<component build-date="10/5/02" build-version="s0239.28"
  name="activity" spec-version="5.0"/>
```

extension.id.**extension**

Any number of extension files that each indicate the presence of an extension that you install as a user extension, as part of a service engagement, or as installed by a third party product. The *<extension.id>*.extension files are not created, logged, or removed by WebSphere Application Server products.

fix-id.**efix**

Any number of interim fix files that each indicate the presence of an installed fix.

ptf-id.**ptf**

Any number of files, that each indicate the presence of an installed fix pack.

XML files in the /properties/version/history directory that store version history information files: event.history

One file that lists update events that have occurred. An update event is an operation that installs or uninstalls an interim fix or a fix pack. The file is sorted by the date and time of the events that are listed.

The following XML files in the /properties/version/history directory describe fixes and fix packs that are currently installed. These XML files are related to installation items by the primary ID information, which is shown in the following examples as italicized text.

fix-id.**efixDriver**

Interim fix-driver defining information

fix-id.**efixApplied**

Interim fix installation details

ptf-id.**ptfDriver**

Fix pack-driver defining information

ptf-id.**ptfApplied**

Fix pack installation details

Reports

You can view product information by examining files in the *properties/version* directory, including the *properties/version/history* directory.

WebSphere Application Server provides the ability to generate two types of reports about the data in the files, **Version** reports and **History** reports. The following report-generation scripts are available in the `install_root/bin` directory.

Product version reports

The following report generation scripts extract data from XML data files in the `properties/version` folder:

- `versionInfo` script
Lets you use parameters to create a version report on Linux and UNIX-based platforms, or on Windows platforms.
- `genVersionReport` script
Generates the `versionReport.html` report file in the `bin` directory on Linux and UNIX-based platforms, or on Windows platforms. The report includes the list of components, fixes, and fix packs.

Product history reports

The following report generation scripts extract data from XML data files in the `properties/version/history` folder.

historyInfo.bat

Lets you use parameters to create a history report of installed and uninstalled fixes and fix packs, on Windows platforms. You can also specify a component name to create a report that shows the history for that component.

Parameters:

-format

`text | html`

Selects the format of the report. The default is TEXT.

-file *fileName*

Specifies the output file name. The report goes to standard output (stdout) by default.

-updateID *ID*

Specifies the ID of an interim fix or fix pack update. When specified, the product history report displays events for only the specified update. When not specified, the report displays events for all updates.

-component *componentName*

Specifies the name of a component. When specified, the product history report displays events for only the named component. When not specified, the report displays events for all components.

historyInfo.sh

Lets you use parameters to create a history report on UNIX-based platforms. Parameters are the same as for the Windows version.

genHistoryReport.bat

Generates the `historyReport.html` report file in the `bin` directory on Windows platforms. The report includes all updates and components.

genHistoryReport.sh

Generates the `historyReport.html` report file in the `bin` directory on UNIX-based platforms. The report includes all updates and components.

Logs and component backups

WebSphere Application Server products use two other directories when performing update operations, for logging and backups:

`install_root/properties/version/log`

Product updates log directory prior to Fix Pack 2.

WebSphere Application Server products store log files to document component, interim fix and fix pack operations and updates.

install_root/logs/update

Product updates log directory beginning with Fix Pack 2.

Beginning with the version of the updateInstaller program that is bundled with Fix Pack 2, log files are in the *install_root/logs/update* directory.

install_root/properties/version/backup

Product updates backup directory

WebSphere Application Server products back up components before applying interim fixes and fix packs. If you uninstall an interim fix or fix pack, WebSphere Application Server products restore the backed-up component JAR file.

File naming convention

Time stamp

YYYYMMDD_HHMMSS

For example: 20020924_211832 is 24-Sep-2002, 9:18:32 pm, GMT. All time stamps are in GMT.

ID

Interim fix ID or fix pack ID

For example: apar6789c is an interim fix ID; PTF_1 is a fix pack ID.

Operation

install | uninstall

Interim fix log file names

timeStamp_fixId_operation.log

For example, prior to Fix Pack

2: *properties/version/log/20020924_211832_apar6789c_install.log* and
properties/version/log/20020924_211912_apar6789c_uninstall.log

At Fix Pack 2 or later, the updateInstaller creates these logs:

install_root/logs/update/20020924_211832_apar6789c_install.log and
install_root/logs/update/20020924_211912_apar6789c_uninstall.log

Interim fix component log file names

timeStamp_fixId_componentName_operation.log

For example, prior to Fix Pack 2:

properties/version/log/20020924_211832_apar6789c_ras_install.log and
properties/version/log/20020924_211912_apar6789c_ras_uninstall.log

At Fix Pack 2 or later, the updateInstaller creates these logs:

install_root/logs/update/20020924_211832_apar6789c_ras_install.log and
install_root/logs/update/20020924_211912_apar6789c_ras_uninstall.log

Fix pack log file names

timeStamp_ptfId_operation.log

For example, prior to Fix Pack 2:

properties/version/log/20030324_211832_was50_fp1_install.log and
properties/version/log/20030325_211912_was50_fp1_uninstall.log

At Fix Pack 2 or later, the updateInstaller creates these logs:

install_root/logs/update/20030324_211832_was50_fp2_install.log and
install_root/logs/update/20030325_211912_was50_fp2_uninstall.log

Fix pack component log file names

timeStamp_ptfId_componentName_operation.log

For example, prior to Fix Pack 2:

properties/version/log/20030324_211832_was50_fp1_ras_install.log and
properties/version/log/20030325_211912_was50_fp1_ras_uninstall.log

At Fix Pack 2 or later, the updateInstaller creates these logs:
install_root/logs/update/20030324_211832_was50_fp2_ras_install.log and
install_root/logs/update/20030325_211912_was50_fp2_ras_uninstall.log

Backup JAR file names

timeStamp_ptfId_componentName_undo.jar or *timeStamp_fixId_componentName_undo.jar*

For example: 20020924_211832_apar6789c_ras_undo.jar Do not delete a backup JAR file. You cannot remove a component update if the corresponding backup JAR file is not present.

Update processing might also use a temporary directory, if necessary. A Java property specifies this directory as described in the next section.

Storage locations

Product information files are located relative to the WebSphere Application Server product install_root, or the server root directory.

Default file paths are:

Version directory

install_root/properties/version or *server_root/properties/version*

History directory

install_root/properties/version/history

Updates log directory

Prior to Fix Pack 2: *install_root/properties/version/log*

The version of the updateInstaller that is bundled with Fix Pack 2 and later fix packs, stores log files in the *install_root/logs/update* directory.

Updates backup directory

install_root/properties/version/backup

DTD directory

install_root/properties/version/dtd

Temporary directory

Specified by the java.io.tmpdir Java system property

Operational description

WebSphere Application Server products update the product version history information while performing events that install or uninstall fixes or fix packs. Events that might occur include:

- A WebSphere Application Server product adds an interim fix file (with an extension of .efix) to the version directory to indicate that an interim fix is currently installed.
- A WebSphere Application Server product removes an interim fix file from the version directory when it uninstalls the corresponding fix.
- A WebSphere Application Server product adds an interim fix driver file (with an extension of .efixDriver) to the history directory when an interim fix is installed. An interim fix driver file contains defining information for a fix.
- A WebSphere Application Server product removes an interim fix driver file when it removes the corresponding fix.
- A WebSphere Application Server product adds an interim fix application file (with an extension of .efixApplied) to the history directory when it installs an interim fix. An interim fix application file contains information that identifies component updates that have been applied for a fix. The application file also provides links to component log and backup files.
- A WebSphere Application Server product removes an interim fix application file when it removes the corresponding fix.
- A WebSphere Application Server product adds a fix pack, with an extension of .ptf, to the version directory to indicate that a fix pack is currently installed.
- A WebSphere Application Server product removes a fix pack file from the version directory when it uninstalls the corresponding fix pack.

- A WebSphere Application Server product adds a fix pack driver file (with an extension of .ptfDriver) to the history directory when it installs a fix pack. A fix pack driver file contains defining information for a fix pack.
- A WebSphere Application Server product adds a fix pack application file (with an extension of .ptfApplied) to the history directory when it installs a fix pack. A fix pack application file contains information that identifies component updates that have been applied for a fix pack. The application file also provides links to component log and backup files.
- A WebSphere Application Server product makes entries in the history file, event.history, when it installs or uninstalls an update.
- A WebSphere Application Server product stores a parent event for each interim fix that it installs or uninstalls.
- A WebSphere Application Server product stores a parent event for each fix pack that it installs or uninstalls.
- A WebSphere Application Server product stores child component events for each component update that it installs or uninstalls, beneath the corresponding interim fix or fix pack parent event.
- A WebSphere Application Server product stores one log file in the logs/update directory as it installs or uninstalls one interim fix or fix pack.
- A WebSphere Application Server product stores one log file in the logs/update directory as it installs or uninstalls an interim fix or fix pack, in response to each component update that occurs.
- A WebSphere Application Server product stores a component backup file in the backup directory for each component update that it installs.
- A WebSphere Application Server product removes a component backup file from the backup directory for each component update that it uninstalls.

Data dictionary

Type Family: websphere product family

File Types:

websphere

File Type:

websphere

Elements:

name	string	required
version	string	required

Persistence:

versionDir/platform.websphere

Type Detail:

The websphere file denotes the presence of WebSphere family products.

Element Detail:

websphere.name	The WebSphere product family name.
websphere.version	The WebSphere product family version.

Type Family: product

File Types: product
component
extension

File Type: product

Persistence: *versionDir/id*.product

Elements:	id	string	required
	name	string	required
	version	string	required
	build-info	complex	required

Type Detail:

A product file is placed to denote the presence of a specific WebSphere family product.

The product ID is embedded in the product file name.

Element Detail:

product.id	The id of the product.
product.name	The name of the product.
product.version	The version of the product.
product.build-info	An element containing build information for the product.

Element Type: build-info

Elements:	date	date	required
	level	string	required

Type Detail:

A build-info instance details the build of a specific installed WebSphere family product.

Element Detail:

build-info.date	The date on which the product was build.
build-info.level	The level code of the product's build.

File Type: component

Persistence: *versionDir/name.component*

Elements:	name	string	required
	spec-version	string	required
	build-version	string	required
	build-date	date	required

File Detail:

A component file denotes the presence of a specific component. The component name is embedded in the component file name.

Element Detail:

component.name	The name of the component.
component.spec-version	The specification version of the component.
component.build-version	The build level of the component.
component.build-date	The build date of the component.

File Type: extension

Persistence: *versionDir/id.extension*

Elements:	id	string	required
	name	string	required

File Detail:

An extension file denotes the presence of a specific extension. The extension's id is embedded in the extension file name.

The elements of an extension file are minimally specified. The listed elements are required. Additional elements may be present as determined by the actual installed extension.

Element Detail:

extension.id The id of the extension.
 extension.name The name of the extension.

Type Family: update

File Types: efix
 ptf
 efix-applied
 ptf-applied

File Type: efix

Persistence: *versionDir/id.efix*

Elements:	id	string	required
	apar-number	string	optional
	pmr-number	string	optional
	short-description	string	required
	long-description	string	required
	is-temporary	boolean	required
	build-version	string	required
	build-date	date	required
	component-update	complex	min=1, max=unbounded
	platform-prereq	complex	min=0, max=unbounded
	product-prereq	complex	min=0, max=unbounded
	efix-prereq	complex	min=0, max=unbounded
	custom-property	complex	min=0, max=unbounded

Type Detail:

An efix file denotes the presence of some portion of a specific fix. The ID of the interim fix is embedded in the file name.

An efix file contains all interim fix data, such as description, a listing of component updates, and prerequisite information.

Almost always, when installing a fix, all of the potential component updates within the interim fix are required to be installed.

A separate application file must be examined to determine the components which have been updated for a particular fix.

A list of custom properties may be provided. These are provided for future use.

Element Detail:

efix.id The id of the fix.

efix.short-description A short description of the fix.

efix.long-description A long description of the fix.

efix.is-trial A flag indicating whether or not an interim fix is considered a trial fix. Generally, a trial interim fix will be followed up with a more permanent fix.

efix.expiration-date A date on which the interim fix is obsolete.

efix.build-version The build version of the fix. This is distinct from

	the build version of component updates contained within the fix.
efix-build-date	The build date of the fix. This is distinct from the build version of the component updates contained within the fix.
efix.apar-info	A list of APAR's which are associated with the fix.
efix.component-update	A list of updates for components. For a fix, these are usually all required, and are all patch updates. At least one component update must be present.
efix.efix-prereq	A list of prerequisite fixes for the fix. Note that prerequisite fixes may be negative (see below). This list may be (and is often) empty.
efix.plaform-prereq	A list of platforms on which the interim fix may be installed. This list may be empty, in which case the interim fix may be installed on all platforms.
efix.product-prereq	A list of products on which the interim fix may be installed. This list may be empty, in which case the interim fix may be installed on all products.
efix.custom-property	A list of properties, provided for future use.

File Type: ptf

Persistence: *versionDir/id.ptf*

Elements:	id	string	required
	short-description	string	required
	long-description	string	required
	build-version	string	required
	build-date	date	required
	component-update	complex	min=1, max=unbounded
	product-update	complex	min=0, max=unbounded
	platform-prereq	complex	min=0, max=unbounded
	product-prereq	complex	min=0, max=unbounded
	included-efix	complex	min=0, max=unbounded
	custom-property	complex	min=0, max=unbounded

Type Detail:

A ptf file denotes the presence of some portion of a specific fix pack. The id of the fix pack is embedded in the fix pack file name.

A ptf file contains all fix pack data, such as description, a listing of component updates, and prerequisite information.

Usually, when installing a fix pack, you can omit certain potential component updates, but only when the corresponding component is not installed.

You must examine a separate application file, to determine which components a particular fix pack has updated.

A fix pack can include updates for a number of fixes.

A list of custom properties might be provided. These are for future use.

Element Detail:

ptf.id	The ID of the fix pack.
ptf.short-description	A short description of the fix pack.

ptf.long-description	A long description of the fix pack.
ptf.build-version	The build version of the fix pack. This is distinct from the build version of component updates contained within the fix pack.
ptf-build-date	The build date of the fix pack. This is distinct from the build version of the component updates contained within the fix pack.
ptf.component-update	A list of updates for components. For a fix pack, these are usually all required, and are all patch updates. At least one component update must be present.
ptf.plaform-prereq	A list of platforms on which you can install the fix pack. This list might be empty. If so, you can install the fix pack on all platforms.
ptf.product-prereq	A list of products on which you can install the fix pack. This list might be empty. If so, you can install the fix pack on all products.
ptf.included-efix	A list of fixes which are included (fixed) by the fix pack.
ptf.custom-proppty	A list of properties, provided for future use.

Element Type: component-update

Elements:	component-name	string	required
	update-type	enum	required [enumUpdateType]
	is-required	boolean	required
	is-recommended	boolean	required
	is-optional	boolean	required
	is-external	boolean	required
	root-property-file	anyURL	optional
	root-property-name	string	optional
	root-property-value	anyURL	optional
	is-custom	boolean	required
	primary-content	string	required
	component-prereq	complex	min=0, max=unbounded
	final-version	complex	optional
	custom-property	complex	min=0, max=unbounded

Type Detail:

A component update represents a potential component update which is packaged in an update (an interim fix or a fix pack).

An component update may be required, in which case the parent update may not be installed unless the component update can be installed. (A component update can be installed if the corresponding component is installed.)

A component update may be a custom update, in which case the content which was provided must be an executable file. Otherwise, the content which is provided must be an update JAR file.

A component update has a type. A final version may be required according to the update type.

Element Detail:

component-update.component-name	The name the component which is to be updated.
component-update.update-type	The type of the component update, one of 'add', 'replace', 'remove', or 'patch'. Final version

information must be provided when the update type is 'add' or 'replace'.

component-update.is-required	A flag which, when true, specifies that the parent update may not be applied unless this component update is applied.
component-update.is-recommended	A flag which, when true, specifies that this component update, although optional, should be installed.
component-update.is-optional	A flag which, when true, specifies that this update may be omitted even if its corresponding component is installed.
component-update.is-external	A flag which, when true, specifies that this component update may live outside of the usual install root.
component-update.root-property-file	For a component with an external root, this properties file provides the root value.
component-update.root-property-name	For a component with an external root, this named property provides the root value.
component-update.root-property-value	For a component with an external root, this value provides the default root value.
component-update.is-custom	A flag which, when true, specifies that the update is a custom update. When true, the content must be an executable program. When false, the content must be an update JAR.
component-update.primary-content	The name of the content which is provided for the update. This will be an entry which is packaged in the 'components' directory of the update.
component-update.component-prereq	A list of component versions, one of which must be present for this update to be installed. When this list is empty, any component version is allowed.
component-update.final-version	Final version information for the component. A final version is required when the update operation is 'add' or 'replace'.
component-update.custom-property	A list of properties, provided for future use.

Element Type: apar-info

Elements:	number	string	required
	date	date	required
	short-description	string	required
	long-description	string	optional

Type Detail:

An apar-info object provides information about an APAR which is associated

with a fix, usually indicating that the interim fix provides an interim fix for the APAR.

Element Detail:

apar-info.number The number of the associated APAR.
apar-info.date The date of the APAR.
apar-info.short-description A short description of the APAR.
apar-info.long-description An optional long description of the APAR.

Element Type: efix-prereq

Elements:	efix-id	string	required
	is-negative	boolean	required
	install-index	int	optional

Type Detail:

An efix-prereq instance denotes an interim fix that must be present (or, if negative, must be absent) for the parent interim fix to be installed.

efix-prereq instances can specify a cycle, in which case the prerequisite specification is treated as a corequisite specification.

The following chart summarizes the interpretation of prerequisite information for two fixes:

fix1	fix2	
-	-	Install the fixes without regard to each other.
fix2	-	fix1 must be installed after fix2 is installed.
-	fix1	fix2 must be installed after fix1 is installed.
fix2	fix1	fix1 and fix2 must be installed together.
!fix2	-	fix1 may not be installed after fix2 is installed.
-	!fix1	fix2 may not be installed after fix1 is installed.
!fix2	!fix1	fix1 and fix2 may not ever be installed together.
!fix2	fix1	This is an erroneous specification.
fix2	!fix1	This is an erroneous specification.

The installation index element provides ordering information for corequisite fixes that must be installed in a particular order.

Element Detail:

fix-prereq.efix-id The id of the prerequisite fix.
fix-prereq.is-negative A flag which indicates if the prerequisite interim fix is required or prohibited. If false, you must install the interim fix before installing the parent fix. If true, you must not install the interim fix before you install the parent fix.
fix-prereq.install-index An optional index number which is used to order corequisite fixes.

Element Type: product-update

Elements:	product-id	string	required
	product-name	string	required
	build-version	string	required
	build-date	date	required
	build-level	string	required

Type Detail:

A product update specifies a replacement to a product file.

The product update information matches the information in product files.

Multiple product updates may be present, in which case each matching product is updated.

Element Detail:

product-update.product-id The id of the product that is updated.

product-update.product-name The name of the product.

product-update.build-version The build version of the product.

product-update.build-date The build date of the product.

product-update.build-level The build level of the product.

Element Type: component-prereq

Elements:	component-name	string	required
	spec-version	string	required
	build-version	string	required
	build-date	date	required

Element Type: platform-prereq

Elements:	architecture	string	required
	os-platform	string	optional
	os-version	string	optional

Type Detail:

A platform prerequisite instance denotes a platform which must be present for an update to be installed. The element values are according to the values supplied for the matching java properties.

Note that when multiple platform prerequisites are specified, these prerequisites have an OR relationship: At least one of the platform prerequisites must be satisfied.

Element Detail:

platform-prereq.architecture The name of an architecture which must be present.

platform-prereq.os-platform The name of an operating system which must be present. This element is optional. When absent, the architecture is checked, but the os-platform and os-version are not.

platform-prereq.os-version The version of a the operating system which must be present. This element is optional. When absent, the architecture and os-platform are checked, but os-version is not. (When os-platform is absent, os-version should not be set.)

Element Type: product-prereq

Elements:	product-id	string	required
	build-version	string	optional
	build-date	date	optional
	build-level	string	optional

Type Detail:

A product prerequisite specifies that a particular product must be present for an update to be installed.

Note that when multiple product prerequisites are specified, these prerequisites have an OR relationship: At least one of the product prerequisites must be satisfied.

Note that all of the elements are required. When multiple products having the same id are supported by an update, multiple product prerequisites must be specified.

Element Detail:

product-prereq.product-id The id of the product which must be present.
product-prereq.build-version The version of the product which must be present.
product-prereq.build-date The build date of the product which must be present.
product-prereq.build-level The level date of the product which must be present.

Element Type: component-prereq

Elements:	component-name	string	required
	spec-version	string	required
	build-version	string	required
	build-date	date	required

Type Detail:

A version prerequisite specifies that a particular component version must be present for an update to be installed.

Note that when multiple version prerequisites are specified, these prerequisites have an OR relationship: At least one of the version prerequisites must be satisfied.

Element Detail:

version-prereq.component-name The name of the component which must be present.
version-prereq.spec-version The specification version of the component which must be present.
version-prereq.build-version The version of the component which must be present.
version-prereq.build-date The build date of the component which must be present.

Element Type: included-efix

Elements:	efix-id	string	required
-----------	---------	--------	----------

Type Detail:

An included-efix identifies an interim fix by ID and indicates that the fix is included in the fix pack.

Element Detail:

included-efix.efix-id The ID of the interim fix that the fix pack includes.

Element Type: custom-property

Elements:	property-name	string	required
-----------	---------------	--------	----------

property-type	string	optional
property-value	string	optional

Type Detail:

A custom property encodes a key-value pair, with an optional type element. Custom properties are provided for future use.

Element Detail:

custom-property.property-name	The name of the custom property.
custom-property.property-type	An optional type of the custom property. The semantics of this type are defined by user of the property value.
custom-property.property-value	The value of the custom property.

File Type: efix-applied

Persistence: *versionDir/id.efixApplied*

Elements:	efix-id	string	required
	component-applied	complex	min=0, max=unbounded

Type Detail:

An efix-applied collection specifies what components have been updated for the interim fix as specified by the efix-id.

Element Detail:

efix-applied.efix-id	The id of the interim fix for which applies are recorded.
efix-applied.component-applied	The list of recorded applications.

File Type: ptf-applied

Persistence: *versionDir/id.ptfApplied*

Elements:	ptf-id	string	required
	component-applied	complex	min=0, max=unbounded

Type Detail:

A ptf-applied collection specified what components have been updated for the fix pack as specified by the fix pack ID.

Element Detail:

ptf-applied.efix-id	The ID of the fix pack for which applies are recorded.
ptf-applied.component-applied	The list of recorded applications.

Element Type: component-applied

Elements:	component-name	string	required
	update-type	enum	required [enumUpdateType]
	is-required	boolean	required
	is-optional	boolean	required
	is-external	boolean	required
	root-property-file	anyURL	optional
	root-property-name	string	optional
	root-property-value	string	optional
	is-custom	boolean	required

log-name	anyURL	required
backup-name	anyURL	required
time-stamp	date	required
initial-version	complex	optional
final-version	complex	optional

Type Detail:

An applied instance is present to indicate the application of an update for a particular interim fix or fix pack to a particular component. (The particular interim fix or fix pack is as specified by the applied's parent.) An applied provides sufficient information to undo itself.

The elements of an applied are copies of values from update events.

Element Detail:

component-applied.component-name	The name of the component which was updated.
component-applied.update-type	The type of the component update.
component-applied.is-required	A true value specifies that the parent update requires this component update.
component-applied.is-optional	A flag which, when true, specifies that the parent update does not require this component update, even if the component is installed.
component-applied.is-external	A flag which, when true, specifies that this component update was applied to a location different than the usual <code>install_root</code> .
component-applied.root-property-file	For an update against a component having an external root, this properties file provides the root value.
component-applied.root-property-name	For an update against a component having an external root, this named property provides the root value.
component-applied.root-property-value	For an update against a component having an external root, this is a record of the actual external root.
component-applied.is-custom	A flag which, when true, specifies that the application was a custom update. When true, an executable program was applied. When false, the contents of an update JAR were applied.
component-applied.log-name	The name of the log file that was generated by this application.
component-applied.backup-name	The name of the backup file which was generated by this application.
component-applied.time-stamp	The time of this application (the ending time of the corresponding update event).
component-applied.initial-version	The version of the component before the application. This version will be null if the application was an add.
component-applied.final-version	The version of the component after application. This will be null if the update was a removal.

Element Type: initial-version

Elements:	component-name	string	required
	spec-version	string	required
	build-version	string	required
	build-date	string	required

Type Detail:

A initial-version instance is used to describe a component level as the initial version of a component.

Element Detail:

initial-version.component-name The name of the component.

initial-version.spec-version The new specification version for the component following the update.

initial-version.build-version The new build version for the component.

initial-version.build-date The new build date for the component.

Element Type: final-version

Elements:	component-name	string	required
	spec-version	string	required
	build-version	string	required
	build-date	string	required

Type Detail:

A final-version instance is used to supply a component level for a component which has been added or replaced.

Element Detail:

final-version.component-name The name of the new component.

final-version.spec-version The new specification version for the component following the update.

final-version.build-version The new build version for the component.

final-version.build-date The new build date for the component.

Enum Type: enumUpdateType

Values:	0 add
	1 replace
	2 remove
	3 patch

Type Detail:

An update type instance specifies the type of an update. An 'add' update adds a component into an installation. A 'replace' update replaces a particular version of a component with a different version of that component. A 'remove' update removes a component. A 'patch' update performs a limited update to a component, in particular, without changing the version of the component.

When adding a component, that component may not already be present.
When replacing or removing a component, that component must be present.
When patching a component, that component must be present.

When replacing or removing a component, or when patching a component, usually, at least one version prerequisite will be specified for the component update.

Value Detail:

enumUpdateType.add Specifies that an update adds a component.
enumUpdateType.replace Specifies that an update replaces a component.
enumUpdateType.remove Specifies that an update removes a component.
enumUpdateType.patch Specifies that an update modifies a component, but does not change its version.

Type Family: history

File Type: event-history

Persistence: *historyDir/event.history*

Elements: update-event complex min=0, max=unbounded

Type Detail:

One event history is provided for a websphere product family installation. This event history contains history of update events, corresponding with the actual update events for that product family.

Element Detail:

event-history.update-event The list of update events for the websphere product family. The top level events are fix and fix pack events, each containing one or more component events.

Element Type: update-event

Elements:	event-type	enum	required	[enumEventType]
	parent-id	string	required	
	id	string	required	
	update-type	enum	required	[enumUpdateType]
	is-required	boolean	required	
	is-optional	boolean	required	
	is-external	boolean	required	
	root-property-file	anyURL	optional	
	root-property-name	string	optional	
	root-property-value	string	optional	
	is-custom	boolean	required	
	primary-content	anyURI	required	
	event-action	enum	required	[enumEventAction]
	log-name	anyURI	required	
	backup-name	anyURI	required	
	start-time-stamp	dateTime	required	
	end-time-stamp	dateTime	optional	
	status	enum	optional	[enumEventResult]
	status-message	string	optional	
	initial-version	complex	optional	
	final-version	complex	optional	
	update-event	complex	optional	

Type Detail:

An update event denotes a single update action, applying to either a fix, a fix pack, or to a component, according to the set event type.

Interim fix (efix) and fix pack (ptf) type events each have a collection of component events.

Currently, component events have no child events.

Element Detail:

update-event.event-type	The type of this event, either an interim fix or fix pack (ptf) type event, or a component type event.
update-event.parent-id	This element is present only for component events. The ID of the parent interim fix or fix pack of this event.
update-event.id	The ID of the fix, fix pack, or component that was updated, interpreted according to the type of the event.
update-event.update-type	The type of update for component events.
update-event.is-required	A flag which, when true, specifies that this component update is required.
update-event.is-optional	A flag which, when true, specifies that this component update is optional, even if the component is installed.
update-event.is-external	A flag which, when true, specifies that this update used an external root.
update-event.root-property-file	For an update of an external component, this properties file contains the external root value.
update-event.root-property-name	For an update of an external component, the property having this name specifies the external root value.
update-event.root-property-value	For an update of an external component, the root value.
update-event.is-custom	A flag that, when true, specifies that the application was a custom update. When true, an executable program was applied. When false, the contents of an update JAR file were applied.
update-event.primary-content	The URL of the primary content for the update.
update-event.event-action	The type of action for this event.
update-event.log-name	The name of the log file that was generated for this event.
update-event.backup-name	The name of the backup file that was generated for this event.
update-event.start-time-stamp	The XML timestamp of the starting time of the event. This timestamp follows the XML timestamp format, meaning that time zone information is included.
update-event.end-time-stamp	The XML timestamp of the ending time of the

event. This timestamp follows the XML timestamp format, meaning that time zone information is included. When absent, the update operation corresponding to the parent event failed with a non-recoverable exception.

update-event.status The result of the update.

update-event.status-message Message text provided in addition to the basic status code. Exception text is provided through the status-message when an update fails.

update-event.initial-version This element is not used unless the update is a component type update. The initial version of the component which was updated. This element is absent when the update is an add type update.

update-event.final-version This element is not used unless the update is a component type update. The final version of the component which was updated. This element is absent when the update is a remove type update.

update-event.update-event A collection of child events. This collection is used for interim fix and fix pack type events. This collection is empty for component type events.

Element Type: initial-version

Elements:	spec-version	string	required
	build-version	string	required
	build-date	string	required

Type Detail:

A initial-version instance is used to describe a component level as the initial version of a component.

Element Detail:

initial-version.spec-version The new specification version for the component following the update.

initial-version.build-version The new build version for the component.

initial-version.build-date The new build date for the component.

Element Type: final-version

Elements:	spec-version	string	required
	build-version	string	required
	build-date	string	required

Type Detail:

A final-version instance is used to supply a component level for a component which has been added or replaced.

Element Detail:

final-version.spec-version The new specification version for the component following the update.

final-version.build-version The new build version for the component.

final-version.build-date The new build date for the component.

Enum Type enumEventType

Values: 0 Interim fix (efix)
 1 Fix pack (ptf)
 2 Component

Type Detail:

An event type instance specifies the type of an update event, which is either an interim fix (efix) event, a fix pack (ptf) event or a component event. The interpretation of particular event elements depends on the set event type.

Value Detail:

enumEventType.efix Specifies that an event is for an interim fix update.

enumEventType.ptf Specifies that an event is for a fix pack update.

enumEventType.component Specifies that an event is for a component update.

Enum Type: enumEventAction

Values: 0 Install
 1 Uninstall
 2 Selective install
 3 Selective uninstall

Type Detail:

An event action instance specified the operation performed by an update, which can be an install or uninstall operation, and which may be a selective operation. Component operations are always either install or uninstall type operations, only interim fix and fix pack operations may be selective operations.

A selective operation is an installation which is applied to a preset list of components. In particular, potential component updates may be skipped, and component updates which were already applied may be reapplied.

A selective uninstall operation is used to back out an update which was cancelled by the user.

Value Detail:

enumEventAction.install Specifies that an event is an install operation.

enumEventAction.uninstall Specifies that an event is an uninstall operation.

enumEventAction.selective-install Specifies that an event is an install operation with a preset list of components that are updated.

enumEventAction.selective-uninstall Specifies that an event is an uninstall operation with a preset list of components that are updated.

Enum Type: enumUpdateType

Values: 0 Add
 1 Replace
 2 Remove
 3 Patch

Type Detail:

An update type instance specifies the type of a component update. An 'add' update adds a component into an installation.

A 'replace' update replaces a particular version of a component with a different version of that component.

A 'remove' update removes a component.

A 'patch' update performs a limited update to a component, in particular, without changing the version of the component.

To add a new component, the component must not exist.

To replace or remove a component, the component must exist.

To patch a component, the component must exist.

When replacing or removing a component, or when patching a component, usually, at least one version prerequisite is specified for the component update.

Value Detail:

<code>enumUpdateType.add</code>	Specifies that an update adds a component.
<code>enumUpdateType.replace</code>	Specifies that an update replaces a component.
<code>enumUpdateType.remove</code>	Specifies that an update removes a component.
<code>enumUpdateType.patch</code>	Specifies that an update modifies a component, but does not change its version.

Enum Type: `enumEventResult`

Values:	0 Succeeded
	1 Failed
	2 Cancelled

Type Detail:

An event result instance denotes a particular result for an update event. The result indicates success, failure, or cancellation.

Value Detail:

<code>enumEventResult.succeeded</code>	Specifies that the operation was successful.
<code>enumEventResult.failed</code>	Specifies that the operation failed.
<code>enumEventResult.cancelled</code>	Specifies that the operation was cancelled.

versionInfo command

The **versionInfo** command generates reports from data it extracts from XML files in the `properties/version` folder.

Product version and history information

The `/properties/version` directory in the installation root contains important data about the product and its installed components, such as the build version and build date. This information is included in *product.product* and *component.component* files. The `/properties/version/history` directory in the installation root contains a collection of records for installed interim fixes and fix packs. This information is included in *interim fixID.efixApplied*, *interim fixID.efixDriver*, *fix packID.ptfApplied*, and *fix packID.ptfDriver* files. A driver file has useful information about the entire contents of an interim fix or fix pack. The applied file has relevant information about the interim fixes or fix packs that are currently applied. `Event.history` files are also present. They contain a detailed log about updates you have applied, either successfully or unsuccessfully. Time-stamped, detailed logs record each update process in the `/properties/version/log` directory of the installation root.

You can view product information by examining files in the `properties/version` directory, including the `properties/version/history` directory. WebSphere Application Server also provides the ability to generate two types of reports about the data in the files, **Version** reports and **History** reports.

Restriction: There is one restriction. Do not use the `versionInfo.sh` or `versionInfo.bat` script while installing or uninstalling the product, or while installing or uninstalling an interim fix or fix pack.

Product version reports

The following report generation scripts extract data from XML data files in the `properties/version` folder:

- `versionInfo` script
Lets you use parameters to create a version report on Linux and UNIX-based platforms, or on Windows platforms.
- `genVersionReport` script
Generates the `versionReport.html` report file in the `bin` directory on Linux and UNIX-based platforms, or on Windows platforms. The report includes the list of components, fixes, and fix packs.

Location of the command file

The command file is a script. On Linux and UNIX platforms, the command file is named `versionInfo.sh` in the `install_root/bin` directory. On Windows platforms, the command file is named `versionInfo.bat` in the `install_root\bin` directory.

Syntax for the `versionInfo` command on a Linux or UNIX-based platform

The command syntax is:

```
versionInfo.sh [ -format text | html ]  
               [ -file output file ]  
               [ -long ]  
               [ -efixes ]  
               [ -efixDetail ]  
               [ -ptfs ]  
               [ -ptfDetail ]  
               [ -components ]  
               [ -componentDetail ]
```

```
versionInfo.sh [ -help | -? | /help | /? | -usage ]
```

Issue the command from the `install_root/bin` directory.

Syntax for the `versionInfo` command on a Windows platform

The command syntax is:

```
versionInfo [ -format text | html ]  
            [ -file output file ]  
            [ -long ]  
            [ -efixes ]  
            [ -efixDetail ]  
            [ -ptfs ]  
            [ -ptfDetail ]  
            [ -components ]  
            [ -componentDetail ]
```

```
versionInfo [ -help | -? | /help | /? | -usage ]
```

Issue the command from the `install_root\bin` directory.

Parameters

-? or /? (Windows only)

Displays command syntax.

-components

Adds a list of installed components to the report.

-componentDetail

Adds details about installed components to the report.

-efixes

Adds a list of applied interim fixes to the report.

-efixDetail

Adds details about applied interim fixes to the report.

-file *fileName*

Specifies the output file name. The report goes to standard output (stdout) by default.

-format text | html

Selects the format of the report. The default is "text".

-help or /help (Windows only)

Displays command syntax.

-long

Creates the long version of the report.

-ptfDetail

Adds details about applied fix packs to the report.

-ptfs

Adds a list of applied fix packs to the report.

-usage

Displays command syntax.

Logging

As the tool runs, it creates reports instead of log entries. Reports appear on the console unless directed to a file with the `-file filename` parameter. There is no default file name. You must specify a file name to generate a file.

When issued on a Windows platform, from the bin directory of the Network Deployment product that has no interim fixes or fix packs applied, the `versionInfo.bat` script displays the following information:

```
D:\Program Files\WebSphere\DeploymentManager\bin>versionInfo
WVER0010I: Copyright (c) IBM Corporation 2002; All rights reserved.
WVER0011I: WebSphere Application Server Release 5.0
WVER0012I: VersionInfo reporter version 1.14, dated 5/9/03
```

```
-----
IBM WebSphere Application Server Product Installation Status Report
-----
```

Report at date and time 2003-10-05T11:01:45-04:00

Installation

```
-----
Product Directory  D:\Program Files\WebSphere\DeploymentManager
Version Directory  ${product.dir}\properties\version
```

```
DTD Directory      ${version.dir}\dtd
Log Directory      D:\Program Files\WebSphere\DeploymentManager\logs\update
Backup Directory   ${version.dir}\backup
TMP Directory      D:\DOCUME~1\ADMINI~1\LOCALS~1\Temp
```

Installation Platform

```
Name      IBM WebSphere Application Server
Version   5.1
```

Technology List

```
ND      installed
```

Installed Product

```
Name      IBM WebSphere Application Server Network Deployment
Version   5.1.0
ID        ND
Build Level b0334.18
Build Date 8/30/03
```

End Installation Status Report

genVersionReport command

The **genVersionReport** command generates the `versionReport.html` report file in the `bin` directory on Linux and UNIX platforms, or on Windows platforms. The report includes the list of components, fixes, and fix packs.

Product version and history information

The `/properties/version` directory in the installation root contains important data about the product and its installed components, such as the build version and build date. This information is included in `[product].product` and `[component].component` files. The `/properties/version/history` directory in the installation root contains a collection of records for installed interim fixes and fix packs. This information is included in `[interim fixID].efixApplied`, `[interim fixID].efixDriver`, `[fix packID].ptfApplied`, and `[fix packID].ptfDriver` files. A driver file has useful information about the entire contents of an interim fix or fix pack. The applied file has relevant information about the interim fixes or fix packs that are currently applied. `Event.history` files are also present. They contain a detailed log about updates you have applied, either successfully or unsuccessfully. Time-stamped, detailed logs record each update process in the `/properties/version/log` directory of the installation root.

You can view product information by examining files in the `properties/version` directory, including the `properties/version/history` directory. WebSphere Application Server also provides the ability to generate two types of reports about the data in the files, **Version** reports and **History** reports.

Product version reports

The following report generation scripts extract data from XML data files in the `properties/version` folder:

- `versionInfo` script
Lets you use parameters to create a version report on Linux and UNIX-based platforms, or on Windows platforms.
- `genVersionReport` script

Generates the `versionReport.html` report file in the `bin` directory on Linux and UNIX-based platforms, or on Windows platforms. The report includes the list of components, fixes, and fix packs.

Location of the command file

The command file is a script. On Linux and UNIX platforms, the command file is named `genVersionReport.sh` in the `install_root/bin` directory. On Windows platforms, the command file is named `genVersionReport.bat` in the `install_root\bin` directory.

Syntax for the `versionInfo` command on a Linux or UNIX-based platform

The command syntax is:

```
genVersionReport.sh
```

Issue the command from the `install_root/bin` directory.

Syntax for the `versionInfo` command on a Windows platform

The command syntax is:

```
genVersionReport.bat
```

Issue the command from the `install_root\bin` directory.

Logging

As the tool runs, it creates the `versionReport.html` report file instead of log entries.

Creating the report

This example shows how to issue the command on a Windows platform, from the `bin` directory of the Network Deployment product:

```
D:\Program Files\WebSphere\DeploymentManager\bin>genVersionReport
WVER0010I: Copyright (c) IBM Corporation 2002; All rights reserved.
WVER0011I: WebSphere Application Server Release 5.0
WVER0012I: VersionInfo reporter version 1.14, dated 5/9/03
```

When the Network Deployment product has no interim fixes or fix packs applied, the `genVersionReport.bat` script creates the following information in the `versionReport.html` report file, which is edited to show only the first few components:

```
IBM WebSphere Application Server Product Installation Status Report
```

```
-----
```

```
Report at date and time 2003-10-05T11:58:40-04:00
```

```
Installation
```

```
Product Directory D:\Program Files\WebSphere\DeploymentManager
Version Directory ${product.dir}\properties\version
DTD Directory ${version.dir}\dtd
Log Directory D:\Program Files\WebSphere\DeploymentManager\logs\update
Backup Directory ${version.dir}\backup
TMP Directory D:\DOCUME~1\ADMINI~1\LOCALS~1\Temp
```

