WebSphere® Application Server V4.0.1 for z/OS and
OS/390

Migration

WebSphere® Application Server V4.0.1 for z/OS and OS/390

Migration

Contents

Figures

Tables

About this book

This book describes migration procedures for WebSphere for z/OS.

Note: The full product name is "WebSphere Application Server V4.0.1 for z/OS and OS/390", hereafter referred to in this text as "WebSphere for z/OS" or the "Application Server."

Who should read this book

This book is intended for those migrating to WebSphere for z/OS from the following WebSphere Application Server products:

- WebSphere Application Server V3.02 (Standard Edition feature) for OS/390 (V3.02SE)
- WebSphere Application Server V3.5 Standard Edition for OS/390 (V3.5 SE)
- WebSphere for z/OS V4.0
- WebSphere Application Server Advanced Edition V4.0 (V4.0 AE)
- It is intended for system programmers, security administrators, network administrators, or database administrators who will be migrating the WebSphere runtime from one release to another.
- It is also intended for application programmers who fulfill the tasks defined in the Sun Microsystems Java 2 Enterprise Edition Specification V1.2 for the roles of Application Component Provider, Application Assembler, and Deployer. For details about those roles and associated responsibilities, refer to the Sun Microsystems J2EE specification, which is available at:

  `http://java.sun.com/`

Where to find related information, tools, and supplements

This is a list of books that are in the WebSphere for z/OS library. They can be found at the following Web site:

`http://www.ibm.com/software/webservers/appserv/`

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Program Directory*, GI10-0680, describes the elements of and the installation instructions for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: License Information*, LA22-7855, describes the license information for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834, describes the planning, installation, and customization tasks and guidelines for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Messages and Diagnosis*, GA22-7837, provides diagnosis information and describes messages and codes associated with WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Operations and Administration*, SA22-7835, describes system operations and administration tasks.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836, describes how to develop, assemble, and install J2EE applications in a WebSphere for z/OS J2EE server.

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling CORBA Applications*, SA22-7848, describes how to develop, assemble, and deploy CORBA applications in a WebSphere for z/OS (MOFW) server.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: System Management User Interface*, SA22-7838, describes the system administration and operations tasks as provided in the Systems Management User Interface.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: System Management Scripting API*, SA22-7839, describes the functionality of the WebSphere for z/OS Systems Management Scripting API product.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Migration*, GA22-7860, describes migration procedures for WebSphere for z/OS.

Here are some other WebSphere Application Server books on that Web site that you might find particularly helpful:

- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835, provides information about running the WebSphere for z/OS plug-in within the HTTP Server address space. You can use this configuration if you want to continue running non-J2EE-compliant Web applications in the V4.0.1 WebSphere for z/OS plug-in within the HTTP Server address space while migrating to the full WebSphere for z/OS run time.
- *Getting Started with WebSphere Application Server*, SC09-4581, provides an overview of the WebSphere Application Server family of products.
- *Building Business Solutions with WebSphere*, SC09-4432

The integrated WebSphere Application Server Advanced Edition and WebSphere Application Server Enterprise Edition InfoCenter includes CORBA (MOFW) information you need to code CORBA (MOFW) components. Go to:

`http://www.ibm.com/software/webservers/appserv/infocenter.html`

For additional WebSphere for z/OS tools and supplements, go to the following Web site:

`http://www.ibm.com/software/webservers/appserv/zos_os390/download.html`

You might also need to refer to information about other z/OS or OS/390 elements and products. All of this information is available through links at the following Internet locations:

`http://www.ibm.com/servers/eserver/zseries/zos/`
`http://www.ibm.com/servers/s390/os390/`

How to send your comments

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

Summary of changes

This is the first edition of this book. Migration information has been moved from *WebSphere Application Server V4.0 for z/OS and OS/390: Installation and Customization, GA22–7834* and *WebSphere Application Server V4.0 for z/OS and OS/390: Assembling J2EE Applications*, SA22–7836 and merged into this book. We've added new information as well. This book is now organized into the following parts:

- "Part 1. General migration considerations" on page 1
- "Part 2. Migrating from WebSphere V3.02SE or V3.5SE to WebSphere V4.0.1" on page 13
- "Part 3. Migrating from WebSphere V4.0 to WebSphere V4.0.1" on page 57
- "Part 4. Migrating applications from WebSphere AE to WebSphere for z/OS V4.0.1" on page 95

Note: This document does not cover WebSphere Application Server Version 3.0.2 Enterprise Edition (EE) migration to WebSphere Application Server V4.0.1 for z/OS and OS/390.