```
Installation Platform
```

```
Name IBM WebSphere Application Server
Version 5.1
```


Technology List
ND installed

Installed Product
Name IBM WebSphere Application Server for Network Deployment
Version 5.1.0
ID ND
Build Level b0334.18
Build Date 8/30/03

Installed Component
Component Name activity
Spec Version 5.0
Build Version b0334.18
Build Date 8/30/03

Installed Component
Component Name activity.impl
Spec Version 5.0
Build Version b0334.18
Build Date 8/30/03

Installed Component
Component Name activity.session
Spec Version 5.0
Build Version b0334.18
Build Date 8/30/03

End Installation Status Report

Uninstalling the product

This task describes how to uninstall WebSphere Application Server Network Deployment from the root user on a Linux platform or a UNIX-based platform, or from a user ID that belongs to the administrator group on a Windows platform.

IBM Support offers tutorials on various WebSphere Application Server installation topics. See WebSphere education on demand: Installation best practices tutorials for more information. One topic describes uninstalling WebSphere Application Server products.

Several factors affect the procedure for uninstalling WebSphere Application Server products:

- Security
- Migrating
- Coexisting
- Using the embedded messaging feature
- Being part of a deployment manager cell

WebSphere Application Server Network Deployment provides an uninstaller program, which you can call with the “uninstall command” on page 275.

Before issuing the command, locate your particular scenario for uninstalling and follow the recommended procedure. The potential exists for losing the embedded messaging feature code for WebSphere

Application Server products that remain on the machine unless you follow the particular procedure. Another possible consequence of not following the correct procedure for uninstalling is losing the configuration of a federated node after migration.

The uninstaller program removes registry entries, uninstalls the product, and removes all related features and products, such as plug-ins. However, the uninstaller program does not remove some files. The uninstaller program does not delete any configuration files that are changed as the result of selecting installation options, or running Samples, for example. The uninstaller program does not delete log files.

Uninstalling products in the proper sequence

Uninstall WebSphere Application Server products in the following sequence when you are uninstalling more than one product:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment and Network Deployment
2. **5.1+** WebSphere Business Integration Server Foundation on a base node and the base product
3. WebSphere Application Server Network Deployment
4. WebSphere Application Server (base product)

If you do not have the Network Deployment product installed, open the information center for the base WebSphere Application Server to find recommended procedures for uninstalling unfederated base product nodes.

Planning for time to uninstall The time required to uninstall a product depends on the number of configured servers. The uninstaller program attempts to stop all running servers. As a result, the time required to uninstall is directly proportional to the number of defined servers. The uninstaller program attempts to contact each configured server and waits for a timeout before assuming that a server is not running. Therefore, uninstalling many servers can take several minutes. The uninstaller program can run slightly faster if all servers are running, which avoids waiting for a timeout condition.

Examining logs from the uninstalling operation

5.1+ Examine the `install_root/logs/WAS.WBISF.uninst.log` file to verify that no file system or other unusual errors occurred. Also examine the `install_root/logs/uninstlog.txt` file if you uninstall the base product or Network Deployment while you are uninstalling Integration Server.

Examine the `install_root/logs/uninstlog.txt` file to verify that no file system or other unusual errors occurred.

1. Close all Web browser processes. Before uninstalling, verify that you have no open Web browsers that are accessing the administrative console. Otherwise, the potential for locked file errors exists.
2. Stop the IBM HTTP Server and any related Java processes. Stop these processes only if the processes are running on the same machine.
3. Disable WebSphere Application Server security. If security is on and Application Servers are running, the uninstaller program cannot detect the status of the Application Servers because security is on. The uninstall procedure fails because it cannot stop the Applications Server processes.

Another scenario involving security is when a federated Application Server is stopped. The uninstall procedure fails because the uninstaller program cannot remove the node from the cell when security is enabled.

4. Stop all Java processes that are related to WebSphere Application Server products on the machine.
 - a. Stop any embedded messaging feature services, such as WebSphere embedded messaging publish and subscribe, `jmsserver`, or WebSphere MQ queue managers, and any related Java processes. Stop these processes only if the processes are running on the same machine.

For example, if you have a federated base product with the embedded messaging server and client feature installed on the same Linux machine, use the following command from the `/opt/WebSphere/AppServer/bin` directory to stop the `jmsserver` process:

```
./stopServer.sh jmsserver
```

- b. Stop any Java processes related to federated WebSphere Application Server processes with the **stopServer** command and the **stopNode** command. Stop all server processes for WebSphere Application Server that are running on the machine. For example, use the following commands from the `/opt/WebSphere/AppServer/bin` directory on a Linux machine to stop the `server1` and `nodeagent` processes:

```
./stopServer.sh server1  
./stopNode.sh
```

If you have another installation instance of WebSphere Application Server, V5.0.x or V5.1, stop all servers that are running on the other instances too.

If servers are running and security is enabled, the uninstaller program cannot shut down the servers and the uninstall procedure fails. Manually stop all servers before uninstalling.

- c. Use the “`wsinstance` command” on page 195 to delete any configuration instances of the product that you are uninstalling. The following example command is on more than one line for formatting purposes. Enter the command on one line to delete the configuration instance named `shasti` from a Linux machine:

```
./wsinstance.sh -name shasti -host planetlinux -delete
```

- d. Stop the deployment manager `dmgr` process with the `stopManager` command. Stop all `dmgr` processes that are running on the machine. For example, issue this command on a Linux machine from the `/opt/WebSphere/DeploymentManager/bin` directory:

```
./stopManager.sh
```

5. Uninstall the product. Select your uninstalling scenario from the following list of options:
 - Uninstall Network Deployment when neither migration nor coexistence are involved.
 - Uninstall a federated base WebSphere Application Server node when neither migration nor coexistence are involved.
 - Uninstall a federated base WebSphere Application Server node while migrating or after migrating.
 - Uninstall base WebSphere Application Server in a coexistence scenario.
6. Optional: Uninstall manually before reinstalling. The uninstaller program leaves some log and configuration files that were changed during installation. You can manually uninstall the product to remove all artifacts of the product instance from the platform. This step is required before you can reinstall the product into the same installation root directory. Follow the reinstallation scenario for V5.1 if you left the embedded messaging feature installed and now you want to reinstall the V5.1 product. Follow the other procedure when you did not leave the embedded messaging feature installed, or when you are removing all product artifacts from V5.1 or V5.0.x.
 - Uninstall manually after uninstalling V5.1:
 - Uninstall manually on all platforms but Solaris and HP-UX before reinstalling.
 - Uninstall manually on Solaris platforms before reinstalling.
 - Uninstall manually on HP-UX platforms before reinstalling.
 - Uninstall manually after uninstalling V5.0.x, or when you are removing all product artifacts from V5.1.

You can uninstall the product.

uninstall command

The **uninstall** command uninstalls the product. Uninstalling the product also selects the feature uninstaller programs, such as the uninstaller program for the IBM HTTP Server if you installed IBM HTTP Server as a feature of the base product. The uninstaller wizard prompts you to uninstall the embedded messaging

feature because other product instances might share the feature. Options on the silent parameter of the uninstaller program control the removal of an existing embedded messaging feature when you uninstall silently.

Before you begin uninstalling the product

If servers are running and security is enabled, the uninstaller program cannot shut down the servers and the uninstall procedure fails. Manually stop all servers before uninstalling the product.

Overview of uninstalling the product

The uninstaller wizard displays a panel when it begins. Click **Next** to begin uninstalling the product when the embedded messaging feature is not installed, or to display a panel with a prompt for uninstalling the embedded messaging feature, if the feature is installed.

If the embedded messaging feature is installed, click the check box to clear the selection and leave the embedded messaging feature installed. You can do this legally only when there is another instance of a WebSphere Application Server product on the machine that also uses the embedded messaging feature and shares the embedded messaging code. Another example of properly retaining the embedded messaging code occurs when migrating to a new version or release of a WebSphere Application Server product that also uses the embedded messaging code.

Leave the check box selected to uninstall the embedded messaging feature.

Click **Next** to begin uninstalling the product.

Location of the command file **5.1 +**

The command file is located in the `_uninst` directory in the installation root directory.

The command file is named `uninstall` for Linux and UNIX-based platforms and `Uninstall.exe` on Windows platforms.

Command syntax **5.1 +**

Command parameters are described at the end of this topic. This section describes the recommended command syntax, which starts the uninstaller wizard.

Command syntax for uninstalling Network Deployment

- **5.1 +** On Linux and UNIX-based platforms, issue the **uninstall** command from the `ND_install_root/_uninst` directory:
`uninstall`
- On Windows platforms, run the uninstaller program in the installation root of the product that you intend to uninstall. Issue a fully qualified command or change directories to the `ND_install_root/_uninst` directory and issue the **uninstall** command.

For example, if the installation root is `C:\WebSphere\NetworkDeployment`, issue one of the following commands:

```
> "C:\WebSphere\NetworkDeployment\_uninst\uninstall"  
> "C:\WebSphere\NetworkDeployment\_uninst\Uninstall.exe"  
C:\WebSphere\NetworkDeployment\_uninst> uninstall  
C:\WebSphere\NetworkDeployment\_uninst> Uninstall.exe
```

Uninstalling the base product or Network Deployment when uninstalling Integration Server

You can uninstall the base product or the Network Deployment product when you uninstall the WebSphere Business Integration Server Foundation product, depending on which product the Integration Server is extending. A panel appears that prompts you to uninstall the extended product. If you select to uninstall the product, the uninstaller program removes the product along with the Integration Server. If you select to leave the extended product, the uninstaller program does not remove the product as it uninstalls the Integration Server.

Logging the results of the uninstall procedure

On Linux and UNIX-based platforms, the return code from the uninstaller program is "1" to indicate success; any other response code indicates failure. There is no return code on Windows platforms.

Examine the `install_root/logs/uninstlog.txt` file to verify that there were no file system or other unusual errors while uninstalling. If there are problems, correct them, and see "Uninstalling manually" on page 292 before reinstalling the product.

After uninstalling a WebSphere Application Server product, delete the installation root directory before reinstalling into the same directory. Deleting the old directory avoids invalid XML configurations that result from retaining old files. To delete all files so that you can reinstall with a clean system, see "Uninstalling manually" on page 292.

Command syntax for using the `-silent` parameter

If you remove the embedded messaging feature when uninstalling V5.1, any instance of WebSphere Application Server, Version 5.0 on your system cannot properly use the embedded messaging function. If you do not remove the embedded messaging feature, you are responsible for verifying that you are (and that you remain) properly licensed to use the embedded messaging feature on this system.

If WebSphere Application Server, Version 5.0 is not installed on this system and you remove WebSphere Application Server, Version 5.1, you may not have any such right to use the embedded messaging feature. By using the silent uninstall method, you indicate your acknowledgment of this condition and your agreement to comply with the relevant licensing terms as they apply to your use of the embedded messaging feature.

Please refer to the license information documents for WebSphere Application Server, Version 5.1 and WebSphere Application Server, Version 5.0 for further details.

Uninstall silently using one of the commands in the following series of tables. Uninstalling silently does not display the uninstaller program wizard.

Use the following commands on a Network Deployment node:

Table 54. Uninstalling Network Deployment silently on Linux and UNIX platforms

Command (from the <code>install_root/_uninst/</code> directory)	Effect on the embedded messaging feature
<code>uninstall -silent</code>	Retains the feature
<code>uninstall -silent -W MQUninstallSelectionPanel.uninstallClient= "false"</code>	Retains the client
<code>uninstall -silent -W MQUninstallSelectionPanel.uninstallClient= "true"</code>	Uninstalls the client

Table 55. Uninstalling Network Deployment silently on Windows platforms

Command (from the <i>install_root_uninst</i> directory)	Effect on the embedded messaging feature
Uninstall.exe -silent	Retains the feature
Uninstall.exe -silent -W MQUninstallSelectionPanel.uninstallClient= "false"	Retains the client
Uninstall.exe -silent -W MQUninstallSelectionPanel.uninstallClient= "true"	Uninstalls the client

Use the following commands on a base WebSphere Application Server node:

Table 56. Uninstalling WebSphere Application Server silently on Linux and UNIX

Command (from the <i>install_root/_uninst/</i> directory)	Effect on the embedded messaging feature
uninstall -silent	Retains the feature
uninstall -silent -W MQUninstallSelectionSequence.active= "false"	Uninstalls the feature
uninstall -silent -W MQUninstallSelectionSequence.active= "true"	Retains the feature
uninstall -silent -W MQUninstallSelectionPanel.uninstallClient= "false"	Retains the client
uninstall -silent -W MQUninstallSelectionPanel.uninstallClient= "true"	Uninstalls the client
uninstall -silent -W MQUninstallSelectionPanel.uninstallServer= "false"	Retains the server and client
uninstall -silent -W MQUninstallSelectionPanel.uninstallServer= "true"	Uninstalls the server and client

Table 57. Uninstalling WebSphere Application Server silently on Windows platforms

Command (from the <i>install_root_uninst</i> directory)	Effect on the embedded messaging feature
Uninstall.exe -silent	Retains the feature
Uninstall.exe -silent -W MQUninstallSelectionSequence.active= "false"	Uninstalls the feature
Uninstall.exe -silent -W MQUninstallSelectionSequence.active= "true"	Retains the feature
Uninstall.exe -silent -W MQUninstallSelectionPanel.uninstallClient= "false"	Retains the client
Uninstall.exe -silent -W MQUninstallSelectionPanel.uninstallClient= "true"	Uninstalls the client
Uninstall.exe -silent -W MQUninstallSelectionPanel.uninstallServer= "false"	Retains the server and client
Uninstall.exe -silent -W MQUninstallSelectionPanel.uninstallServer= "true"	Uninstalls the server and client

vpd.properties file

The installer program for any WebSphere Application Server product uses the Install Shield for Multiplatforms (ISMP) program to install code. ISMP uses the *vpd.properties* file to track WebSphere products that it installs on all platforms but Solaris and HP-UX. The *vpd.properties* file lists program components that are currently installed. The file helps ISMP and the installer programs of WebSphere Application Server products to recognize previous installations of WebSphere Application Server products and to control options for new installations.

Situations that require you to edit the vpd.properties file

Certain situations require you to edit the vpd.properties file before reinstalling a WebSphere Application Server product. The uninstaller programs for WebSphere Application Server products edit the vpd.properties while uninstalling a product, to remove entries for the product and any of its features that might have entries in the file.

It is necessary to remove product entries in the vpd.properties file before you can reinstall the product. Some situations that occur require manually removing entries from the vpd.properties file before you can reinstall a product. These situations include:

- Removing entries for the embedded messaging feature when the feature was left intact but the product was uninstalled
- Bypassing the uninstaller program to uninstall a product manually
- Manually uninstalling a product when the uninstaller program is not present or is not working

If the vpd.properties file has entries for a product you uninstalled, you must edit the file and remove the entries. If you do not edit the vpd.properties file to remove entries for a product or features of a product, you cannot reinstall the product into the same directory structure. If product entries in the vpd.properties file are present, the installer program reads the vpd.properties file, determines that the product is already installed, and displays the panel that prompts you to install additional features into the existing instance or to install a new instance. Unfortunately, the existing instance might not be a valid installation. The installer program does not verify the products that it finds listed in the vpd.properties file.

Location of the vpd.properties file

The location of the vpd.properties file varies per operating platform:

- AIX platforms: The root directory or the usr/lib/objrepos directory
- Linux platforms: The root directory
- Windows platforms: installation directory of the operating system, such as the C:\WINNT directory or the C:\windows directory.

Example of the vpd.properties file

The following example shows the first ten entries for the vpd.properties file for Version 5.0.2 of the base product on a Windows platform. The example shows the entire first line but breaks the line for formatting purposes. The example truncates the rest of the lines in the file. The identifier for the first component is highlighted. The installation location of the first component is also highlighted.

```
WSBACAA|5|0|0|0| |1=Admin Console|Admin Console|
    $L(com.ibm.ws.install.resourcebundle.
    WSResourceBundle,Features.admin.console.description)| | |
D:\WebSphere\AppServer50|0|2|
WSBSRAA|5|0|0|0| |0|0|2|ref_759810|
WSBASAA|5|0|0|0| |0|0|2|ref_58595|1|
WSBADAA|5|0|0|0| |1|0|false|true|3|
WSBACAA|5|0|0|0| |1
WSBAC2AA|5|0|0|0| |1=adminConsoleRegKeyComponent|admin
WSBSR4AA|5|0|0|0| |1=serverConfigWithoutSamplesCompon
WSBMSAA|5|0|0|0| |1=mqSeriesServer|MQSeries Server|$L(
WSBC05AA|5|0|0|0| |1=IsmpLauncherComponent|IsmpLaunche
WSBASAA|5|0|0|0| |1=Admin Scripting|Admin Scripting|$L
WSBAC1AA|5|0|0|0| |1=adminConsoleFilesComponent|adminC
WSBC04AA|5|0|0|0| |1=pbwServerConfigWithMQGood|pbwSamp
WSBSMAA|5|0|0|0| |1=Samples|Samples|$L(com.ibm.ws.inst
WSBSM2AA|5|0|0|0| |1=samplesRegKeyComponent|samplesReg
```

Example of the vpd.properties file after a coexistence or migration installation

The following example shows the first ten entries of the vpd.properties file when you install V5.1 of the base WebSphere Application Server product onto a Windows platform that already has V5.0.2 installed.

The example shows partial entries and breaks the lines for formatting purposes. Highlighting shows differences in the identifiers and the installation directories of the components. V5.1 identifiers have "51" at the end except for the application clients identifier.

The last two lines in the example show how entries identify a second instance of a component with a common identifier. Highlighting in these lines shows how to identify entries for a particular version.

```
WSBJD8AA51|1|3| |1| |1=javaRegistryComponent|javaRegistryUpdate| | | |
  D:\WebSphere\AppServer51|0|0|1|...
WSBACAA|5|0|0|0| |1=Admin Console|Admin Console|...
  D:\WebSphere\AppServer50|0|2|...
WSBSM2AA51|5|1|0|0| |1=samplesRegKeyComponent|...
  D:\WebSphere\AppServer51|0|0|1|
WSBAC2AA|5|0|0|0| |1=adminConsoleRegKeyComponent|...
  D:\WebSphere\AppServer50|0|0|1|...
WSBSR4AA|5|0|0|0| |1=serverConfigWithoutSamplesComponent|...
  D:\WebSphere\AppServer50|0|0|1|
WSBMSAA|5|0|0|0| |1=mqSeriesServer|MQSeries Server|...
  C:\Program Files\IBM\WebSphere MQ|0|1|...
WSBAS2AA51|5|1|0|0| |1=adminScriptingRegKeyComponent|adminScriptingR
WSBSR4AA51|5|1|0|0| |1=serverConfigWithoutSamplesComponent|config| |
WSBC05AA|5|0|0|0| |1=IsmpLauncherComponent|IsmpLauncherComponent| |
WSBAC1AA51|5|1|0|0| |1=adminConsoleFilesComponent|adminConsoleFiles|
.
.
.
WSBMQ4AA|5|0|0|0| |2=mqSeriesPropFilesComponent|...
  D:\WebSphere\AppServer51\properties|0|0|1|...
WSBMQ4AA|5|0|0|0| |1=mqSeriesPropFilesComponent|...
  D:\WebSphere\AppServer50\properties|0|0|1|...
```

Example of the vpd.properties file after uninstalling V5.1 but keeping the embedded messaging feature

The following example shows entries for the vpd.properties file after uninstalling the V5.1 base product but keeping the embedded messaging feature for use with coexisting WebSphere Application Server products. The example shows all entries for V5.1 that remain in the file after uninstalling V5.1.

The example shows partial entries and breaks the lines for formatting purposes. Highlighting identifies the V5.1 entries that still remain in the file. The last line shows an entire entry.

```
WSBACAA|5|0|0|0| |1=Admin Console|Admin Console|...
  D:\WebSphere\AppServer50|0|2|...
WSBAA51|5|1|0|0| |1=WebSphere Application Server|...
  D:\WebSphere\AppServer51...
WSBMQ4AA|5|0|0|0| |2=mqSeriesPropFilesComponent|...
  D:\WebSphere\AppServer51\properties|0|0|1|...
WSBMQAA|5|0|0|0| |2=MQSeries|MQSeries|...
  D:\WebSphere\AppServer51|0|0|1|...
WSBMS2AA|5|0|0|0| |2=mqSeriesWindowsConfig|...
  D:\WebSphere\AppServer51|0|0|1|...
WSBMQ3AA|5|0|0|0| |2=mqSeriesLibFilesComponent|...
  D:\WebSphere\AppServer51\lib|0|0|1|...
WSBMS1AA|5|0|0|0| |2=mqSeriesWindowsInstall|...
  D:\WebSphere\AppServer51|0|0|1|...
WSBMQ2AA|5|0|0|0| |2=mqSeriesBinComponent|...
  D:\WebSphere\AppServer51\bin|0|0|1|...
WSBMQ1AA|5|0|0|0| |2=mqSeriesSetupFileComponent|
```



```
mqSeriesSetupFileComponent| | | |
D:\WebSphere\AppServer51\_uninst\mq|0|0|1|
WSBMQAA|5|0|0|0| |2|0|false| |true|3|WSBMQ1AA|5|0|0|0| |2
```

Editing the vpd.properties file before reinstalling V5.1

If you uninstall V5.1 in a coexistence or migration environment, you can reinstall the product.

There are two options for reinstalling the product:

- Installing into a new directory.
Edit the vpd.properties file to remove the **WSBAA51** entry.
Select another installation path during the installation.
- Edit the vpd.properties file to remove all entries that identify V5.1 and reinstall into the same directory structure as the original V5.1 product.
Edit the vpd.properties file in a flat-file editor. Edit the file on the operating platform where the file resides to avoid problems with line-end characters. Use an editor that shows line breaks so that you can see each identifier at the beginning of a line.

With either option, you can select coexistence and assign non-conflicting ports, or select migration and migrate the configuration from the V5.0.x product to the V5.1 product. You can also select to neither coexist or migrate to run each product when the other is not running.

Perform the following procedure to remove all entries in the vpd.properties file and to reinstall into the original V5.1 installation root directory:

1. Back up the vpd.properties file before you edit it.
2. Search for and delete lines that start with an identifier that ends in "51". For example, delete the following entry from the file:

```
WSBAA51|5|1|0|0| |1=WebSphere Application Server|IBM WebSphere Application Server
D:\WebSphere\AppServer51|0|0|1|WSBAA51|5|1|0|0| |1|0|false|"_uninst" "uninst"
```

3. Search for and delete all multiple instance entries that specify the original installation root. For example, delete the following entries from the file:

```
WSBMQ4AA|5|0|0|0| |2=mqSeriesPropFilesComponent|
mqSeriesPropFilesComponent| | | |
D:\WebSphere\AppServer51\properties|0|0|1|
WSBMQAA|5|0|0|0| |2|0|false| |true|3|WSBMQ4AA|5|0|0|0| |2
WSBMQAA|5|0|0|0| |2=mqSeries|MQSeries|...
D:\WebSphere\AppServer51|0|0|1|...
WSBMS2AA|5|0|0|0| |2=mqSeriesWindowsConfig|...
D:\WebSphere\AppServer51|0|0|1|...
WSBMQ3AA|5|0|0|0| |2=mqSeriesLibFilesComponent|...
D:\WebSphere\AppServer51\lib|0|0|1|...
WSBMS1AA|5|0|0|0| |2=mqSeriesWindowsInstall|...
D:\WebSphere\AppServer51|0|0|1|...
WSBMQ2AA|5|0|0|0| |2=mqSeriesBinComponent|...
D:\WebSphere\AppServer51\bin|0|0|1|...
WSBMQ1AA|5|0|0|0| |2=mqSeriesSetupFileComponent|
mqSeriesSetupFileComponent| | | |
D:\WebSphere\AppServer51\_uninst\mq|0|0|1|
WSBMQAA|5|0|0|0| |2|0|false| |true|3|WSBMQ1AA|5|0|0|0| |2
```

4. Rename the installation root directory to back it up. For example, change the name from D:\WebSphere\AppServer51 to D:\WebSphere\AppServer51Backup.
5. Reinstall the base WebSphere Application Server into the D:\WebSphere\AppServer51 directory.

Identifying entries in the vpd.properties file

Use the following table to help identify product entries. The first three characters of any identifier are unique to a product. Version 5.1 identifiers end with "51" as the last two characters except for the application clients, which is the same for all versions.

Table 58. Identifiers in the vpd.properties file for WebSphere products

Identifier	Product
WSB...	WebSphere Application Server (base product)
WSB...51	WebSphere Application Server, Version 5.1
WSBM...	WebSphere Application Server embedded messaging feature
WSC...	WebSphere Application Server application clients, all releases
WSE...	WebSphere Application Server Enterprise, Version 5.0.x
WSE...51	WebSphere Business Integration Server Foundation, Version 5.1
WSN...	WebSphere Application Server Network Deployment
WSN...51	WebSphere Application Server Network Deployment, Version 5.1
WSNM...	WebSphere Application Server Network Deployment embedded messaging client feature
WSM...	WebSphere MQ (the stand-alone product, not the embedded messaging feature)

Uninstalling Network Deployment

WebSphere Application Server Network Deployment provides an uninstaller program that you can use to uninstall the product. This task describes using the uninstaller program to uninstall the Network Deployment product.

Before uninstalling, verify that you have no open Web browsers that are accessing the administrative console.

Uninstalling products in the proper order Depending on the WebSphere Application Server products you installed, uninstall all that you intend to uninstall in this order:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment
2. **5.1+** WebSphere Business Integration Server Foundation on base nodes
3. WebSphere Application Server Network Deployment
4. WebSphere Application Server (base product)
 1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in "Uninstalling the product" on page 273.
 2. Change directories to the `_uninst` directory before issuing the "uninstall command" on page 275. The command file is located in the `install_root/_uninst` directory on a Linux or UNIX-based platform, and in the `install_root_uninst` directory on a Windows system.

For example, to change directories on a Linux platform, issue this command if your installation root is `/opt/WebSphere/NetworkDeployment`:

```
cd /opt/WebSphere/NetworkDeployment/_uninst
```

On a Windows system, you can use the Add/Remove Programs feature and bypass this step, if you prefer.
 3. Issue the **uninstall** command. The command file is named `uninstall` for Linux and UNIX-based platforms and `Uninstall.exe` on Windows platforms. The command file for V5.0.x on Linux for S/390 platforms* is named `uninstall.sh`. For V5.1 the command file is named `uninstall`, as it is on other Linux platforms.

*Linux for S/390 platforms refers to the Linux distributions available from Linux distributors that run on IBM eServer zSeries and S/390 systems in 31-bit mode.

On Windows platforms, call the **Uninstall.exe** command:

```
ND_install_root\_uninst\Uninstall.exe
```

Call the program directly from the *install_root_uninst* directory. For example, if the installation root is C:\WebSphere\NetworkDeployment, issue the following command:

```
C:\WebSphere\NetworkDeployment\_uninst> Uninstall.exe
```

On Linux and UNIX-based platforms, issue the **uninstall** command from the *install_root/_uninst* directory:

```
uninstall
```

Uninstalling the product also selects the feature uninstaller programs, such as the uninstaller program for the Web services feature.

The uninstaller program on V5.1 gives you the option of leaving the embedded messaging client feature installed. This is a legal option only when there is another instance on the machine of either Network Deployment or the base product that has the client feature installed. The V5.0.x uninstaller program does not have this enhancement of leaving the embedded messaging client code installed. Therefore, the procedure for uninstalling Network Deployment varies slightly, depending on the version you are uninstalling.

The V5.1 uninstaller wizard prompts you whether to uninstall the embedded messaging client feature. The prompt lets you select to keep the embedded messaging client because it might be shared with other product instances. Legally you can keep the feature installed only when there is another instance of a WebSphere Application Server product on the machine that also uses the feature.

The uninstaller wizard begins and displays a panel. In V5.0.x the uninstaller program displays a language selection panel.

4. Click **Next** to begin uninstalling the product. The V5.1 uninstaller wizard prompts you whether to uninstall the embedded messaging feature, if the feature is installed. To obey licensing rules, always remove the embedded messaging feature unless there are other product instances that share the code. The uninstaller wizard displays a summary panel that lists the product and features that you are uninstalling.
5. Click **Next** to continue uninstalling the product.
6. Click **Finish** to close the wizard after the wizard removes the product.
7. Optional: Uninstall manually before reinstalling. This step is required before you can reinstall the product into the same installation root directory. The procedures vary per product that you are uninstalling:
 - Uninstall manually after uninstalling V5.1:
 - Uninstall manually on all platforms but Solaris and HP-UX before reinstalling, as described in “Reinstalling V5.1 after uninstalling and leaving the embedded messaging feature installed” on page 310.
 - Uninstall manually on Solaris platforms before reinstalling, as described in Platform-specific tips for installing and migrating.
 - Uninstall manually on HP-UX platforms before reinstalling, as described in Platform-specific tips for installing and migrating.
 - Uninstall manually after uninstalling V5.0.x, or when you are removing all product artifacts from V5.1.

You can uninstall Network Deployment after which you can reinstall.

Uninstalling base WebSphere Application Server

This task describes using the uninstaller program to uninstall WebSphere Application Server.

Before uninstalling, verify that you have no open Web browsers that are accessing the administrative console.

Uninstalling products in the proper order Depending on the WebSphere Application Server products you installed, uninstall all that you intend to uninstall in this order:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment
2. **5.1+** WebSphere Business Integration Server Foundation on base nodes
3. WebSphere Application Server Network Deployment
4. WebSphere Application Server (base product)

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Change directories to the `_uninst` directory before issuing the **uninstall** command. The command file is located in the `install_root/_uninst` directory on a Linux or UNIX-based platform, and in the `install_root_uninst` directory on a Windows system.

For example, to change directories on a Linux platform, issue this command if your installation root is `/opt/WebSphere/AppServer`:

```
cd /opt/WebSphere/AppServer/_uninst
```

3. Issue the **uninstall** command. The command file is named `uninstall` for Linux and UNIX-based platforms and `Uninstall.exe` on Windows platforms. The command file for V5.0.x on Linux for S/390 platforms* is named `uninstall.sh`. For V5.1 the command file is named `uninstall`, as it is on other Linux platforms.

*Linux for S/390 platforms refers to the Linux distributions available from Linux distributors that run on IBM eServer zSeries and S/390 systems in 31-bit mode.

On Windows platforms, call the **Uninstall.exe** command:

```
install_root\_uninst\Uninstall.exe
```

Call the program directly from the `install_root_uninst` directory. For example, if the installation root is `C:\WebSphere\AppServer`, issue the following command:

```
C:\WebSphere\AppServer\_uninst> Uninstall.exe
```

On Linux and UNIX-based platforms, issue the **uninstall** command from the `install_root/_uninst` directory:

```
uninstall
```

(To uninstall V5.0.x on a Linux for S/390 platform, use the `uninstall.sh` script.)

Uninstalling the product also selects the feature uninstaller programs, such as the uninstaller program for the plug-ins feature.

The V5.1 uninstaller wizard prompts you whether to uninstall the embedded messaging feature when the feature is present. To obey licensing rules, always remove the embedded messaging feature unless there are other product instances on the machine that also share the code. The V5.0.x uninstaller program does not have this enhancement of leaving the embedded messaging feature code installed. Therefore, the procedure for uninstalling WebSphere Application Server varies slightly, depending on the version you are uninstalling.

The uninstaller wizard begins and displays a panel.

In V5.0.x the uninstaller program displays a language selection panel.

4. Click **Next** to begin uninstalling the product. In situations where the embedded messaging feature is installed, the V5.1 uninstaller wizard displays the panel that prompts you whether to uninstall the embedded messaging feature. Click **Next** on that panel to continue uninstalling the product. The uninstaller wizard displays a summary panel that lists the product and features that you are uninstalling.
5. Click **Next** to continue uninstalling the product.
6. Click **Finish** to close the wizard after the wizard removes the product.

You can uninstall WebSphere Application Server.

Uninstalling in a migration scenario

This task describes uninstalling in a migration scenario. Uninstall during the migration procedure when there is an incompatibility such as exists when migrating WebSphere Application Server, Version 5.0.1 with the embedded messaging feature to Version 5.1. Uninstall after migrating Version 5.0.x to Version 5.1 to uninstall one of the two nodes in the migration process, either the V5.0.x node or the V5.1 node.

Several scenarios can occur when uninstalling during migration or after migration. Unless there is an incompatibility with the products you are migrating between, you would leave the old product installed during a time of testing and verification. After you make a decision whether to keep or uninstall the new product, you uninstall either the old product or the new one. Various scenarios that you can follow include:

- Uninstalling during migration because the old product is not compatible with the new one
- Uninstalling a node that does not have the embedded messaging feature
- Uninstalling a base node that has the embedded messaging server and client feature
- Uninstalling a node that has the embedded messaging client feature

Select one of the following optional steps that describes your scenario.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Uninstall the product by selecting one of the following paths:
 - Uninstall during migration because the old product is not compatible with the new one. The general process for uninstalling during migration is described in “Migrating Network Deployment, Version 5.0.0 or Version 5.0.1 with embedded messaging to Version 5.1” on page 330. Uninstall the product with the following procedure.
 - a. Run the uninstall command from the *install_root_uninst* directory of the product that you are uninstalling.
 - b. Run the platform-specific uninstalling procedure to complete the removal by removing all artifacts of the earlier product from the system.
 - c. Complete the migration by installing the new product and running the WASPostUpgrade tool after the installation is complete.
 - Uninstall a Network deployment or federated base node that does not have the embedded messaging feature.
 - Uninstall a federated base node that has the embedded messaging server and client feature.
 - Uninstall a Network deployment or federated base node that has the embedded messaging client feature.

You can uninstall a node during migration or after migration.

Uninstalling a node when the embedded messaging feature is not installed

This task describes the proper procedure for uninstalling a node that does not have the embedded messaging feature installed.

Uninstalling a Network Deployment or a federated base node after migration is affected by the presence of the embedded messaging feature. If the embedded messaging feature is not installed, uninstalling the product is fairly straightforward. This task describes uninstalling when there is no embedded messaging feature.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Uninstall a Network Deployment node.
3. Uninstall a federated base node. “Uninstalling base WebSphere Application Server” on page 283

- a. Change directories to the `bin` directory on the node you are uninstalling. Change directories to the `install_root/bin` directory on a Linux or UNIX-based platform, or to the `install_root\bin` directory on a Windows system.

For example, to change directories on a Linux platform, issue this command if your installation root is `/opt/WebSphere/AppServer`:

```
cd /opt/WebSphere/AppServer/bin
```

- b. Run the **pre_uninst50ws** command to prevent unfederating the node you migrated from or the node you migrated to.

For example, run the following command on a Linux platform:

```
pre_uninst50ws.sh
```

If you are uninstalling V5.0.2 and the script is not present in the V5.0.2 directory, copy it from the V5.1 directory.

If you do not run this script, verify that the deployment manager is not running before uninstalling the product.

- c. Uninstall the base node.

You can uninstall a base or a Network Deployment node that does not have the embedded messaging feature installed.

Uninstalling a base node that has the embedded messaging server and client feature

This task describes uninstalling a base WebSphere Application Server node after migrating from Version 5.0.2 to WebSphere Application Server, Version 5.1. In most situations you uninstall V5.0.2 after migrating to V5.1. However, some situations might require that you uninstall V5.1 while both versions are on the machine.

Before uninstalling, verify that you have no open Web browsers that are accessing the administrative console.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Identify which base node you are uninstalling. Uninstall either the V5.0.2 node and leave the V5.1 node installed, or uninstall the V5.1 node and leave the V5.0.2 node installed. Write down the installation root directory of the node you intend to uninstall if the path names are similar. Examples in this task use the following path names for nodes on a Linux platform:
 - **V5.0.2:** `/opt/WebSphere/AppServer50`
 - **V5.1:** `/opt/WebSphere/AppServer51`

The required steps to uninstall each node are different. Each set of steps is presented as sub-steps in an optional step in this procedure.

3. Change directories to the `bin` directory on the node you are uninstalling. Change directories to the `install_root/bin` directory on a Linux or UNIX-based platform, or to the `install_root\bin` directory on a Windows system.

For example, issue this command to change directories on a Linux platform:

```
cd /opt/WebSphere/AppServer50/bin
```

4. Run the **pre_uninst50ws** command to prevent removing the *migrated-from* node and the *migrated-to* node from the deployment manager cell and to preserve the `jmsserver` process, which is the embedded messaging queue manager for the federated base node.

For example, run the following command on a Linux platform:

```
pre_uninst50ws.sh
```

If you are uninstalling V5.0.2 and the script is not present in the V5.0.2 directory, copy it from the V5.1 directory.

If you do not run this script, verify that the deployment manager is not running before uninstalling the product.

5. Optional: Uninstall the V5.0.2 node.

- a. Change directories to the V5.1 *install_root51/bin* directory.

For example, issue this command to change directories on a Linux platform:

```
cd /opt/WebSphere/AppServer51/bin
```

- b. Run the **pre_uninst502mq** command. The command helps avoid removing the embedded messaging feature code. On Windows platforms, the command creates dummy registry entries. On all platforms, the command creates a dummy queue manager, WAS_COEXIST, that helps retain the embedded messaging feature code. Run the script from the *install_root/bin* directory of the V5.1 installation:

```
pre_uninst502mq.sh
```

- c. Change directories to the *install_root50/_uninst* directory in the V5.0.2 installation root before issuing the **uninstall** command. The command file is located in the *install_root50/_uninst* directory on a Linux or UNIX-based platform, and in the *install_root50_uninst* directory on a Windows system.

For example, use this command to change directories on a Linux platform:

```
cd /opt/WebSphere/AppServer50/_uninst
```

- d. Issue the **uninstall** command. The command file is named `uninstall` for Linux and UNIX-based platforms and `Uninstall.exe` on Windows platforms. The command file for V5.0.x on Linux for S/390 platforms* is named `uninstall.sh`.

*Linux for S/390 platforms refers to the Linux distributions available from Linux distributors that run on IBM eServer zSeries and S/390 systems in 31-bit mode.

On Windows platforms, call the **Uninstall.exe** command:

```
install_root\_uninst\Uninstall.exe
```

Call the program directly from the *install_root_uninst* directory. For example, if the installation root is `C:\WebSphere\AppServer50`, issue the following command:

```
C:\WebSphere\AppServer50\_uninst> Uninstall.exe
```

On Linux and UNIX-based platforms, issue the **uninstall** command from the *install_root/_uninst* directory:

```
uninstall
```

(To uninstall V5.0.x on a Linux for S/390 platform, use the `uninstall.sh` script.)

- e. Change directories to the V5.1 *install_root51/bin* directory.

For example, issue this command to change directories on a Linux platform:

```
cd /opt/WebSphere/AppServer51/bin
```

- f. Run the **post_uninst502mq** command. The command removes the dummy registry entries. Run the script from the *install_root/bin* directory of the V5.1 installation:

```
post_uninst502mq.sh
```

- g. Delete the dummy queue manager if it still exists. Issue the following commands from the *install_root51/bin* directory to delete the dummy queue manager:

```
endmqm WAS_COEXIST  
dltmqm WAS_COEXIST
```

- h. Stop and restart the V5.1 Application Server. For example, run these commands from the *install_root51/bin* directory to recycle the server1 process on a Linux platform:

```
stopServer.sh server1  
startNode.sh
```

```
startServer.sh server1
```

- i. Optional: Uninstall manually before reinstalling. This step is required before you can reinstall the V5.0.x product into the same installation root directory. Otherwise you must reinstall the product into another directory.
6. Optional: Uninstall the V5.1 node.
 - a. Change directories to the V5.0.2 *install_root50/bin* directory. For example, issue this command to change directories on a Linux platform:


```
cd /opt/WebSphere/AppServer50/bin
```
 - b. Change directories to the *install_root51/_uninst* directory in the V5.1 installation root before issuing the **uninstall** command. The command file is located in the *install_root51/_uninst* directory on a Linux or UNIX-based platform, and in the *install_root51_uninst* directory on a Windows system.

For example, use this command to change directories on a Linux platform:

```
cd /opt/WebSphere/AppServer51/_uninst
```
 - c. Issue the **uninstall** command. The command file is named `uninstall` for Linux and UNIX-based platforms and `Uninstall.exe` on Windows platforms.

On Windows platforms, call the **Uninstall.exe** command:

```
install_root51\_uninst\Uninstall.exe
```

Call the program directly from the *install_root51_uninst* directory. For example, if the installation root is `C:\WebSphere\AppServer51`, issue the following command:

```
C:\WebSphere\AppServer51\_uninst> Uninstall.exe
```

On Linux and UNIX-based platforms, issue the **uninstall** command from the *install_root/_uninst* directory:

```
uninstall
```
 - d. Use the uninstaller wizard panel to keep the embedded messaging feature code installed. When the V5.1 uninstaller wizard displays a panel that prompts you whether to uninstall the embedded messaging feature, clear the check box to leave the feature installed so that the V5.0.2 node can continue to use the code.
 - e. After the uninstaller removes the V5.1 code, change directories to the *install_root50/bin* directory.
 - f. Start the V5.0.2 deployment manager with the `startManager` command.
 - g. Stop and restart the V5.0.2 Application Server. For example, run these commands from the *install_root50/bin* to recycle the `server1` process and the `nodeagent` process on a Linux platform:


```
stopServer.sh server1
startNode.sh

startServer.sh server1
```
 - h. Optional: Uninstall V5.1 artifacts before reinstalling V5.1. Select a platform-specific procedure for completing the removal of the product:
 - Uninstall manually on all platforms but Solaris and HP-UX before reinstalling.
 - Uninstall manually on Solaris platforms before reinstalling.
 - Uninstall manually on HP-UX platforms before reinstalling.

This step is required before you can reinstall the product into the same installation root directory.

You can uninstall a base node that has the embedded messaging server and client feature.

Uninstalling when the embedded messaging client feature is installed

This topic describes uninstalling when the embedded messaging client feature is installed on more than one Network Deployment or base node on the machine. When there is more than one node on a machine using the client feature, uninstalling a single node can remove the embedded messaging client code so that none of the nodes can access it. This topic describes how to avoid that situation.

This task describes how to uninstall a V5.0.2 node that has the client feature and a V5.1 node that has the feature. You have the option of uninstalling either node. But you do not have to uninstall both nodes.

Select your scenario in the following procedure.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Uninstall the node. Select your scenario from the following list of procedures:
 - Uninstall a V5.0.2 Network Deployment node or a federated base node that has the embedded messaging client feature.
 - Uninstall a V5.1 Network Deployment node or a federated base node that has the embedded messaging client feature.

You can uninstall a node that has the client feature.

Uninstalling V5.0.2 when the embedded messaging client feature is installed:

This task describes how to uninstall the Network Deployment product or a federated base WebSphere Application Server, Version 5.0.2 product when the product has the embedded messaging client feature.

Uninstalling a V5.0.x node that has the embedded messaging client feature affects all other nodes that also use the feature. The uninstaller program on V5.0.x does not give you the option of leaving the client feature installed when you uninstall a product. However, you can rename the *install_root/_uninst/mq* directory to keep the code for the embedded messaging client feature installed when you uninstall the product.

Two alternatives exist for preserving the embedded messaging client code:

- **Renaming the *_uninst/mq* directory before uninstalling the V5.0.2 product**

If you rename the *_uninst/mq* directory before uninstalling the V5.0.2 product, you do not have to reinstall the embedded messaging client feature because you avoid uninstalling the client. Renaming the directory is the recommended method whenever possible.

- **Reinstalling the embedded messaging client code after uninstalling the V5.0.2 product**

If you uninstall V5.0.2 without renaming the *_uninst/mq* directory, you also uninstall the embedded messaging client code. You can recover by reinstalling the embedded messaging client feature. The general procedure is to uninstall V5.0.2 and to reinstall the client feature from the V5.1 product installation image, which is usually the product CD-ROM. You can also reinstall the client feature from the V5.0.2 product installation image, which is usually where you downloaded and unzipped Fix Pack 2. However, you might have a product CD-ROM for V5.0.2 on some platforms.

Before uninstalling, verify that you have no open Web browsers that are accessing the administrative console.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Optional: Rename the *install_root/_uninst/mq* directory to prevent uninstalling the embedded messaging client feature, uninstall the product, and rename the directory to its original name.
 - a. Rename the *install_root/_uninst/mq* directory to prevent uninstalling the embedded messaging client feature.
 - b. Follow the normal procedure for uninstalling the V5.0.2 deployment manager or federated base node:
 - 1) Follow the normal procedure for uninstalling a V5.0.2 deployment manager node.
 - 2) Follow the normal procedure for uninstalling the V5.0.2 federated or unfederated node.
 - c. Rename the *install_root/_uninst/mq* directory to its correct name.

3. Optional: Reinstall the embedded messaging client feature on a Windows platform. You do not have to reinstall the embedded messaging client if you renamed the *install_root_uninst\mq* directory as described in the earlier step.

- a. Locate the V5.1 product CD-ROM.
- b. Identify the original installation location of the embedded messaging client feature on a Windows machine. Use the administrative console of the V5.1 product to examine the value of the *MQ_INSTALL_ROOT* variable in **Environment > Manage WebSphere variables**.
- c. Optional: Issue the command to reinstall the embedded messaging client code on a Windows platform:

```
source_path\wmsetup /installc parm1 parm2 parm3 parm4
```

source_path

The fully qualified file path to the **wmsetup** command, either on disk or in the *\messaging\client* directory on the product CD-ROM.

parm1

The file path where the embedded messaging feature was originally installed, which is by default *C:\Program Files\IBM\WebSphere MQ* on a Windows platform.

parm2

The source location of embedded messaging software, such as the *\messaging\client* directory on the product CD-ROM.

parm3

The path and file name for recording log messages.

parm4

A code that represents the installation language. For example, 1033 for US English.

For example, you can use the following command by entering it all on one line:

```
E:\WASTEST\messaging\client\Setup\wmsetup /installc  
C:\Program Files\IBM\WebSphere MQ  
I:\Windows\messaging\client  
C:\temp\mq_install.log  
1033
```

4. Issue the command to reinstall the embedded messaging client code on a Linux or UNIX-based platform. You do not have to reinstall the embedded messaging client if you renamed the *install_root/_uninst/mq* directory as described in the earlier step. **Command to use on a Linux or UNIX-based platform:**

```
source_path/wmsetup installc logfile
```

source_path

The fully qualified file path to the **wmsetup** command, either on disk or in the */messaging/client* directory on the product CD-ROM.

logfile

The path and file name for recording log messages.

For example:

```
./wmsetup installc /tmp/mq_install.log
```

You can uninstall a V5.0.2 node that uses the embedded messaging client feature without affecting other nodes that also use the client feature.

Uninstalling V5.1 and leaving the embedded messaging client feature installed:

This task describes how to uninstall the Network Deployment product or a federated base WebSphere Application Server, Version 5.1 product when the product has the embedded messaging client feature. The

uninstaller program on V5.1 gives you the option of leaving the embedded messaging client feature installed when the program uninstalls the Network Deployment product or a federated base WebSphere Application Server product.

Uninstalling a V5.1 node that has the embedded messaging client feature can affect all other nodes that use the feature. The uninstaller program on V5.1 gives you an option that you must use to uninstall a V5.1 node that uses the embedded messaging client feature if you do not want to affect other nodes that also use the feature.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Change directories to the `_uninst` directory before issuing the **uninstall** command. The command file is located in the `install_root/_uninst` directory on a Linux or UNIX-based platform, and in the `install_root_uninst` directory on a Windows system.

For example, to change directories on a Linux platform, issue this command if your installation root is `/opt/WebSphere/NetworkDeployment`:

```
cd /opt/WebSphere/NetworkDeployment/_uninst
```

On a Windows system, you can use the Add/Remove Programs feature and bypass this step, if you prefer.

3. Issue the **uninstall** command. The command file is named `uninstall` for Linux and UNIX-based platforms and `Uninstall.exe` on Windows platforms.

On Windows platforms, call the **Uninstall.exe** command:

```
ND_install_root\_uninst\Uninstall.exe
```

Call the program directly from the `install_root_uninst` directory.

For example, if the installation root is `C:\WebSphere\NetworkDeployment`, issue the following command:

```
C:\WebSphere\NetworkDeployment\_uninst> Uninstall.exe
```

On Linux and UNIX-based platforms, issue the **uninstall** command from the `install_root/_uninst` directory:

```
uninstall
```

Uninstalling the product also selects the feature uninstaller programs, such as the uninstaller program for the embedded messaging client feature.

The uninstaller wizard begins and displays a panel.

4. Click **Next** to begin uninstalling the product. In situations where the embedded messaging client feature is installed, the V5.1 uninstaller wizard displays another panel that prompts you whether to uninstall the embedded messaging feature.
5. Select the check box to clear it so that you do not uninstall the embedded messaging client feature. On that panel, click **Next** to continue uninstalling the product. The uninstaller wizard displays a summary panel that lists the product and features that you are uninstalling.
6. Click **Next** to continue uninstalling the product.
7. Click **Finish** to close the wizard after the wizard removes the product.
8. Optional: Uninstall manually before reinstalling. This step is required before you can reinstall the product into the same installation root directory:
 - Uninstall manually on all platforms but Solaris and HP-UX before reinstalling.
 - Uninstall manually on Solaris platforms before reinstalling.
 - Uninstall manually on HP-UX platforms before reinstalling.

You can uninstall the V5.1 Network Deployment product or a federated base WebSphere Application Server product and leave the embedded messaging client feature installed for other product instances to use.

Uninstalling in a coexistence scenario

This task describes how to uninstall a product in a coexistence scenario, when various releases of IBM WebSphere Application Server products might be on the same machine, each running in a uniquely named node. You can use the uninstaller program to uninstall each product.

Several scenarios can occur when uninstalling coexisting products. All of them have to do with whether the product has the embedded messaging feature installed. Steps in this task describe various scenarios you can follow:

- Uninstalling a node that does not have the embedded messaging feature
- Uninstalling a base node that has the embedded messaging server and client feature
- Uninstalling a node that has the embedded messaging client feature

Select one of the following optional steps that describes your scenario.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Uninstall a Network Deployment node that does not have the embedded messaging client feature installed.
3. Uninstall a federated base WebSphere Application Server node that does not have an embedded messaging feature installed.
4. Uninstall a federated base WebSphere Application Server node that has the embedded messaging server and client feature installed.
5. Uninstall a Network Deployment or a federated base node that has the embedded messaging client feature installed.

You can uninstall a WebSphere Application Server product in a coexistence scenario.

Uninstalling manually

This task describes how to remove all artifacts of an instance of the product so that you have a clean machine with regard to the instance. Repeat the procedure for each instance of the product that you intend to remove.

Uninstalling manually occurs after running the uninstaller program, or because you cannot run the uninstaller program because the program might affect federated nodes, coexisting products, or features. The following for manually uninstalling the product are specific to the platform on which the product is installed.

Before uninstalling, verify that you have no open Web browsers that are accessing the administrative console. Otherwise, there is the potential for locked file errors as you uninstall.

Even if you run the uninstaller program to remove registry entries, uninstall the product code, and remove all related features and products, such as plug-ins, there are some files that the uninstaller program does not remove. The uninstaller program does not delete any configuration files that are changed as the result of selecting installation options, or running Samples, for example. The uninstaller program does not delete log files. The uninstaller program for the base product and the Network Deployment product also leaves a readme file in the `_uninst` directory after it runs, to describe how to completely uninstall the product.

1. Stop any embedded messaging feature services, such as WebSphere Embedded Messaging Publish And Subscribe, `jmsserver`, or WebSphere MQ queue managers, and any related Java processes.
2. Stop the IBM HTTP Server and any related Java processes.
3. Stop any WebSphere Application Server Java processes with the **stopManager** or **stopServer** commands.

4. Back up all configuration and log files. The uninstaller program leaves some log and configuration files that were changed during installation. Before manually uninstalling the product, copy all configuration and log files that are left behind in the WebSphere, the IBMHTTPServer, and the IBM/MQSeries directories. You can refer to your copy of the configuration and log files after you reinstall to compare the configuration data, if necessary.
5. Use one of the following platform-specific uninstalling procedures to delete the directories that are left behind for your product instance, and to perform other tasks that let you reinstall the product on your operating system. You cannot reinstall into the old directories. Deleting the directories and registry entries lets you reinstall with a clean system. If you do plan to reinstall the product, it is important that you follow the procedure for your platform to completely remove all artifacts of the previous installation. The following topics describe how to uninstall the product from supported operating systems. The IBM WebSphere Application Server supported hardware, software, and APIs Web site describes which operating systems are supported.
 - Manually uninstalling on AIX platforms
 - Manually uninstalling on HP-UX platforms
 - Manually uninstalling on Linux
 - Manually uninstalling on Solaris
 - Manually uninstalling on Windows

You can completely uninstall the product after which you can reinstall.

Manually uninstalling on AIX platforms

This task describes how to uninstall the product on an AIX platform.

Use the following procedure to remove all remnants of a WebSphere Application Server product. This produces a clean system that allows you to reinstall the product. Verify that you are removing only the product instance that you intend to remove and not coexisting products, or products that you might have migrated and do not intend to delete. It is important to know the installation root directory for the product instance that you are removing.

Uninstall WebSphere Application Server products in the following sequence when you are uninstalling more than one product:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment
2. WebSphere Application Server Network Deployment
3. **5.1+** WebSphere Business Integration Server Foundation on the base product
4. WebSphere Application Server (base product)

In the following steps:

- The default *install_root* for WebSphere Application Server is the `/usr/WebSphere/AppServer` directory.
 - The default *install_root* for Network Deployment is the `/usr/WebSphere/DeploymentManager` directory.
1. Use the **killall -9 java** command to kill all Java processes that are running.

If there are Java processes running that are not related to WebSphere Application Server products, and it is not possible to stop them, stop all WebSphere Application Server product-related processes. Use the following command to determine all processes that are running:

```
ps -ef | grep java
```

Stop all WebSphere Application Server-related processes with the **kill -9 java_pid_1 java_pid_2...java_pid_n** command.
 2. Halt any running MQ queue managers.
 - a. Type **dspmq** to show the state of any queue managers.
 - b. Type **endmqm -i** for each running queue manager.
 - c. Type **\$ ipcs -a** to check for any IPCs.
 - d. Type **\$ ipcrm -[qms] [ID]** to delete the IPCs.

3. Type `kill -9 amq_pid_1 amq_pid_2 ... amq_pid_n` to stop any MQ processes that are running. You can use a `ps -e | grep #AMQ_COMMAND#` command structure to specify the command for amq to find each amq pid.

4. Search for related packages. Either use `smit` to remove packages, or search for and remove packages manually:

- Type `smit` to clean up remnants.
 - a. Click **Software Installation and Maintenance**.
 - b. Click **Software Maintenance and Utilities**.
 - c. Click **Remove Installed Software**.
 - d. Click **LIST** by Software name.
 - e. Search for packages that contain the words **IBM** or **WS** to identify all WebSphere Application Server product-related packages, including those that belong to **IBM HTTP Server** and the embedded messaging feature, which is based on **IBM WebSphere MQ** technology.
 - f. Change the **PREVIEW ONLY** option to **NO**.
 - g. Click **OK**.
- Manually search for, and remove packages that are only in the instance that you intend to delete.

Do not remove IBM WebSphere MQ packages if you have installed IBM WebSphere MQ as a separate product.

Do not remove embedded messaging packages if you intend to leave the embedded messaging feature installed on the machine.

If you do intend to remove WebSphere MQ or embedded messaging packages, include them in the commands you are running to remove packages. For example, type these commands to search for, and remove related packages:

a. Search for related packages:

- 1) `ls1pp -l | grep WS` to show packages for WebSphere Application Server products and the IBM HTTP Server product
- 2) `ls1pp -l | grep wemps` to show packages related to the embedded messaging feature or to IBM WebSphere MQ.
- 3) `ls1pp -l | grep mqjava` to show packages related to the embedded messaging feature or to IBM WebSphere MQ.
- 4) `ls1pp -l | grep mqm` to show packages related to the embedded messaging feature or to IBM WebSphere MQ.

You can also run the following command from root to remove the embedded messaging feature from your system:

```
installp -u wemps mqjava mqm
```

Reply `y[es]` to all prompts.

However, do not remove any `mqm` or `mqjava` packages if you have installed IBM WebSphere MQ separately or if you have other WebSphere Application Server products that use the embedded messaging feature.

The examples below show typical package names that might appear on a system with the base WebSphere Application Server product.

If no packages appear when using these commands, skip the next step.

b. Type `smit remove packagename1 packagename2 packagename3 ...` to remove any WebSphere Application Server-related packages. Do not remove packages for any products that you intend to keep. You can identify packages that belong to V5.1 by the 51 on the end of `WSB` packages and `WSN` packages. `WSBM` and `WSNM` packages are for the embedded messaging feature.

Do not remove IBM WebSphere MQ packages if you have installed IBM WebSphere MQ as a separate product, or if you have installed the embedded messaging feature and other product instances are using the feature.

Do not remove IBM `WEMPS` packages unless you are certain that there is no embedded messaging data to preserve.

5. Change the directory to the `usr` directory.

6. Type `rm -rf WebSphere` to delete this WebSphere Application Server-related directory, if the only subdirectories are the AppServer and DeploymentManager directories and if the only product instances they contain are those that you intend to delete.
7. Change directory to the `usr/opt` directory.
8. Type `rm -rf wemps` to delete the embedded messaging feature directory if you are certain that there is no embedded messaging data to preserve. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.
9. Change the directory to the `/usr` directory.
10. Type `rm -rf IBMHttpServer` to delete the IBM HTTP Server directory.
11. If you do not have IBM WebSphere MQ installed as a separate product on this machine, type `rm -rf mqm` to delete the embedded messaging feature directory if you are certain that there is no embedded messaging data to preserve. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.

If you installed IBM WebSphere MQ as a separate product on this host to use as the messaging provider, and if you do not want to continue using WebSphere MQ, you can uninstall the product as described in the WebSphere MQ information.

12. Edit the `vpd.properties` file, as described in “vpd.properties file” on page 278. Certain situations require you to edit the `vpd.properties` file before reinstalling a WebSphere Application Server product. Edit the `vpd.properperties` file to remove entries for the WebSphere Application Server product that you are uninstalling and to remove entries for any of its features.

Do not delete or rename the `vpd.properties` file because the InstallShield for MultiPlatforms (ISMP) program uses it for other products that it installs. If the `vpd.properties` file has no other entries, you can delete the file.

When the process completes, the product is completely uninstalled. You are now ready to reinstall.

Example of displaying package names beginning with mqm, for the embedded messaging feature

```
==>ls1pp -l | grep mqm
```

```
mqm.base.runtime      5.3.0.1 COMMITTED WebSphere MQ Runtime for
mqm.base.sdk          5.3.0.1 COMMITTED WebSphere MQ Base Kit for
mqm.client.rte        5.3.0.1 COMMITTED WebSphere MQ Client for AIX
mqm.java.rte          5.3.0.1 COMMITTED WebSphere MQ Java Client and
mqm.msg.De_DE          5.3.0.1 COMMITTED WebSphere MQ Messages - German
mqm.msg.Es_ES          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.Fr_FR          5.3.0.1 COMMITTED WebSphere MQ Messages - French
mqm.msg.It_IT          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.Ja_JP          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.Zh_CN          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.Zh_TW          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.de_DE          5.3.0.1 COMMITTED WebSphere MQ Messages - German
mqm.msg.en_US          5.3.0.1 COMMITTED WebSphere MQ Messages - U.S.
mqm.msg.es_ES          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.fr_FR          5.3.0.1 COMMITTED WebSphere MQ Messages - French
mqm.msg.it_IT          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.ja_JP          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.ko_KR          5.3.0.1 COMMITTED WebSphere MQ Messages - Korean
mqm.msg.pt_BR          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.zh_CN          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.msg.zh_TW          5.3.0.1 COMMITTED WebSphere MQ Messages -
mqm.server.rte        5.3.0.1 COMMITTED WebSphere MQ Server
```

Example of displaying package names beginning with wemps, for the embedded messaging feature

```
==>ls1pp -l | grep wemps
```

```
wemps.base.runtime    2.1.0.0 COMMITTED WebSphere Embedded Messaging
```

Example of displaying package names beginning with WS, for WebSphere Application Server-related products

```
==>ls1pp -l | grep WS
```

WSBAA	5.0.0.0	COMMITTED	ISMP installed entry
WSBAAAA	5.0.0.0	COMMITTED	Installs tools for assembly
WSBAC1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBACAA	5.0.0.0	COMMITTED	Includes adminconsole.ear, the
WSBADAA	5.0.0.0	COMMITTED	Installs the Administrative
WSBAS1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBASAA	5.0.0.0	COMMITTED	Includes wsadmin, the
WSBAT1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBATAA	5.0.0.0	COMMITTED	Includes a GUI-based tool for
WSBAU1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBAUAA	5.0.0.0	COMMITTED	Includes Apache ANT, a
WSBCO1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBCO4AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBCO5AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBCOAA	5.0.0.0	COMMITTED	ISMP installed entry
WSBDT1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBDTAA	5.0.0.0	COMMITTED	Includes a command-line
WSBES1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBES3AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBESAA	5.0.0.0	COMMITTED	Includes samples for
WSBGK2AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBGKAA	5.0.0.0	COMMITTED	ISMP installed entry
WSBIHAA	1.3.28.0	COMMITTED	Installs an Apache-powered Web
WSBIHAB	1.3.28.0	COMMITTED	ISMP installed entry
WSBJA1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBJAAA	5.0.0.0	COMMITTED	Installs WebSphere public
WSBJD3AA	1.3.1.0	COMMITTED	ISMP installed entry
WSBJD7AA	1.3.1.0	COMMITTED	ISMP installed entry
WSBJD9AA	1.3.1.0	COMMITTED	ISMP installed entry
WSBJDAA	1.3.1.0	COMMITTED	ISMP installed entry
WSBLA1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBLAAA	5.0.0.0	COMMITTED	Includes a graphical utility
WSBMQ1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBMQ2AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBMQ3AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBMQAA	5.0.0.0	COMMITTED	Installs Java Messaging
WSBMS2AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBMS4AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBMS5AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBMSAA	5.0.0.0	COMMITTED	Includes JMS
WSBPL1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBPLAA	5.0.0.0	COMMITTED	Installs plugins to configure
WSBPTAA	5.0.0.0	COMMITTED	Installs tools for performance
WSBSM1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBSMAA	5.0.0.0	COMMITTED	Includes samples, including
WSBSR1AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBSR5AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBSR6AA	5.0.0.0	COMMITTED	ISMP installed entry
WSBSRAA	5.0.0.0	COMMITTED	Installs the application

Manually uninstalling on HP-UX platforms

This task describes how to uninstall the product on an HP-UX platform.

Use the following procedure to remove all remnants of a WebSphere Application Server product. This produces a clean system that allows you to reinstall the product. Verify that you are removing only the product instance that you intend to remove and not coexisting products, or products that you might have migrated and do not intend to delete. It is important to know the installation root directory for the product instance that you are removing.

Uninstall WebSphere Application Server products in the following sequence when you are uninstalling more than one product:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment
2. WebSphere Application Server Network Deployment
3. **5.1+** WebSphere Business Integration Server Foundation on the base product
4. WebSphere Application Server (base product)

In the following steps:

- The default *install_root* for WebSphere Application Server is the /opt/WebSphere/AppServer directory.
- The default *install_root* for Network Deployment is the /opt/WebSphere/DeploymentManager directory.
- The default directory for IBM HTTP Server is /opt/IBMHttpServer.
- The default directories for the embedded messaging feature, which is based on WebSphere MQ, are /opt/wemps and /opt/mqm.

1. Use the **killall -9 java** command to kill all Java processes that are running.

If there are Java processes running that are not related to WebSphere Application Server products, and it is not possible to stop them, stop all WebSphere Application Server product-related processes. Use the following command to determine all processes that are running:

```
ps -ef | grep java
```

Stop all WebSphere Application Server-related processes with the **kill -9 java_pid_1 java_pid_2...java_pid_n** command.

2. Halt any running WebSphere MQ queue managers.
 - a. Type **dspmq** to show the state of any queue managers.
 - b. Type **endmqm -i** for each running queue manager.
 - c. Type **\$ ipcs -a** to check for any IPCs.
 - d. Type **\$ ipcrm -[qms] [ID]** to delete the IPCs.
3. Search for mqm processes by typing: **ps -eaf | grep mqm** or **ps -eaf | grep ^MQ***. Type **kill -9 amq_pid_1 amq_pid_2 ... amq_pid_n** to stop any MQ processes that are running. Do not remove IBM WebSphere MQ packages if you have installed IBM WebSphere MQ as a separate product. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories. You can use a **ps -e | grep #AMQ_COMMAND#** command structure to specify the command for amq to find each amq pid.
4. Search for other related packages. Do not remove IBM WebSphere MQ packages if you have installed IBM WebSphere MQ as a separate product. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, which is based on WebSphere MQ, do not remove the WebSphere MQ packages or directories.

If you do intend to remove embedded messaging feature packages, use HP-UX System Administration Manager (SAM) to remove packages for the instance that you intend to delete. Then search for the packages to verify their removal.

- a. Start the SAM utility and verify that your DISPLAY and TERM environment variables are set properly.
 - 1) Click **Software management**.
 - 2) Click **View installed software**.
 - 3) Look for these two entries in the SD list:
 - MQSERIES -> B.11.530.01 WebSphere MQ for HP-UX
 - MQSERIES -> B.11.530.03 WebSphere MQ Update (U485562) for HP-UX
 - MQSERIES -> B.11.530.04 WebSphere MQ Update (U486875) for HP-UX
 - 4) Close the SD list.
 - 5) Click **Remove local host software**.
 - 6) On the **SD Remove list**, remove these instances, if they exist:
 - MQSERIES -> B.11.530.01 WebSphere MQ for HP-UX
 - MQSERIES -> B.11.530.03 WebSphere MQ Update (U485562) for HP-UX

- MQSERIES -> B.11.530.04 WebSphere MQ Update (U486875) for HP-UX
 - a) Highlight the MQSERIES instances on the SD List.
 - b) Click **Actions > Remove...**
 - c) On **Remove Analysis** dialog, verify that **Status** is **ready**.
 - d) Click **OK**.
 - e) Check the result message displayed on the dialog.
 - f) If the removal fails, follow the instructions that appear.
 - 7) Return to the **SD Remove List**.
 - 8) Click **IBM HTTP Server, MQSERIES, WEMPS, WSBA, WSNA, gsk7bas**. You can identify packages that belong to V5.1 by the 51 on the end of WSB packages and WSN packages. WSBM and WSNM packages are for the embedded messaging feature.
 - 9) Click **Actions > Mark for remove**.
 - 10) Click **Actions > Remove**.
 - 11) Click **OK** in the **Remove analysis** dialog box.
 - 12) Click **Logs** to display real-time removal of selected packages.
 - 13) Click **Done** when all packages are removed.
 - 14) Exit SAM.
- b. Verify that you removed the packages that you intend to remove. Type these commands to search for related packages:
- 1) `swlist | grep WS` to show packages for WebSphere Application Server products and the IBM HTTP Server product.
 - 2) `swlist | grep WEMPS` to show packages for the embedded messaging feature or for the IBM WebSphere MQ product. You can delete these packages. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.
 - 3) `swlist | grep MQ` to show packages for the embedded messaging feature or for the IBM WebSphere MQ product.
5. Type `rm -rf /opt/WebSphere/AppServer/` to remove WebSphere Application Server directories in the instance installed in the `/opt/WebSphere/AppServer` installation root. Do not remove installation root directories for products that you intend to keep.
 6. Type `rm -rf /opt/WebSphere/AppServer/ /opt/WebSphere/DeploymentManager/` to remove multiple directories. Again, do not remove directories for products that you intend to keep.
 7. Type `rm -fr /var/wemps /opt/wemps` if you are certain that there is no embedded messaging data to preserve. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.
 8. Type `rm -rf IBMHttpServer` to delete the IBM HTTP Server directory.
 9. If you do not have IBM WebSphere MQ installed as a separate product on this machine, and if you do not have the embedded messaging server and client feature installed on this machine, type `rm -fr /var/mqm /opt/mqm` if you are certain that there is no embedded messaging data to preserve. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.
- If you installed IBM WebSphere MQ as a separate product on this host to use as the messaging provider, and if you do not want to continue using WebSphere MQ, you can uninstall the product as described in the WebSphere MQ information.

When the process completes, the product is completely uninstalled. You are now ready to reinstall.

Manually uninstalling on Linux platforms

This task describes uninstalling the product on Linux platforms.

Use the following procedure to remove all remnants of a WebSphere Application Server product. This produces a clean system that allows you to reinstall the product. Verify that you are removing only the

product instance that you intend to remove and not coexisting products, or products that you might have migrated and do not intend to delete. It is important to know the installation root directory for the product instance that you are removing.

Uninstall WebSphere Application Server products in the following sequence when you are uninstalling more than one product:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment
2. WebSphere Application Server Network Deployment
3. **5.1+** WebSphere Business Integration Server Foundation on the base product
4. WebSphere Application Server (base product)

In the following steps:

- The default *install_root* for WebSphere Application Server is the `/opt/WebSphere/AppServer` directory.
- The default *install_root* for the Network Deployment program is the `/opt/WebSphere/DeploymentManager` directory.
- The default directory for IBM HTTP Server is `/opt/IBMHttpServer`.
- The default directories for the embedded messaging feature, which is based on WebSphere MQ, are `/opt/wemps` and `/opt/mqm`.

1. Use the **killall -9 java** command to kill all Java processes that are running.

If there are Java processes running that are not related to WebSphere Application Server products, and it is not possible to stop them, stop all WebSphere Application Server product-related processes. Use the following command to determine all processes that are running:

```
ps -ef | grep java
```

Stop all WebSphere Application Server-related processes with the **kill -9 java_pid_1 java_pid_2...java_pid_n** command.

2. Halt any running WebSphere MQ queue managers.
 - a. Type **dspmq** to show the state of any queue managers.
 - b. Type **endmqm -i** for each running queue manager.
 - c. Type **\$ ipcs -a** to check for any IPCs.
 - d. Type **\$ ipcrm -[qms] [ID]** to delete the IPCs.
3. Type **kill -9 amq_pid_1 amq_pid_2 ... amq_pid_n** to stop any MQ processes that are running. You can use a `ps -e | grep #AMQ_COMMAND#` command structure to specify the command for amq to find each amq pid.
4. Search for related packages. Type these commands to search for related packages:
 - **rpm -qa | grep WS** to show packages for the WebSphere Application Server products and the IBM HTTP Server product
 - **rpm -qa | grep MQ** to show packages related to the embedded messaging feature or to IBM WebSphere MQ
 - **rpm -qa | grep wemps** to show packages related to the embedded messaging feature or to IBM WebSphere MQ

Do not remove packages for WebSphere Application Server products that you are not uninstalling. You can identify packages that belong to V5.1 by the 51 on the end of WSB packages and WSN packages. WSBM and WSNM packages are for the embedded messaging feature.

Do not remove IBM WebSphere MQ packages if you have installed IBM WebSphere MQ as a separate product.

Do not remove embedded messaging packages if you intend to leave the embedded messaging feature installed on the machine.

5. If there are packages to delete, type **rpm -e packagename** to remove any WebSphere Application Server-related packages for the product you are uninstalling.

You can remove more than one package with the following command:

```
rpm -qa | grep package_prefix | xargs rpm -e
```

However you should verify that everything in the list is something you want to remove before adding | xargs rpm -e to the command.

If you do intend to remove WebSphere MQ or embedded messaging packages, include them in the commands you are running to remove packages. For example, run these commands as root:

```
rpm -e MQSeriesClient-5.3.0-1
rpm -e MQSeriesMsg_Zh_CN-5.3.0-1
rpm -e MQSeriesMsg_Zh_TW-5.3.0-1
rpm -e MQSeriesMsg_de-5.3.0-1
rpm -e MQSeriesMsg_es-5.3.0-1
rpm -e MQSeriesMsg_fr-5.3.0-1
rpm -e MQSeriesMsg_it-5.3.0-1
rpm -e MQSeriesMsg_ja-5.3.0-1
rpm -e MQSeriesMsg_ko-5.3.0-1
rpm -e MQSeriesMsg_pt-5.3.0-1
rpm -e MQSeriesRuntime-5.3.0-1
rpm -e MQSeriesSDK-5.3.0-1
rpm -e MQSeriesJava-5.3.0-1
rpm -e MQSeriesServer-5.3.0-1
rpm -e MQSeriesJava-5.3.0-1
rpm -e wemps-runtime-2.1.0-0
rpm -e wemps-msg-De_DE-2.1.0-0
rpm -e wemps-msg-Es_ES-2.1.0-0
rpm -e wemps-msg-Fr_FR-2.1.0-0
rpm -e wemps-msg-It_IT-2.1.0-0
rpm -e wemps-msg-Ja_JP-2.1.0-0
rpm -e wemps-msg-Ko_KR-2.1.0-0
rpm -e wemps-msg-Pt_BR-2.1.0-0
rpm -e wemps-msg-Zh_CN-2.1.0-0
rpm -e wemps-msg-Zh_TW-2.1.0-0
```

If there is a problem with package dependencies, you can use the following command to remove the packages:

```
rpm -e packagename --nodeps --justdb
```

The nodeps option skips the dependency check. The justdb option updates only the package database, and not the file system. Using only the nodeps option can cause a failure in package removal if there is any mismatch in the dependent file system (files and directories).

6. Type **rm -rf /opt/WebSphere/AppServer/** to remove WebSphere Application Server directories in the instance installed in the /opt/WebSphere/AppServer installation root. Do not remove installation root directories for products that you intend to keep.
7. Type **rm -rf /opt/WebSphere/AppServer/ /opt/WebSphere/DeploymentManager/** to remove multiple directories. Again, do not remove directories for products that you intend to keep.
8. Type **rm -fr /var/wemps /opt/wemps** if you are certain that there is no embedded messaging data to preserve. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.
9. Type **rm -rf IBMHttpServer** to delete the IBM HTTP Server directory.
10. If you do not have IBM WebSphere MQ installed as a separate product on this machine, and if you do not have the embedded messaging server and client feature installed on this machine, type **rm -fr /var/mqm /opt/mqm** if you are certain that there is no embedded messaging data to preserve. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.

If you installed IBM WebSphere MQ as a separate product on this host to use as the messaging provider, and if you do not want to continue using WebSphere MQ, you can uninstall the product as described in the WebSphere MQ information.

11. Edit the vpd.properties file.

Do not delete or rename the vpd.properties file because the InstallShield for MultiPlatforms (ISMP) program uses it for other products that it installs.

When the process completes, the product is completely uninstalled. You are now ready to reinstall.

Example of displaying package names beginning with MQ, for the embedded messaging feature

```
==>rpm -qa | grep MQ
MQSeriesMsg_Zh_CN-5.3.0-1
MQSeriesMsg_Zh_TW-5.3.0-1
MQSeriesMsg_ko-5.3.0-1
MQSeriesClient-5.3.0-1
MQSeriesMsg_de-5.3.0-1
MQSeriesMsg_es-5.3.0-1
MQSeriesMsg_fr-5.3.0-1
WSBMQ1AA-5.0-0
WSBMQ2AA-5.0-0
WSBMQ3AA-5.0-0
MQSeriesMsg_it-5.3.0-1
MQSeriesMsg_ja-5.3.0-1
MQSeriesMsg_pt-5.3.0-1
MQSeriesSDK-5.3.0-1
MQSeriesJava-5.3.0-1
MQSeriesServer-5.3.0-1
MQSeriesRuntime-5.3.0-1
```

Example of displaying package names beginning with wemps, for the embedded messaging feature

```
==>rpm -qa | grep wemps
wemps-msg-De_DE-2.1.0-0
wemps-msg-Es_ES-2.1.0-0
wemps-msg-Fr_FR-2.1.0-0
wemps-msg-It_IT-2.1.0-0
wemps-msg-Ja_JP-2.1.0-0
wemps-msg-Ko_KR-2.1.0-0
wemps-msg-Pt_BR-2.1.0-0
wemps-msg-Zh_CN-2.1.0-0
wemps-msg-Zh_TW-2.1.0-0
wemps-runtime-2.1.0-0
```

Example of displaying package names beginning with WSB, for the base WebSphere Application Server product

```
# rpm -qa | grep WSB
WSBSR1AA51-5.1-0
WSBMQ4AA-5.0-0
WSBTV1AA51-5.1-0
WSBJD5AA51-1.3-1
WSBDM1AA51-5.1-0
WSBJD7AA51-1.3-1
WSBDT1AA51-5.1-0
WSBMS3AA-5.0-0
WSBAU1AA51-5.1-0
WSBMQ3AA-5.0-0
WSBPL1AA51-5.1-0
WSBMS6AA-5.0-0
WSBPS1AA51-5.1-0
WSBC05AA51-5.1-0
WSBSR4AA51-5.1-0
WSBC01AA51-5.1-0
WSBSM1AA51-5.1-0
WSBAC1AA51-5.1-0
WSBMQ1AA-5.0-0
WSBJD9AA51-1.3-1
WSBGK2AA51-5.1-0
WSBES1AA-5.0-0
WSBJA1AA51-5.1-0
```

```
WSBLA1AA51-5.1-0
WSBAS1AA51-5.1-0
WSBMQ2AA-5.0-0
WSBIH1AA51-1.3-28
```

Example of displaying package names beginning with WSN, for the WebSphere Application Server Network Deployment product

```
# rpm -qa | grep WSN
WSNC02AA51-5.1-0
WSNWG1AA51-5.1-0
WSNJD5AA51-1.3-1
WSNCF1AA51-5.1-0
WSNUR1AA51-5.1-0
WSNJD9AA51-1.3-1
WSNMQ1AA51-5.1-0
WSNCF4AA51-5.1-0
WSNMC5AA-5.0-0
WSNSR1AA51-5.1-0
WSNC01AA51-5.1-0
```

Manually uninstalling on Solaris platforms

This task describes uninstalling the product on the Solaris operating platform.

Use the following procedure to remove all remnants of a WebSphere Application Server product. This produces a clean system that allows you to reinstall the product. Verify that you are removing only the product instance that you intend to remove and not coexisting products, or products that you might have migrated and do not intend to delete. It is important to know the installation root directory for the product instance that you are removing.

Uninstall WebSphere Application Server products in the following sequence when you are uninstalling more than one product:

1. **5.1 +** WebSphere Business Integration Server Foundation on Network Deployment
2. WebSphere Application Server Network Deployment
3. **5.1 +** WebSphere Business Integration Server Foundation on the base product
4. WebSphere Application Server (base product)

In the following steps:

- The default *WAS_install_root* for WebSphere Application Server is the `/opt/WebSphere/AppServer` directory.
 - The default *install_root* for Network Deployment is the `/opt/WebSphere/DeploymentManager` directory.
 - The default directory for IBM HTTP Server is `/opt/IBMHttpServer`.
1. Use the **killall -9 java** command to kill all Java processes that are running.
If there are Java processes running that are not related to WebSphere Application Server products, and it is not possible to stop them, stop all WebSphere Application Server product-related processes. Use the following command to determine all processes that are running:

```
ps -ef | grep java
```

Stop all WebSphere Application Server-related processes with the **kill -9 java_pid_1 java_pid_2...java_pid_n** command.
 2. Halt any running WebSphere MQ queue managers.
 - a. Type **dspmq** to show the state of any queue managers.
 - b. Type **endmqm -i** for each running queue manager.
 - c. Type **\$ ipcs -a** to check for any IPCs.
 - d. Type **\$ ipcrm -[qms] [ID]** to delete the IPCs.
 3. Issue the following command to stop any MQ processes that are running.

```
kill -9 amq_pid_1 amq_pid_2 ... amq_pid_n
```

4. Search for related packages. Type the following commands from any directory to search for related packages:

- `pkginfo | grep WSB` to show packages for WebSphere Application Server products and the IBM HTTP Server product
- `pkginfo | grep wemps` to show packages related to the embedded messaging feature or to IBM WebSphere MQ. You can delete these packages.
- `pkginfo | grep mqm` to show packages related to the embedded messaging feature or to IBM WebSphere MQ.

Do not remove IBM WebSphere MQ packages if you have installed IBM WebSphere MQ as a separate product.

If no packages appear when using these commands, skip the next step.

5. Change directories to the directory where package information is registered.

```
cd /var/sadm/pkg
```

6. Issue the following command to remove any WebSphere Application Server-related packages.

Do not remove packages for WebSphere Application Server products that you are not uninstalling. You can identify packages that belong to V5.1 by the 51 on the end of WSB packages and WSN packages. WSBM and WSNM packages are for the embedded messaging feature.

```
pkgrm packagename1 packagename2 packagename3 ...
```

Do not remove embedded messaging packages if you intend to leave the embedded messaging feature installed on the machine.

Do not remove `mqjava`, `mqm-upd04`, or `mqm` packages if you have installed IBM WebSphere MQ as a separate product. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, which is based on WebSphere MQ, do not remove the embedded messaging or `mqjava`, `mqm-upd04`, or `mqm` packages or directories.

If you do not have IBM WebSphere MQ installed as a separate product on this machine, and if you do not have the embedded messaging server and client feature installed on this machine, you can remove `wemps` packages.

Remove the `mqm` package last, or you cannot remove any other embedded messaging feature packages. You will have to reinstall the embedded messaging feature to remove the packages. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.

If you do intend to remove WebSphere MQ or embedded messaging packages, include the packages in the commands you are running to remove packages. For example, issue the following command as root to remove the embedded messaging feature:

```
pkgrm wemps mqjava mqm-upd04 mqm
```

Reply `y[es]` to all prompts.

Alternatively, you can be in any directory and use the **pkginfo** command to remove any package, such as those that begin with `wemps`:

```
pkginfo | grep wemps | awk '{print $2}' | xargs -i pkgrm -n {}
```

Issue the following commands from the `/var/sadm/pkg` directory to search for and remove any WebSphere Application Server product-related packages that are registered in the `/var/sadm/pkg` directory:

- a. Verify that you are in the correct directory: `cd /var/sadm/pkg`
- b. `ls |grep WSB|xargs -i pkgrm -n {}` for the base WebSphere Application Server product
- c. `ls |grep WSN|xargs -i pkgrm -n {}` for the Network Deployment product
- d. `ls |grep WSC|xargs -i pkgrm -n {}` for the WebSphere Application Server Java client
- e. `ls |grep wemps|xargs -i pkgrm -n {}` for the WebSphere embedded messaging publish and subscribe function
- f. `ls |grep mqjava|xargs -i pkgrm -n {}` for IBM WebSphere MQ
- g. `ls |grep mqm|xargs -i pkgrm -n {}` for IBM WebSphere MQ

A known problem in ISMP causes ISMP to rename packages beginning with **WS** to **ISWS**. Search for and remove ISWS packages as well:

- a. Verify that you are in the correct directory: `cd /var/sadm/pkg`
- b. `ls |grep ISWSB|xargs -i pkgrm -n {}` for the base WebSphere Application Server product
- c. `ls |grep ISWSN|xargs -i pkgrm -n {}` for the Network Deployment product
- d. `ls |grep ISWSC|xargs -i pkgrm -n {}` for the WebSphere Application Server Java client
- e. `ls |grep wemps|xargs -i pkgrm -n {}` for the WebSphere embedded messaging publish and subscribe function
- f. `ls |grep mqjava|xargs -i pkgrm -n {}` for IBM WebSphere MQ
- g. `ls |grep mqm|xargs -i pkgrm -n {}` for IBM WebSphere MQ

If there is a problem removing the packages, remove the related package directories in the `/var/sadm/pkg` directory.

7. Issue the following command to remove WebSphere Application Server directories in the `/opt/WebSphere/AppServer` installation root.

```
rm -rf /opt/WebSphere/AppServer/
```

Do not remove installation root directories for products that you intend to keep.

8. Issue the following command to remove multiple directories.

```
rm -rf /opt/WebSphere/AppServer/ /opt/WebSphere/DeploymentManager/
```

Again, do not remove directories for products that you intend to keep.

9. Issue the following command if you are certain that there is no embedded messaging data to preserve.

```
rm -fr /var/wemps /opt/wemps
```

If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.

10. Issue the following command to delete the IBM HTTP Server directory.

```
rm -rf IBMHttpServer
```

11. If you do not have IBM WebSphere MQ installed as a separate product on this machine, and if you do not have the embedded messaging server and client feature installed on this machine, issue the following command if you are certain that there is no embedded messaging data to preserve.

If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging packages or directories.