Part 1. General migration considerations

This part of the book contains general migration considerations from previous releases of WebSphere to WebSphere V4.0.1. and contains the above chapters.

1

Chapter 1. Migration overview

Your plan for migrating to the new level of WebSphere for z/OS should include information from a variety of sources. These sources of information describe topics such as coexistence, service, hardware and software requirements, installation and migration procedures, and interface changes.

The following documentation, which is supplied with your product order, provides information about installing your OS/390 system.

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Program Directory*, GI10-0680

  This document leads you through specific installation steps and is provided with your WebSphere for z/OS product order.

- *ServerPac Installing Your Order*

  This is the order-customized, installation book for using the ServerPac Installation method. Be sure to review the product information in the appendixes, which describes data sets supplied, jobs or procedures that have been completed for you, and product status. IBM may have run jobs or made updates to PARMLIB or other system control data sets. These updates could affect your migration.

Within this book, you can find information about the specific updates and considerations that apply to this release of WebSphere for z/OS.

- "Chapter 2. Migration roadmap" on page 7

  This section identifies the migration paths that are supported with the current level of WebSphere for z/OS. It also describes the additional publications that can assist you with your migration to the current level.

- "Chapter 11. New function in WebSphere for z/OS V4.0.1" on page 65

  This section describes the specific updates that were made to WebSphere for z/OS for the current release. For each item, this section provides an overview of the change, a description of any migration and coexistence tasks that may be considered, and where you can find more detailed information in the WebSphere for z/OS library or other element libraries.

Terms you need to know

This section describes some terms you may need to know as you use this book.

Migration Activities that relate to the installation of a new version or release of a program to replace an earlier level. Completion of these activities ensures that the applications and resources on your system will function correctly at the new level.

Coexistence Two or more systems at different levels (for example, software, service or operational levels) that share resources. Coexistence includes the ability of a system to respond in the following ways to a new function that was introduced on another system with which it shares resources: ignore a new function, terminate gracefully, support a new function. The following are examples of configurations in which resource sharing can occur:

- WebSphere Application Server Standard Edition for OS/390 V3.02 and WebSphere for z/OS
- WebSphere Application Server Standard Edition for OS/390 V3.5 and WebSphere for z/OS

Exploitation   Activities related to taking advantage of optional functional enhancements for a release.

Interoperability

Two or more systems on differing platforms that communicate with each other. For example, a client on a WebSphere distributed platform interoperates with a server on WebSphere for z/OS.

Developing a migration strategy

The recommended steps for migrating to a new release of WebSphere for z/OS are:

1. Become familiar with the supporting migration and installation documentation for the release.

   You should determine what updates are needed for products that are supplied by IBM, system libraries, and non-IBM products. Review "Chapter 2. Migration roadmap" on page 7 and "Chapter 1. Migration overview" on page 3 for information about WebSphere for z/OS.

2. Develop a migration plan for your installation.

   When planning to migrate to a new release of WebSphere for z/OS, you must consider high-level support requirements, such as machine and programming restrictions, migration paths, and program compatibility.

3. Obtain and install any required program temporary fixes (PTFs) or updated versions of the operating system.

   Call the IBM Software Support Center to obtain the pre-requisite products and preventive service planning (PSP) upgrade for WebSphere for z/OS, which provides the most current information about PTFs for WebSphere for z/OS. Check RETAIN again just before testing WebSphere for z/OS. For information about preventive service planning, refer to *WebSphere Application Server V4.0.1 for z/OS and OS/390: Program Directory*, GI10-0680. Although the *Program Directory* contains a list of the required PTFs, the most current information is available from the IBM Software Support Center.