```
rm -fr /var/mqm /opt/mqm
```

If you installed IBM WebSphere MQ as a separate product on this host to use as the messaging provider, and if you do not want to continue using WebSphere MQ, you can uninstall the product as described in the WebSphere MQ information.

When the process completes, the product is completely uninstalled. You are now ready to reinstall.

Example of displaying package names beginning with mqm, for the embedded messaging feature

```
pkginfo |grep mqm
```

```
application mqm          WebSphere MQ for Sun Solaris
```

Example of displaying package names beginning with wemps, for the embedded messaging feature

```
pkginfo |grep wemps
```

```
application wemps          WebSphere Embedded Messaging Publish and Subscribe Edition
```

Example package names for the base package

application ISWSBIH1A	ihsComponent
application ISWSBJD9A	javaCommonConfigComponent
application ISWSBJD9A.2	javaCommonConfigComponent
application ISWSBLA1A	logAnalyzerComponent
application ISWSBLAAA	LogAnalyzer
application WSBA	WebSphere Application Server
application WSBAAAA	Application And Assembly Tools
application WSBA1AA	adminConsoleFilesComponent
application WSBA1AA	Admin Console
application WSBA1AA	Admin
application WSBA1AA	adminScriptingFilesComponent
application WSBA1AA	Admin Scripting
application WSBA1AA	applicationAssemblyToolComponent
application WSBA1AA	Application Assembly Tool
application WSBA1AA	antUtilityComponent
application WSBA1AA	Ant Utility
application WSBA1AA	commonFiles
application WSBA1AA	pbwServerConfigWithMQGood
application WSBA1AA	IsmplauncherComponent
application WSBA1AA	commonFeature
application WSBA1AA	DCMStdComponent
application WSBA1AA	Dynamic Cache Monitor
application WSBA1AA	deployToolComponent
application WSBA1AA	Deploy Tool
application WSBA1AA	messagingSampleFileComponentBean
application WSBA1AA	mqSeriesSamples
application WSBA1AA	gskitUnixComponent
application WSBA1AA	gskit4SolarisComponent
application WSBA1AA	gskitFeature
application WSBA1AA	ihsFeature
application WSBA1AA	javadocComponent
application WSBA1AA	Javadoc
application WSBA1AA	javaSolarisComponent
application WSBA1AA	javaUninstallComponent
application WSBA1AA	Java
application WSBA1AA	mqSeriesSetupFileComponent
application WSBA1AA	mqSeriesBinComponent
application WSBA1AA	mqSeriesLibFilesComponent
application WSBA1AA	MQSeries
application WSBA1AA	mqSeriesSunConfig
application WSBA1AA	mqSeriesServer
application WSBA1AA	component8
application WSBA1AA	Plugins
application WSBA1AA	perfServletComponent
application WSBA1AA	Performance Servlet
application WSBA1AA	PerformanceAndAnalysisTools
application WSBA1AA	samplesComponent
application WSBA1AA	Samples
application WSBA1AA	serverStdComponent
application WSBA1AA	serverConfigWithoutSamplesComponent
application WSBA1AA	Server
application WSBA1AA	tivoliViewerComponent
application WSBA1AA	TivoliPerformanceViewer

Example package names for the Network Deployment package

application ISWSNMQ1A	mqClientFiles
application ISWSNMQAA	mqFeature
application WSNA	WebSphere Network Deployment
application WSNC1AA	configFiles
application WSNC4AA	configSunLinuxWithMQComponent
application WSNCFAA	Config
application WSNC01AA	commonFiles
application WSNC02AA	IsmplauncherComponent
application WSNC0AA	commonFeature
application WSNC4AA	javaSolarisComponent
application WSNCJAA	Java

application WNSR1AA	serverStdComponent
application WNSRAA	Server
application WSNUR1AA	uddiComponent
application WSNURAA	UDDI
application WSNWG1AA	wsgwComponent
application WSNWGAA	Web Services Gateway
application WSNWSAA	Web Services

Example package names for a V5.1 system with the base product and the Network Deployment product

application IBMHTTPSV	IBM HTTP Server
application ISIHS.bas	HTTPServer base
application ISIHS.con	component23
application ISIHS.htt	feature13
application ISIHS.ins	component26
application ISIHS.ssl	component57
application ISWSBIH1A	ihComponent
application ISWSBJD9A	javaCommonConfigComponent
application ISWSBLA1A	logAnalyzerComponent
application ISWSBLAAA	LogAnalyzer
application ISWSNJD9A	javaCommonConfigComponent
application ISWSNMQ1A	mqClientFiles
application ISWSNMQAA	mqFeature
application WSBA51	WebSphere Application Server
application WSBAAA51	Application And Assembly Tools
application WSBAC1AA51	adminConsoleFilesComponent
application WSBACAA51	Admin Console
application WSBADAA51	Admin
application WSBAS1AA51	adminScriptingFilesComponent
application WSBASAA51	Admin Scripting
application WSBAU1AA51	antUtilityComponent
application WSBAUAA51	Ant Utility
application WSBC01AA51	commonFiles
application WSBC05AA51	IsmpLauncherComponent
application WSBC0AA51	commonFeature
application WSBDM1AA51	DCMStdComponent
application WSBDMAA51	Dynamic Cache Monitor
application WSBDT1AA51	deployToolComponent
application WSBDTAA51	Deploy Tool
application WSBE1AA	messagingSampleFileComponentBean
application WSBE5AA	mqSeriesSamples
application WSBGK2AA51	gskitUnixComponent
application WSBGK3AA51	gskit4SolarisComponent
application WSBGKAA51	gskitFeature
application WSBIHAA51	ihFeature
application WSBJA1AA51	javadocComponent
application WSBJAAA51	Javadoc
application WSBJD4AA51	javaSolarisComponent
application WSBJD7AA51	javaUninstallComponent
application WSBJDAA51	Java
application WSBMC4AA	mqSeriesClientUnixInstall
application WSBMQ1AA	mqSeriesSetupFileComponent
application WSBMQ2AA	mqSeriesBinComponent
application WSBMQ3AA	mqSeriesLibFilesComponent
application WSBMQAA	MQSeries
application WSBMS4AA	mqSeriesUnixInstall
application WSBMS6AA	mqSeriesSunConfig
application WSBMSAA	mqSeriesServer
application WSBPL1AA51	component8
application WSBPLAA51	Plugins
application WSBPS1AA51	perfServletComponent
application WSBPSAA51	Performance Servlet
application WSBPTAA51	PerformanceAndAnalysisTools
application WSBSM1AA51	samplesComponent
application WSBSMAA51	Samples

application WSBSR1AA51	serverStdComponent
application WSBSR4AA51	serverConfigWithoutSamplesComponent
application WSBSRAA51	Server
application WSBTV1AA51	tivoliViewerComponent
application WSBTVAA51	TivoliPerformanceViewer
application WSNAA51	WebSphere Network Deployment
application WSNCF1AA51	configFiles
application WSNCF4AA51	configSunLinuxWithMQComponent
application WSNCFAA51	Config
application WSNCO1AA51	commonFiles
application WSNCO2AA51	IsmplauncherComponent
application WSNCOAA51	commonFeature
application WSNJD4AA51	javaSolarisComponent
application WSNJDAA51	Java
application WSNSR1AA51	serverStdComponent
application WSNSRAA51	Server
application WSNUR1AA51	uddiComponent
application WSNURAA51	UDDI
application WSNWG1AA51	wsgwComponent
application WSNWGAA51	Web Services Gateway
application WSNWSAA51	Web Services
application gsk4bas	gsk4bas
application gsk7bas	Certificate and SSL Base Runtime (gsk7bas)
application mqm	WebSphere MQ for Sun Solaris
application mqm-upd04	WebSphere MQ for Sun Solaris 2 - U486877
application wemps	WebSphere Embedded Messaging Publish and Subscribe Edition

Manually uninstalling on Windows platforms

This task describes how to uninstall the product on a Windows platform.

Use the following procedure to remove all remnants of a WebSphere Application Server product. This produces a clean system that allows you to reinstall the product. Verify that you are removing only the product instance that you intend to remove and not coexisting products, or products that you might have migrated and do not intend to delete. It is important to know the installation root directory for the product instance that you are removing.

There are three items you must delete to remove a WebSphere Application Server product or feature: files, registry entries, and the MSI record (when you have installed the embedded messaging feature). If you are uninstalling in a coexistence environment, there are registry entries only for the installation image that you installed last. If you are uninstalling any version but the last image you installed, you need to delete only the file structure for the installation image you are deleting. You do not need to delete registry entries or the MSI record. If you are using the embedded messaging feature in other product instances on the machine, you need to delete only the file structure for the installation image you are deleting. You do not need to delete the MSI record.

Uninstall WebSphere Application Server products in the following sequence when you are uninstalling more than one product:

1. **5.1+** WebSphere Business Integration Server Foundation on Network Deployment
2. WebSphere Application Server Network Deployment
3. **5.1+** WebSphere Business Integration Server Foundation on the base product
4. WebSphere Application Server (base product)

In the following steps:

- The default installation root for WebSphere Application Server is the *drive*:\Program Files\WebSphere\AppServer directory.
- The default installation root for Network Deployment is the *drive*:\Program Files\WebSphere\DeploymentManager directory.

- The default directory for the embedded messaging feature is the *drive:\Program Files\IBM\WebSphere MQ* directory.
 - The default directory for the IBM Key Management utility (for Web server support) is the *drive:\Program Files\IBM\gsk7* directory.
1. Verify that you have an Emergency Recovery Disk. Instructions for creating this disk are in the Windows help documentation.
 2. Use the *regback.exe* program from the Windows Resource Kit to back up the Registry.
 3. If you have not installed IBM WebSphere MQ as a separate product on this machine, and if you do not have other product instances that are using the embedded messaging feature, verify that the embedded messaging feature has been uninstalled. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging registry entries or directories.
 - a. If you are not using the embedded messaging feature or the WebSphere MQ product on other instances on the machine, use the Add/Remove program to verify if the following programs are still listed:
 - IBM WebSphere EMPS
 - IBM WebSphere MQ
 - b. If either of these programs is still listed, complete the following steps: If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not complete these steps to remove the embedded messaging registry entries or directories.
 - 1) Delete any broker or queue manager defined for WebSphere Application Server.
 - To delete any broker defined for WebSphere Application Server, run the **wempsdeletebroker.exe** program, if it still exists:


```
drive:\IBM\WebSphere MQ\WEMPS\bin\wempsdeletebroker.exe brokername -q
```

The *-q* option deletes the Broker's WebSphere MQ queue manager. For example:

```
drive:\IBM\WebSphere MQ\WEMPS\bin\wempsdeletebroker.exe WAS_nodename_server1 -q
```
 - To delete any Queue Manager defined for WebSphere Application Server, run the **dltmqm** command, if it still exists:


```
drive:\IBM\WebSphere MQ\bin\dltmqm.exe queuemanagername
```
 - 2) If you are not using the embedded messaging feature or the WebSphere MQ product on other instances on the machine, use the Add/Remove utility to remove the following programs, if listed:
 - IBM WebSphere EMPS
 - IBM WebSphere MQ

If you installed IBM WebSphere MQ as a separate product, do not remove it now. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging registry entries or directories.
 - c. If you are not using the embedded messaging feature or the WebSphere MQ product on other instances on the machine, remove the WebSphere MQ Windows tray icon if it present.

The MQ Windows tray icon in the lower right corner indicates that a WebSphere MQ process (**amqmtbrn.exe**) is running. Remove the tray icon to close the process.

To remove the tray icon, right click the icon and click **Hide**.
4. Delete WebSphere Application Server product registry entries only if the instance you are removing was the last instance installed. This step is optional in a coexistence environment. Perform this step only if you are removing the image that you last installed.

If you are not using the embedded messaging feature or the WebSphere MQ product on other instances on the machine, remove any WebSphere Application Server product or feature that appears in the Add/Remove Programs panel, which is available from the Control Panel.

5. Invoke **regedit.exe** from a command prompt, to edit the Windows system registry. This step is optional in a coexistence environment. Perform this step only if you are removing the image that you last installed.

Handle the Registry with care

Note: You can easily make a mistake while using the **regedit.exe** editor to view and edit registry contents. The editor does not warn you of editing errors, which can be extremely dangerous. A corrupt Registry can disrupt your system to the point where your only option is to reinstall the Windows operating system.

- a. Use **Ctrl-F** to search for all instances of **WebSphere**, **IBM HTTP Server**, or **IBM MQSeries**, to determine whether you should delete each entry. You might not be able to remove all of the entries related to WebSphere Application Server, which is not a problem. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging registry entries or directories.
- b. Expand and select keys related to **WebSphere Application Server** and its features, **IBM HTTP Server**, and **IBM WebSphere MQ**.

Delete the following keys:

- **5.1+** **HKEY_LOCAL_MACHINE\ SOFTWARE\ IBM\ HTTP Server\ 1.3.28**
- **5.1+** **HKEY_LOCAL_MACHINE\ SOFTWARE\ IBM\ WebSphere Application Server\ 5.1.x.y**, where *x* and *y* identify the specific modification and service levels
- **5.1+** **HKEY_CURRENT_USER\ Software\ Microsoft\ Windows\ CurrentVersion\ Explorer\ MenuOrder\ Start Menu\ Programs\ IBM WebSphere\ Application Server v5.1**
- **5.1+** **HKEY_CURRENT_USER\ HKEY_USERS\ S-1-5-21-3097617641-2550431553-14499587-500\ Software \ Microsoft\ Windows\ CurrentVersion\ Explorer\ MenuOrder\ Start Menu\ Programs\ IBM Websphere\ Application Server v5.1**
- **HKEY_LOCAL_MACHINE\ SOFTWARE\ IBM\ WebSphereEmbeddedMessagingPublishAndSubscribe**

If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove this embedded messaging registry entry.

- **5.1+** **HKEY_LOCAL_MACHINE\ SOFTWARE\ IBM\ WebSphere Network Deployment\ 5.1.x.0**
- **HKEY_LOCAL_MACHINE\ SOFTWARE\ IBM\ WebSphere Network Deployment\ 5.0.x.0**
- If you have installed IBM WebSphere MQ as a separate product on this machine, or if you have other instances of WebSphere Application Server products on the same machine that use the embedded messaging feature, do not remove the following embedded messaging registry entry: **HKEY_LOCAL_MACHINE\ SOFTWARE\ IBM\ MQSeries\ CurrentVersion**

- c. Click **Edit > Delete** from the menu bar for each related key.
 - d. Click **Yes** when asked to confirm deletion of the key.
 - e. Click **Registry > Exit** from the menu bar when you are finished.
6. Restart the Windows platform. Reboot before reinstalling any of the products.
 7. If you have installed IBM WebSphere MQ as a separate product on this machine, or if you have other instances of WebSphere Application Server products on the same machine that use the embedded messaging feature, do not perform this step. If you intend to delete the IBM WebSphere MQ program, or to delete the embedded messaging feature from this machine, use the Microsoft MSI Cleanup Utility to remove MRI records. The utility is available from the Microsoft Web site as **msicuu.exe**. Click on the **msicuu.exe** file after downloading it, to install the utility. Once installed, the utility appears on the program menu.

When MSI starts, it lists all products that it knows about. To uninstall using this technique:

- a. Select a product and click **Remove** to remove its MSI record. These WebSphere MQ product entries might be present:
 - IBM WebSphere EMPS
 - IBM WebSphere MQ

Do not remove the entry for IBM WebSphere MQ if you installed the product separately. If you have other instances of WebSphere Application Server products on the same machine, and if they use the embedded messaging feature, do not remove the embedded messaging registry entries or directories.

8. Delete the Network Deployment installation root directory, *drive:\WebSphere\DeploymentManager*, and all subdirectories if it is the installation root directory for the instance you are uninstalling.
9. Delete the IBM HTTP Server installation root, which is usually in the *drive:\Program Files\IBMHttpServer* directory by default.
10. Delete the *drive:\IBM\WebSphere MQ\WEMPS* directory if there are no other product instances on this machine that are using the embedded messaging feature.
11. Depending on whether you installed IBM WebSphere MQ as a separate product and whether you installed the embedded messaging feature on any other installation instances on the machine, delete the WebSphere MQ directory. If you do not have IBM WebSphere MQ installed as a separate product on this machine and if you do not use the embedded messaging feature on another installation instance on the machine, delete the *drive:\IBM\WebSphere MQ* directory for the embedded messaging feature.

If you installed IBM WebSphere MQ as a separate product on this host to use as the messaging provider, and do not want to continue using WebSphere MQ, you can uninstall the product as described in the WebSphere MQ information.

12. Delete the *drive:\Program Files\IBM\gsk7* directory to remove the IBM Key Management utility.
13. Delete the *drive:\WINDOWS\IsUninst.exe* file.
14. Delete the *drive:\WINDOWS\gsk7bui.exe* file.
15. Edit the *vpd.properties* file.

Do not delete or rename the *vpd.properties* file because the InstallShield for MultiPlatforms (ISMP) program uses it for other products that it installs.

16. Restart your machine if you removed the embedded messaging feature.

When you finish, the product is completely uninstalled. You are now ready to reinstall.

Reinstalling V5.1 after uninstalling and leaving the embedded messaging feature installed

This task describes how to reinstall WebSphere Application Server Network Deployment, Version 5.1 with the embedded messaging feature after uninstalling the product and leaving the embedded messaging feature installed. Follow this procedure to reinstall the product into its original directory.

If you uninstall V5.1 and leave the embedded messaging feature installed to work with other instances on the system, you must follow this procedure before reinstalling V5.1. The uninstaller program on V5.1 gives you the option of leaving the embedded messaging feature installed when there is another instance that has the feature installed. But the V5.0.2 uninstaller program does not have this enhancement of leaving the embedded messaging feature. Therefore, there are different procedures for preparing to reinstall, depending on the version you are reinstalling.

1. Stop any browsers, delete any configuration instances, and stop any Java processes related to WebSphere Application Server products as described in “Uninstalling the product” on page 273.
2. Uninstall manually after uninstalling V5.1. Select your scenario from the following list:
 - Uninstall V5.1 manually on Solaris platforms before reinstalling, as described in Platform-specific tips for installing and migrating.

- Uninstall V5.1 manually on HP-UX platforms before reinstalling, as described in Platform-specific tips for installing and migrating.
- Perform the manual uninstalling procedure for Version 5.1 nodes for platforms other than Solaris and HP-UX, as described in “Uninstalling manually” on page 292 and use the following guidelines:

Guidelines for running the procedure

Important: When you edit the `vpd.properties` file, do not delete entries in the file for the embedded messaging feature or for V5.0.x products.

The `vpd.properties` file topic describes how to edit the file and how to identify entries in the file. For example:

- Any entry that ends with **51** is one that you want to delete. Such entries are for the V5.1 product. Remove all such entries before reinstalling.
- Look for the original installation root directory of the V5.1 product in an entry to verify that the entry is describing a component in the V5.1 product. These are the entries that you want to delete.

3. Reinstall the V5.1 product, as described in “Installing the product” on page 43. Select the embedded messaging feature when you reinstall.

You can reinstall the V5.1 product with the embedded messaging feature after uninstalling V5.1 but leaving the feature installed.

Migrating and coexisting

This topic describes migrating, which is copying the configuration from a previous release of a WebSphere Application Server product into a new release. This topic also describes coexisting, which is running a new release of a WebSphere Application Server product on the same machine at the same time as you run an earlier release.

IBM Support offers tutorials on various WebSphere Application Server installation topics. See WebSphere education on demand: Installation best practices tutorials for more information. One topic describes migration. Another topic describes installing a coexisting WebSphere Application Server product.

Determine if an existing version of WebSphere Application Server is installed on the machine where you plan to install your Version 5.1 product.

If you have a previous version, you must decide whether to copy the configuration and applications of the previous version to the new version. Migration does not uninstall the previous version. The earlier release is still functional but you cannot run both releases at the same time. To run an earlier release and the new release at the same time is coexistence. You must provide non-default port assignments to support coexistence, by selecting the **coexistence** option during installation. See “Migration and coexistence overview” on page 314 for more information.

Coexistence is described as an optional step in the following procedure. Migration is also optional but the steps are not listed as optional. Use the following procedure to migrate applications and configurations:

1. Prepare to migrate or update product prerequisites and corequisites to supported versions.

Refer to the IBM WebSphere Application Server supported hardware, software, and APIs site for current requirements.

5.1+ Examples include upgrading to the level of DB2 supported in V5.1 and the embedded messaging feature.

V5.0.0 and V5.0.1 use a level of embedded messaging that is incompatible with the level supported by V5.0.2 or V5.1. To solve this problem, upgrade V5.0.0 or V5.0.1 to V5.0.2 by applying Fix Pack 2.

2. Prepare to use the migration function of the Installation wizard, to migrate silently, or to migrate using the migration tools. You have the option of letting the Installation wizard issue the commands to use the migration tools or you can issue the commands after installation.

The Installation wizard detects a previously installed version, and displays a coexistence panel. If you choose not to coexist with the existing product, the Installation wizard displays a migration panel. You can select either option or no option.

- a. Start the administrative server of WebSphere Application Server Standard Edition (V3.5.x) or WebSphere Application Server Advanced Edition (V3.5.x or V4.0.x). When you start the administrative server, the migration tools can use the XMLConfig tool to export the configuration data repository. It is not necessary that you start the administrative server for WebSphere Application Server Advanced Single Server Edition, V4. The migration tools use the XML Metadata Interchange (XMI) configuration files directly.
- b. Select the migration option and click **Next** when using the Installation wizard. When using the silent installation method, select the appropriate option. Enter a fully qualified path for the backup directory, where the migration tools store and read the saved configuration and other files. During migration, the wizard backs up the old version of the administrative configuration and the user data files, but does not uninstall the old version. The wizard automatically configures the new installation with the previous configuration data and copies some application data to the newer version.

The Installation wizard uses the WASPreUpgrade migration tool to export the current administrative configuration. After this phase completes, the wizard runs the second phase, which uses the WASPostUpgrade migration tool to convert the backed-up administrative configuration into the new V5.1 configuration. You can stop the administrative server of the earlier version after the migration is complete. You must stop the administrative server of the earlier version before starting the V5.1 WebSphere Application Server.

The installation program prompts you for the following information during migration:

- Backup directory.
 - Currently installed WebSphere Application Server directory. You must install V5.1 in a different directory.
- c. Plan to complete the migration after the installation is complete. Here is an overview of the tasks that you must perform to complete the migration:
 - 1) Stop all servers from the earlier version.
 - 2) Load existing applications into your development environment and fix any known problems.
 - 3) Start the Application Servers in the new installation.
 - 4) Deploy the changed applications in a test environment.
 - 5) Test all applications thoroughly.
 - 6) Follow normal test procedures as you move the test environment into production.

The Installation wizard backs up the earlier version but does not uninstall it, exports the current configuration, and migrates the configuration to the new installation. If errors result when using the WASPreUpgrade tool, the Installation wizard does not run the WASPostUpgrade tool. The wizard displays the WASPreUpgrade.log and the WASPostUpgrade.log files if errors occur.

- d. “Migrating V3.5.x and V4.0.x to Network Deployment” on page 326 using the Installation wizard or during a “Installing silently” on page 174.
 - e. **5.1+** “Migrating configuration data” on page 324 if you prefer a more incremental approach. The Migrating topic describes migrating from V4.x (little to consider) or V3.5.x (several things to consider). The topic describes differences between V3.5.x and V5.1 that result from the full compliance of V5.1 with Java 2 Platform, Enterprise Edition (J2EE) specifications.
3. Set up multiple versions of WebSphere Application Server to coexist. No run-time conflicts can exist for multiple instances and versions of WebSphere Application Server to run at the same time on the same machine. Potential conflicts can occur in these areas:
 - Prerequisites and corequisites, including the level of the embedded messaging feature
 - Operating system registration
 - Environmental conflicts
 - File system

- Port assignments
- a. **5.1+** “Setting up Version 3.5.x and Version 5 coexistence” on page 357.
- b. **5.1+** “Setting up Version 4.0.x and Version 5.x coexistence” on page 358.
- c. **5.1+** “Setting up Version 5 coexistence” on page 360

5.1+ The migration function migrates all of your resources and applications, but does not migrate entities in your `classes` directory. As the Installation wizard for the WebSphere Application Server product migrates applications to Version V5.1, it also migrates any applications that use IBM WebSphere Business Integration Server Foundation Edition, Version 4.0.x.

The base migration removes (from the server configuration) those parts of your configuration that are dependent upon the Integration Server services.

Migration saves the following files in the backup directory.

For Version 3.5.x:

- `bin/setupCmdLine.sh` (or `bin/setupCmdLine.bat` for Windows platforms)
- `classes` (not saved for iSeries)
- `deployableEJBs` (Advanced Edition only)
- `deployedEJBs` (Advanced Edition only)
- `hosts`
- `properties`
- `servlets`

For Version 4.0.x:

- `bin/setupCmdLine.sh` (or `bin/setupCmdLine.bat` for Windows platforms)
- `classes` (not saved for iSeries)
- `config`
- `installableApps`
- `installedApps` (by default unless overridden within a specified developer configuration file)
- `installedConnectors` (Version 4.x Advanced Edition only)
- `properties` (including iSeries as of V5.1 migration tooling)

For Version 5.0.x:

- `classes` (not saved for iSeries)
- `config`
- `installableApps`
- `installedApps`
- `properties`

You can coexist with, or migrate the applications and configuration from a previous version of WebSphere Application Server.

5.1+ Migration from V5.0.x to V5.1 does not require extensive tuning but does require that you use the new migration tools. Use the tools to migrate an embedded messaging queue manager and a federated node.

5.1+ Migration from V4.x to V5.1 does not require extensive tuning. Migration from V3.5.x to V5.1 does require that you examine the migrating applications. See “Migrating configuration data” on page 324 for a description of fine tuning a migration. Part of the procedure for using the migration tools includes a description of what to tune after using the tools. After migrating, return to “Installing the product” on page 43.

For more information on coexistence among releases, see “Migration and coexistence overview” on page 314.

Migration and coexistence overview

WebSphere Application Server contains migration tools that provide migration functionality. The installation wizard can call the migration tools, or you can call them at a later time. The migration tools migrate applications and configuration information to the new version, as described in Migrating and Configuration mapping during migration.

You can also find information about migrating the configuration and the applications from a previous version of WebSphere Application Server to Version 5.0.x in the IBM Redbook, Migrating to WebSphere V5.0: An End-to-End Migration Guide, SG24-6910-00.

5.1 +

Table 59. Overview of migrating from release to release

Migration path	Description
V5.0.x to V5.1	<p>The migration from V5.0.x to V5.1 is routine if there is no embedded messaging feature and if the node is unfederated. You can use the installer program to migrate and have little or no post-migration tuning to perform. When the embedded messaging feature is installed, you must perform one of the following tasks:</p> <ul style="list-style-type: none">• Upgrade V5.0.0 or V5.0.1 to V5.0.2 by applying Fix Pack 2 before migrating to V5.1, and run special scripts before and after uninstalling the V5.0.2 node• Use the migration tools to save the V5.0.0 or V5.0.1 configuration data, uninstall V5.0.0 or V5.0.1, install V5.1, and use the migration tools again to restore the configuration data. <p>Upgrading and using the special scripts is necessary to avoid using incompatible levels of the embedded messaging feature, and to retain the queue manager and persistent data. After migrating a federated node, uninstalling the V5.0.x node can also cause the V5.1 node to unfederate unless you run the migration tools, <code>pre_uninst50ws</code> and <code>post_uninst50ws</code>, before and after uninstalling the V5.0.x node.</p>
V4.0.x to V5.1	<p>The migration tools perform a fairly routine migration from V4 to V5.1. For example, Java 2 Platform, Enterprise Edition (J2EE) 1.2 enterprise archive (EAR) files in V4 work in V5.1 of WebSphere Application Server, which also supports the J2EE 1.3 specification. Similarly, it is not necessary to redeploy enterprise Java bean (EJB) 1.1 Java archive (JAR) files when moving them from V4 to V5.x, which also supports EJB 2.0 JAR files.</p>
V3.5.x to V5.1	<p>The migration from V3.5 to V5.1 involves significant changes in application structures, development, and deployment. The migration tools assist in this transition by migrating system configurations and creating J2EE artifacts, including mapping previous security settings to J2EE security roles. These security mappings let you access migrated assets during the transition. The migration tools create initial J2EE enterprise applications based on V3.5.x configurations. However, because of the significant changes in the application structures, carefully test and fine tune migrated applications using development and deployment tools.</p>

You can select from three combinations of migration and coexistence options in the installation wizard or when customizing the response file for silent installation:

- Migrate only
- Coexist only
- Neither migrate nor coexist

If you neither migrate nor coexist with an earlier version of WebSphere Application Server, you are choosing to ignore the previous installation. You can run only one version at a time because of conflicting default port assignments and the potential for conflicting levels of the embedded messaging feature. It is possible that both versions might run at the same time without conflict if you use non-default ports in the

earlier version and if they both use the same level of embedded messaging. To resolve conflicting port assignments, the coexistence panel lets you assign ports for Version 5 to ensure that it can run with an earlier version.

If you use the embedded messaging feature, the earlier version must use the same level of embedded messaging. For example, Version 5.0.2 and V5.1 use the CSD04 level of embedded messaging. V5.0.0 and V5.0.1 use earlier versions. V5.0.2 and later are the only versions that can coexist with V5.1, when embedded messaging is installed. If you want an instance of V5.0.0 or V5.0.1 to coexist with an instance of V5.0.2 or V5.1, upgrade the V5.0.x instance to V5.0.2 by applying Fix Pack 2.

You can specify port assignments for coexistence on the installation wizard coexistence panel, by editing configuration files, by wsadmin scripting, or by using the **Servers > Application Servers > server1 > End Points** administrative console page.

Migrating and coexisting have roughly opposite goals. The goal of migration is to reconstruct your earlier version in a nearly identical V5.1 environment, including applications, configuration settings, universal resource identifier (URI) descriptors, and Web server context roots. The installation wizard can use the migration tools to migrate the previous configuration and applications. The goal of coexistence is to create an environment that is not in conflict with an earlier version, so far as port settings are concerned. The installation wizard displays a panel where you can set non-conflicting port values that allow the V5.x product to coexist with the earlier version, without port conflicts. This allows both nodes to start and run at the same time. Coexistence processing changes the following configuration files:

- The `virtualhosts.xml` file:
 - HTTP Transport Port
 - IBM HTTP Server Port
 - HTTPS Transport Port
 - HTTP Administrative Console Port
 - HTTPS Administrative Console Secure Port
- The `serverindex.xml` file:
 - Bootstrap Port
 - SOAP Connector Address
 - DRS Client Address
 - JMS Server Queued Address
 - JMS Server Direct Address
 - SAS SSL ServerAuth Address
 - CSIV2 ServerAuth Listener Address
 - CSIV2 MultiAuth Listener Address
- The `server1/server.xml` file
 - HTTP Transport Port
 - HTTPS Transport Port
 - HTTP Administrative Console Port
 - HTTPS Administrative Console Secure Port
 - JMS Server Security Port

See Default coexistence settings for port numbers for more information.

After choosing coexistence, start `server1` and use it to change any ports that are in conflict, so that there is no conflict with the earlier version. Then you can run both versions at the same time.

Consider these issues in a migration or coexistence scenario:

- Conflicting context roots when attempting to share the same Web server.
 - A procedure for configuring a Web server for sharing between WebSphere Application Server versions is described in the information center for the base WebSphere Application Server product.
- Installing WebSphere Application Server multiple times on the same machine.

- Having multiple V5.0.0 or V5.0.1 instances use the embedded messaging feature and attempting to upgrade one instance to V5.0.2, or attempting to install V5.1 with the embedded messaging feature. The service level of the embedded messaging feature for V5.0.0 and V5.0.1 is not compatible with the service level for V5.0.2 or V5.1 (CSD04).

Configuration mapping during migration

This topic describes what changes during migration, which always involves migrating a single instance to another single instance on the same machine or a separate machine. An example is a server group node from a Version 4.0.x environment migrating to a Version 5.x node that you later federate into a deployment manager cell.

Many migration scenarios are possible. The Version 5.x migration tools map objects and attributes to the Version 5.x environment when you restore a configuration from a previous version.

Bootstrap port

Migration maps a default bootstrap NameServer port setting, 900, from V3.5.x and V4.0.x Advanced Edition to the V5.x NameServer default, 2809. The migration tools map a non-default value directly into the V5.x environment.

For Advanced Single Server Edition migration, the bootstrap NameServer port maps to the NameServer value of the Application Server defined in the server configuration file.

Command line parameters

The migration tools convert appropriate command line parameters to Java virtual machine (JVM) settings in the server process definition. Most settings are mapped directly. Some settings, such as memory heap sizes, are not migrated because their roles in the V5.x configuration either do not exist, have different meanings, or have different scopes.

Cluster members

Version 4.0.x server groups are converted to Version 5.x clusters. Migration configures cluster members differently than stand-alone servers. When migrating a server group, all servers in the group are assigned to the HTTP transport port number of the server group, regardless of whether or not they each had a unique port number. Therefore, after migration, all the Application Servers in the new cluster use the same HTTP transport port. Migrated cluster members cannot run until federated into a Network Deployment cell. You can use the administrative console to assign unique port numbers, which lets you run the server without federating it.

Default Server

The name of the default server in Version 5.x is server1. All objects previously owned by the DefaultServer of the prior version, are owned by server1 of Version 5.x after migration.

Enterprise applications for cluster members

Migration does not deploy enterprise applications on cluster members when migrating from Version 3.5.x or Version 4.0.x. You must manually deploy these applications on the cluster using scripting or the deployment manager administrative console.

Automated migration of enterprise applications that are installed in a directory other than the default installedApps directory is not feasible in Version 5.x. Versions 4.0, 5.0, and 5.1 each allow enterprise applications to be installed in a location other than the default installedApps directory. However, for architectural reasons, this property is not migrated from previous versions of WebSphere Application Server to Version 5.0 or Version 5.1. The migration process instead redeploys the enterprise application to the default installedApps directory in the new version. Enterprise applications that are not installed in the default directory must be reinstalled in their new Version 5.x environment.

Java database connectivity (JDBC) drivers and data sources

Version 5.x significantly redefines JDBC and data source objects. The migration tools map V3.5.x and V4.0.x data sources to Version 5.x data sources, using predecessor settings as input variables. The data source that is used is the WebSphere Application Server V4.0.x data source that uses the ConnectionManager architecture.

Migration after federation

You cannot migrate a previous version configuration to a Version 5.x Application Server node after the node is federated into a deployment manager cell. The master copy of the configuration no longer resides on the Version 5.x Application Server node. A master configuration is on the deployment manager node.

You can perform the migration by removing the Application Server node from the cell, migrating, and refederating the node to the cell. As the node is federated, the deployment manager copies the migrated configuration into the master configuration that it maintains for the Application Server node.

5.1+ Migration of a V5.0.x node to a V5.1.x node

You can migrate a V5.0.x node that belongs to a cell without removing the node from the cell.

Migrate the deployment manager first, before migrating any base nodes in the cell.

Table 60. Migration tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Avoiding the use of a V5.0.x deployment manager after migrating to V5.1

Use the same cell name when migrating Network Deployment from V5.0.x to V5.1.x. If you use a different cell name, federated nodes cannot successfully migrate to the Network Deployment V5.1.x cell.

Migrating a base WebSphere Application Server node that is within a cell to V5.1.x also migrates the node agent to V5.1.x. A cell can have some V5.1.x nodes and other nodes that are at V5.0.x levels.

Models, clones, and server groups

Version 3.5.x models and clones and V4.0.x server groups are redefined in V5.x as clusters. Application servers are the only objects supported as models and cluster members in V5.x. This change is a significant difference from V3.5.x, in which many objects can be models and clones. All models and clones relating to Application Servers are mapped to clusters in V5.

Special mapping occurs during the migration of all the other objects that you could previously clone. All clones are treated as simple objects and map as if they are not cluster members. Models that are not Application Server models are ignored and unmapped.

Name bindings

Version 5.x has a new naming structure. All references, such as Enterprise JavaBeans (EJB) references that were valid in previous versions no longer work in Version 5.x. However, you can use the administrative console to add a name binding that maps an old name into the new Version 5.x naming structure. For example, the name of the Version 3.5.x or 4.0.x enterprise bean reference can be both the name of the binding and the Java Naming and Directory Interface (JNDI) name in the Version 5.x name space.

For an example, see *Migrating to WebSphere V5: An End-to-End Migration Guide*, which is available from the Redbooks Web site at <http://www.ibm.com/redbooks>.

Node name

A Version 3.5.x and a Version 4.0.x repository can contain more than one node name and associated children. The WASPostUpgrade tool processes only those objects and children that match the node name of the migrating node. The tool identifies node names in the configuration files that it is migrating and selects any nodes names in a configuration file that match the long network name or short network name of the migrating machine.

PageList servlet

The configuration of the PageList servlet has changed in Version 5.x. Direct use of the servlet has been deprecated. The PageList servlet is available as part of the servlet extension configuration in the Web archive (WAR) file. All references are updated to the servlet configuration supported in Version 5.

You can also use the Assembly Toolkit, which is available on the IBM WebSphere Application Server Toolkit (ASTK) CD-ROM to modify the servlet configuration.

If you use or extend the PageList servlet, you might see an error similar to the following example when running a migrated application that uses the servlet:

```
Error 500: No PageList information is configured for servlet  
EmpInfoApp.SearchByDept
```

Use the Assembly Toolkit to correct the error, by moving the usage or extension to your migrated Enterprise archive (EAR) file:

1. Start the Assembly Toolkit to load the EAR file that generates the error.
2. Open the Web modules within the EAR file.
3. Expand the Web module that generates the error.
4. Open the Web components and find the one that generates the error.
5. Expand the Servlets. The **PageList Extensions** option displays.
6. Add your extension information.
7. Save the EAR file and redeploy it.

Properties directory, classes directory, and the lib/app directory

5.1 + Migration copies files from prior version directories into the Version 5.1.x configuration. See the following section for more information.

Property file migration from Version 3.5.x and Version 4.0.x

5.1 + V5.1 migration does migrate property files from V3.5.x and V4.0.x if these files are also present in WebSphere Application Server, V5.1. Specifically, property-file migration includes these files:

- converter.properties
- encoding.properties (If the "ko" setting is incorrect, no migration occurs.)
- sas.client.props
- TraceSettings.properties
- uddi.properties (V4.0,x only)

You must manually convert settings in other property files to the equivalent V5.1 configuration.

Property file migration from Version 5.0.x to Version 5.1.x

WebSphere Application Server V5.1 migrates all the property files that are installed with V5.0.x by merging settings into the V5.1.x property files with these exceptions:

- j2c.properties
- samples.properties

Migration does not overlay property files.

Samples

No migration of Samples from previous versions is available. Equivalent Version 5.x Samples are available to use.

Security

Java 2 Security is enabled by default in Version 5.x. Security enablement might cause some applications to run on Version 4.0 and not run on Version 5.x. Several techniques are available that you can use to define different levels of Java 2 Security in Version 5.x. One is to create a was.policy file as part of the application, to enable all security permissions. The migration tools call the **wsadmin** command to add an existing was.policy file in the Version 5.x properties directory to enterprise applications as they migrate. The migration tools perform this task while moving Version 4.0 applications into Version 5.

Global security that uses Lightweight Third Party Authentication (LTPA) in Versions 3.5.x and 4.0.x is migrated to the base WebSphere Application Server product and to the Network Deployment product. However, although global security was enabled in Versions 3.5.x and 4.0.x, it is disabled during migration to Version 5.x.

If you add this node later to an IBM WebSphere Application Server Network Deployment, Version 5.x configuration, you can enable and use the LTPA configuration. Use the administrative console to generate keys for the migrated LTPA mechanism. After generating the keys, you can enable global security.

The Global security feature that uses *localos* authentication mechanisms in Versions 3.5.x and 4.0.x is migrated to the Network Deployment product. However, although global security was enabled in Versions 3.5.x and 4.0.x, it is disabled during migration to Version 5.x. The Network Deployment product does not support the authentication mechanism known as SWAM. Migration sets the authentication mechanism in Version 5.x to LTPA. Use the administrative console to generate keys for the migrated LTPA. After generating the keys, you can enable global security.

Version 4.0.x introduced properties to support tuning the JNDI search timeout value along with LDAP reuse connection. These two properties are now settings in the Security Center of the Version 5.x administrative console. Version 4.0.x property values are not migrated to Version 5.x settings.

- The `jndi.LDAP.SearchControl.TimeLimit` property is equivalent to the Version 5.x Search Timeout setting, which is 300 by default in Version 5.
- The `jndi.LDAP.URLContextImplementation` property is equivalent to the Version 5.x Reuse Connection setting, which is true by default in Version 5.

Use the Version 5.x administrative console to change these settings to match your Version 4 property values, if necessary.

Servlet package name changes

The package that contains the `DefaultErrorReporter`, `SimpleFileServlet`, and `InvokerServlet` servlets has changed for Version 5.x. In Versions 3.5.x and 4.0.x, the servlets are in the `com.ibm.servlet.engine.webapp` class. In Version 5, the servlets are in the `com.ibm.ws.webcontainer.servlet` class.

Stdin, stdout, stderr, passivation, and working directories

The location for these directories is typically within the installation directory of a previous version. The default location for `stdin`, `stdout`, and `stderr` is the `logs` directory of the Version 5.x installation root. The migration tools attempt to migrate existing passivation and working directories. Otherwise, appropriate Version 5.x defaults are used.

Using common directories between versions in a coexistence scenario can cause problems.

Transport ports

The migration tools migrate all ports. The tools warn about port conflicts in a log when a port already exists. You must resolve port conflicts before running the servers that are in conflict, at the same time.

The default transport type of the servlet engine in Version 3.5.x is Open servlet engine (OSE). Because Version 5.x no longer supports OSE transport, the migration tools map these transports to HTTP transports, using the same port assignments.

You must manually add `VirtualHost` alias entries for each port.

Web modules

The specification level of the Java 2 Platform, Enterprise Edition (J2EE) that Version 5.x implements requires behavior changes in the Web container for setting the content type. If a default servlet writer does not set the content type, not only does the Version 5.x Web container no longer default to it, the Web container returns the call as "null". This situation might cause some browsers to display resulting Web container tags incorrectly. Migration sets the `autoResponseEncoding` IBM extension to true for Web modules as it migrates enterprise applications. This action prevents the problem.

Version 3.5.x to Version 5.x migration

The migration tools assist in the transition from Version 3.5.x to Version 5, by migrating system configurations and creating J2EE artifacts, including J2EE security roles mapping. The migration tools create initial J2EE enterprise applications based on Version 3.5.x configurations. However, because of the significant change in application structures, plan to carefully test and fine tune migrated applications, using development and deployment tools, to determine exactly how the applications function in Version 5.x.

Analyze the `WASPostUpgrade.log` file for detailed information about migrated enterprise beans. The J2EE programming model specifies an architecture for how applications are created and deployed. Because applications in Version 3.5.x do not have the same architecture, the `WASPostUpgrade` tool recreates applications. It creates all migrated Web resources and enterprise beans in J2EE applications. It maps all enterprise applications from the Version 3.5.x installation into J2EE applications with the same name, deployed in the same server.

The `WASPostUpgrade` tool maps Web resources and enterprise beans that are not included in an enterprise application, into a default J2EE application that includes the name of the server. The tool maps Web applications to J2EE WAR files. The tool deploys enterprise beans as EJB 1.1 beans in J2EE JAR files. The tool combines resources in a J2EE EAR file and deploys it in the Version 5.x configuration. Some differences exist between the EJB 1.0 and EJB 1.1 specifications, but in most cases, EJB 1.0 beans can run successfully as EJB 1.1 beans.

Mapping details for V3.5.x to Version 5.x migration

- **data sources.xml**

You can use a Version 3.5.x `data sources.xml` file to augment data source configuration settings. Version 3.5.x stores the file in the `properties` directory. The migration tools migrate an existing `data sources.xml` file by merging properties in the file into the data source and JDBC driver configuration.

- **Enterprise applications**

The Version 3.5.x enterprise application entries are optional. The entries group sets of objects together for security definitions. The enterprise bean and Web application portions of the enterprise application point to their respective entries in other portions of the XML file.

Processing each enterprise application creates a J2EE application with the same name. The enterprise bean and Web application entries become pointers to the definitions of enterprise beans and Web applications. Entry details help build a J2EE application.

For enterprise bean files, the JAR file definition helps locate the JAR files to redeploy and add to the J2EE application. Document root entries in the Web application help locate the resources used within the Web application, such as HTML pages and JSP pages, which become part of the WAR file within the J2EE application. Class path entries in the Web application help locate servlets and JAR files, which become part of the WAR file within the J2EE application. Version 3.5.x migration to V5.x creates J2EE 1.2 compatible enterprise applications that contain EJB 1.1, Servlet 2.2, and JSP 1.1 level modules. This migration provides the most straight forward compatibility and enables interoperability with previous WebSphere Application Server versions.

- **Enterprise beans**

Version 3.x supports only the EJB 1.0 Components Specification level. Version 5.x supports EJB 1.1 and 2.0 components. However, many EJB 1.0 beans can successfully deploy as EJB 1.1 beans. The migration tools redeploy enterprise beans automatically as part of the application migration phase. Check the `WASPostUpgrade.log` file for deployment details of these enterprise beans. Make any necessary changes and redeploy.

No redeployment is required when moving EJB 1.1 JAR files from Version 4.

Specify only one backend data store vendor per JAR file. If enterprise beans use different backend data stores, package them into separate JAR files.

- **J2EE security**

The security authorization model in Version 3.5.x is based on the notion of enterprise application and method groups. The cross product of the enterprise application and the method groups is a WebSphere Application Server permission. The J2EE specification includes an authorization model based on roles.

To convert from the WebSphere Application Server permission model in Version 3.5.x to the role-based authorization model in Version 5, the migration tools create a one-to-one mapping from a WebSphere Application Server permission to a new role under that application. Therefore, for each enterprise application and each method group in Version 3.5.x, the migration tools create a role in Version 5, contained in the J2EE application deployment descriptor. The authorized subjects for each role are contained in an authorization table found in the J2EE application binding.

The J2EE specification includes an authorization model based on roles. WebSphere Application Server interprets the role to mean a set of permissions to access a resource. In the case of an enterprise bean method invocation, the permission to access the method on a particular bean is specified by a method permission. This method permission is associated with one or more roles in the deployment descriptor of the bean JAR file.

In the case of accessing Web resources, the permission to access a Web URI and invoke an HTTP method on that URI is specified in terms of Web resource collections and security constraints in the J2EE specification. The deployment descriptor in the WAR file of the Web application contains the security constraints and Web resource collections.

- **JSP levels**

Version 5.x runs JSP 1.0 and 1.1 objects as JSP 1.2 objects, which is the only supported level.

- **Servlet redirector**

Version 5.x does not support the servlet redirector from previous versions. The migration tools ignore these objects.

- **Servlet package name changes when migrating from V3.5.x to V5.x**

If the Version 3.5.x configuration defines the SimpleFileServlet servlet, this servlet is not migrated. The migration tools set the FileServingEnabled attribute in the `ibm-web-ext.xml` Web module file to `true`.

If the Version 3.5 configuration defines the InvokerServlet servlet, the servlet is not migrated. The migration tools set the ServeServletsByClassnameEnabled attribute in the `ibm-web-ext.xml` Web module file to `true`.

If the Version 3.5.x configuration defines the DefaultErrorReporter servlet, the servlet is migrated into the `web.xml` Web module file. Migration uses the new package to set the class name.

- **Transports**

The default transport type of the servlet engine in Version 3.5.x is Open servlet engine (OSE). Because Version 5.x no longer supports OSE transport, the migration tools map these transports to HTTP transports, using the same port assignments. You must manually add VirtualHost alias entries for each port.

Version 4.0.x to Version 5.x migration

This migration is much less complicated than moving from V3.5.x to V5.x. The V4.0.x configuration is already at the J2EE 1.2 specification level. Although Version 5.x is at the J2EE 1.3 specification level, J2EE 1.2 objects are supported.

- **Enterprise beans**

No redeployment is required when moving EJB 1.1 JAR files from Version 4.0.

Specify only one backend data store vendor per JAR file. If enterprise beans use different backend data stores, package them into separate JAR files.

- **JMS Resources**

All JMS resources from Version 4.0 are mapped into generic JMS resources in the Version 5.x configuration. Reconfigure JMS resources that use IBM WebSphere MQ as IBM WebSphere MQ-specific resources. MQ JMS resources have better integration with system management. You do not need to manually define entries in the name space. You can see the backing MQ queue definitions through MQ JMS entries.

- **JSP precompiling**

In Version 4.0.x, the classes generated from JSP pages are in a package based on the directory structure of the WAR file. Any JSP at the top of the context root is in the unnamed package. JSP pages in subdirectories of the root are in packages named after the subdirectories. In Version 5, the classes generated from JSP pages are all in the `org.apache.jsp` package. Therefore, the class files are not compatible between versions.

When migrating an enterprise application from Version 4.0.x to Version 5, recompile the JSP pages to regenerate the class files into the correct packages.

The migration tools provide this support, by using the `-preCompileJSPs` option of the **wsadmin** tool during the installation of the application.

Use the same option to install any Version 4.0.x enterprise applications that you manually move to Version 5.x.

- **J2EE security**

You can apply security in two Version 4.0.x locations to enterprise applications. Information in the repository has precedence over information in the enterprise application bindings. The migration tools migrate information in the repository to the enterprise application.

- **Secure Sockets Layer (SSL) migration**

The following SSLConfig attributes that point to user-defined key files are migrated from WebSphere Application Server Advanced Edition, Version 4.0.x to V5.x as follows:

V4.0.x settings

```
<key_file_name>dir_name/WASLDAPKeyring.jks</key_file_name>  
<trust_file_name>dir_name/WASLDAPKeyring.jks</trust_file_name>
```

The `dir_name` variable identifies the original location of the WASLDAPKeyring.jks file.

V5.x settings

```
keyFileName="dir_name/WASLDAPKeyring.jks"  
trustFileName="dir_name/WASLDAPKeyring.jks"
```

The `dir_name` variable identifies the original location of the WASLDAPKeyring.jks file.

V4.0.x settings

```
<key_file_name>${WAS_HOME}/keys/WASLDAPKeyring.jks</key_file_name>  
<trust_file_name>${WAS_HOME}/keys/WASLDAPKeyring.jks</trust_file_name>
```

V5.x settings

```
keyFileName="${USER_INSTALL_ROOT}/keys/WASLDAPKeyring.jks"  
trustFileName="${USER_INSTALL_ROOT}/keys/WASLDAPKeyring.jks"
```

The migration tools do not copy the key files (for example, .jks, or .kdb) to the corresponding directory in the base WebSphere Application Server product or the Network Deployment product. You must complete the migration of the SSL configuration by copying qualifying key store files to Version 5.x directories.

If the key-file-name and trust-file-name attributes point to the DummyServerKeyFile.jks file in the WebSphere Application Server Advanced Edition V4.0.x configuration, the key-file-name and trust-file-name attributes are not migrated to V5.x. Instead the V5.x default value of `${USER_INSTALL_ROOT}/etc/DummyServerKeyFile.jks` is left unchanged.

- **Servlet package name changes when migrating from Version 4.0 to Version 5.x**

If the `web.xml` Web module file for Version 4.0 defines the SimpleFileServlet servlet, the migration tools update the class name to reflect the Version 5.x package. The tools also set the FileServing Enabled attribute to true.

If the `web.xml` file defines the InvokerServlet servlet, the migration tools update the class name to reflect the Version 5.x package. The tools also set the ServeServletsByClassnameEnabled attribute to true.

If the `web.xml` file defines the DefaultErrorReporter servlet, the migration tools update the class name to reflect the Version 5.x package.

Version 5.0.x to Version 5.1 migration

Migrating V5.0.x to V5.1 is much less complicated than migrating from V4.0.x or V3.5.x. Both sides

of the migration use the same underlying definitions. The task involves mapping configuration files from the V5.0.x to the V5.1 configuration and copying installed applications into the new product. The migration tools support the migration of federated nodes and support the full migration of a Network Deployment node.

Java heap size for migrating EAR files

When migrating all 5.0.x EAR files to V5.1 using the wsadmin tool, the WASPostUpgrade tool uses the default maximum Java heap size value of 64MB to install the EAR files.

If a V5.0.x EAR file fails to install during migration because the Java heap size is not large enough, you see a message similar to the following error:

```
java.lang.OutOfMemoryError JVMXE006:OutOfMemoryError
```

Increase the maximum Java heap size. Then use the following information to manually install the EAR file. Assume that the new maximum heap size is represented by number signs (###) in the following example. If the `install_app_name.jacl` file is in the `$WAS50_backup` directory created by migration, run the following command from the `bin` directory of the V5.1 WebSphere Application Server. (The command appears on more than one line for clarity.)

```
wsadmin -conntype NONE
        -javaoption
        -Xmx###m
        -f $WAS50_backup/install_app_name.jacl
```

If the `install_app_name.jacl` file does not exist in the `$WAS50_backup` directory created by migration, run the following command from the `install_root/bin` directory of the Version 5.1 base WebSphere Application Server product or the Network Deployment product.

Installing the application on WebSphere Application Server, Version 5.1

Assume that:

Installation root

`C:\WebSphere\AppServer`

Number signs (###)

Maximum heap size value

`EAR_file_name`

The name of the EAR file

`app_name`

The name of the application.

`server_name`

The name of the server on which the EAR file installs

`node_name`

The name of the node on which the server is configured

The command appears on more than one line for clarity.

```
wsadmin -conntype NONE
        -javaoption
        -Xmx###m
        -c "$AdminApp install
           C:\\WebSphere\\AppServer\\installableApps\\
           EAR_file_name
        {-nodeployejb
         -appName app_name
         -server server_name
         -node node_name}"
```

Installing the application on WebSphere Application Server Network Deployment, Version 5.1

Assume that:

Installation root

C:\WebSphere\DeploymentManager

Number signs (###)

Maximum heap size value

EAR_file_name

The name of the EAR file

app_name

The name of the application.

cluster_name

The name of the cluster on which the EAR file should be installed

The command appears on more than one line for clarity.

```
wsadmin -conntype NONE
        -javaoption
        -Xmx###m
        -c "$AdminApp install
           C:\WebSphere\DeploymentManager\installableApps\
           EAR_file_name>
        {-nodeployejb
         -appname app_name
         -cluster cluster_name}"
```

No migration of configuration instances from V5.0.x to V5.1

The migration tools do not support the migration of configuration instances from V5.0.x to V5.1

Migrating configuration data

You can migrate administrative configurations with the Installation wizard or with the migration tools, as this task describes. If you decide to use the migration tools, do not select the Migration check box on the Migration panel of the Installation wizard.

If you use an earlier version of WebSphere Application Server, the system administrator might have fine-tuned various application and server settings for your environment. It is important to have a strategy for migrating these settings with maximum efficiency and minimal loss.

You can perform incremental migration of V3.5.x or V4.0.x nodes by calling the migration tools multiple times, each time specifying a different configuration file. Various reasons exist for having multiple configuration files. Whatever the reason, migrating one configuration file at a time lets you test applications incrementally before continuing to the next configuration file.

5.1 + Before using the migration tools, consult the V5.1 Release Notes document to understand what fixes you must apply to earlier versions. Applying fixes to an earlier version might also apply fixes to files that have a role in the migration. Apply any fixes to ensure the most effective migration of configurations and applications possible.

5.1 + The migration tools in V5.1 support migration from all supported versions of WebSphere Application Server, including V5.0.x.

IBM provides a set of migration tools for migrating administrative configurations from V3.5.x, V4.x, or V5.0.x to the Network Deployment product.

The overall migration process when you issue the commands to use the migration tools instead of letting the installation program issue the commands is:

1. Save the current configuration and necessary files with the WASPreUpgrade migration tool.
2. **5.1 +** Install the Version V5.1 product without selecting the automated migration option.

3. Restore the configuration from the earlier release with the WASPostUpgrade migration tool.

5.1+ WASPostUpgrade uses the **backupConfig** command to save the existing V5.1 configuration before performing migration. The results are stored in the *install_root/temp* directory. You can use the **restoreConfig** command to restore the backup, if required.

1. **5.1+** Open the information center for WebSphere Business Integration Server Foundation to find installation and migration instructions for the base product and the Network Deployment product when you are migrating Enterprise to Integration Server.

2. **5.1+** Migrate the Network Deployment product. Select one of the following migration scenarios for information about how to migrate configuration data to a V5.1 Network Deployment node:

- Migrate V3.5.x and V.4.0.x to Network Deployment V5.1.
- Migrate V5.0.x of Network Deployment to V5.1.

3. Start the deployment manager before migrating base nodes that are federated.

5.1+ Start the V5.1 deployment manager. During migration, the V5.1 deployment manager must be running for the migration tools to:

- Update the configuration for each federated node
- Request full synchronization

If the V5.1 deployment manager is not running, failures can occur.

Table 61. Migration tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Recovering from configuration errors when the deployment manager was not running during migration.

4. Migrate base WebSphere Application Server nodes. **Guidelines for migrating federated nodes**

Rule	Description
A federated node has no control over its feature configuration.	The deployment manager owns the configuration of all federated nodes, including the configuration of features on those nodes. When migrating a federated node, the configuration on the base node is owned by the deployment manager and cannot be changed.
Unfederate a node to let the node own its configuration, if you must add features to the node.	It is always true that you must unfederate a node before adding features. You then add the node back to the cell to cause the deployment manager to add the changed configuration of the base node.
Install the V5.1 node with features that make it appear to be a clone of the node that you are migrating.	When migrating, select the same features for the new node that are in the existing node. Adding or changing features can cause configuration problems in a federated node. The most trouble-free path is to select the same features.
It is possible to migrate from one machine to another.	One of the tasks introduced in this topic describes how to migrate a federated node to V5.1 on a different machine.

If you are migrating a federated node, select the same type of installation that you chose when you installed the earlier release. If you selected a full installation type, select a full installation type now. If you selected a custom installation type, select a custom installation type now and select the same features that you did for the V5.0.x release. Federated nodes have configurations that are controlled by the deployment manager that controls the cell.

5.1+ Migrating a federated node installs the same features for the V5.1 node that are in the configuration for the V5.0.x node.

If the node that you are migrating is part of a deployment manager cell, migrate the V5.0.x deployment manager to V5.1 first, before continuing this procedure. The deployment manager node must always be at the highest release level within the cell.

Select one of the following migration scenarios for information about how to migrate configuration data to base WebSphere Application Server nodes:

- Migrate V3.5.x or V4.0.x of WebSphere Application Server to V5.1.
 - **5.1+** Migrating V3.5.x or V4.0.x of WebSphere Application Server to a remote V5.1 machine.
 - **5.1+** Migrate Version 5.0.x of WebSphere Application Server to Version 5.1.
 - **5.1+** Migrate Version 5.0.x of WebSphere Application Server to a remote Version 5.1 machine.
 - Migrate from an operating system that is no longer supported.
 - **5.1+** Add or change features on a federated node while performing a manual reconfiguration. Use this procedure to add or change features.
5. After migrating each base node, start each node. Use the `startNode` command from the `install_root/bin` directory of each base Application Server to start the nodeagent process:

```
startNode.sh
```

Occasionally, for example after rebooting an Application Server machine, you must restart the node agent server on the Application Server node, by running the `startNode.sh` or `startNode.bat` script. To keep your Application Server nodes running without accessing the `bin` directory of each one, use the operating system to monitor and restart the nodeagent process on each Application Server node, as described in “Automatically restarting server processes” on page 176. (You can also set up the `dmgr` server as a managed process on the deployment manager node.) Adding a node automatically issues the `startNode.sh` or `startNode.bat` script for the node.

6. Configure the Application Server after migration. Configuring the Application Server after migration is a way of verifying the results of the migration tools. You can also use Configuration mapping during migration to verify the results of the migration. The topic has a detailed description of how the migration tools migrate objects, and what you should verify.

You can use the migration tools to migrate from one version of WebSphere Application Server to another.

Migrating V3.5.x and V.4.0.x to Network Deployment

You can migrate Version 3.5.x or Version 4.0.x of WebSphere Application Server to WebSphere Application Server Network Deployment, V5.x.

Migrating WebSphere Application Server Advanced Edition to WebSphere Application Server Network Deployment requires migrating all nodes in the collection to WebSphere Application Server. When migrating a model or server group from an earlier version to a Version 5 cluster, you should first migrate the model or server group node to a Version 5 base WebSphere Application Server node, which also migrates applications that are on the previous version node. Migrated applications are in the `installableapps` folder in the `install_root` of the base WebSphere Application Server node.

Install the Network Deployment product and then follow the instructions in this task to migrate the model or server group configuration to the deployment manager. Migration creates the appropriate number of Version 5 clusters. To install a migrated application on a cluster, you copy the EAR file from the `installableapps` folder of the base Version 5 WebSphere Application Server node.

Your current configuration from a previously installed version contains one or more node configurations, one for each node in the repository. You have a choice of where to install the Network Deployment product. You can install it either:

- On one of the nodes where you are installing and migrating applications to WebSphere Application Server.

This choice maps the single domain with multiple nodes to a single Version 5 cell with the same number of nodes. Network Deployment is on one of the nodes. This placement provides a configuration that is equivalent to the previously installed version.

- On an additional machine that does not have WebSphere Application Server.

This choice maps the single domain with multiple nodes to a single Version 5 cell with the same number of nodes, but with Network Deployment on an additional node.

You can vary the order in which you migrate these nodes. This task describes the most direct method.

After federating an Application Server node into a deployment manager cell, you cannot use the migration tools on the Application Server node. To use these tools again, remove the node from the cell, use the tools, and add the node to the cell again. You can, however, use the migration tools on a Network Deployment node after base Application Server nodes have been added. Use the `-nodeName` parameter on the **WASPostUpgrade** command to specify the Deployment Manager node.

You can also perform a silent migration to Network Deployment during a silent installation.

1. Perform a typical or custom WebSphere Application Server installation on all nodes in the repository. The installation procedure is described in the information center for the base WebSphere Application Server product.
2. Migrate each Application Server node to the base WebSphere Application Server product, as described in the WebSphere Application Server information center. You must migrate each node that you intend to add to the Network Deployment configuration.
3. Install the Network Deployment product.
 - When installing Network Deployment on a WebSphere Application Server node that you migrated:
 - If the previous version is still installed and running, select the migration option during the installation.
 - If the previous version is no longer installed, perform a manual migration against the **WASPreUpgrade** backup directory on this or another migrated node. Issue the **WASPostUpgrade backupDirectory** command from the `bin` directory of the `install_root`.
 - When installing Network Deployment on a machine that did not have an earlier version of WebSphere Application Server, perform a manual migration against the **WASPreUpgrade** backup directory on another node that you migrated from the previous domain. Issue the **WASPostUpgrade backupDirectory** command from the `bin` directory of the `install_root`.

You can also install the Network Deployment product silently, and migrate an earlier version configuration silently. You must supply values for silent migration options when you tailor the options response file:

- a. Direct the installation program to migrate a previous installation by specifying the `-W previousVersionDetectedBean.previousVersionDetected="true"` parameter.
- b. Identify the specific version to migrate. Use one of the following values to identify the specific version to migrate:
 - For WebSphere Application Server Advanced Edition, Versions 3.5.x and 4.0.x, use one of the following parameters:
 - `-W previousVersionPanelBean.selectedVersionEdition="AE"`
 - `-W previousVersionPanelBean.selectedVersionEdition="advanced"`
 - For WebSphere Application Server Advanced Single Server Edition, Version 4.0.x, use the `-W previousVersionPanelBean.selectedVersionEdition="AEs"` parameter.
 - For WebSphere Application Server Standard Edition, Version 3.x, use the `-W previousVersionPanelBean.selectedVersionEdition="standard"` parameter.
- c. Specify the location where the previous version is installed in the `-W previousVersionPanelBean.selectedVersionInstallLocation="/opt/WebSphere/AppServer"` parameter.
- d. Specify the path to the configuration file.

- For WebSphere Application Server Advanced Edition, Versions 3.x and 4.0.x, use the `-W previousVersionPanelBean.selectedVersionConfigFile="/opt/WebSphere/AppServer/config/admin.config"` parameter.
 - For WebSphere Application Server Advanced Single Server Edition, Version 4.0.x, use the `-W previousVersionPanelBean.selectedVersionConfigFile="/opt/WebSphere/AppServer/config/server-cfg.xml"` parameter.
 - For WebSphere Application Server Standard Edition, Version 3.x, use the `-W previousVersionPanelBean.selectedVersionConfigFile="/opt/WebSphere/AppServer/config/server-cfg.xml"` parameter.
- e. Specify the version number of the previous version, such as "4.0", "4.0.1", or "3.5", in the `-W previousVersionPanelBean.previousVersionSelected="4.0"` parameter.
 - f. Indicate that you are selecting the version you have identified with the `-W previousVersionPanelBean.migrationSelected="true"` parameter.
 - g. Specify the backup directory where WASPreUpgrade is to store information from the previous version with the `-W migrationInformationPanelBean.migrationBackupDir="/tmp/migrationbackup"` parameter.
 - h. Specify the directory for storing migration logs with the `-W migrationInformationPanelBean.migrationLogfileDir="/tmp/migrationlogs"` parameter.
 - i. Enter a node name, host name, and cell name for the new Version 5 installation. The node name is used for administration must be unique within the group of nodes that comprise the cell. The host name is the DNS name or IP address for this computer. The cell name is a logical name for the group of nodes.
 - 1) Use the `-W nodeNameBean.nodeName="nodenameManager"` parameter to specify a unique node name. You can use the short host name of the machine as the node name portion of the name and append Manager to it to compose the name.
 - 2) Use the `-W nodeNameBean.cellName="nodenameNetwork"` parameter to specify the cell name. You can use the host name as the node name portion of the name and append Network to it to compose the name.
 - 3) Use the `-W nodeNameBean.hostName="hostNameOrIPAddress"` parameter to specify the host name for the machine, which can be the fully qualified network name, the short name, or the IP address.

You can also configure the options response file for coexistence with an earlier version by specifying non-conflicting port assignments in the file.

4. Start the deployment manager on the Network Deployment node. On the system that has Network Deployment installed, start the deployment manager in either of these ways:
 - Click **Programs > WebSphere Application Server > Deployment Manager** from the Start menu on a Windows platform.
 - Issue the **startManager** command from the `bin` directory of the `install_root`.
5. Use the deployment manager administrative console to add each WebSphere Application Server node to the cell. The following procedure supports adding migrated nodes to a deployment manager cell, whether the nodes are cloned, stand-alone, or a combination of the two.
 - a. Use the Network Deployment administrative console to add each node to the Network Deployment cell.
 - b. Install any clustered applications on the appropriate cluster, using either the administrative console or the **wsadmin** command. Use the EAR file that the migration tools create when migrating the earlier version to the Version 5 base WebSphere Application Server. You can find the EAR file in the `installableApps` directory that is in the base product `install_root`.

The deployment manager automatically propagates enterprise applications to application server nodes in the cluster.

After federating an Application Server node into the Network Deployment cell, the configuration repository for the cell contains the master copy of the node configuration. From that point on, the

deployment manager maintains the permanent configuration for the node. Changes you make with the deployment manager administrative console directly update the configuration files stored on the deployment manager node.

6. Use First Steps to perform the installation verification test (IVT).

Occasionally, for example after rebooting an Application Server machine, you must restart the nodeagent server on the Application Server node, by running the **startNode** command from the `bin` directory of the Application Server `install_root`. To keep your Application Server nodes running, without having to access the `bin` directory of each one, use the operating system to monitor and restart the nodeagent process on each Application Server node, as described in “Automatically restarting server processes” on page 176. (You can also set up the dmgr server as a managed process on the deployment manager node.) Adding a node automatically issues the **startNode** command for the node.

Migrating Network Deployment, Version 5.0.x to Version 5.1

The migration tools of WebSphere Application Server Network Deployment, Version 5.1 support migrating the configuration of Network Deployment, Version 5.0.x to Network Deployment, Version 5.1.

This topic describes migrating a V5.0.x Network Deployment node to V5.1.

Migrate V5.0.x of WebSphere Application Server Network Deployment to V5.1 before migrating the base nodes that comprise the cell. The Network Deployment node must always be at the highest release and fix level within a cell, to allow it to manage all nodes in the cell. Nodes in the cell can be at any V5 level.

You can also perform a silent migration to Network Deployment during a silent installation.

Use the following procedure to migrate the deployment manager node.

1. Stop the V5.0.x deployment manager. Use the `stopManager` command from the `install_root/bin` directory. For example, use the following command on a Linux platform:

```
stopManager.sh
```

If you have security enabled, specify the `-user` and `-password` parameters of the command.

You can migrate a V5.0.x node without stopping it. But it is not necessary to have the node running to migrate its configuration. The migration tools can retrieve all the configuration data while the node is stopped. And you must stop the node before you can start the V5.1 node that you are installing. So, you can stop the node now.

Table 62. Migration tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Avoiding the use of a V5.0.x deployment manager after migrating to V5.1

2. Use the migration tools to migrate the V5.0.x configuration to V5.1. The steps vary depending on whether you are migrating a V5.0.0 or V5.0.1 node with the embedded messaging client installed.
 - Migrate V5.0.0 or V5.0.1 of Network Deployment with embedded messaging to V5.1. If you have V5.0.0 or V5.0.1 with the embedded messaging client installed, use this procedure.
 - If you have V5.0.0 or V5.0.1 without the embedded messaging client installed, or if you have V5.0.2 with or without the embedded messaging client installed, perform the following steps:
 - a. Install the V5.1 product. Select the migration option when it appears. If the migration option does not appear, cancel the installation.

In some cases, such as when installing a non-English version, the installation wizard might not detect a previous version. You can force the migration panel to appear, by starting the installation with an option on the **Install.exe** or **install** command.

For example, on Linux and UNIX-based platforms, use this command from the CD-ROM mount point:

```
./install -W previousVersionDetectedBean.previousVersionDetected="true"
```

- b. Verify the installation of the V5.1 deployment manager. Use the First Steps tool when it appears at the end of installing the product, or run the installation verification test yourself, if the First Steps tool does not appear for some reason.
- c. Optional: Uninstall the V5.0.x deployment manager. Do not uninstall the V5.1 node.
Perform this step only after you are certain that you have successfully migrated the configuration of the node you intend to delete.

The migration tools do not migrate V5.0.0 or V5.0.1 Samples to the V5.1 Network Deployment node. A synchronization removes the Samples from the V5.0.0 or V5.0.1 nodes.

The migration tools do migrate V5.0.2 Samples to the V5.1 Network Deployment node, to support cell members at the V5.0.2 level. If the migration did not migrate the V5.0.2 Samples, a synchronization would remove the Samples from the V5.0.2 nodes. After migrating all federated nodes to V5.1, the V5.0.2 Samples remain in the configuration. You can remove the V5.0.2 Samples at that time or replace them with the V5.1 samples.

Migrating Network Deployment, Version 5.0.0 or Version 5.0.1 with embedded messaging to Version 5.1:

The migration tools of WebSphere Application Server Network Deployment, Version 5.1 support migrating Network Deployment, Version 5.0.0 or Version 5.0.1 with embedded messaging to Version 5.1.

This topic describes migrating WebSphere Application Server Network Deployment, Version 5.0.0 or Version 5.0.1 with the embedded messaging client feature installed, to Version 5.1. If you are migrating a V5.0.0 or V5.0.1 deployment manager without the embedded messaging client, or if you are migrating a V5.0.2 deployment manager, see “Migrating Network Deployment, Version 5.0.x to Version 5.1” on page 329.

The embedded messaging feature places certain restrictions on migrating V5.0.0 or V5.0.1 nodes to V5.1. If embedded messaging is not involved, you can migrate any level of V5.0.x to a V5.1 during the installation of the V5.1 Network Deployment product. If the embedded messaging client is installed on the V5.0.0 or V5.0.1 deployment manager node, you must first either upgrade the node to V5.0.2, or use this task to perform the manual migration procedure for V5.0.0 or V5.0.1 deployment manager nodes with the embedded messaging client.

1. Stop the V5.0.0 or V5.0.1 deployment manager. Use the stopManager command from the `install_root/bin` directory. For example, use the following command on a Linux platform:
`stopManager.sh`

If you have security enabled, specify the `-user` and `-password` parameters of the command.

You can migrate a V5.0.0 or V5.0.1 node without stopping it. But it is not necessary to have the node running to migrate its configuration. The migration tools can retrieve all the configuration data while the node is stopped. And you must stop the node before you can start the V5.1 node that you are installing. So, you can stop the node now.

Table 63. Migration tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Avoiding the use of a V5.0.x deployment manager after migrating to V5.1

2. Obtain the V5.1 product CD-ROM. On this CD is the `migration/bin` directory. This directory contains a special environment that you can use to run the WASPreUpgrade tool without installing V5.1.
3. Save the current configuration using the WASPreUpgrade script from the `/migration/bin` directory of the V5.1 product CD-ROM. Save the configuration in the `migration-specific-backup` directory on the V5.0.0 or V5.0.1 machine.

`WASPreUpgrade /opt/tmp/migration-specific-backup /opt/websphere/deploymentmanager`

The WASPreUpgrade tool provides status to the screen and to log files in the /migration-specific-backup directory. ASCII log file names start with the text "WASPreUpgrade" and include a date timestamp.

4. Uninstall the V5.0.0 or V5.0.1 deployment manager node. Use the following steps to uninstall the node:
 - a. Run the uninstall command from the *install_root*_uninst directory of the product that you are uninstalling.
 - b. Run the platform-specific uninstalling procedure to complete the removal by removing all artifacts of the earlier product from the system.
5. Install V5.1 of Network Deployment without selecting the migration option, if it appears.
6. Add the V5.0.0 or V5.0.1 configuration to the V5.1 configuration. Use the WASPostUpgrade tool in the *install_root*/bin directory of the V5.1 installation root directory to add the V5.0.0 or V5.0.1 configuration to the V5.1 configuration.

```
WASPostUpgrade /opt/tmp/migration-specific-backup
```

You can use the migration tools to migrate V5.0.0 or V5.0.1 of Network Deployment with the embedded messaging client to V5.1.

Migrating Version 3.5.x or Version 4.0.x of WebSphere Application Server to Version 5.x

You can use the migration tools to migrate configuration data from Version 3.5.x or Version 4.0.x of WebSphere Application Server to WebSphere Application Server V5.x.

If you use an earlier version of WebSphere Application Server, the system administrator might have fine-tuned various application and server settings for your environment. It is important to have a strategy for migrating these settings with maximum efficiency and minimal loss.

You can migrate administrative configurations with the installation wizard or manually, as this task describes. If you decide to migrate manually, do not select the migration check box on the installation wizard migration panel.

You can perform incremental manual migration by calling the migration tools multiple times, each time specifying a different configuration file. There are various reasons for having multiple configuration files. Whatever the reason, migrating one configuration file at a time lets you test applications incrementally before continuing to the next configuration file.

Manual migration provides a more incremental migration approach than the complete migration that the installation wizard provides. IBM provides a set of migration tools for migrating administrative configurations to the base WebSphere Application Server product from either edition of Version 3.5.x, or from Version 4.x. The overall migration process is to back up the current configuration and necessary files, install the Version 5 product, and restore the configuration.

Before using the migration tools, consult the Release Notes document to understand what fixes you must apply to earlier versions. Applying fixes to an earlier version might also apply fixes to files that have a role in the migration. Apply any fixes to ensure the most effective migration of configurations and applications possible.

5.1+ Typically you can use the WASPreUpgrade and WASPostUpgrade migration tools from V5.1 of WebSphere Application Server to upgrade from either V3.5 or V4.0 to V5.1 on the same machine. If your scenario includes migrating a V3.5.x or a V4.0.x configuration on one machine to WebSphere Application Server V5.1 on another machine, use the alternate procedure described in "Migrating Version 3.5.x or Version 4.0.x of WebSphere Application Server to a remote Version 5.1 machine" on page 333.

This topic describes using the V5.x migration tools to migrate the following products:

- WebSphere Application Server Single Server Edition, V3.5

- WebSphere Application Server Advanced Edition, V3.5
- WebSphere Application Server Advanced Edition, V4.0
- WebSphere Application Server Advanced Single Server Edition V4.0 (the steps vary slightly)

The WASPreUpgrade tool saves the existing V3.5 or V4.0 configuration into a *migration-specific-backup* directory. The WASPostUpgrade tool uses this directory to add the old configuration settings to the new V5.x environment.

1. Obtain the new product CD-ROM. On this CD is the *migration/bin* directory. This directory contains a special environment that you can use to run the WASPreUpgrade tool without installing the product.
2. Save the current configuration using the WASPreUpgrade script from the *migration/bin* directory of the product CD-ROM. Save the configuration in the *migration-specific-backup* directory:

```
WASPreUpgrade /usr/tmp/migration-specific-backup /usr/websphere/appserver yourNodeName
```

For all scenarios except V4.0.x Advanced Single Server Edition, verify that the administrative server of the existing environment is running. The WASPreUpgrade tool provides status to the screen and to log files in the *migration-specific-backup* directory. ASCII log file names start with the text WASPreUpgrade and include a date and timestamp.

The WASPreUpgrade tool saves all files from the following directories in the existing V3.5.x or the V4.0.x configuration to the *backup* directory:

For Version 3.5.x

- bin
- classes
- deployableEJBs (Advanced Edition only)
- deployedEJBs (Advanced Edition only)
- hosts
- properties
- servlets

For Version 4.x

- bin
- classes
- config (Version 4.0.x Advanced Single Server Edition only)
- installableApps
- installedApps
- installedConnectors (Version 4.0.x Advanced Edition only)
- properties

The WASPreUpgrade tool saves selected files from the V3.5.x or the V4.0.x *install_root/bin* directory. It also exports the existing Application Server configuration from the V3.5.x or the V4.0.x repository. The WASPreUpgrade tool calls the XMLConfig tool to export the existing V3.5 or the V4.0 repository to the *websphere_backup.xml* file in the *migration-specific-backup* directory.

V4.0.x Advanced Single Server Edition does not require the administrative server to run at the time of migration. The WASPreUpgrade tool copies the *server-cfg.xml* file from the *install_root/config* directory to the *migration-specific-backup/config* directory.

If errors occur while running the WASPreUpgrade tool, you might have to apply fixes to the V3.5 or the V4.0 installation to successfully complete the export step. See the IBM Support page for the latest fixes that might be applicable. When viewing this information from the information center, click **Support** to link to the IBM Support page.

3. Install V5.1 of the WebSphere Application Server Version product. Do not select the migration option, if it appears.

After each use of WASPostUpgrade, verify V5.1 port settings in two files:

- Verify the `BOOTSTRAP_ADDRESS` port assignment for `server1` in the `serverindex.xml` file

If the `BOOTSTRAP_ADDRESS` port of the earlier version is 900, migration maps this to 2809. If the `BOOTSTRAP_ADDRESS` port of the earlier version is not 900, migration maps the value to `server1` in an Advanced Edition migration, or to the actual server name in an Advanced Single Server Edition migration.

- Verify the HTTP Transport port assignments in the `server.xml` file

WASPostUpgrade processing adds the HTTP Transport ports from the earlier version to the Version 5.1 `server.xml` file. This means that `server1` contains duplicate HTTP Transport port assignments, from both the coexistence panel and the previous version *Default Server*.

4. Migrate the previous configuration to the new installation with the WASPostUpgrade tool in the `install_root/bin` directory of the V5.1 installation. The WASPostUpgrade tool migrates V3.5.x or V4.0.x configuration information created by the WASPreUpgrade tool to the V5.1 installation. Because the V5.1 product adheres to the J2EE programming model and V3.5.x does not, significant changes are required to apply the V3.5.x configuration to a V5.1 installation.

The WASPostUpgrade tool does not migrate Samples or the administrative console application because there are already Samples and an administrative console application in V5.1. The WASPostUpgrade tool records detailed information specific to each enterprise bean it deploys, in the `WASPostUpgrade.log` file.

5. Stop the administrative server of the earlier version if it is running, before running the V5.1 node.
6. Configure WebSphere Application Server after migration. Configuring WebSphere Application Server after migration is a way of verifying the results of the migration tools. You can also use Configuration mapping during migration to verify the results of the migration. The topic has a detailed description of how the migration tools migrate objects, and what you should verify.

You can use the migration tools to perform a manual migration from Version 3.5.x or Version 4.0.x of WebSphere Application Server to Version 5.1.

After you test and verify that the applications and configuration data you moved to the V5.1 node is successful, you can uninstall the V3.5.x or the V4.0.x Application Server as described in the information centers for those releases. Click the **Library** link at the bottom of any V5.1 information center topic to locate the information centers for the other releases.

Migrating Version 3.5.x or Version 4.0.x of WebSphere Application Server to a remote Version 5.1 machine

You can use the migration tools to perform a manual migration between two machines.

Typically you can use the WASPreUpgrade and the WASPostUpgrade migration tools from V5.1 of WebSphere Application Server to upgrade from either V3.5 or V4.0 to V5.x on the same machine.

However, some scenarios require that you migrate the V3.5 or the V4.0 configuration on one machine to V5.x on a different machine. One of these scenarios is when you install new machines for your latest V5.1 environment but need to migrate your existing V3.5.x or V4.0.x configuration from other machines.

This topic describes using the V5.1 migration tools to migrate the following products:

- WebSphere Application Server Single Server Edition, V3.5
- WebSphere Application Server Advanced Edition, V3.5
- WebSphere Application Server Advanced Edition, V4.0
- WebSphere Application Server Advanced Single Server Edition, V4.0 (the steps vary slightly)

The WASPreUpgrade tool saves the existing V3.5 or V4.0 configuration into a *migration-specific-backup* directory. The WASPostUpgrade tool uses this directory to add the old configuration settings to the new V5.1 environment.

1. Obtain the V5.1 product CD-ROM. On this CD is the `migration/bin` directory. This directory contains a special environment that you can use to run the WASPreUpgrade tool without installing V5.1.

2. Save the current configuration using the WASPreUpgrade script from the `/migration/bin` directory of the V5.1 product CD-ROM, which you must mount to the V3.5 or V4.0 machine. Save the configuration in the `migration-specific-backup` directory on the V3.5 or V4.0 machine.