4. For coexistence of your current version of WebSphere for z/OS and WebSphere Application Server V4.0.1 for z/OS and OS/390, choose different mount points. See "Choosing mountpoints for WebSphere V4.0.1" on page 17 for more information.

5. Determine whether you need to install WebSphere V4.0.1 for the first time, or migrate to WebSphere V4.0.1. IBM recommends that you use the customization dialog which supports either. For installing for the first time, see "Installing and customizing your first runtime" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. For migrating to V4.0.1, see "Steps for performing a warm start from WebSphere V4.0 to WebSphere V4.0.1" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

6. Determine whether you can do a rolling warm start in a sysplex or if you need or can afford to do a disruptive upgrade. More information can be found on these in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

7. Install the product using *WebSphere Application Server V4.0.1 for z/OS and OS/390: Program Directory*, GI10-0680 or the *ServerPac Installing Your Order* documentation.

8. Contact programmers who are responsible for updating applications at your installation.

   Verify that your installation's applications will continue to run, and, if necessary, make changes to ensure compatibility with the new release.

9. Decide among differing possibilities for migrating your applications.

10. If necessary, customize the new function for your installation.

11. Exercise the new functions.

Reviewing changes to WebSphere for z/OS processing

As you define your installation's migration plan, consider how the new and changed WebSphere for z/OS support might affect the following areas of WebSphere for z/OS processing. For each item described in "Chapter 1. Migration overview" on page 3, you should review the "What this change affects" and "Migration procedures" sections to determine how, or if, the support affects the tasks that are performed at your installation.

Administration	Administrators must be aware of how changes introduced by a new product release can affect an installation's data processing resources. Changes to real and virtual storage requirements, performance, security, and integrity are of interest to administrators or to system programmers who are responsible for making decisions about the computing system resources used with a program.
Application development	Application development programmers must be aware of new functions introduced in a new release of WebSphere for z/OS. To ensure that existing programs run as before, your application programmers need to know about any changes in application programming interfaces and processing requirements. This book provides an overview of the changes that might affect existing application programs.
Auditing	Typically, auditors are responsible for ensuring proper access control and accountability for their installation. This book identifies any changes to security options, audit records, and report generation utilities.
Customization	To meet the specific requirements of your installation, you can customize WebSphere for z/OS functions to take advantage of new support after the product is installed. For example, you can tailor WebSphere for z/OS to improve performance. This book lists changes to WebSphere for z/OS that might require your installation to tailor the product, either to ensure that WebSphere for z/OS runs as before or to accommodate new security controls that your installation may need.

General user	This book provides an overview of the changes that might affect existing procedures for general users.
Operations	The new WebSphere for z/OS release might introduce changes to its operating characteristics, such as changed commands, new or changed messages, or in the methods of implementing new functions. This book identifies those changes for which you should provide user education before running this release of the product.

Reviewing changes to WebSphere for z/OS interfaces

When defining your installation's migration plan, also consider that WebSphere for z/OS interfaces may also be affected by the new or changed functions that are introduced in this release. These interfaces include:
- Commands
- Database templates
- Messages
- Panels
- SMF Records
- Utilities

Chapter 2. Migration roadmap

This section describes the migration paths that are supported by the current release of WebSphere for z/OS. It also provides information about how you can obtain the WebSphere for z/OS migration information from previous releases.

You can migrate to WebSphere Application Server V4.0.1 for z/OS and OS/390 from the following releases:

- WebSphere Application Server Standard Edition for OS/390 V3.02 (hereafter referred to as "WebSphere V3.02SE").
- WebSphere Application Server Standard Edition for OS/390 V3.5 (hereafter referred to as "WebSphere V3.5SE").
- WebSphere Application Server V4.0 for z/OS and OS/390
- WebSphere Application Server Advanced Edition (hereafter referred to as "WebSphere AE"

You can also migrate J2EE applications from other platforms to WebSphere Application Server V4.0.1 for z/OS and OS/390.

The roadmaps in this section provide an overview of each migration.

WebSphereV3.02SE or V3.5SE to WebSphere V4.0.1 summary

The migration from Standard Edition V3.02 and Standard Edition V3.5 is nearly the same, with the following exceptions:

- If you are migrating from Standard Edition V3.02, you can either migrate your applications directly to V4.0.1, or you can migrate them to Standard Edition V3.5, and then, over time, to V4.0.1. V3.5 applications can be run in a V4.0.1 environment provided you specify the fully qualified name of the V3.5 was.conf file as the second parameter on the ServerInit directive in the hosting Web server's httpd.conf configuration file.

  For more information about migrating to WebSphere Application Server Standard Edition for OS/390 V3.5, see *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835.

- To migrate directly to WebSphere for z/OS V4.0.1 from Standard Edition V3.02, the following are steps to be considered:
  - "Choosing mountpoints for WebSphere V4.0.1" on page 17
  - "Determining system requirements for WebSphere V4.0.1" on page 18
  - "Chapter 5. Installing the WebSphere Application Server V4.0.1 for z/OS and OS/390 runtime" on page 21
  - Migrate your Java environment and your applications to the following levels:
    - Update your JDK to SDK 1.3
    - Update your servlets to Java™ Servlet Specification V2.2
    - Update your JSPs to JavaServer Pages V1.1 specification
    - Repackage your applications as a .war file
- To migrate your applications from Standard Edition V3.5 to WebSphere for z/OS V4.0.1, you must make sure
  - Your servlets are written to Java Servlet Specification V2.2

– Your JSPs are written to JavaServer Pages V1.1 specification
– Your applications are packaged as a .war file

For information about...	See . . .
Operating system and database requirements	"Operating system and database requirements" on page 24
Process/execution model differences	"Process/execution model differences" on page 28
Application assembly and deployment differences	"Application assembly and deployment differences" on page 31
WebSphere HTTP session state database repository	"WebSphere HTTP session state database repository" on page 33
Security mechanism	"Security mechanism" on page 35
Common Connector Framework support	"Common Connector Framework support" on page 38 See the "Download Connector Support" for J2EE Connector Architecture based Beta connectors, located at: `http://www.ibm.com/software/webservers/appserv/download_v4z.html`
Accessing CICS	"Accessing CICS" on page 40
Accessing IMS	"Accessing IMS" on page 42
Accessing DB2 for OS/390 through JDBC	"Accessing DB2 for OS/390 through JDBC V2.0 Standard Extension DataSource APIs" on page 44
JRas support	"JRas support" on page 46
J2EE services for Web applications	"J2EE services for Web applications" on page 48
RunAs	"RunAs" on page 83

WebSphere V4.0 to WebSphere V4.0.1 summary

The migration from WebSphere V4.0 to WebSphere V4.0.1 has the following considerations:

- Depending on whether you are in a monoplex or sysplex, there are various choices you may have to make to migrate from WebSphere V4.0 to WebSphere V4.0.1. For more information, see the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0." in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

- You need to determine whether you used the customization dialog in configuring your WebSphere V4.0. There are various configuration choices you may have to make based on whether you used the customization dialog in configuring V4.0. See Table 15 on page 63 for more information on some decision paths that may help you in your installation and configuration to V4.0.1.

- To migrate directly to WebSphere for z/OS V4.0.1 from WebSphere for z/OS V4.0, the following are steps to be considered:
  - Creating the proper HFS structure for upgrades found in "Migrating your runtime from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1" on page 62
  - Backing up your WebSphere for z/OS V4.0 system, also found in "Migrating your runtime from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1" on page 62

- No application changes are required if the warmstart process is utilized when migrating to WebSphere for z/OS V4.0.1 from WebSphere for z/OS V4.0.