```
WASPreUpgrade /opt/tmp/migration-specific-backup /opt/websphere/appserver adminNodeName
```

For all scenarios except V4.0.x Advanced Single Server Edition, verify that the administrative server of the existing environment is running. The WASPreUpgrade tool provides status to the screen and to log files in the `migration-specific-backup` directory. ASCII log file names start with the text WASPreUpgrade and include a date and timestamp.

The WASPreUpgrade tool saves selected files from the V3.5.x or V4.0.x `/bin` directory. It also exports the existing Application Server configuration from the V3.5.x or V4.0.x repository. The WASPreUpgrade tool calls XMLConfig to export the existing V3.5 or V4.0 repository to the `websphere_backup.xml` file in the `migration-specific-backup` directory.

V4.0.x Advanced Single Server Edition does not require the administrative server to run at the time of migration. The WASPreUpgrade tool copies the `server-cfg.xml` file from the `install_root/config` directory to the `migration-specific-backup/config` directory.

If errors occur while running the WASPreUpgrade tool, you might have to apply fixes to the V3.5 or V4.0 installation to successfully complete the export step. See the IBM Support page for the latest fixes that might be applicable. When viewing this information from the information center, click **Support** to link to the IBM Support page.

3. Copy the `migration-specific-backup` directory from the V3.5 or V4.0 machine to the V5.x machine. Use the **ftp** command, shared storage, or some other mechanism to copy the file to the new machine. Perform the following steps on the machine with V5.x of WebSphere Application Server.
4. Copy the `migration-specific-backup/websphere_backup.xml` or the `migration-specific-backup/config/server-cfg.xml` file and store it as an archive. You edit the original file in the next step.
5. Edit the `migration-specific-backup/websphere_backup.xml` or the `/migration-specific-backup/config/server-cfg.xml` file to correct machine-dependent settings. Make the following changes in the file:
 - a. Change the node name in the `migration-specific-backup/websphere_backup.xml` file. There is no node name in the `migration-specific-backup/config/server-cfg.xml` file. If you are using the same node name for the V5.x machine that you use for the original V3.5 or the V4.0 configuration, do not change the name. Otherwise, you must change all occurrences of the V3.5 or the V4.0 node name to the node name you are using for WebSphere Application Server V5.1. The node name occurs in many XML stanzas throughout the file. Failing to change all occurrences results in an incomplete migration of data.
 - b. Change the path names in the `migration-specific-backup/websphere_backup.xml` or the `migration-specific-backup/config/server-cfg.xml` file. The configuration file refers to path names in many XML stanzas throughout the file. Update any reference to a file outside of the V3.5 or V4.0 directory structure to the equivalent directory on the new machine, even if you must create an equivalent directory. The implication of configuring a matching environment means that you might have to copy the original directory to the V5.1 machine. Or you might have to install the appropriate software.
 - c. Check files in the `properties` directories for references that contain path names. In particular, edit the `migration-specific-backup/properties/sas.client.props` and the `migration-specific-backup/properties/TraceSettings.properties` files to correct machine-dependent settings: Make the following changes in the file:
 - 1) Change the path values of any property in the file.
Each property file contains properties that refer to paths. Update any reference to a file outside of the V3.5 or V4.0 directory structure to the equivalent directory on the new machine, even if you must create an equivalent directory.
 - 2) Correct specification styles for path values that are dependent on the operating system.
You must correct path specifications if they differ from what works on the machine running V5.1.

- d. Correct specification styles for path names that are dependent on the operating system. You must correct path specifications if they differ from what works on the machine running V5.1. For example, if you are moving from V3.5.x or V4.0.x on a Windows platform to V5.1 on a Linux platform, change any Windows-specific path in the configuration file to use the Linux path style. Change `c:\mystuff\somopath` to `/opt/mystuff/somopath`.
 - e. Change user IDs and passwords to match security requirements. You might have to change user IDs and passwords if they are not identical to those in use on the V5.1 machine.
To change an encoded password to a clear-text password, change `<password>{xor}LCoxayht</password>` to `<password>mypassword</password>`.
 - f. Change other machine-specific information. The configuration might refer to other software products or configurations that do not exist on the new machine. For example, the old machine might have a database. The V5.x configuration should still point to the database on the old machine, possibly. Modify the data source to point to the database on the V3.5 or the V4.0 machine.
6. Install V5.1 of the WebSphere Application Server without selecting the migration option.
 7. Add the V3.5 or the V4.0 configuration to the V5.x configuration. Use the WASPostUpgrade tool in the `install_root/bin` directory of V5.x to add the V3.5 or the V4.0 configuration to the V5.x configuration.
`WASPostUpgrade /opt/tmp/migration-specific-backup`

The WASPostUpgrade tool records detailed information specific to each enterprise bean it deploys, in the `migration-specific-backup/WASPostUpgrade.log` file.

8. Configure WebSphere Application Server after migration. Configuring WebSphere Application Server after migration is a way of verifying the results of the migration tools. You can read Configuration mapping during migration to learn more about the results of migration. This topic has a detailed description of how the migration tools migrate objects, and what you should verify.

You can migrate WebSphere Application Server from V3.5.x or V4.0.x to a remote V5.1 machine.

Migrating Version 5.0.x of WebSphere Application Server to a remote Version 5.1 machine

You can use the migration tools to migrate between two machines.

Typically you can use the WASPreUpgrade and the WASPostUpgrade migration tools from WebSphere Application Server, Version 5.1 to upgrade from Version 5.0.x to Version 5.1 on the same machine.

However, some scenarios require that you migrate the V5.0.x configuration on one machine to V5.1 on a different machine. One of these scenarios is when you install new machines for your V5.1 environment but need to migrate your existing V5.0.x configuration from other machines.

If the node you are migrating is part of a deployment manager cell, migrate the V5.0.x deployment manager to V5.1 migrate the V5.0.x deployment manager to V5.1 as described in the information center for Network Deployment first, before continuing this procedure. The deployment manager node must always be at the highest release level within the cell.

This task describes how to use the V5.1 migration tools to migrate WebSphere Application Server, Version 5.0.x to Version 5.1 on a separate machine.

The WASPreUpgrade tool saves the existing V5.0.x configuration into a `migration-specific-backup` directory. The WASPostUpgrade tool uses this directory to add the old configuration settings to the new V5.1 environment.

1. Obtain the V5.1 product CD-ROM. On this CD is the `migration/bin` directory. This directory contains a special environment that you can use to run the WASPreUpgrade tool without installing V5.1.
2. Save the current configuration using the WASPreUpgrade script from the `migration/bin` directory of the V5.1 product CD-ROM, which you must mount to the V5.0.x machine. Save the configuration in the `migration-specific-backup` directory on the V5.0.x machine.

```
WASPreUpgrade /opt/tmp/migration-specific-backup /opt/websphere/appserver
```

The WASPreUpgrade tool provides status to the screen and to log files in the *migration-specific-backup* directory. ASCII log file names start with the text WASPreUpgrade and include a date and timestamp.

3. Copy the *migration-specific-backup* directory from the V5.0.x machine to the V5.1 machine. Use the **ftp** command, shared storage, or some other mechanism to copy the directory to the new machine.
4. Install V5.1 of WebSphere Application Server without selecting the migration option. Install the same features as the earlier release.
5. Add the V5.0.x configuration to the V5.1 configuration. Use the WASPostUpgrade tool in the `install_root/bin` directory of the V5.1 installation to add the V5.0.x configuration to the V5.1 configuration.

```
WASPostUpgrade /opt/tmp/migration-specific-backup
```

The WASPostUpgrade tool records detailed information specific to each enterprise bean it deploys, in the *migration-specific-backup/WASPostUpgrade.log* file.

6. Modify the configuration using the WebSphere Application Server 5.1 administration interfaces. Make these changes:
 - a. Change user IDs and passwords to match security requirements. You might have to change user IDs and passwords if they are not identical to those in use on the V5.0.x machine.
 - b. Change other machine-specific information. The configuration might refer to other software products or configurations that do not exist on the new machine. For example, the old machine might have a database. Modify the data source to point to the database on the old machine.

You can migrate WebSphere Application Server from V5.0.x to a remote V5.1 machine.

Migrating Version 5.0.x of WebSphere Application Server to Version 5.1

The migration tools of WebSphere Application Server, Version 5.1 support migrating configuration data from V5.0.x to V5.1.

This task describes how to migrate the configuration of a V5.0.0 or a V5.0.1 node without the embedded messaging feature installed, or a V5.0.2 node with or without embedded messaging, to V5.1. This task describes how to use the V5.1 migration tools to perform the migration.

If the node you are migrating is part of a deployment manager cell, migrate the V5.0.x deployment manager to V5.1 first, before continuing this procedure. The deployment manager node must always be at the highest release level within the cell.

If your scenario includes migrating a V5.0.x configuration on one machine to WebSphere Application Server V5.1 on another machine, use the alternate procedure described in “Migrating Version 5.0.x of WebSphere Application Server to a remote Version 5.1 machine” on page 335.

If your scenario includes migrating V5.0.0 or V5.0.1 WebSphere Application Server with embedded messaging installed, use the alternate procedure described in Migrating V5.0.0 or V5.0.1 WebSphere Application Server with embedded messaging to V5.1..

1. Stop the V5.0.x deployment manager and start the V5.1 migrated deployment manager if the deployment manager is not already running.

During migration the V5.1 deployment manager must be running for the migration tools to:

- Update the configuration for each federated node
- Request full synchronization

If the V5.1 deployment manager is not running, failures can occur.

Table 64. Migration tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Recovering from configuration errors when the deployment manager was not running during migration.

2. Stop all of the V5.0.x Application Servers that are running on the node. Use the stopServer command from the *install_root/bin* directory. For example, issue the following commands to stop server1 and server2 on a Linux platform:

```
stopServer.sh server1
stopServer.sh server2
```

If security is enabled, specify the `-user` and `-password` parameters on the **stopServer** command.

You can migrate a V5.0.x node without stopping it. But it is not necessary to have the node running to migrate its configuration. The migration tools can retrieve all the configuration data while the node is stopped. You must stop the node before you can start the V5.1 node that you are installing.

3. Stop the messaging service provider, if the embedded messaging server feature is installed. Run the **stopServer** command from the *install_root/bin* directory of the V5.0.x Application Server. For example, issue the following command to stop the jmsserver process on a Linux platform:

```
stopServer.sh jmsserver
```

If security is enabled, specify the `-user` and `-password` parameters on the **stopServer** command.

4. Stop the node agent process. Run the stopNode command from the *install_root/bin* directory of the V5.0.x Application Server. For example, issue the following command to stop the nodeagent process on a Linux platform:

```
stopNode.sh
```

If security is enabled, specify the `-user` and `-password` parameters on the **stopNode** command.

5. Install the V5.1 product. Select the migration option when it appears. If the migration option does not appear, cancel the installation.

In some cases, such as when installing a non-English version, the installation wizard might not detect a previous version. You can force the migration panel to appear, by starting the installation with an option on the **Install.exe** or the **install** command.

For example, on Linux and UNIX-based platforms, use this command from the CD-ROM mount point:

```
install -W previousVersionDetectedBean.previousVersionDetected= "true"
```

On the migration panel, specify the same value for the node name that the V5.0.x product uses. The V5.0.x product and the V5.1 product must have identical node names. The host names can be identical but it is not a requirement.

The installer program calls the WASPreUpgrade migration tool before installing the new product, to save the configuration from the old product in a backup directory. The installer program calls the WASPostUpgrade migration tool after installing the new product, backing up the configuration of the new product and then adding the configuration of the old product from the backup directory to the configuration of the new product.

Read the messages from the migration tools that appear on the installer program panels. Each migration tool also records events in its log file.

6. Close the First Steps tool when it appears. Do not verify the installation of the V5.1 Application Server because of node differences when migrating a federated node.
7. Start the node agent for the V5.1 node. Your newly migrated V5.1 node is federated if the V5.0.x node is federated. Use the startNode command from the *install_root/bin* directory of the V5.1 Application Server. For example, issue the following command to start the nodeagent process on a Linux platform:

```
startNode.sh
```

If security is enabled, specify the `-user` and `-password` parameters on the **startNode** command.

8. Start all V5.1 Application Servers. Issue the startServer command from the install_root/bin directory of the V5.1 installation. For example, issue the following commands to start server1 and server2 on a Linux platform:

```
startServer.sh server1  
startServer.sh server2
```

If security is enabled, specify the -user and -password parameters on the **startServer** command.

The server1 and server2 Java processes begin.

9. Start the messaging service provider if the node is part of a deployment manager cell and the embedded messaging server feature is installed. Issue the following command from the install_root/bin directory of the V5.1 installation.

```
startServer.sh jmsserver
```

If security is enabled, specify the -user and -password parameters on the **startServer** command.

The jmsserver Java process begins.

You can migrate a Version 5.0.x node to Version 5.1.

Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1:

You can use the migration tools of WebSphere Application Server, Version 5.1 to migrate from V5.0.0 or V5.0.1 to V5.1, when V5.0.0 or V5.0.1 has the embedded messaging feature installed.

This task describes how to migrate V5.0.0 or V5.0.1 with the embedded messaging feature installed, to V5.1. If V5.0.0 or V5.0.1 does not have the embedded messaging feature installed, see “Migrating Version 5.0.x of WebSphere Application Server to Version 5.1” on page 336.

Typically you can use the WASPreUpgrade and WASPostUpgrade migration tools from V5.1 of WebSphere Application Server to upgrade from V5.0.0 or V5.0.1 to V5.1 on the same machine. However, if your scenario includes migrating a V5.0.0 or V5.0.1 configuration on one machine to WebSphere Application Server V5.1 on another machine, use the alternate procedure described in “Migrating Version 5.0.x of WebSphere Application Server to a remote Version 5.1 machine” on page 335.

If the node you are migrating is part of a deployment manager cell, migrate the V5.0.x deployment manager to V5.1 first, before continuing this procedure. The deployment manager node must always be at the highest release level within the cell.

If you prefer, you can upgrade from V5.0.0 or V5.0.1 to V5.0.2 and use the migration function of the V5.1 installer to migrate from V5.0.2 to V5.1. If you prefer not to upgrade, you can use this procedure to manually migrate V5.0.0 or V5.0.1 to V5.1, when you have the embedded messaging feature installed on the V5.0.0 or V5.0.1 node.

A few problems can occur when you migrate a node that has the server function of the embedded messaging feature. The problems occur because the version of the embedded messaging feature that runs on V5.0.0 and V5.0.1 is not the same version (CSD04) that runs on V5.0.2 and V5.1.

1. Obtain the V5.1 product CD-ROM. On this CD is the migration/bin directory. This directory contains a special environment that you can use to run the WASPreUpgrade tool without installing V5.1.
2. Save the current configuration using the WASPreUpgrade script from the /migration/bin directory of the V5.1 product CD-ROM. Save the configuration in the *migration-specific-backup* directory.

```
WASPreUpgrade /opt/tmp/migration-specific-backup /opt/websphere/appserver
```

The WASPreUpgrade tool provides status to the screen and to log files in the */migration-specific-backup* directory. ASCII log file names start with the text WASPreUpgrade and include a date and timestamp.

3. If the V5.0.0 or V5.0.1 node is federated, stop the V5.1 deployment manager and the V5.0.x deployment manager. Run the stopManager command from the *install_root/bin* directory of each deployment manager node. For example, use the following command to stop the deployment manager process on a Linux or UNIX-based platform:

```
./stopManager.sh
```

The script stops the deployment manager, which prevents either deployment manager from removing the base node from the cell configuration, when you uninstall the base node in the next step.

4. Optional: Uninstall the V5.0.0 or V5.0.1 node. Use the following steps to uninstall the node:
 - a. Run the uninstall command from the *install_root_uninst* directory of the product that you are uninstalling.
 - b. Run the platform-specific uninstalling procedure to complete the removal by removing all artifacts of the earlier product from the system.
5. If the base node is a federated node, start the V5.1 deployment manager process. Run the startManager command from the *install_root/bin* directory of the V5.1 deployment manager node. For example, use the following command to start the deployment manager process on a Linux or UNIX-based platform:

```
startManager.sh
```

6. Install V5.1 of WebSphere Application Server without selecting the migration option, if it appears.
7. Add the V5.0.0 or V5.0.1 configuration to the V5.1 configuration. Use the WASPostUpgrade tool in the *install_root/bin* directory of the V5.1 installation to add the V5.0.0 or V5.0.1 configuration to the V5.1 configuration.

```
WASPostUpgrade /opt/tmp/migration-specific-backup
```

The WASPostUpgrade tool records detailed information specific to each enterprise bean it deploys, in the */migration-specific-backup/WASPostUpgrade.log* file.

8. Check the status of the migration. During migration the V5.1 deployment manager must be running for the migration tools to:
 - Update the configuration for each federated node
 - Request full synchronization

If the V5.1 deployment manager is not running, failures can occur.

Table 65. Migration tip

Operating platform	Tip in “Platform-specific tips for installing and migrating” on page 131
All platforms	Recovering from configuration errors when the deployment manager was not running during migration.

You can perform a manual migration instead of using the installation wizard to automatically migrate from V5.0.0 or V5.0.1.

Migrating from an operating system that is no longer supported

You can migrate an earlier WebSphere Application Server Version 3.5.x or Version 4.0.x release that is running on an operating system that Version 5.x does not support.

1. Start up the WebSphere Application Server Version 3.5.x or Version 4.0.x Administrative Server.
2. Run the **WASPreUpgrade** command line migration tool. There are two options. You can run the command from the *migration\bin* (or *migration/bin*) directory in the *platform_root* of the Version 5 CD-ROM. Or, you can copy the files in the directory on the CD-ROM to a directory you create on your hard drive.

Identify the Version 3.5.x or 4.0.x release, and identify a backup directory where the command stores configuration files and migrating applications from the earlier version. See the *WASPreUpgrade* topic for command syntax.

- a. Run the command from the `migration\bin` (or `migration/bin`) directory in the *platform_root* of the Version 5 CD-ROM. Identify the backup directory and the location of the configuration files.

```
CD_drive:\WASPreUpgrade backupDirectory filepath\WebSphere\AppServer yourNodeName
```

If this works, go to Step 4. If this does not work for some reason, perform steps 2B through 2F.

- b. Make a **migration** directory on your hard drive.
- c. Copy the `WASPreUpgrade.bat` (or `WASPreUpgrade.sh`) and the `setupCmdLine.bat` (or `setupCmdLine.sh`) files from the `migration/bin` directory in the *platform_root* of the Version 5 CD-ROM, to the directory you created on your hard drive.
- d. Edit the `setupCmdLine.bat` (or `setupCmdLine.sh`) file in your new directory. Change the following variables:
 - **WAS_HOME** to point to the fully qualified path to the migration directory you created
 - **JAVA_HOME** to point to the fully qualified path to your IBM Developer Kit or the Java directory
- e. Ensure that the executable bit is on for the `setupCmdLine.sh` and `WASPreUpgrade.sh` files in the directory that you created on your hard drive.
- f. Run the command from the migration directory you created. Identify the backup directory and the location of the configuration files.

```
yourMigrationDirectory\WASPreUpgrade backupDirectory filepath\WebSphere\AppServer yourNodeName
```

3. Shut down the WebSphere Application Server Version 3.5.x or Version 4.0.x release by stopping all server nodes in the configuration, including the administrative server node.
4. Tar or zip the backup directory and FTP it to another system.
5. Install the new operating system, keeping the same host name. If possible, keep the system name and passwords the same as the old system. Place any database files related to applications you are migrating in the same path as the previous system. In general, try to keep paths the same. However, do not install Version 5 in the same directory as the previous version. If you do change paths and names, you can edit the XML configuration files to reflect the changes. Make such changes before running the **WASPostUpgrade** command below.
6. FTP the backup directory from the other system and unzip it.
7. Install WebSphere Application Server, Version 5. Do not select the migration option, if it appears.
8. Run the **WASPostUpgrade** command line migration tool, from the `bin` directory of the Version 5 `install_root`. Specify the backup directory (and any non-standard configuration file name in the directory) that the **WASPreUpgrade** command created. See the *WASPostUpgrade* topic for proper command syntax.

```
install_root\bin\WASPostUpgrade WAS_HOME\migration
```

Migration tools

This topic introduces the migration tools that WebSphere Application Server provides. All of the migration tools are in the `install_root/bin` directory after installation. The `WASPreUpgrade.sh` or `WASPreUpgrade.bat` scripts also ship in the `/migration/bin` directory on the product CD-ROM so that you can store the configuration of an existing release before installing the V5.1 product. It is important to use the migration tools for the version of Application Server that you are installing. The tools change over time. The tools on the product CD-ROM provide the necessary function for migrating from a previous release of Application Server to the one on the product CD-ROM. The tools on the CD-ROM match the product on the CD-ROM. If you use migration tools from an earlier release of Application Server, you are likely to encounter a problem with the migration.

clientUpgrade.sh (and clientUpgrade.bat)

Upgrades the client application to a new release level.

pre_uninst50ws.sh (and pre_uninst50ws.bat)

Prevents a federated node that you migrate to a new version, from removing the new node from the cell when you uninstall the old node.

post_uninst50ws.sh (and post_uninst50ws.bat)

Restores the normal uninstaller environment that allows the uninstaller to remove a base node from a cell prior to uninstalling the base node.

pre_uninst502mq.sh (and pre_uninst502mq.bat)

Prevents the uninstaller from uninstalling the embedded messaging feature and the message queue when it uninstalls a V5.0.2 node that you have migrated to V5.1.

post_uninst502mq.sh (and post_uninst502mq.bat)

Restores the normal uninstaller environment that allows the uninstaller to uninstall a base V5.0.2 node and delete the embedded messaging feature and its queue manager.

WASPreUpgrade.sh (and WASPreUpgrade.bat)

Saves the applications and configuration data from a previous installation of WebSphere Application Server to a backup directory. The WASPostUpgrade script restores the configuration data from the directory to the new installation. The installer calls the WASPreUpgrade.sh script during installation, if you select migration. You can also use the command to perform a manual migration, after installing the new version.

WASPostUpgrade.sh (and WASPostUpgrade.bat)

Restores the configuration data from a previous release. WASPostUpgrade reads the data from the backup directory where the WASPreUpgrade script stored the data. The installer calls the WASPostUpgrade.sh script during installation, if you select migration. You can also use the command to perform a manual migration, after installing the new version.

The clientUpgrade command:

Use the **clientUpgrade** command to migrate application clients.

Note: WebSphere Application Server Version 5.1 does not support the Windows NT platform.

The command file is located in the bin subdirectory of the *WAS_install_root*, or the *ND_install_root* directory. By default, the *WAS_install_root* for WebSphere Application Server and WebSphere Application Server Enterprise is:

- Windows NT or Windows 2000 operating platforms - *drive*\WebSphere\AppServer directory
- AIX or UNIX-based operating platforms - /usr/WebSphere/AppServer directory

By default, the *ND_install_root* for WebSphere Application Server Network Deployment is:

- Windows NT or Windows 2000 operating platforms - *drive*\WebSphere\DeploymentManager directory
- AIX or UNIX-based operating platforms - /usr/WebSphere/DeploymentManager directory

Linux and UNIX-based platforms:

```
clientUpgrade.sh EAR_file [-clientJar client_jar ]
                  [-traceString trace_spec [-traceFile file_name ]]
```

Windows platforms:

```
clientUpgrade.bat EAR_file [-clientJar client_jar ]
                  [-traceString trace_spec [-traceFile file_name ]]
```

Parameters

Supported arguments include the following:

EAR_file

Use this parameter to specify the fully qualified path to the EAR file that contains client JAR files to process.

-clientJar

Use this optional parameter to specify a JAR file for processing. If not specified, the program transforms all client JAR files in the EAR file.

-traceString -traceFile

Use these optional parameters to gather trace information for IBM Service personnel. Specify a trace_spec of "`*=all=enabled`" (with quotation marks) to gather all trace information.

-logFileLocation

Use this optional parameter to specify an alternate location to store the log output.

The following example demonstrates correct syntax:

```
clientUpgrade EAR_file -clientJar ejbJarFile
```

pre_uninst50ws command:

The **pre_uninst50ws** command is a V5.1 migration tool for use when you migrate a federated V5.0.x node to V5.1. The script changes the normal uninstaller program environment to prevent the uninstaller program from removing the V5.0.x or V5.1 node from the deployment manager cell while uninstalling the base V5.0.x or V5.1 node. The script also prevents the uninstaller program from removing the jmsserver process, which is the embedded messaging queue manager for the federated node.

A problem can occur when you uninstall an earlier version of the base WebSphere Application Server product that you migrated to Version 5.1. If the earlier node is federated, migrating the federated node to Version 5.1 gives you a federated V5.1 node, too. The new node has the same node name as the earlier node, and is a member of the same cell. The problem is that the uninstaller program issues a **removeNode** command when it uninstalls a federated node. So, if you uninstall the earlier version, the uninstaller issues a call to the deployment manager that requests the removal of the node from the cell. You are left with a V5.1 node that is configured as if it were a member of the cell and a deployment manager that does not recognize the node.

The `pre_uninst50ws` script overcomes the problem by renaming the `_nodeuninst.sh` or the `_nodeuninst.bat` script and creating a dummy script that prevents the uninstaller program from logging an error and stopping. Because the uninstaller program is not calling the script that removes the node from the cell, uninstalling the earlier version does not also unfederate the node from the cell. This lets you uninstall the previous node and still have a federated V5.1 node.

The "post_uninst50ws command" on page 344 restores the environment to allow the uninstaller program to remove a federated node from its cell while uninstalling the base node. If you are completely uninstalling the previous version, it is not necessary to run the `post_uninst50ws` script. If you follow the platform-specific instructions for uninstalling the previous version, you remove the `_nodeuninst.sh` or the `_nodeuninst.bat` script. So there is nothing left for the script to rename.

Location of the command file

The command file is located in the `install_root/bin` directory of the V5.1 node.

If the installation wizard performs the initial migration, the wizard copies the scripts to the `install_root/bin` directory of the V5.0.x node that it is migrating. If you migrate the node by calling the migration tool commands, you might have to copy the following files to the `install_root/bin` directory of the V5.0.x node from the `install_root/bin` directory of the V5.1 node:

- `pre_uninst50ws.sh` and `post_uninst50ws.sh` on a Linux or UNIX-based platform
- `pre_uninst50ws.bat` and `post_uninst50ws.bat` on a Windows platform

Recovering when you do not use the `pre_uninst50ws` script

If you uninstall a federated base node without running the `pre_uninst50ws.sh` script, you can recover by recreating whatever configuration data is lost. Lost configuration data includes any configuration that occurred while the original node was federated. Use the `addNode.sh` or the `addNode.bat` script in the `bin` directory of the installation root of the base Application Server node to add the new node back to the deployment manager cell. Do not use the `-includeapps` parameter to include applications from the node. The installer program included the applications into the deployment manager cell during installation. The `removeNode.sh` script does not delete applications from the cell.

Using the command when removing a V5.0.x node

The script is new as of Version 5.1.

1. Copy the script files described previously if the `pre_uninst50ws` command file is not in the `install_root/bin` directory of the WebSphere Application Server, Version 5.0.x instance.
2. Run the `pre_uninst50ws` command from the `install_root/bin` directory of the WebSphere Application Server, Version 5.0.x instance.
3. Uninstall the WebSphere Application Server, Version 5.0.x instance.

Run the `post_uninst50ws` script from a node only when you have run the `pre_uninst50ws` script on the node and decide not to uninstall the node.

Using the command when removing a V5.1 node

Use the `pre_uninst50ws` script if you uninstall V5.1 before uninstalling V5.0.2, to make V5.0.2 operational again.

For example, suppose you install V5.1 in spite of missing patches on your operating system and there are problems that require you to uninstall V5.1 to reinstall it. In the meantime, you bring up your V5.0.2 node to have a running WebSphere Application Server node. So you start up the V5.0.2 deployment manager, and the V5.0.2 node before uninstalling V5.1.

Uninstalling V5.1 removes the node from the deployment manager cell if and only if the V5.1 deployment manager is running. If the V5.0.2 deployment manager is running, the `remove-node` part of the uninstalling procedure fails because a V5.1 node cannot communicate with a V5.0.2 deployment manager. The V5.1 `jmsserver` process is not removed either.

However, if you start a V5.1 deployment manager, start the V5.0.2 node, and then uninstall the V5.1 node without running the `pre_uninst50ws` script, there are problems. Removing the V5.1 node causes the same effect as removing the V5.0.2 node. The deployment manager removes the configuration of the node. Follow this procedure to remove the V5.1 node without a problem.

1. Run the `pre_uninst50ws` script from the `install_root/bin` directory of the WebSphere Application Server, Version 5.1 instance.
2. Run the `pre_uninst502mq` script from the `install_root/bin` directory of the WebSphere Application Server, Version 5.0.2 instance if the embedded messaging feature is installed.
3. Uninstall the WebSphere Application Server, Version 5.1 instance.

Run the `post_uninst50ws` script from a node only when you have run the `pre_uninst50ws` script on the node and decide not to uninstall the node.

`pre_uninst50ws.sh` command syntax for Linux and UNIX-based platforms

The command syntax is as follows:

```
pre_uninst50ws.sh
```

pre_uninst50ws.bat command syntax for Windows platforms

pre_uninst50ws.bat

Parameters

No parameters are associated with this command.

Migrating from a V5.0.x base node to a V5.1 node

Issue the **pre_uninst50ws** command before uninstalling the V5.0.x product, from the *install_root/bin* directory of the product you are uninstalling. This command renames the `_nodeuninst.sh` or the `_nodeuninst.bat` script and creates a dummy, non-operational script. For example, on a Linux platform, issue the following command:

```
pre_uninst50ws.sh
```

Workaround for uninstalling a federated node without the scripts

It is possible to use the WASPreupgrade tool to save the configuration of a V5.0.x instance before installing the V5.1 instance. Such a scenario is described in “Migrating Version 5.0.0 or Version 5.0.1 of WebSphere Application Server with embedded messaging to Version 5.1” on page 338. In such a scenario, you do not have access to the following files:

- `pre_uninst50ws.sh` or `pre_uninst50ws.bat`
- `post_uninst50ws.sh` or `post_uninst50ws.bat`

You can perform either of two workarounds to avoid removing the node from the cell:

- Stop the deployment manager so that it does not lose the configuration for the node that you are uninstalling.
- Rename the program on the node that you are uninstalling to prevent the node from requesting that the deployment manager remove the node from the cell:
 1. Rename the `_nodeuninst.sh` or the `_nodeuninst.bat` script.
 2. Create a dummy script named `_nodeuninst.sh` or `_nodeuninst.bat` that is not operational. The dummy script can be an empty file. This prevents the uninstaller program from logging an error and stopping.
 3. Uninstall the V5.0.x product.
 4. Delete the dummy file and restore the `_nodeuninst.sh` or the `_nodeuninst.bat` script.

post_uninst50ws command:

The **post_uninst50ws** command restores the normal uninstaller program environment in which the uninstaller program removes a federated base node from a cell prior to uninstalling the base WebSphere Application Server node. Run the `post_uninst50ws` script from a node only when you have run the `pre_uninst50ws` script on the node and decide not to uninstall the node.

Overview

The `post_uninst50ws` tool restores the environment created by the use of the “`pre_uninst50ws` command” on page 342. The `post_uninst50ws` tool allows the uninstaller program to remove a federated node from its cell while uninstalling the base node.

If you are completely uninstalling the previous version, it is not necessary to run the `post_uninst50ws` script. If you follow the platform-specific instructions for uninstalling the previous version, as described in “Uninstalling manually” on page 292, you remove the `_nodeuninst.sh` or the `_nodeuninst.bat` script along with all other artifacts. So there is nothing left for the script to rename.

Use the `post_uninst50ws` migration tool to reverse the effects of the `pre_uninst50ws` tool when you run the `pre_uninst50ws` tool and then decide not to uninstall the product.

Location of the command file

The command file is located in the `install_root/bin` directory of the V5.1 node.

If the installation wizard performs the initial migration, the wizard copies the scripts to the `install_root/bin` directory of the V5.0.x node that it is migrating. If you migrate the node by calling the migration tool commands, you might have to copy the following files to the `install_root/bin` directory of the V5.0.x node from the `install_root/bin` directory of the V5.1 node:

- `pre_uninst50ws.sh` and `post_uninst50ws.sh` on a Linux or UNIX-based platform
- `pre_uninst50ws.bat` and `post_uninst50ws.bat` on a Windows platform

`post_uninst50ws.sh` command syntax for Linux and UNIX-based platforms

The command syntax is as follows:

```
post_uninst50ws.sh
```

`post_uninst50ws.bat` command syntax for Windows platforms

```
post_uninst50ws.bat
```

Parameters

No parameters are associated with this command.

***pre_uninst502mq* command:**

The **`pre_uninst502mq`** command is a V5.1 migration tool for use when you migrate a V5.0.2 node with embedded messaging to V5.1. The script changes the normal uninstaller program environment to prevent the uninstaller program from removing the embedded messaging feature while uninstalling the base V5.0.2 node. Normally the uninstaller program removes the embedded messaging feature while uninstalling the product.

When you migrate the V5.0.2 base node to V5.1, the two installation instances share the same node name, the same embedded messaging feature code, and the same messaging queue manager. Uninstalling either V5 installation instance removes the instance and the embedded messaging feature. The `pre_uninst502mq` script prevents the removal of embedded messaging when the V5.0.2 installation instance is uninstalled. The V5.1 uninstaller program has an option that you can use to prevent the removal of the embedded messaging feature.

The uninstaller program normally stops and removes the default embedded messaging queue manager process before uninstalling the feature. If another embedded messaging process is running on a Linux or UNIX-based machine, the uninstaller program cannot remove the embedded messaging feature. The `pre_uninst502mq` script creates the `WAS_COEXIST` queue manager to prevent the removal of the embedded messaging feature code on Linux and UNIX-based platforms. The uninstaller program does not remove the `WAS_COEXIST` queue manager, which is active and therefore causes the uninstaller program to not remove the embedded messaging feature code.

Location of the command file

The command file is located in the `install_root/bin` directory. The command file is a script named `pre_uninst502mq.sh` on Linux and UNIX-based platforms and `pre_uninst502mq.bat` on Windows platforms.

pre_uninst502mq.sh command syntax for Linux and UNIX-based platforms

The command syntax is as follows:

```
pre_uninst502mq.sh
```

pre_uninst502mq.bat command syntax for Windows platforms

```
pre_uninst502mq.bat
```

Parameters

No parameters are associated with this command.

Migrating a V5.0.2 base node to a V5.1 node

Issue the **pre_uninst502mq** command after migrating the configuration of a V5.0.2 node that has the embedded messaging feature installed, installing the V5.1 product, and before uninstalling the V5.0.2 node. After issuing the command, uninstall the V5.0.2 node. Because you are using the `pre_uninst502mq` script, uninstalling the V5.0.2 node prevents the removal of the embedded messaging feature code.

After using the `pre_uninst502mq` script and uninstalling the V5.0.2 product on Linux and UNIX-based platforms, perform this procedure:

1. Run the `post_uninst502mq` script to remove the WAS_COEXIST queue manager.
2. On Linux and UNIX-based platforms, run the following commands to verify that the WAS_COEXIST queue manager is deleted. Run the commands as root from the `install_root/bin` directory:

```
endmqm WAS_COEXIST  
dltmqm WAS_COEXIST
```
3. Run the `createmq` command from the `install_root/bin` directory to create the default queue manager:

UNIX platforms:

```
createmq.sh install_root  
          cellName  
          nodeName  
          server1
```

Windows platforms:

```
createmq.bat install_root  
          cellName  
          nodeName  
          server1                  mq_dir  
          broker_dir
```

The command that the installer program used to create your original queue manager is in the `install_root/logs/createMQ.nodeName_server1.log` file.

To create the default queue manager on a Linux node named `wasnode`, for example, run the following command:

```
createmq.sh /opt/WebSphere/AppServer wasnode wasnode server1
```

On an unfederated node, the value of the cell name parameter is the node name.

post_uninst502mq command:

The **post_uninst502mq** command restores the normal uninstaller program environment, which lets the uninstaller program uninstall a base V5.0.2 node and delete the embedded messaging feature code. Run this command only after using the `pre_uninst502mq` tool.

When you migrate the V5.0.2 base node to V5.1, the two installation instances share the same node name, the same embedded messaging feature code, and the same messaging queue manager. Uninstalling either V5 installation instance removes the instance and the embedded messaging feature. The `pre_uninst502mq` tool prevents the removal of the embedded messaging feature code when the V5.0.2 installation instance is uninstalled. The V5.1 uninstaller program has an option that you can use to prevent the removal of the embedded messaging feature code.

The uninstaller program normally stops and removes the default embedded messaging queue manager process before uninstalling the feature. If another embedded messaging process is running on a Linux or UNIX-based machine, the uninstaller program cannot remove the embedded messaging feature. Therefore, the `pre_uninst502mq` script creates the `WAS_COEXIST` queue manager to prevent the removal of the embedded messaging feature code on Linux and UNIX-based platforms. The uninstaller program does not remove the `WAS_COEXIST` queue manager, which is active and therefore causes the uninstaller program to not remove the embedded messaging feature code.

On Windows platforms, the `pre_uninst502mq` script prevents the removal of the embedded messaging feature code by creating dummy registry entries. After using the `pre_uninst502mq` script and uninstalling the V5.0.2 product, run the `post_uninst502mq` script on Windows platforms to remove the dummy registry entries.

Location of the command file

The command file is located in the `install_root/bin` directory. The command file is a script named `post_uninst502mq.sh` on Linux and UNIX-based platforms and `post_uninst502mq.bat` on Windows platforms.

post_uninst502mq.sh command syntax for Linux and UNIX-based platforms

The command syntax is as follows:

```
post_uninst502mq.sh
```

post_uninst502mq.bat command syntax for Windows platforms

```
post_uninst502mq.bat
```

Parameters

No parameters are associated with this command.

Migrating a V5.0.2 base node to a V5.1 node

Issue the `pre_uninst502mq` command after migrating the configuration of a V5.0.2 node that has the embedded messaging feature installed, installing the V5.1 product, and before uninstalling the V5.0.2 node. After issuing the command, uninstall the V5.0.2 node. Because you are using the `pre_uninst502mq` script, uninstalling the V5.0.2 node prevents the removal of the embedded messaging feature code.

After using the `pre_uninst502mq` script and uninstalling the V5.0.2 product on Linux and UNIX-based platforms, perform this procedure:

1. Run the `post_uninst502mq` script to remove the `WAS_COEXIST` queue manager.
2. On Linux and UNIX-based platforms, run the following commands to verify that the `WAS_COEXIST` queue manager is deleted. Run the commands as root from the `install_root/bin` directory:

```
endmqm WAS_COEXIST
dltmqm WAS_COEXIST
```
3. Run the `createmq` command from the `install_root/bin` directory to create the default queue manager:

UNIX platforms:

```
createmq.sh install_root
             cellName
             nodeName
             server1
```

Windows platforms:

```
createmq.bat install_root
              cellName
              nodeName
              server1           mq_dir
              broker_dir
```

The command that the installer program used to create your original queue manager is in the *install_root/logs/createMQ.nodeName_server1.log* file.

To create the default queue manager on a Linux node named *wasnode*, for example, run the following command:

```
createmq.sh /opt/WebSphere/AppServer wasnode wasnode server1
```

On an unfederated node, the value of the cell name parameter is the node name.

WASPreUpgrade command:

The **WASPreUpgrade** command is a migration tool that saves the configuration and applications of a previous version or release to a Version 5.1 WebSphere Application Server node or Network Deployment node.

Location of command file

The command file is located in the *install_root/bin* directory. The command file is a script named *WASPreUpgrade.sh* for Linux and UNIX-based platforms or *WASPreUpgrade.bat* for Windows platforms.

WASPreUpgrade.sh command syntax for Linux and UNIX-based platforms

The command syntax is as follows:

```
WASPreUpgrade.sh backupDirectory
                  currentWASDirectory
                  [adminNodeName]
                  [-nameServiceHost host_name [-nameServicePort port_number ]]
                  [-import xmiDataFile ]
                  [-traceString trace_spec [-traceFile file_name ]]
```

WASPreUpgrade.bat command syntax for Windows platforms

```
WASPreUpgrade backupDirectory
               currentWASDirectory
               [adminNodeName]
               [-nameServiceHost host_name [-nameServicePort port_number ]]
               [-import xmiDataFile ]
               [-traceString trace_spec [-traceFile file_name ]]
```

Parameters

The first two arguments are required and positional. The third argument is required and positional only when upgrading from WebSphere Application Server, Version 3.5.x or Advanced Edition, Version 4.0.x.

Supported arguments include:

adminNodeName

Optional, positional name of the node containing the administrative server for the currently installed product. The value of this argument must match the node name given in the topology tree on the Topology tab of the administrative console for the currently installed product. The WASPreUpgrade tool calls the XMLConfig tool using this parameter. This parameter is only required when upgrading from WebSphere Application Server Standard Edition, Version 3.5.x, WebSphere Application Server Advanced Edition, Version 3.5.x, or WebSphere Application Server Advanced Edition, Version 4.0.x.

No equivalent parameter is available in the silent installation options response file.

backupDirectory

Required positional name of the directory in which the WASPreUpgrade tool stores the saved configuration and files, and from which the WASPostUpgrade tool later reads the configuration and files. The WASPreUpgrade tool creates the directory if the directory does not already exist.

This parameter is equivalent to the `-W migrationInformationPanelBean.migrationBackupDir="/tmp/migrationbackup"` parameter in the silent installation options response file.

currentWASDirectory

Required positional name of the installation root for the current V3.5.x, V4.x, or V5.0.x installation. This version can be either WebSphere Application Server Standard Edition, V3.5.x, WebSphere Application Server Advanced Edition, V3.5.x, any form of WebSphere Application Server, V4.0.x, and any form of V5.0.x, including WebSphere Application Server - Express.

This parameter is equivalent to the `-W previousVersionPanelBean.selectedVersionInstallLocation="/opt/WebSphere/AppServer"` parameter in the silent installation options response file.

-import *xmiDataFile*

The name of the WebSphere Application Server Advanced Single Server Edition, Version 4.0 XML Metadata Interchange (XMI) configuration file to process. This parameter is optional because the program uses the `config\server-cfg.xml` file by default.

When migrating a configuration that uses anything other than the default `server-cfg.xml` file name, you must use the `-import` option along with a path to point to the non-default XMI configuration file. You also must use the `-import` and `path` option when running the WASPostUpgrade tool later, to point to the non-default XMI configuration file in the directory created by the WASPreUpgrade tool.

This parameter is equivalent to the `-W previousVersionPanelBean.selectedVersionConfigFile="/opt/WebSphere/AppServer/config/server-cfg.xml"` parameter in the silent installation options response file.

-nameServiceHost -nameServicePort

When specified, the WASPreUpgrade tool passes these optional parameters to the XMLConfig tool. Use these parameters to override the default host name and port number used by the XMLConfig tool.

No equivalent parameter is available in the silent installation options response file.

-traceString *trace_spec* -traceFile *file_name*

Optional parameters to gather trace information for IBM Service personnel. Specify a trace specification of `"*=all=enabled"` (with quotation marks) to gather all trace information.

No equivalent parameter is available in the silent installation options response file.

Logging

The WASPreUpgrade tool displays status to the screen while it runs. The tool also saves a more extensive set of logging information in the *backup* directory. You can view the `WASPreUpgrade.log` file with a text editor.

Migrating from V3.5.x Standard Edition or Advanced Edition, or from V4.0.x Advanced Edition

The following example specifies a backup directory named `backupDirectory`, and identifies the root of the existing installation as `d:\WebSphere\AppServer`.

```
WASPreUpgrade backupDirectory d:\WebSphere\AppServer yourNodeName
```

Migrating from V4.0.x Advanced Single Server Edition with multiple backup directories

This example shows how to migrate incrementally, to migrate separate configuration files from a single node with a single installation of WebSphere Application Server Advanced Single Server Edition. To migrate more than one configuration file, you must run the WASPreUpgrade tool multiple times to multiple backup directories because not all of the applications are in the same `installedApps` directory. For this reason, using a single backup directory for all runs of the WASPreUpgrade tool is not recommended. Use a separate backup directory for each run. The intent of this example is to show how to migrate a single node with multiple configuration files.

1. Run the following WASPreUpgrade commands to migrate applications A, B, C, D, and E, which reside in three separate application directories. Server assumptions include:

- The Application Server uses the default configuration file, `server-cfg.xml`, as well as `myServer1-cfg.xml` and `01dServer-cfg.xml`.

```
> WASPreUpgrade C:\WAS4ABBACKUP G:\WebSphere\AppServer
> WASPreUpgrade C:\WAS4CDBACKUP G:\WebSphere\AppServer
  -import G:\WebSphere\AppServer\config\myServer1-cfg.xml
> WASPreUpgrade C:\WAS4EBACKUP G:\WebSphere\AppServer
  -import G:\WebSphere\AppServer\config\01dServer-cfg.xml
```

2. Run the following WASPostUpgrade commands to restore the applications and configurations to the Version 5 Application Server:

```
> WASPostUpgrade C:\WAS4ABBACKUP
> WASPostUpgrade C:\WAS4CDBACKUP -import C:\WAS4CDBACKUP\myServer1-cfg.xml
> WASPostUpgrade C:\WAS4EBACKUP -import C:\WAS4EBACKUP\01dServer-cfg.xml
```

Migrating from V5.0.0 or V5.0.1 with the embedded messaging feature **5.1+**

This example shows how to migrate a single instance of base WebSphere Application Server, V5.0.0. Verify that you have stopped all Java processes related to the WebSphere Application Server product that you are migrating. Run the WASPreUpgrade tool against the `installedApps` directory.

1. Run the following WASPreUpgrade.bat command to migrate all applications in the `installedApps` directory of the V5.0.0 Application Server, which has an installation root of `C:\Program Files\WebSphere\AppServer`.

```
> WASPreUpgrade C:\WAS500BACKUP C:\Program Files\WebSphere\AppServer
```

2. Run the following WASPostUpgrade commands to restore the applications and configurations to the Version 5 Application Server:

```
> WASPostUpgrade C:\WAS500BACKUP
```

WASPostUpgrade command:

The **WASPostUpgrade** command is a migration tool for adding the configuration and applications of a previous version or release to the current WebSphere Application Server node. The configuration includes migrated applications. The tool adds all migrated applications into the `install_root/installedApps` directory of the current product. The tool locates the saved configuration that the WASPreUpgrade tool saves through a parameter you use to specify the `backup` directory.

Location of command file

The command file is located in the `install_root/bin` directory. The command file is a script named `WASPostUpgrade.sh` for Linux and UNIX-based platforms or `WASPostUpgrade.bat` for Windows platforms.

WASPostUpgrade.sh command syntax for Linux and UNIX-based platforms

The command syntax is as follows:

```
WASPostUpgrade.sh backupDirectory
    [-import xmi_data_file]
    [-cellName cell_name]
    [-nodeName node_name]
    [-serverName server_name]
    [-webModuleAdditionalClasspath classpath]
    [-documentRootLimit number]
    [-substitute "key1=value1 [; key2=value2; [...]]"]
    [-portBlock port_starting_number]
    [-backupConfig true | false]
    [-replacePorts true | false]
    [[-traceString trace_spec [-traceFile file_name]]
```

The first argument is required.

WASPostUpgrade.bat command syntax for Windows platforms

```
WASPostUpgrade backupDirectory
    [-import xmi_data_file]
    [-cellName cell_name]
    [-nodeName node_name]
    [-serverName server_name]
    [-webModuleAdditionalClasspath classpath]
    [-documentRootLimit number]
    [-substitute "key1=value1 [; key2=value2; [...]]"]
    [-portBlock port_starting_number]
    [-backupConfig true | false]
    [-replacePorts true | false]
    [[-traceString trace_spec [-traceFile file_name]]
```

The first argument is required.

Parameters

Supported arguments include:

5.1+ -backupConfig true | false

An optional parameter used to back up the existing configuration of the Version 5.1 instance before adding the saved configuration from the earlier release to the Version 5.1 instance. The default is **true**, to use the **backupConfig** command to save a copy of the Version 5.1 configuration into the `install_root/temp` directory. Use the **restoreConfig** command to restore that configuration as required.

No equivalent parameter is available in the silent installation options response file.

backupDirectory

Required name of the directory in which the WASPreUpgrade tool stores the saved configuration and files. The WASPostUpgrade tool reads the configuration and files stored in this directory. The WASPreUpgrade tool creates this directory if it does not already exist.

This parameter is equivalent to the `-W migrationInformationPanelBean.migrationBackupDir="/tmp/migrationbackup"` parameter in the silent installation options response file.

-cellName *cell_name*

An optional parameter to specify the cell name for the program to update. If not specified, the program inspects the configuration for cell names. When one cell name exists, the program uses it. Otherwise, the program returns an error.

This parameter is equivalent to the `-W nodeNameBean.cellName="nodenameNetwork"` parameter in the silent installation options response file.

-documentRootLimit *number*

An optional parameter to specify the number of files that the program copies from the document-root field of Web-application. It is only applicable to Version 3.5.x upgrades. If not specified, the default is 300.

No equivalent parameter is available in the silent installation options response file.

-import *xmi_data_file*

The name of the WebSphere Application Server Advanced Single Server Edition, Version 4.0 XML Metadata Interchange (XMI) configuration file to process. This parameter is optional because the program uses the `config\server-cfg.xml` file by default.

This parameter is ignored when migrating from V5.0.x, including WebSphere Application Server-Express, V5.

When migrating a configuration that uses anything other than the default `server-cfg.xml` file name, you must use the `-import` option along with the path to the non-default XML configuration file in the directory created by the WASPreUpgrade tool.

This parameter is equivalent to the `-W previousVersionPanelBean.selectedVersionConfigFile="/opt/WebSphere/AppServer/config/server-cfg.xml"` parameter in the silent installation options response file.

-nodeName *node_name*

An optional parameter to specify the node name for the program to update. If not specified, the program inspects the configuration for node names. When one node name exists, the program uses it. Otherwise, the program returns an error.

This parameter is equivalent to the `-W nodeNameBean.nodeName="nodenameManager"` parameter in the silent installation options response file.

-portBlock *port_starting_number*

An optional parameter used to specify the starting value to use when creating ports.

No equivalent parameter is available in the silent installation options response file.

5.1 + -replacePorts true | false

An optional parameter used to define how to migrate virtual host and server transport ports. The default is `false`, to not replace default port definitions. By default the migration adds configuration data from the previous version to the existing data in the V5.1.x configuration. In some cases existing port definitions from the earlier release were carefully set to avoid port conflicts with other products. In such cases, it is likely that you would want to migrate the settings into V5.1.x. Use the `-replacePorts` parameter to totally replace settings in the V5.1.x environment with the settings from the previous version. Select `true` to replace all virtual host alias port settings during migration. If migrating from V5.0.x or later, transport settings in existing servers are replaced by the settings from the previous version.

-serverName *server_name*

An optional parameter to specify the server name for the program to update. If not specified, the program inspects the configuration for server names. When one server name exists, with an Application Server name that matches, the program uses it. Otherwise, the program creates a new server.

-substitute *key1=value1;key2=value2;...*

Optional argument passed to the XMLConfig tool. Specify values for security variables to substitute (for example, `-substitute "NODE_NAME=admin_node;APP_SERVER=default_server"`).

In the input XML data file, each key appears as `key` for substitution. This argument substitutes any occurrence of `$NODE_NAME$` with `admin_node` and `APP_SERVER` with `default_server` in the input XML file.

If the substitution string contains semicolons, use `$semiColon$` to distinguish the string from the ";" delimiter. On Linux and UNIX-based platforms, add an escape character to each dollar sign (\$) within the substitution string (for example, `\$semiColon\$`).

This parameter is applicable for configurations saved from Advanced Edition, Versions 3.5.x and 4.0.x.

No equivalent parameter is available in the silent installation options response file.

-traceString *trace_spec* **-traceFile** *file_name*

An optional parameters to gather trace information for IBM Service personnel. Specify a trace specification of `"*=all=enabled"` (with quotation marks) to gather all trace information.

No equivalent parameter is available in the silent installation options response file.

-webModuleAdditionalClasspath *classpath*

An optional parameter to specify the path or the path and file names of specific directories or files that you do not want copied into the Web archive (WAR) file. Instead, the program adds the paths and the files to the Web Module extension (`ibm-web-ext.xmi`) `additionalClassPath` attribute. This parameter is applicable only when migrating a Version 3.5.x installation.

No equivalent parameter is available in the silent installation options response file.

Logging

The WASPostUpgrade tool displays status to the screen while running. This tool also saves a more extensive set of logging information in the `logs` directory. You can view the `WASPostUpgrade.log` file with a text editor.

Migrating from V3.5.x Standard Edition or Advanced Edition, or from V4.0.x Advanced Edition

The following example specifies a backup directory named `backupDirectory`, and identifies the root of the existing installation as `d:\WebSphere\AppServer`.

```
WASPreUpgrade backupDirectory d:\WebSphere\AppServer yourNodeName
```

Migrating from V4.0.x Advanced Single Server Edition with multiple backup directories

This example shows how to migrate separate configuration files incrementally from a single node with a single installation of WebSphere Application Server Advanced Single Server Edition. To migrate more than one configuration file, you must run the WASPreUpgrade tool multiple times to multiple backup directories because not all of the applications are in the same `installedApps` directory. For this reason, using a single backup directory for all the runs of the WASPreUpgrade tool is not recommended. Use a separate backup directory for each run. The intent of this example is to show how to migrate a single node with multiple configuration files.

1. Run the following WASPreUpgrade commands to migrate applications A, B, C, D, and E, which reside in three separate application directories. Server assumptions include:

- The Application Server uses the `server-cfg.xml` default configuration file, as well as the `myServer1-cfg.xml` and the `OldServer-cfg.xml` files.

```
> WASPreUpgrade "C:\WAS4ABBACKUP" G:\WebSphere\AppServer
> WASPreUpgrade "C:\WAS4CDBACKUP" G:\WebSphere\AppServer
  -import G:\WebSphere\AppServer\config\myServer1-cfg.xml
> WASPreUpgrade "C:\WAS4EBACKUP" G:\WebSphere\AppServer
  -import G:\WebSphere\AppServer\config\OldServer-cfg.xml
```

2. Run the following WASPostUpgrade commands to restore the applications and configurations to the Version 5.1 Application Server:

```
> WASPostUpgrade C:\WAS4ABBACKUP
> WASPostUpgrade "C:\WAS4CDBACKUP" -import "C:\WAS4CDBACKUP\myServer1-cfg.xml"
> WASPostUpgrade "C:\WAS4EBACKUP" -import "C:\WAS4EBACKUP\OldServer-cfg.xml"
```

Migrating from V5.0.0 or V5.0.1 with the embedded messaging feature **5.1+**

This example shows how to migrate a single instance of base WebSphere Application Server, V5.0.0. Verify that you have stopped all Java processes related to the WebSphere Application Server product that you are migrating. Run the WASPreUpgrade tool against the installedApps directory.

1. Run the following WASPreUpgrade.bat command to migrate all applications in the installedApps directory of the V5.0.0 Application Server, which has an installation root of C:\Program Files\WebSphere\AppServer.

```
> WASPreUpgrade "C:\WAS500BACKUP" C:\Program Files\WebSphere\AppServer
```

2. Run the following WASPostUpgrade commands to restore the applications and configurations to the Version 5 Application Server:

```
> WASPostUpgrade "C:\WAS500BACKUP"
```

Adding or changing features on a federated node

This task describes a manual procedure workaround for changing features on a federated node that you are migrating .

When migrating a federated node, the configuration on the base node is owned by the deployment manager and cannot be changed. It is always true that you must unfederate a node before adding features. When you unfederate the node, you lose any of the configuration for the node that you performed with the administrative console of the deployment manager while the node was federated. After installing additional features, add the node back to the cell to cause the deployment manager to add the configuration of the base node. When migrating a node, always select the same features for the new node that are in the existing node. Adding or changing features can cause configuration problems in a federated node.

If you intend to install additional features, use the following procedure to avoid component regression problems. It is important that you understand that there is no way to add features to a base node without unfederating it from its cell and losing its configuration. The configuration for the base node is restored back to its original configuration, before it was federated and before you changed it with the administrative console of the deployment manager. This means that you will lose the configuration for any additional servers that you created on the base node from the deployment manager. (The additional servers will not exist any more.) You will lose any other configuration you performed from the deployment manager administrative console. Although you can use the **backupConfig** command to save the configuration, you cannot restore it with the **restoreConfig** command because the configuration you restore does not include any features that you add to the base node.

You can use the **backupConfig** command to save the Network Deployment configuration so that you can restore it if necessary. You can also refer to the configuration as you use the administrative console of the deployment manager to reconfigure the base node.

If the node you are migrating is part of a deployment manager cell, migrate the V5.0.x deployment manager to V5.1 first, before continuing this procedure. The deployment manager node must always be at the highest release level within the cell.

5.1+ Use the following procedure to unfederate the V5.0.x node, so that you can add or change features during the migration of the cell to V5.1.

1. Cancel the installation of the base WebSphere Application Server, if you have already begun the installation.
 - a. Click **Cancel**. Confirm that you intend to cancel the installation.
 - b. Open a command window and change directories to the V5.0.x install_root/bin directory.
2. Unfederate the node. Use the removeNode.sh or the removeNode.bat script. For example, use the following command to remove the node on a Linux platform:

```
./removeNode.sh
```

Wait for the successful conclusion of the command.

- Optional: Uninstall the V5.0.x node. Click the platform-specific uninstalling procedure in the topic to remove product artifacts from the machine. Verify that you are manually removing artifacts from the V5.0.x node and not another installation instance that might be on the machine. Leave files related to the embedded messaging feature if other installation instances also use embedded messaging.
- Install the product. Select the custom installation type. Select any features you intend to use.
- Verify the installation of the Application Server. Use the First Steps tool when it appears at the end of installing the product, or run the installation verification test yourself, if the First Steps tool does not appear for some reason.
- Verify that the deployment manager is running. If it is not, start the deployment manager with the `startManager.sh` or the `startManager.bat` script, issued from the `install_root/bin` directory of the deployment manager node.
- Add the V5.x node to the deployment manager cell. Open a command window and change directories to the V5.x `install_root/bin` directory of the base WebSphere Application Server product. Use the `addNode.sh` or the `addNode.bat` script to federate the V5.1 node. For example, use the following command to add the node on a Linux platform:

```
./addNode.sh deploymentNode 8879 -includeapps
```

Use the name of the deployment manager node in place of the `deploymentNode` variable in this example. Use the `-includeapps` parameter to copy all V5.1 Sample applications to the deployment manager. The default port is SOAP port 8879, which must match the deployment manager port.

Wait for the successful conclusion of the command.

The **addNode** command starts the `nodeagent` process for the V5.1 node.

- Start the messaging service provider if the embedded messaging server feature is installed. Use the `startServerer.sh` or the `startServer.bat` script from the `install_root/bin` directory of the V5.1 Application Server. For example, use the following command on a Linux platform:

```
./startServer.sh jmsserver
```

- Use the administrative console of the deployment manager to configure the base node.

You can install additional features on a federated node by unfederating the V5.0.x node, uninstalling fix packs and interim fixes, and reinstalling the node with additional features. Adding features to a federated node always involves removing the node from the cell, which loses the cell-controlled configuration. If the deployment manager is at a later release level, such as V5.1, uninstall the V5.0.x node if you must add features and install a V5.1 node. Configure the V5.1 node manually with the administrative console to achieve the intended configuration.

Coexistence support

Coexistence, as it applies to WebSphere Application Server products, is the ability of multiple installations of WebSphere Application Server to run on the same machine at the same time. Multiple installations include multiple versions and multiple instances of one version. Coexistence also implies various combinations of Web server interaction.

Version 5.x WebSphere Application Server products can coexist with the following supported versions. The installation wizard looks for these existing installations to determine if it should prompt for coexistence information:

- IBM WebSphere Application Server Standard Edition and Advanced Edition, Version 3.5.5 and up
- IBM WebSphere Application Server Advanced Server Single Edition and Advanced Edition, Version 4.0.2 and later
- IBM WebSphere Business Integration Server Foundation Edition, Version 4.1 and later
- IBM WebSphere Application Server, Version 5.0.0 and later
- IBM WebSphere Application Server Network Deployment, Version 5.0.0 and later
- IBM WebSphere Application Server Enterprise, Version 5.0.0 and later
- IBM WebSphere Application Server, Version 5.1.0
- IBM WebSphere Application Server Network Deployment, Version 5.1.0

All combinations of V3.5.x products, V4.x products, and V5.x products can coexist so long as there are no port conflicts. There are some coexistence limitations for V5 products that have the embedded messaging feature installed, as described later. These limitations are described as *No EM* in the following description.

5.1+ V3.5.5 and later products and V4.0.2 and later products can coexist with V5.x products:

Table 66. Multiversion coexistence scenarios

Installed product	V5.0.0 and V5.0.1			V5.0.2 and V5.1		
	Application Server	Network Deployment	Enterprise	Application Server	Network Deployment	Enterprise V5.0.2
Standard Edition V3.5.5 and later	Supported	Supported	Supported	Supported	Supported	Supported
Advanced Edition V3.5.5 and later	Supported	Supported	Supported	Supported	Supported	Supported
Advanced Single Server Edition V4.0.2 and later	Supported	Supported	Supported	Supported	Supported	Supported
Advanced Edition V4.0.2 and later	Supported	Supported	Supported	Supported	Supported	Supported
Enterprise Edition V4.1 and later	Supported	Supported	Supported	Supported	Supported	Supported
WebSphere Application Server clients V4.0.x	No EM	No EM	No EM	No EM	No EM	No EM
Application Server, V5.0.0	Supported	Supported	Supported	No EM	No EM	No EM
Network Deployment V5.0.0	Supported	Supported	Supported	No EM	No EM	No EM
Enterprise, V5.0.0	Supported	Supported	Supported	No EM	No EM	No EM
WebSphere Application Server clients V5.0.0	Supported	Supported	Supported	No EM	No EM	No EM
Application Server V5.0.1	Supported	Supported	Supported	No EM	No EM	No EM
Network Deployment V5.0.1	Supported	Supported	Supported	No EM	No EM	No EM
Enterprise V5.0.1	Supported	Supported	Supported	No EM	No EM	No EM
WebSphere Application Server clients V5.0.1	Supported	Supported	Supported	No EM	No EM	No EM
Application Server V5.0.2	No EM	No EM	No EM	Supported	Supported	Supported
Network Deployment V5.0.2	No EM	No EM	No EM	Supported	Supported	Supported
Enterprise V5.0.2	No EM	No EM	No EM	Supported	Supported	Supported
WebSphere Application Server clients V5.0.2	No EM	No EM	No EM	Supported	Supported	Supported

Table 66. Multiversion coexistence scenarios (continued)

Installed product	V5.0.0 and V5.0.1			V5.0.2 and V5.1		
	Application Server	Network Deployment	Enterprise	Application Server	Network Deployment	Enterprise V5.0.2
Application Server V5.1	No EM	No EM	No EM	Supported	Supported	Supported
Network Deployment V5.1	No EM	No EM	No EM	Supported	Supported	Supported
WebSphere Application Server clients V5.1	No EM	No EM	No EM	Supported	Supported	Supported

5.1+ No EM: This status in the preceding tables describes an incompatibility between the embedded messaging features in different levels of Version 5. Version 5.0.0 and V5.0.1 support the embedded messaging feature at a service level less than CSD04, which is what V5.0.2 and V5.1 support. Because all V5 instances that coexist on a machine must use the same level of embedded messaging, there is a coexistence problem when two V5 instances use different levels of embedded messaging. If you want V5.0.0 to coexist with V5.1, for example, upgrade V5.0.0 to V5.0.2 by applying Fix Pack 2. See *Upgrading V5.0.0 or V5.0.1 to V5.0.2*.

In addition to multiversion coexistence, WebSphere Application Server also lets you install multiple times on one machine (multiple installation instances), or install once and have multiple configurations (multiple configuration instances).

Multiple Version 5 instances on one machine include:

- Multiple Application Server instances from multiple installations of the base WebSphere Application Server product
- Multiple Application Server configuration instances from a single installation of the base product
- Multiple deployment manager instances from multiple installations of the Network Deployment product
- Multiple deployment manager configuration instances from a single installation of the Network Deployment product

See “Federating multiple Version 5 installation instances” on page 362 for information about mixed node cells, where a Version 5.1 deployment manager cell can include Version 5.0.x base nodes.

Setting up Version 3.5.x and Version 5 coexistence

You must migrate prerequisite and corequisite programs to the levels required by WebSphere Application Server, Version 5. You must also identify ports in use in Version 3.5.x before you begin the Version 5 installation, to avoid possible conflicts during coexistence. The first two steps in this task describe these activities.

You can install WebSphere Application Server Version 3.5.5 (and later) and Version 5 on the same node. When the Version 5 installation wizard detects the Version 3.5.x installation, it displays the migration and coexistence panel, where you can select either option, or neither option. If you select coexistence, the installation wizard displays the port number panel, to ensure you install Version 5 without port conflicts.

Silent installation also supports configuring for coexistence silently. You can specify non-conflicting port assignments in the options response file.

By default, there are port conflicts between Version 3.5.x and Version 5 that you must resolve. Also, if you migrate more than two Version 3.5.5 nodes to Version 5, there are port conflicts that you must resolve, as described in the *Setting up Version 5 coexistence* topic.

1. Migrate prerequisite and corequisite programs to the levels required by WebSphere Application Server, Version 5.

Refer to the IBM WebSphere Application Server supported hardware, software, and APIs site for current requirements.

2. Resolve port conflicts.

Refer to the “Port number settings in WebSphere Application Server versions” on page 363 topic to see a list of default port numbers, and where they are defined.

Inspect the configuration of the Version 3.5.x product, either WebSphere Application Server Advanced Edition, or WebSphere Application Server Standard Edition.

If necessary, use the following command to examine existing node and port settings when the administrative server is running.

```
xmlConfig -export config.xml -nodeName theNodeName
```

Review the output xml file to look for `<node action="update" name="nodename">` to determine the node name. You can also find port number assignments in the output.xml file. The installation wizard displays a default set of coexistence port numbers as suggested Version 5 port numbers. Change the values to ports that are not in use. The installation wizard uses whatever values you approve.

Change conflicting HTTP transport ports manually, if necessary. See “Changing HTTP transport ports” on page 202.

3. Associate a Web server with each WebSphere Application Server:
 - Use a separate Web server for each WebSphere Application Server.
 - a. Create a Web server instance using the Web server documentation.
 - b. Select the appropriate Web server plug-in feature during WebSphere Application Server installation, and identify the Web server configuration file location. (For example, identify the location of the `httpd.conf` file for IBM HTTP Server.)
 - Use the same Web server for both WebSphere Application Server versions.

Upgrade the Web server to the common level supported by both versions of the Application Server.

Prepare the same Web server to support both Application Server versions:

 - Migrate IBM HTTP Server, as described in the “Migrating IBM HTTP Server to support multiple WebSphere Application Server versions” (`tins_websIHS`) topic in the base WebSphere Application Server information center.
 - Migrate iPlanet, now known as the Sun ONE Web server (`tins_websiPI`).
 - Migrate Lotus Domino (`tins_websDom`).
 - Migrate Microsoft IIS (`tins_websIIS`).

4. Fix any problems with environmental variables on Windows platforms.

For example, installing WebSphere Application Server, Version 5 updates the system variable `PATH`, potentially affecting tools with the same name across installations. To run tools with conflicting names, alter the `PATH` environment variable in a command window and place the directory for the former installation before the directory for the latter installation. For example, `PATH=E:\WebSphere\AppServer\35\bin;%PATH%`. Then, invoke the tools from the `bin` directory.

Setting up Version 4.0.x and Version 5.x coexistence

You must migrate prerequisite and corequisite programs to the levels required by WebSphere Application Server, Version 5.x. You must also identify ports in use in Version 4.0.x before you begin the Version 5.x installation, to avoid possible conflicts during coexistence. The first two steps in this task describe these activities.

You can install WebSphere Application Server Version 4.0.x and Version 5 on the same node. When the Version 5 installation wizard detects the Version 4.0.x installation, it displays the migration and coexistence panel, where you can select either option, both options, or neither option. If you select coexistence, the installation wizard displays the coexistence panel to ensure you install Version 5 without port conflicts.

Silent installation also supports configuring for coexistence silently. You can specify non-conflicting port assignments in the options response file, as described in “Installing silently” on page 174.

By default, there are port conflicts between Version 4.0.x and Version 5 that you must resolve. Also, if you migrate more than two Version 4.0.x nodes to Version 5, there are port conflicts that you must resolve, as described in the *Setting up Version 5 coexistence* topic.

1. Migrate prerequisite and corequisite programs to the levels required by WebSphere Application Server, Version 5.

Refer to the IBM WebSphere Application Server supported hardware, software, and APIs site for current requirements.

2. Resolve port conflicts.

Refer to the Port number settings in WebSphere Application Server versions topic to see a list of default port numbers, and where they are defined.

Inspect the configuration of the previous version:

- **For WebSphere Application Server Advanced Single Server Edition, Version 4.0.x:** Inspect the `server-cfg.xml` file to get port values for the configuration.
- **For WebSphere Application Server Advanced Edition, Version 4.0.x:** Inspect the `admin.config` file to get port values for the configuration. When the administrative server is running, use this command:

```
xmlConfig -export config.xml -nodeName theNodeName
```

Review the `config.xml` file to look for `<node action="update" name="nodename">` to find the appropriate node and port number assignments in the file. The installation wizard displays a default set of coexistence port numbers as suggested Version 5 port numbers. Change the values to ports that are not in use. The installation wizard uses whatever values you approve.

Change conflicting HTTP transport ports manually, if necessary.

3. Associate a Web server with each WebSphere Application Server.

- Use a separate Web server for each WebSphere Application Server.
 - a. Create a Web server instance using the Web server documentation.
 - b. Select the appropriate Web Server plug-in feature during WebSphere Application Server installation, and identify the Web server configuration file location. (For example, identify the location of the `httpd.conf` file for IBM HTTP Server.)
- Use the same Web server for both WebSphere Application Server versions.

To use the same Web server for both Application Server versions, you must first upgrade the Web server to the common level supported by both versions of the application server.

Follow this procedure to use the same Web server for both WebSphere Application Server versions.

- a. Select the appropriate Web Server plug-in feature during WebSphere Application Server installation, and identify the Web server configuration file location. (For example, identify the location of the `httpd.conf` file for the IBM HTTP Server that is associated with Version 4.0.x.)
- b. Edit the Web server configuration file to remove entries for Version 4.0.x, as described in the *Migrating plug-ins, one machine at a time* (`tins_websmig2`) topic in the base Application Server information center.

The WebSphere Application Server, Version 5 plug-in acts as the routing agent to route requests to both versions.

- c. Generate the plug-in configuration files for both versions of the Application Server.
- d. Edit the Version 4.0.x `plugin-cfg.xml` file and the Version 5 `plugin-cfg.xml` file, to merge their entries into a combined file of all Web context roots.

Web context roots must be unique across Application Server versions. If there is a common context root across versions, requests are served by the plug-in instance that was last loaded.

If you have both Version 4.0.x Samples and Version 5 Samples in the file, you can access only the Version 5 Samples after merging the configuration files because there is one context root for Samples.

Whenever you modify the `plugin-cfg.xml` file for either version, you must manually merge the files again, to make a master file.

This is described in the *Preparing to install and configure a Web server (tins_webserver)* topic in the base Application Server information center.

Whenever you modify the `plugin-cfg.xml` file for either version, you must manually merge the files again, to make a master file.

- e. Replace the original `plugin-cfg.xml` file of the Version 5 installation with the master file.
4. Fix any problems with environmental variables on Windows platforms. For example, installing WebSphere Application Server, Version 5.1 updates the system variable `PATH`, potentially affecting tools with the same name across installations. To run tools with conflicting names, alter the `PATH` environment variable in a command window and place the directory for the former installation before the directory for the latter installation. For example, `PATH=E:\WebSphere\AppServer\40\bin;%PATH%`. Then, invoke the tools from the `bin` directory.

Setting up Version 5 coexistence

This task describes how to install a Version 5.1 product to coexist with another installation instance of Version 5.1 or Version 5.0.x.

After installing the WebSphere Application Server product, you can install it again on the same machine.

Select the new installation option from the installation wizard panel, to install a new instance instead of adding features to the last installation, and to install into a separate installation directory. Select the coexistence option to provide nonconflicting ports.

Each installation of the base product is a stand-alone Application Server (`server1`) with its own set of unique configuration files.

Each installation of the Network Deployment product is a stand-alone deployment manager (`dmgr`) with its own set of unique configuration files for its cell.

Be aware of multiple instance limitations, between:

- Incompatible service levels of the embedded messaging feature.

If you attempt to install V5.0.2 or V5.1 when V5.0.0 or V5.0.1 with the embedded messaging feature is already on the machine, you cannot select the embedded messaging feature. An incompatibility exists between the service level versions of embedded messaging. The level of embedded messaging for V5.0.0 and V5.0.1 differs from the level supported on V5.0.2 and V5.1. Upgrade the V5.0.0 or V5.0.1 node to V5.0.2 before installing another instance of V5.0.2 or V5.1.

- The base WebSphere Application Server product and the Network Deployment product.

The deployment manager and the Application Server have two port conflicts in the default configuration settings for both products. Each product is configured to use port 9090 as the administrative console port and port 9043 as the administrative console secure port by default. It is not recommended that you run both the deployment manager and the Application Server on the same production machine, unless you have a machine with the capacity to handle the cumulative requirements of both products. However, if you do so, select the coexistence option during installation to change these port settings. There are no other port conflicts if you install a total of two instances, one base WebSphere Application Server product and one Network Deployment product.

If you install more than two instances, the third and subsequent installations require that you change all port numbers on the coexistence panel, to avoid potential conflicts.

- The base WebSphere Application Server product and another installation of the base WebSphere Application Server product, or the Network Deployment product and another installation of the Network Deployment product.

You can install another instance of either product, by selecting to do so at the prompt, and by changing the installation directory.

The most recent installation is the only one in the operating system registry.

If you install more than two total instances of either product (more than one of each, or two of either and none of the other), the third and subsequent installations require that you change all port numbers on the coexistence panel to avoid potential conflicts.

To uninstall a product instance, always use the operating system remove program function, such as the Add/Remove program on Windows platforms. To uninstall an unregistered instance, use the **Uninstall.exe** or **uninstall** command in the `_uninstall` directory of the `install_root` that matches the instance you intend to remove.

Reasons to use multiple installation instances include:

- You can achieve complete isolation between each WebSphere Application Server instance. You can uninstall one instance independently of the others.
- You can federate each installation instance of the base WebSphere Application Server product to a deployment manager cell.
- You can install the base WebSphere Application Server more than once on the same machine.
- You can install Network Deployment more than once on the same machine.
- **5.1+** You can install V5.0.x Network Deployment and the V5.1 product on the same machine. There are some limitations involving the embedded messaging feature. The level of embedded messaging for V5.0.0 and V5.0.1 differs from the level supported on V5.0.2 and V5.1.

Reasons to not use multiple installation instances include:

- The machine might have a hard disk space constraint.
- You can use the operating system registry to locate the last installed instance of a WebSphere Application Server product only.

When you install any product a second time, the last installation is the one that appears in the registry.

- Uninstalling the last instance removes any record of the product in the registry.

Suppose you have installed three instances of the base WebSphere Application Server product. You use the remove program function of the operating system to uninstall the registered third copy of the base product. A registry record no longer exists that indicates the existence of the other two installation instances. Other applications cannot use a query of the operating system registry to detect the presence of either base WebSphere Application Server product instance.

The Creating multiple Version 5 configuration instances topic describes installing the base WebSphere Application Server product or the Network Deployment product once and creating multiple configuration instances.

Use the following procedure to install multiple installation instances.

1. Upgrade V5.0.0 or V5.0.1 with embedded messaging to V5.0.2.
2. Use the installation wizard to install another installation. If you intend to share a single Web server among installations, install a Web server plug-in feature during one installation only, as described in the next step.
3. Share a Web server among multiple installation instances.
 - a. Select a Web server plug-in feature during one installation only.
 - b. Generate the plug-in configuration files for every installation instance.
 - c. Edit the `plugin-cfg.xml` configuration files to merge them into one master configuration.
 - d. Replace the original `plugin-cfg.xml` file with the master file on the Application Server where you selected the Web server plug-in feature.

You can access samples from only one of the installation instances.

4. Federate multiple installation instances into a deployment manager cell.
5. Create additional servers in a multiple instance or coexistence environment.

6. Change port assignments in configuration files if you have a node that you cannot start because of port conflicts.

Upgrading a Version 5.0.0 or Version 5.0.1 product to Version 5.0.2

Install Fix Pack 2 from the Support site for WebSphere Application Server, to upgrade Version 5.0.0 or Version 5.0.1 to Version 5.0.2. Upgrading to Version 5.0.2 also upgrades the service level of the embedded messaging feature to CSD04. This upgrade is important for migrating to Version 5.1, and also for coexisting with Version 5.0.2 and Version 5.1 instances. Version 5.0.0 and Version 5.0.1 cannot coexist with Version 5.0.2 and Version 5.1 when the embedded messaging feature is installed.

Apply Fix Pack 2 to the Application Server, to upgrade the service level of the embedded messaging feature.

Fix packs are available on the Support site for WebSphere Application Server. Download all of the ZIP files for Fix Pack 2 that you require. A separate ZIP file is available for the client and for each WebSphere Application Server product.

1. Locate the download page for Fix Pack 2 on the Support site. See WebSphere Application Server Version 5.0 Fix Pack 2 (Version 5.0.2) at <http://www-1.ibm.com/support/docview.wss?rs=860&context=SW600&q=Fix+Pack+2&uid=swg24005012> for information about applying Fix Pack 2.
2. Download the ZIP files for Fix Pack 2 to a read/write directory. Installing from a read-only directory is not supported at this time. The directory must have no spaces in its name.
3. Unzip the file into the directory. In addition to Fix Pack 2, the ZIP file also contains the update installer program, which you use to install the Fix Pack. The update installer program is also known as the `updateInstaller` program or the update installation wizard.
4. Use the update installer to install Fix Pack 2, as described in “Installing interim fixes, cumulative fixes, and fix packs” on page 202. Use either the wizard interface, with the **updateWizard** command, or its silent, command-line interface, with the **updateSilent** command.
5. Refer to the readme file for Fix Pack 2 to continue. The readme file is included in the ZIP file for the fix pack. There is also another readme file for update installer.

The download page for Fix Pack 2 at the Support site contains the most current version of the readme files, which you can download separately from the ZIP files for Fix Pack 2.

You can upgrade V5.0.0 or V5.0.1 to V5.0.2 by installing Fix Pack 2.

Federating multiple Version 5 installation instances

This article describes federating or joining a base WebSphere Application Server node to a WebSphere Application Server Network Deployment cell.

There are two ways to federate a base WebSphere Application Server node into a deployment manager cell:

- Using the deployment manager administrative console, as described in Managing nodes.
- Using the **addNode** command line script from the `bin` directory of the node you are federating

Always use the `-includeapps` parameter of the **addNode** command unless you included the applications during a previous federation of the base WebSphere Application Server node into the cell. See the description of the **addNode** command for more information about its parameters. The administrative console also provides an option for including applications on the base node into the cell.

Federating from the deployment manager administrative console requires a running base WebSphere Application Server.

Consider a scenario where you install both the base WebSphere Application Server product and the Network Deployment product on the same development machine. Installing a production Application Server on the same machine as the deployment manager is not recommended unless the machine has the

capacity to handle both jobs. To let both products run at the same time, you must install the Network Deployment product using the coexistence panel, to select ports that do not conflict with the base WebSphere Application Server ports. After installation, stop the Application Server if it is running. Start the deployment manager server. Issue the **addNode** command from the bin directory of the base node. The deployment manager federates the base node and instantiates the nodeagent server process on the base Application Server node.

A coexistence environment might have multiple base WebSphere Application Server product installations on one machine. You can federate each installation into the same cell, or into different cells. Whenever the deployment manager federates a node into its cell, it configures the nodeagent server process for the node with a set of default ports. If the base node has the embedded messaging feature, the deployment manager also configures a JMS provider, `jmsserver`, which is a server process with another set of default ports.

Plan to configure the node agent and the JMS provider with port assignments that differ from the defaults, to verify that there is no conflict with coexisting installation instances that have similar configurations.

Mixed node cells

Ensure that the fix level of the Version 5 WebSphere Application Server product matches or is lower than the fix level of the Network Deployment product. For example, in a mixed node environment you can federate a Version 5.0.2 base node into a Version 5.1 deployment manager cell but you cannot federate a Version 5.1 base node into a V5.0.x deployment manager cell.

1. Use the **stopNode** command to stop any running node agents that are using default ports on the same physical machine.
Issue the command from the `bin` directory of the Application Server installation root.
2. Federate the Application Server node into the deployment manager cell, using the **addNode** command from the `bin` directory of the node to federate. This command automatically instantiates the nodeagent process, with default ports.
For example, suppose you federate an installation instance that has a node name of *MyNode*.
3. Log on to the deployment manager administrative console. Go to **System Administration > Node Agents > nodeagent > End Points**. Change port numbers for all end points and save the changes.
4. Log on to the deployment manager administrative console, if you have installed the embedded messaging feature on the node you federated:
 - a. Go to **System Administration > Node Agents > jmsserver > Security Port Endpoints**. Change port numbers for all end points and save the changes.
 - b. Go to **System Administration > Node Agents > jmsserver > End Points**. Change port numbers for all the end points and save the changes.
5. Synchronize the changes using **System Administration > Nodes**. Select the node, `MyNode`, and click **Synchronize**.
6. Stop the node agent on the Application Server node using the **stopNode** command line script. Restart the node agent using the **startNode** command.
7. Change port assignments in the configuration files if you have an Application Server or deployment manager node that you cannot start because of port conflicts.

Port number settings in WebSphere Application Server versions

This topic provides reference information about identifying port numbers in versions of WebSphere Application Server, as a means of determining port conflicts that might occur when you intend for an earlier version to coexist or interoperate with Version 5.1.

Version 3.5.x port numbers

Resolve port conflicts by inspecting the configuration of the Version 3.5.x product, either WebSphere Application Server Advanced Edition or WebSphere Application Server Standard Edition. When the administrative server is running, use this command to examine current node and port settings:

```
xmlConfig -export config.xml -nodeName theNodeName
```

Look for the OSE Remote port assignments.

Table 67. WebSphere Application Server Version 3.5.x default port definitions in the admin.config file

Port name	Standard Edition	Advanced Edition
	Value	
Isdport (Location Service Daemon)	9000	9000
bootstrapPort	900	900
LSDSSL (With security on)	9001	9001

Version 4.0.x port numbers

For WebSphere Application Server Advanced Single Server Edition, Version 4.0.x: Inspect the server-cfg.xml file to find the Web container HTTP transports port values for the configuration.

For WebSphere Application Server Advanced Edition, Version 4.0.x: When the administrative server is running, use this command to extract the configuration from the database:

```
xmlConfig -export config.xml -nodeName theNodeName
```

Look for the Web container HTTP transports port assignments.

Table 68. Port definitions for WebSphere Application Server V4.0.x

Port name	Value	Advanced Edition	EnterpriseEdition	Advanced Single Server Edition
		File		
bootstrapPort	900			
IsdPort	9000	admin.config	admin.config	
LSDSSLPort	9001			
HTTP transport port	9080			
HTTPS transport port	9443			server-cfg.xml
Admin Console HTTP transport port	9090	database	database	
ObjectLevelTrace	2102			
diagThreadPort	7000			

Version 5.x port numbers

You can use the administrative console to configure the Version 5.x Application Server, to resolve conflicting ports.

Table 69. Port definitions for WebSphere Application Server Version 5.x

Port name	WebSphere Application Server	Network Deployment	File
	Value		
HTTP_TRANSPORT	9080	Not applicable	
HTTPS Transport Port (HTTPS_TRANSPORT)	9443	Not applicable	server.xml and virtualhosts.xml
HTTPS Admin Console Port (HTTP_TRANSPORT_ADMIN)	9090	9090	
HTTPS Admin Console Secure Port (HTTPS_TRANSPORT_ADMIN)	9043	9043	
Internal JMS Server (JMSSERVER_SECURITY_PORT)	5557	Not applicable	server.xml
JMSSERVER_QUEUED_ADDRESS	5558	Not applicable	serverindex.xml
JMSSERVER_DIRECT_ADDRESS	5559	Not applicable	
BOOTSTRAP_ADDRESS	2809	9809	serverindex.xml
SOAP_CONNECTOR-ADDRESS	8880	8879	serverindex.xml
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	0	9401	serverindex.xml
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	0	9403	serverindex.xml
CSIV2_SSL_MULTIAUTH_LISTENER_ADDRESS	0	9402	
IBM HTTP Server Port	80	Not applicable	virtualhosts.xml, plugin-cfg.xml, and <i>IHSinstall_root/conf/</i> <i>httpd.conf</i>
IBM HTTPS Server Admin Port	8008	Not applicable	<i>IHSinstall_root/conf/</i> <i>admin.conf</i>
CELL_DISCOVERY_ADDRESS	Not applicable	7277	serverindex.xml
ORB_LISTENER_ADDRESS	9100	9100	serverindex.xml
CELL_MULTICAST_DISCOVERY_ADDRESS	Not applicable	7272	serverindex.xml

When you federate an Application Server node into a deployment manager cell, the deployment manager instantiates the nodeagent server process on the Application Server node. The nodeagent server uses the following port assignments by default:

Table 70. Port definitions for the V5.x nodeagent server process

Port name	Value	File
BOOTSTRAP_ADDRESS	2809	
ORB_LISTENER_ADDRESS	9900	
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	9901	
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS	9101	
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	9201	serverindex.xml
NODE_DISCOVERY_ADDRESS	7272	
NODE_MULTICAST_DISCOVERY_ADDRESS	5000	
DRS_CLIENT_ADDRESS	7888	
SOAP_CONNECTOR_ADDRESS	8878	

When you federate an Application Server node with the embedded messaging server feature into a deployment manager cell, the deployment manager instantiates a JMS server process, `jmsserver`, on the Application Server node. The JMS server uses the following port assignments by default:

Table 71. Port definitions for V5.x `jmsserver` process

Port name	Value	File
JMSSERVER_DIRECT_ADDRESS	5559	
JMSSERVER_QUEUED_ADDRESS	5558	serverindex.xml
SOAP_CONNECTOR_ADDRESS	8876	
JMSSERVER SECURITY PORT	5557	server.xml

Default coexistence settings for port numbers

This topic provides reference information for identifying port numbers in WebSphere Application Server, Version 5.x in coexistence mode.

You must verify that you are using ports that do not conflict with other products and programs running on your node. These ports are the default coexistence ports. You can display the default coexistence ports when the installation wizard detects a previous installation of Application Server and displays the check box for selecting coexistence ports. Select the check box to show the ports. Change the ports to avoid conflicts.

You can also force the installation wizard to display the coexistence ports by using the following parameter on the installation command. The example shows the installation command for Linux and UNIX-based systems. You can find the commands for supported operating systems in "Installing the product" on page 43.

You can also force the appearance of the coexistence panel to change conflicting port number assignments. For example, the AIX WebSM system management server listens on port 9090 by default. To avoid a conflict with the administrative console port (HTTP_TRANSPORT_ADMIN) assignment, which is also 9090 by default, force the coexistence panel to appear using this command:

```
./install -W coexistenceOptionsBean.showCoexistence="true"
```

Default port numbers for coexistence for Version V5.1

After installing a product with the port numbers that you select on the coexistence panel (or in the response file for a silent installation), you can use the administrative console to configure port numbers and resolve conflicting ports.

Table 72. Default port settings for coexistence for V5.1

Port name	WebSphere Application Server	Network Deployment	File
	Value		
HTTP_TRANSPORT	9085	Not applicable	
HTTPS Transport Port (HTTPS_TRANSPORT)	9444	Not applicable	
HTTP Admin Console Port (HTTP_TRANSPORT_ADMIN)	9091	9091	server.xml virtualhosts.xml
HTTPS Admin Console Secure Port (HTTPS_TRANSPORT_ADMIN)	9044	9044	
Internal JMS Server (JMSSERVER_SECURITY_PORT)	5567	5567	server.xml

Table 72. Default port settings for coexistence for V5.1 (continued)

JMSSERVER_QUEUED_ADDRESS	5568	5568	serverindex.xml
JMSSERVER_DIRECT_ADDRESS	5569	5569	
BOOTSTRAP_ADDRESS	2809	9817	serverindex.xml
SOAP_CONNECTOR-ADDRESS	8880	8889	serverindex.xml
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	0	9819	serverindex.xml
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	0	9502	
CSIV2_SSL_MULTIAUTH_LISTENER_ADDRESS	0	9503	serverindex.xml
IBM HTTP Server Port	80	Not applicable	virtualhosts.xml, plugin-cfg.xml, and <i>IHSinstall_root/conf/</i> httpd.conf
IBM HTTPS Server Admin Port	8008	Not applicable	<i>IHSinstall_root/conf/</i> admin.conf
CELL_DISCOVERY_ADDRESS	Not applicable	7278	
ORB_LISTENER_ADDRESS	9100	9703	serverindex.xml
CELL_MULTICAST_DISCOVERY_ADDRESS	Not applicable	7278	

Detailed definitions of coexistence ports

Use the tables as a reference for finding information about the ports that appear on the coexistence panel of the installation wizard.

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: HTTP Transport Port Field name in the base WebSphere Application Server server.xml configuration file: HTTPTransport_1 EndPoint_1 Field name in the Network Deployment server.xml configuration file: Not applicable virtualhosts.xml: HostAlias_1 Field name in the silent installation response file: coexistencePanelBean.httpTransportPort Base WebSphere Application Server, default value: 9080 Network Deployment, default value: Not applicable 	9085

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: HTTPS Transport Port Field name in the base WebSphere Application Server server.xml configuration file: HTTPTransport_2 EndPoint_2 Field name in the Network Deployment server.xml configuration file: Not applicable virtualhosts.xml: HostAlias_3 Field name in the silent installation response file: coexistencePanelBean.httpsTransportPort Base WebSphere Application Server, default value: 9443 Network Deployment, default value: Not applicable 	9444

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: HTTP Admin Console Port Field name in the base WebSphere Application Server server.xml configuration file: HTTPTransport_3 EndPoint_3 Field name in the Network Deployment server.xml configuration file: HTTPTransport_1 EndPoint_1 virtualhosts.xml: HostAlias_4 Field name in the silent installation response file: coexistencePanelBean.adminConsolePort Default value: 9090 	9091

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: HTTPS Admin Console Secure Port Field name in the base WebSphere Application Server server.xml configuration file: HTTPTransport_4 EndPoint_4 Field name in the Network Deployment server.xml configuration file: HTTPTransport_1 EndPoint_2 virtualhosts.xml: HostAlias_5 Field name in the silent installation response file: coexistencePanelBean.secureAdminConsolePort Default value: 9043 	9044

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: Bootstrap port Field name in the base WebSphere Application Server serverindex.xml configuration file: BOOTSTRAP_ADDRESS EndPoint_1 Field name in the Network Deployment serverindex.xml configuration file: BOOTSTRAP_ADDRESS EndPoint_2 Field name in the silent installation response file: coexistencePanelBean.bootstrapPort Base WebSphere Application Server, default value: 2809 Network Deployment, default value: 9809 	Base: 2810, ND: 9917

1. You can use the connector type and port number parameters of certain commands to identify the bootstrap port number that you specify for the remote method invocation (RMI) connector address in a coexistence scenario. For example, to use the **wsadmin** command in a coexistence environment, you can specify either of the following commands:

```
wsadmin.bat/sh -conntype RMI -port portNumber
wsadmin.bat/sh -conntype SOAP -port portNumber
```

Alternatively, you can update the wsadmin.properties file located in the install_root/properties directory to contain the new port values. If you update this file, you do not need to specify the port parameter when using the wsadmin tool.

The SOAP connection type uses the SOAP connector address port number described next.

If you edit the serverindex.xml file, verify that the host name is the valid host name for the machine and not the generic localhost identifier.

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: SOAP Connector Address Field name in the base WebSphere Application Server serverindex.xml configuration file: SOAP_CONNECTOR-ADDRESS EndPoint_2 Field name in the Network Deployment serverindex.xml configuration file: SOAP_CONNECTOR-ADDRESS EndPoint_4 Field name in the silent installation response file: coexistencePanelBean.soapConnectorAddress Base WebSphere Application Server, default value: 8880 Network Deployment, default value: 8879 	Base: 8881, ND: 8889

Port names, default value, and description	Value
--	-------

You can use the connector type and port number parameters of certain commands to identify the port number that you specify for the SOAP connector address in a coexistence scenario. For example, to use the **wsadmin** command in a coexistence environment, you can specify either of the following commands:

```
wsadmin.bat/sh -conntype RMI -port portNumber
wsadmin.bat/sh -conntype SOAP -port portNumber
```

Alternatively, you can update the `wsadmin.properties` file located in the `install_root/properties` directory to contain the new port values. If you update this file, you do not need to specify the port parameter when using the `wsadmin` tool.

The remote method invocation (RMI) connection type uses the bootstrap port number described previously.

Port names, default value, and description	Value
--	-------

- Field name in the installation wizard: **JMS server queued address** 5568
- Field name in the base WebSphere Application Server `serverindex.xml` configuration file: `JMSSERVER_QUEUED_ADDRESS EndPoint_4`
- Field name in the Network Deployment `serverindex.xml` configuration file: **Not applicable**
- Field name in the silent installation response file: `coexistencePanelBean.jmsServerQueuedAddress`
- Default value: 5558

Port names, default value, and description	Value
--	-------

- Field name in the installation wizard: **JMS server security port** 5567
- Field name in the base WebSphere Application Server `server.xml` configuration file: `JMSServer_1 Address_3`
- Field name in the Network Deployment `server.xml` configuration file: **Not applicable**
- Field name in the silent installation response file: `coexistencePanelBean.jmsServerSecurityPort`
- Default value: 5557

Port names, default value, and description	Value
--	-------

- Field name in the installation wizard: **JMS Server Direct Address** 5569
- Field name in the base WebSphere Application Server `serverindex.xml` configuration file: `JMSSERVER_DIRECT_ADDRESS EndPoint_5`
- Field name in the Network Deployment `serverindex.xml` configuration file: Not applicable
- Field name in the silent installation response file: `coexistencePanelBean.jmsServerDirectAddress`
- Default value: 5559

Port names, default value, and description	Value
--	-------

- Field name in the installation wizard: **SAS SSL ServerAuth Address** Base: 0, ND: 9819
- Field name in the `serverindex.xml` configuration file: `SAS_SSL_SERVERAUTH_LISTENER_ADDRESS EndPoint_6`
- Field name in the silent installation response file: `coexistencePanelBean.sasSSLServerAuthAddr`
- Base WebSphere Application Server, default value: 0; With security, assign your own non-conflicting value: *nnnn*
- Network Deployment default value: 9401 but is not enabled until you enable security

Assign a coexistence port other than 0, which dynamically takes a free port. You must have a fixed port that clients can use to find the Secure Authentication Service (SAS) Secure Sockets Layer (SSL) ServerAuth address. No mechanism exists for clients to use to find a dynamically assigned port.

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: CSIV2 ServerAuth Listener Address Field name in the base WebSphere Application Server <code>serverindex.xml</code> configuration file: <code>CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS EndPoint_7</code> Field name in the Network Deployment <code>serverindex.xml</code> configuration file: <code>CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS EndPoint_8</code> Field name in the silent installation response file: <code>coexistencePanelBean.csivServerAuthListenerAddr</code> Base WebSphere Application Server, default value: 0; With security, assign your own non-conflicting value: <i>nnnn</i> Network Deployment, default value: 9403 but is not enabled until you enable security 	Base: 0, ND: 9502

Assign a coexistence port other than 0, which dynamically takes a free port. You must have a fixed port that clients can use to find the Common Secure Interoperability Specification, Version 2 (CSlv2) ServerAuth listener address. No mechanism exists for clients to use to find a dynamically assigned port.

Port names, default value, and description	Value
<ul style="list-style-type: none"> Installation wizard name: CSIV2 MultiAuth Listener Address Field name in the base WebSphere Application Server <code>serverindex.xml</code> configuration file: <code>CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS EndPoint_8</code> Field name in the Network Deployment <code>serverindex.xml</code> configuration file: <code>CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS EndPoint_7</code> Field name in the silent installation response file: <code>coexistencePanelBean.csivMultiAuthListenerAddr</code> Base WebSphere Application Server, default value: 0; With security, assign your own non-conflicting value: <i>nnnn</i> Network Deployment, default value: 9402 but you must enable security to enable this value 	Base: 0, ND: 9503

Assign a coexistence port other than 0, which dynamically takes a free port. You must have a fixed port that clients can use to find the Common Secure Interoperability Specification, Version 2 (CSlv2) MultiAuth listener address. No mechanism exists for clients to use to find a dynamically assigned port.

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the installation wizard: ORB Listener Port Field name in the base WebSphere Application Server <code>serverindex.xml</code> configuration file: Not applicable Field name in the Network Deployment <code>serverindex.xml</code> configuration file: <code>ORB_LISTENER_ADDRESS EndPoint_5</code> Field name in the silent installation response file: Not applicable - See note. Base WebSphere Application Server default value: Not applicable Network Deployment default value: 9100 	Base: 9109, ND V5.0.x: 9101, ND V5.1.x: 9703

Setting the object request broker (ORB) listener port to a value of 0 or greater causes all object references in the name space to specify their connection to the server and not depend on the location service daemon (LSD) to redirect connections to the server. An `ORB_LISTENER_ADDRESS` value equal to 0 starts the server on any available port and does not use the LSD. A value greater than 0 starts the server on the specified port and does not use the LSD.

Set the ORB listener port after a silent installation of the Network Deployment product.

Port names, default value, and description	Value
<ul style="list-style-type: none"> Field name in the Network Deployment installation wizard: Cell Discovery Address Field name in the base WebSphere Application Server installation wizard: Not applicable Field name in the <code>install_root\config\cells\nodeName\nodes\cellName\serverindex.xml</code> configuration file: <code>NamedEndPoint_1 endPointName= "CELL_DISCOVERY_ADDRESS"</code> Field name in the silent installation response file option: Not applicable Default value: 7277 	ND 5.0.x: 9989, ND 5.1.x: 7278

Port names, default value, and description	Value
Set the Cell discovery address after a silent installation of the Network Deployment product.	

Configuring WebSphere Application Server after migration

The installation wizard automatically configures IBM WebSphere Application Server, Version 5.1 and all other bundled products. There is no need for additional configuration if you do not migrate from an earlier version.

If you use the installation wizard to migrate a previous installation of WebSphere Application Server, Version 3.5.x, there are some items to review before considering your environment fully configured.

1. Check the `WASPostUpgrade.log` file for deployment details of the Version 3.5.x enterprise beans. Make any necessary changes and redeploy.

Note: No redeployment is required when moving EJB 1.1 JAR files from Version 4.

The J2EE programming model specifies an architecture for how applications are created and deployed. The `WASPostUpgrade` tool recreates applications because Version 3.5.x applications do not have the same architecture as Version 5.1 applications. The new Version 5.1 J2EE applications contain all migrated Web resources and enterprise beans. All Version 3.5.x enterprise applications become Version 5.1 J2EE applications with the same name, deployed in the same server.

The `WASPostUpgrade` tool maps Web resources and enterprise beans that are not included in an enterprise application, into a default J2EE application that includes the name of the server. The tool maps Web applications to J2EE WAR files. The tool deploys enterprise beans as EJB 1.1 beans in J2EE JAR files. The tool combines resources in a J2EE EAR file and deploys it in the Version 5.1 configuration. There are some differences between the EJB 1.0 and EJB 1.1 Specifications, but in most cases, EJB 1.0 beans can run successfully as EJB 1.1 beans.

Version 3.5.x supports only the EJB 1.0 components specification level. Version 5.1 supports EJB 1.1 components. However, many EJB 1.0 beans can successfully deploy as EJB 1.1 beans. The migration tools redeploy enterprise beans automatically as part of the application migration phase.

Note: After federating an Application Server node into a deployment manager cell, you cannot use the migration tools on the Application Server node. To use these tools again, remove the node from the cell, use the tools, and add the node to the cell again.

2. Examine any Lightweight Third Party Authentication (LTPA) security settings you might have used, and apply the settings in the WebSphere Application Server security settings.

The `WASPostUpgrade` tool migrates applicable security settings in the Version 3.5.x environment to J2EE security attributes.

Global security that uses Lightweight Third Party Authentication (LTPA) authentication in Versions 3.5.x and 4.0.x is migrated to the base WebSphere Application Server product and to the Network Deployment product. However, although global security was enabled in Versions 3.5.x and 4.0.x, it is disabled during migration to Version 5.1.

If you add this node later to an IBM WebSphere Application Server Network Deployment, Version 5.1 configuration, you can enable and use the LTPA configuration. Use the administrative console to generate keys for the migrated LTPA authentication mechanism. After generating the keys, you can enable global security.

Global security that uses *localos* authentication mechanisms in versions 3.5.x and 4.0.x is migrated to the Network Deployment product. However, although global security was enabled in Versions 3.5.x and 4.0.x, it is disabled during migration to Version 5.1. The Network Deployment product does not support the SWAM authentication mechanism. Migration sets the authentication mechanism in Version 5.1 to LTPA. Use the administrative console to generate keys for the migrated LTPA authentication mechanism. After generating the keys, you can enable global security.

3. Check the `WASPostUpgrade.log` file in the `logs` directory, for details about JSP 0.91 objects that the migration tools do not migrate.

Version 5.1 does not support JSP 0.91 objects. The migration tools do not migrate JSP objects configured to run as JSP 0.91 objects. The migration tools do, however, recognize the objects in the output and log them. Version 5.1 runs JSP 1.0 and 1.1 objects as JSP 1.2 objects, which is its only supported level.

4. Review Version 3.5.x models and clones to identify non-migrated objects that you must recreate in Version 5.1.

Version 3.5.x models and clones and Version 4.0.x server groups have been dramatically redefined in Version 5.1 as clusters and cluster members. Application servers are the only objects supported as models and cluster members in Version 5.1. This change is a significant difference from Version 3.5.x, in which many objects are models and clones. All models and clones relating to Application Servers are mapped to cluster members in Version 5.1.

During the migration of all other objects that you could previously clone, special mapping occurs. All clones are treated as simple objects and are mapped as if they are not cluster members. Models that are not Application Server models are ignored and not mapped.

Version 4.0.x Server Groups are converted to Version 5.1 clusters.

5. Identify and use the migration tools to migrate non-migrated nodes in Version 3.5.x and Version 4.0.x repositories that have multiple nodes.

A Version 3.5.x and a Version 4.0.x repository can contain more than one node name and its associated children. The `WASPostUpgrade` tool processes only those objects and children that match the migrating node. This determination is made by checking the names of nodes in configuration files with fully qualified and non-qualified network names of the migrating machine.

6. Examine any applications that use IBM extensions.

In many cases, IBM provides additional features and customization options that extend the specification level even further. If your existing Version 3.x applications use IBM extensions from earlier product versions, you might need to perform mandatory or optional migration to use the same kinds of extensions in the Version 5.1.

Migration from Version 4.0.x requires little conversion.

7. Update J2EE resources in client JAR files to the new resource format with the `ClientUpgrade` tool. J2EE applications might exist on the client, if the client has client JAR files with J2EE resources.

8. Migrate Version 3.5.x XML applications to supported XML APIs.

If your XML applications use XML for Java API, Version 2.0.x or earlier, you must migrate them to API Version 3.1 or the equivalent open-source version. Although there are inherent performance improvements in later versions of the XML for Java API, you can gain additional performance by explicitly using nonvalidating parsers in application environments where you can trust the data.

The most significant change is that the TX-compatible APIs are no longer available. The Document API retains the XML manipulation APIs that were in `TXDocument`, but you must rewrite the following functionality:

- Creating and loading an XML parser: Use a Java API for XML Processing (JAXP) factory class.
- Writing out the Document Object Model (DOM) tree: Use a serializer. One drawback to the DOM Level 2 implementation in this level of the XML for Java API is that the grammar (DTD or schema) is no longer a node in the DOM tree, so you cannot write it out. As a result, only external grammars are recommended. You can query the system ID of the root element and use it to retrieve the name from the statement. After the tree is written to an XML file, you can read the file as text and insert a statement.

In addition to the XML API changes, it is important to understand that the J2EE Version 1.3 specification mandates the use of JAXP 1.1, DOM Version 2, SAX Version 2 and SAX2 extensions, and XSL Transformations (XSLT) 1.0. JAXP Version 1.2, DOM Version 3, and SAX Version 3 are not allowed in products that are compliant with the J2EE Version 1.3 specification. This prohibition exists because the newer versions were *experimental* at the time of the J2EE Version 1.3 specification.

Because WebSphere Application Server is compliant with the J2EE Version 1.3 specification, WebSphere Application Server has support for JAXP Version 1.1, DOM Version 2 and SAX Version 2 only.

You must only recompile a Version 4.0.x XML application to migrate it to the Version 5.1 level.

9. Configure WebSphere Application Server to use a database. For example, you can configure WebSphere Application Server to use DB2, as described in “Configuring WebSphere Application Server for DB2 access” on page 375.
10. Review your Java virtual machine settings to verify that you are using a heap size of at least 50 for improved startup performance.

If you have used a smaller heap size in the past, you can use the default heap size, which is now 50.

Now you are finished with pre-test configuration. You might have to fine tune your WebSphere Application Server environment as you test it. Test all redeployed applications before moving them into production.

XML parser for Java code

On November 9, 1999, the Apache Software Foundation announced the creation of the `xml.apache.org` project for Open Source Extensible Markup Language (XML) solutions.

As part of that announcement, IBM announced that it was donating its XML4J, XML4C, and LotusXSL technologies to the `xml.apache.org` project. The parsing technologies have been renamed Xerces, and the LotusXSL technology has been renamed Xalan.

IBM is shifting its XML parsing development resources to work on the Xerces parsers. The objective is to use the Xerces code base as the foundation for XML4J and XML4C. This version of XML4J is based on the Apache Xerces2 code base.

What is the difference between XML4J and Xerces?

IBM has tested this version of XML4J in addition to the testing done by the `xml.apache.org` project. The names of the main Java archive (JAR) files have changed from `xml4j.jar` to `xmlParserAPIs.jar` and `xercesImpl.jar`, and from `xml4jSamples.jar` to `xercesSamples.jar`.

This version of XML4J contains one major API change and a few relatively minor API changes from the last major release. Its major features are many bugfixes and performance improvements, as well as the addition of a few features.

XML4J 4.2.x ships with the official W3C DOM HTML Recommendation API. The major change to the API was the removal of `setSelected` from the `HTMLOptionElement` interface. To workaround this change from XML4J 4.1.x, use `setAttribute` and `removeAttribute` from the interface to modify the selected attribute.

Which part of the API is public, and which is subject to change?

To answer this question, you must understand the terms used for application program interface (API) status:

- **Public** - The typical client developer codes for this API. Any severe problem in this API is addressed. Also, the majority of this API is defined as “public” and has reached World Wide Web Consortium (W3C) Recommendation status or a similar status in the XML Core Working Group. Not much change in these interfaces is expected. A W3C Recommendation indicates that a specification is stable, contributes to Web interoperability, and has been reviewed by all W3C members, who are in favor of supporting its adoption by the industry.
- **Experimental** - These interfaces and classes reflect the latest W3C specifications and Simple API for XML (SAX) specifications from the XML Core Working Group. Because these specifications are not

finalized, the interfaces are subject to change. As an experimental specification is adopted at the highest level, such as at the W3C Recommendation level, the specification is upgraded to the Public category.

- **Internal** - These classes are considered internal to Xerces, even though they might be public and have public methods. Developers with complex and specific needs, such as building an XML parser can use these classes. However, the architecture is subject to change.

The following table summarizes API status by interface content:

API Status	API contents (packages, interfaces, classes and methods)	Comments
Public	DOM Level 1 interfaces (found in org.w3c.dom, org.w3c.dom.html) and DOM Level 2 (DOM2) interfaces (found in org.w3c.dom and subpackages) The DOM2 interfaces are implemented in the same interfaces as DOM1 as new methods. SAX Level 1 interfaces (found in org.xml.sax. and subpackages) and SAX Level 2 (SAX2) interfaces (found in org.xml.sax. and subpackages)	DOM L1, DOM L2, SAX1, and SAX2 Interfaces are stable.
Experimental	DOM Level 3 (DOM3) and Core DOM Level 3 (DOM3) Abstract Schemas and Load and Save (found in org.apache.xerces.dom3 and subpackages).	DOM L3 is in working draft status. XML4J provides a subset of DOM L3 support.
Internal	All other packages are internal.	The internal Xerces architecture can change.

For more information see XML4J Information.

As an Apache Open Source project, the Xerces community is interested in your questions and feedback regarding the entire API, not just the part that is designated as public.

If you have specific questions, patches, or feedback regarding the Xerces API or code, visit the Apache Web site, and sign up for the mailing list. Send more basic questions or questions specific to XML4J to the AlphaWorks discussion forum.

Which APIs do I use for new development?

Use the org.apache.xerces.parsers.* classes for new development. The four compatibility parser classes are replaced by the following classes:

```
org.apache.xerces.parsers.SAXParser
org.apache.xerces.parsers.DOMParser
```

Validation control has become a feature of the parser, and no longer requires separate classes.

The future direction for the parser instantiation classes is a parser instantiation API that results from the W3C DOM Level 3 effort, which is just getting underway.

To make sure that your code is as stable as possible, use the interfaces that are specified in the Public section of the preceding table (for example DOM1 and SAX1). Updates are made to the table to reflect the new interfaces and the classes that are given Public status (for example, DOM2 and SAX2).

Is IBM making any additional support guarantees with XML4J?

IBM is not making any additional support guarantees for XML4J. In particular, IBM is not certifying that XML4J is Y2K compliant. XML4J makes no internal date calculations.

Migrating applications to use the XML4J 4.2.2 parser and the XSLT4J 2.5.4 transformer

The Java API for XML Processing (JAXP) specification defines a pluggability mechanism for a SAX or a DOM parser using the javax.xml.parsers APIs. Transformers are pluggable using the javax.xml.transform APIs.

The IBM SDK 1.4.1 bundles in Version 5.1 include an XML4J 4.2.2 parser and an XSLT4J 2.5.4 transformer. You can use a different implementation of JAXP in an application. Package the parser and transformer in the application.

You can change an application to remove its dependency on the API in a previous version of the parser or the transformer from an earlier version of WebSphere Application Server. Package the JAR files in the application.

In both cases, set the class loader delegation mode to PARENT_LAST on the application or Web module.

Recommendation: Have your applications use the JAXP API instead of using the parser or transformer implementation API directly.

You must recompile a V4.0.x XML application or a V5.0.x XML application to migrate it to the Version 5.1 level.

Configuring WebSphere Application Server for DB2 access

This topic describes configuring WebSphere Application Server on a Linux or UNIX-based platform to use multiple DB2 clients, including a V7 client and a V8.1 client.

In WebSphere Application Server Version 4, environment variables required for accessing a DB2 database, such as CLASSPATH, native library path, db2instance name and so on, are included or initialized in the WebSphere Application Server admin.config file or in the setupCmdLine.sh and the startServer.sh command scripts. Because WebSphere Application Server V5.x does not store administration repository information in a relational database management system (RDBMS), a database is not a prerequisite. Therefore, the installation and configuration of V5.x do not involve database configuration.

With the introduction of DB2 UDB Version 8.1, you can install both DB2 Version 7 and Version 8.1, or multiple DB2 Version 8.1 instances on the same Linux or UNIX machine. This topic describes configuring V5.x on a Linux or UNIX-based platform to use multiple DB2 clients, including a V7 client and a V8.1 client.

5.1 + Version 5.1 of WebSphere Application Server requires that you upgrade DB2 client and server nodes to DB2 V7.2 (Fix Pack 11) or DB2 V8.1.3 (Fix Pack 3) for IBM Developer Kit for Java 1.4.1 compatibility. A DB2 client at V8.1 with Fix Pack 3 can connect to a V7.2 database server, but not for XA, which some applications require. See the IBM WebSphere Application Server supported hardware, software, and APIs Web site at <http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html> for more information. You can also open the page by clicking **Prerequisites** on the LaunchPad.

For more information about migrating to DB2 V8.1.3, see Migration to Version 8 at <http://www-3.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8infocenter.d2w/report?target=mainFrame&fn=c0009326.htm>. (This Web address has spaces for formatting. Remove all spaces when pasting the address into a browser.)

If your WebSphere Application Server V5.x machine has only a Version 7 client or a Version 8.1 client installed, and all DB2 data sources defined in WebSphere Application Server access DB2 databases through this client, source the `db2profile` file in the login profile of your V5.x instance owner, invoke the script `db2_home/java12/usejdbc2` to use the JDBC2 drivers instead of the default JDBC1 drivers, and put the `DB2 lib` directory in the `java.library.path` variable.

For example, assume that the following values are true:

WebSphere Application Server instance owner, that is the administrative user who starts WebSphere Application Server	adm0001
DB2 client instance owner	db2inst1
DB2 client instance home	/export/home/db2inst1

1. Update the `.profile` of user `adm0001`.

Add the following line:

```
. /export/home/db2inst1/sql1lib/db2profile
```

You can also add the line to the `setupCmdLine.sh` script of WebSphere Application Server.

2. Specify the JDBC provider class path.

You have two ways to specify the JDBC provider class path on the DB2 JDBC Provider definition panel of the WebSphere Application Server administration console:

- Leave the default value (`${DB2_JDBC_DRIVER_PATH}/db2java.zip`) as is. Click **Environment > Manage WebSphere Variables** in the administrative console. Set the `DB2_JDBC_DRIVER_PATH` variable to the value `/export/home/db2inst1/sql1lib/java12`. This approach is preferred.
- Enter the class path, such as `/export/home/db2inst1/sql1lib/java12/db2java.zip`.

Windows platforms support only one DB2 installation. The DB2 environment variables are populated in the system environment automatically. WebSphere Application Server does not need to set these environment variables.

3. Set DB2-required environment variables for a particular Application Server.
 - a. On the administrative console of WebSphere Application Server, click **Application Servers > server > Configuration > Additional Properties > Process Definition > Additional Properties > Environment Entries > New** to create a new entry.
 - b. Create a `DB2INSTANCE` entry. Enter `DB2INSTANCE` in the **Name** field, and the instance name, such as `db2v7` in the **Value** field.
 - c. Create a library path entry. Create the appropriate library path:
 - **AIX:** `LIBPATH` entry
 - **Linux:** `LD_LIBRARY_PATH` entry
 - **Solaris:** `LD_LIBRARY_PATH` entry
 - **HP-UX:** `SHLIB_PATH` entry

Use the value of the DB2 native library path, such as `/export/home/db2v7/sql1lib/java12:/export/home/db2v7/sql1lib/lib` for a V7 client, or `/export/home/db2v8/sql1lib/lib` for a V8.1 client.

The `DB2 lib` directory must be on the `java.library.path` path. Otherwise the Application Server cannot load the `db2jdbc.so` library and cannot work with DB2. Even with the `DB2 lib` directory on the `java.library.path` path, you must invoke the `/home/db2inst1/db2profile` environment into the root shell before you start the Application Server.

4. Configure the WebSphere Application Server V5.x product to use both DB2 V7 and V8.1 clients.

In cases where both a DB2 V7 client and a V8.1 client are installed on a WebSphere Application Server V5.0.x machine, or multiple V8.1 clients are installed on a WebSphere Application Server V5.0.x or V5.1.x machine, and you intend to use two or more clients in your V5.x Application Servers, you can set DB2 environment variables based on each Application Server, instead of setting them globally as shown previously.

DB2 UDB V8 uses a new client and server communications mechanism that is based on distributed relational database architecture. While the new communications mechanism provides a number of advantages, it also introduces restrictions on the communications capability between DB2 V7 and V8 products.

Because a number of restrictions exist when using a V8 client to communicate with a V7 server, this configuration is not recommended. However, a V7 client can access a V8 server without difficulty.

If you have a WebSphere Application Server application that accesses both a V7 server and a V8 server, only one V7 client is required on the Application Server instance. Use this client to access both the V7 server and the V8 server.

You can choose to use a V7 client to access a V7 server, and use a V8 client to access a V8 server. This choice results in two versions of DB2 clients on the same WebSphere Application Server machine.

When you use a V7 client to access a V8 server, you must explicitly bind V7 packages to the V8 server. To bind packages in the V7 client environment, make a connection to the V8 server and bind both the `db2cli.lst` and the `db2ubind.lst` files. For example:

```
cd /home/db2inst1/sql1lib/bnd
db2 connect to v8server
db2 bind @db2cli.lst
db2 bind @db2ubind.lst
db2 terminate
```

One WebSphere Application Server node can support multiple Application Server instances. Each Application Server is essentially a single Java run-time environment, which is one Java virtual machine (JVM). Each JVM can have its own set of environment variables that differ from other Application Server instances.

You can set DB2 environment variables per Application Server instance to let each Application Server instance communicate with a single DB2 client instance. The client instance can have multiple databases cataloged.

Such a configuration means that you cannot have one Application Server instance communicating with both a V7 client and a V8.1 client. However, you can create another Application Server instance to communicate with a different DB2 client.

5. Match the appropriate data source to the application component.

When mapping a data source to an application component, do not mismatch data sources from different DB2 client instances.

For example, if you set the `server1` Application Server instance to run in the DB2 V7 client instance, `server1` application components can use only data sources defined under the same V7 client JDBC driver.

WebSphere Application Server V5.x supports defining a JDBC provider on different scopes: cell, node (the default) and server. If you have different DB2 client instances, consider defining them on the server level instead of on the node level. Server level definitions avoid possible mismatches between the data sources and different DB2 JDBC providers.

You can configure DB2 for use on WebSphere Application Server. Tune your WebSphere Application Server environment as you test it. Test all redeployed applications before moving them into production.

Installation: Resources for learning

Use the following links to find relevant supplemental information about installation. The information resides on IBM and non-IBM Internet sites, whose sponsors control the technical accuracy of the information.

These links are provided for convenience. Often, the information is not specific to the IBM WebSphere Application Server product, but is useful all or in part for understanding the product. When possible, links are provided to technical papers and Redbooks that supplement the broad coverage of the release documentation with in-depth examinations of particular product areas.

View links to additional information about:

- Planning, business scenarios, and IT architecture
- Programming model and decisions
- Programming instructions and examples
- Programming specifications
- Administration
- Support

Planning, business scenarios, and IT architecture

- IBM WebSphere Application Server supported hardware, software, and APIs
The official site for determining product prerequisites for hardware, software and APIs for all WebSphere Application Server products.
- IBM developerWorks WebSphere
The home of technical information for developers working with WebSphere products. You can download WebSphere software, take a fast path to developerWorks zones, such as VisualAge Java or WebSphere Application Server, learn about WebSphere products through a newcomers page, tutorials, technology previews, training, and Redbooks, get answers to questions about WebSphere products, and join the WebSphere community, where you can keep up with the latest developments and technical papers.
- IBM WebSphere Application Server library and information centers Web site
The IBM WebSphere Application Server Library Web site contains links to all WebSphere Application Server information centers, for all versions. It also lets you access each information center in your native language.
- IBM WebSphere Application Server home page
The IBM WebSphere Application Server home page contains useful information, including support links and downloads for fixes, APARs, tools, and trials.
- IBM WebSphere software platform home page
The IBM WebSphere software platform home page introduces WebSphere products and describes how companies can easily transform to an e-business, with software that can grow as fast as the business it supports.
- Migrating to WebSphere V5.0: An End-to-End Migration Guide, SG24-6910-00
This IBM Redbook is the definitive migration guide for migrating earlier versions of WebSphere Application Server to Version 5. The Redbook adds a broader scope, including planning for application migration and WebSphere Studio Application Developer tooling and samples. Read this book to formulate an optimal migration strategy.
- The power of Edge of Network technology in IBM WebSphere Application Server Version 5
A description of WebSphere Application Server Edge Components Version 5.
- WebSphere Application Server - Express, V5
A description of WebSphere Application Server Express, Version 5.
- WebSphere Application Server, Version 5
A description of the base product, WebSphere Application Server, Version 5.
- IBM WebSphere Business Integration Server Foundation, Version 5
A description of IBM WebSphere Business Integration Server Foundation, Version 5.
- Information center for WebSphere Application Server Edge components
The information center for WebSphere Application Server Edge components contains complete documentation for the Caching Proxy and the Load Balancer in these PDF online books, *WebSphere Application Server Concepts, Planning, and Installation for Edge Components*, the *WebSphere*

Application Server Caching Proxy Administration Guide, the *WebSphere Application Server Load Balancer Administration Guide*, and the *WebSphere Application Server Programming Guide for Edge Components*.

- Linux on IBM eserver zSeries and S/390: ISP/ASP Solutions

This IBM Redbook describes how Linux can be combined with z/VM on zSeries and S/390 hardware. This combination of hardware and operating systems enables Internet Service Providers (ISP) and Application Service providers (ASP) to more efficiently provide services.

The book assumes a broad definition of ASP that includes production enterprise solutions as simple as file serving. In a world of discrete servers, when a new resource is required, workload can either be added to an existing server or a new one can be purchased. Often a new server is installed and the server farm grows in the enterprise.

S/390 and zSeries hardware, microcode and software allow physical resources to be made virtual among Linux systems. Virtual physical resources allow many hundreds of Linux systems to exist on a single server. Running multiple Linux images as guests of VM/ESA or z/VM is a smart choice.

- Performance Analysis for Java Web sites

A guide to developing applications that do not have performance and scalability problems when they get deployed to production. The *Performance Analysis for Java Web Sites* book walks you through planning, designing, and testing to verify that you are not surprised when you release your new site. All of the examples use WebSphere Application Server products and applications.

Stacy Joines, Ruth Willenborg, and Ken Hygh have helped many customers with performance-related issues, and have authored this book to capture their experiences, and to help you test and tune your applications.

The book is available in book stores and on Amazon.

- developerWorks: Patterns for e-business: Redbooks listing

This Web page lists links to pattern resources under these categories:

- Current patterns Redbooks
- Superseded patterns Redbooks (valid for back-level product versions)
- Independent analyst reports
- Patterns CD order offer
- Back-level version patterns Web site (zip downloads and old Flash tutorial)
- Customer references
- White papers
- Multimedia presentations and screen cams
- Webcasts
- Patterns development kit
- WebSphere technical exchange presentations

- developerWorks: IBM Patterns for e-business

The IBM developerWorks site is the source for IBM patterns for e-business, a set of tested, reusable intellectual assets that you can use to design and implement your e-business network and architecture!

- Design for Scalability - An Update

This White paper is from the IBM High Volume Web Sites team. The White paper describes component selection and management techniques you can use to make your Web site ready to adapt to increasing traffic. These techniques are the product of IBM experiences while working with customers seeking to improve the performance and availability of some of the largest Web sites in the world.

Abstract: Optimizing for scalability remains a significant challenge for e-businesses as they balance the demands for availability, reliability, security, and high performance. Vendors are responding with infrastructure options and supporting hardware and software platforms that address these requirements. This update identifies current products and emerging trends that are most likely to improve the scalability of your e-business infrastructure.

- IBM WebSphere Application Server V5.0 System Management and Configuration: WebSphere Handbook Series

This IBM Redbook provides the knowledge needed to implement WebSphere Application Server V5.0, Network Deployment runtime environment, to package and deploy Web applications, and to perform

ongoing management of the WebSphere environment. It is one in a series of handbooks, designed to give you in-depth information on the entire range of WebSphere Application Server products.

This book provides an overview of the architecture, topology options, and new features of WebSphere Application Server Version 5 and WebSphere Application Server Network Deployment Version 5. It takes you through the installation steps needed to install each topology. Platform-specific chapters are included for installation on Windows, AIX, and Solaris.

The book then takes you through the process of configuring WebSphere Application Server. It is organized in the same manner as the WebSphere Application Server Administrative Console. It also includes information on packaging and deploying applications. It concludes with information on troubleshooting runtime problems.

- IBM WebSphere V4.0 Advanced Edition Handbook

This Redbook describes base application topologies and product mappings for WebSphere Application Server. Refer to the IBM Redbooks Web site for the latest update.

- The User centered design (UCD) for different project types, part 1

This Web page is the first of two articles posted to IBM developerWorks that describes useful application design activities for different types of projects.

- The User centered design (UCD) for different project types, part 2

This Web page is the latest of two articles that describes design activities that IBM scientists have found most useful in various types of projects. This article defines user interface design elements, including the design prototype, use case model, and design specification document.

Programming model and decisions

- Designing e-business Solutions for Performance

This White paper describes how the design or implementation of an e-business application can affect performance.

- Managing Web Site Performance

This White paper contains tips and techniques for developers building applications that use session persistence. It also helps administrators to tune the WebSphere Application Server product appropriately for these applications.

Programming instructions and examples

- IBM developerWorks

IBM developerWorks contains many excellent resources for developers, including tutorials on Web development-related topics. There is an excellent tutorial on the JDBC API.

- IBM Redbooks

The IBM Redbooks site contains many WebSphere Application Server related documents.

Programming specifications

- J2EE information

For more information about J2EE specifications, visit the Sun site.

- sun.net.inetaddr.ttl property

The following Java 2 SDK, Standard Edition 1.4 Web site describes the private sun.net.inetaddr.ttl property, which works in both Java 2 SDK, Standard Edition 1.3 (WebSphere Application Server V5.0.0, V5.0.1, and V5.0.2) and Java 2 SDK, Standard Edition 1.4 (WebSphere Application Server V5.1).

- java.net.URLConnection class

The *Networking* section of this Java 2 SDK, Standard Edition 1.4 Web site describes a change in the behavior of the java.net.URLConnection class.

Administration

- Best Practices Zone on WSDD

The WebSphere Best Practices Zone is a collection of best practices for administering WebSphere Application Server. Over time, the zone is intended to grow to include best practices for using other WebSphere software products, and to cover more topics. Use the feedback mechanism to submit your best practice suggestions.

- The IBM Glossary of Computing Terms

This glossary defines technical terms used in many IBM products. It is not a comprehensive resource of all IBM computing terms. This resource is provided for information purposes only and is updated periodically. IBM takes no responsibility for the accuracy of the information it contains.

Support

- WebSphere education on demand: Installation best practices tutorials

IBM Support offers tutorials on various WebSphere Application Server installation topics. Topics include:

- Installing - where you install a new product or install a coexisting product (on the same machine as an existing product)
- Upgrading - where you use the update installer program to apply service to an existing installation
- Migrating - where you have an older product and you want to install a newer one but retain as many characteristics of the old system as possible
- Uninstalling - where you want to remove an existing product and reinstall the product or a newer version

- Ten Steps to Getting Support for WebSphere Application Server

If you are new to a product, you might have difficulty finding all the information you need. And if you come across a problem, where do you go for help? Whether you are a new user looking for introductory information, or an experienced user looking for a workaround for a specific defect, you can benefit immediately from extensive Web-based support from IBM. It enables you to download fix packs, search on keywords, look up FAQs, Hints and Tips, and so forth. Always use this Web resource before contacting IBM Support directly.

- WebSphere Application Server Support page

Take advantage of the Web-based Support and Service resources from IBM to quickly find answers to your technical questions. You can easily access this extensive Web-based support through the IBM Software Support portal at URL <http://www.ibm.com/software/support/> and search by product category, or by product name. For example, if you are experiencing problems specific to WebSphere Application Server, click **WebSphere Application Server** in the product list. The WebSphere Application Server Support page appears.

- AIX Fix Distribution Service Web site

A Web facility for downloading AIX Version 4 and AIX Version 3 fixes, with a limited search engine designed with the assumption that you know what fix you need. If you do not know what fix you need, there is a pointer at the Web site to the APAR Database Facility. You can also contact your authorized IBM business partner or IBM Support Center.

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, New York 10594 USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks and service marks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

- AIX
- AS/400
- CICS
- Cloudscape
- DB2
- DFSMS
- Domino
- Everyplace
- iSeries
- IBM
- IMS
- Informix
- iSeries
- Language Environment
- Lotus
- MQSeries
- MVS
- OS/390
- OS/400
- Power Architecture
- PowerPC
- RACF
- Rational
- Redbooks
- RMF
- SecureWay
- SupportPac
- Tivoli
- ViaVoice
- VisualAge
- VM/ESA
- VTAM
- WebSphere
- z/OS
- zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

LINUX is a trademark of Linus Torvalds in the U.S., other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel Inside (logos), MMX and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SET and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product and service names may be trademarks or service marks of others.



Printed in USA

SC31-6371-04