For information about the following areas of processing:

- Administration
- Application development
- Auditing
- Customization
- General user
- Operations
- Interfaces

see "Part 3. Migrating from WebSphere V4.0 to WebSphere V4.0.1" on page 57 for the tasks needed to migrate from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1. This section also contains the new V4.0.1 functions and contains the following on each new function:
- Description
- Summary of the WebSphere for z/OS tasks or interfaces that might be affected
- Coexistence considerations, if any, that are associated with the item
- Migration procedures, if any, that are associated with the item
- References to other publications that contain additional detailed information

WebSphere AE to WebSphere V4.0.1 summary

See "Part 4. Migrating applications from WebSphere AE to WebSphere for z/OS V4.0.1" on page 95 for the tasks needed to migrate from WebSphere AE to WebSphere V4.0.1.

Summary of SE V3.02, SE V3.5, V4.0, and V4.0.1 WebSphere Application Server characteristics and functional differences

Table 1 shows the various releases of the WebSphere for z/OS family and the specification and functional differences between them.

Table 1. Specification and functional differences between WebSphere for z/OS releases

Specifications	WebSphere V3.02SE	WebSphere V3.5SE	WebSphere V4.0	WebSphere V4.0.1

Java related	• Requires SDK 1.1.8
	• Supports specification levels:
	  – Servlet 2.1
	  – JSP 1.0
	  – JDBC 1.1
		• Requires SDK 1.3
		• Supports specification levels:
		  – Servlet 2.1/2.2
		  – JSP .091/1.0/1.1
		  – JDBC 1.2
			• Requires SDK 1.3
			• Supports J2EE 1.2 levels:
			  – Servlet 2.2
			  – JSP 1.1
			  – JDBC 2.0
			  – EJB 1.1
			  – JNDI 1.2
			  – JTA 1.0
			  – RMI/IIOP 1.0
				• Requires SDK 1.3
				• Supports J2EE 1.2 levels:
				  – Servlet 2.2
				  – JSP 1.1
				  – JDBC 2.0
				  – EJB 1.1
				  – JNDI 1.2
				  – JTA 1.0
				  – RMI/IIOP 1.0
				  – JMS 1.1
				  – JavaMail 1.1
				  – JAP 1.0
				  – Client container

Table 1. Specification and functional differences between WebSphere for z/OS releases (continued)

Web Services	N/A	N/A	N/A	Introduces Web Services equivalent to those provided by:
				
				 • WebSphere AE V4.0
				  – HTTP to Stateless Session beans
				 • Requires SDK 1.3 PTF available 10/2001
				 • Supports specification levels:
				  – SOAP 1.1
				  – Apache SOAP V2.2

- **WebSphere V3.02SE** provides a JAVA runtime for executing Web applications consisting of servlets/JSPs.
- **WebSphere V3.5SE** adds support for Servlet/JSP specification levels required by the J2EE 1.2 specifications, allowing customers to take advantage of the latest levels and to start early migration of applications in preparation for movement to J2EE 1.2 servers.
- **WebSphere V4.0** introduces a J2EE server that provides support for enterprise applications consisting of Web applications and enterprise java beans (EJBs). Support for Servlets, JSPs, and EJBs is compliant to the J2EE 1.2 architecture and provides all the benefits of this architecture. Tooling supports the J2EE 1.2 required packaging scheme. The J2EE server, unlike prior JAVA runtimes provided by WAS SE, provides container managed services for items such as transactions and security, simplifying the application programmers job. In addition the J2EE server utilizes the z/OS infrastructure to provide QOS expected on the 390 platform.
- **WebSphere V4.0.1** finishes delivery of all functional items required for certification. This includes delivery of JMS, JAVAMail, and client container support. In addition, WebSphere 4.0.1 provides the initial delivery of Web Services.

Table 2 summarizes the characteristics for the releases of WebSphere Application Server for OS/390 and WebSphere Application Server V4.0.1 for z/OS and OS/390.

Table 2. Summary of SE V3.02, SE V3.5, and V4.0 J2EE server characteristics, for migration purposes

Characteristic	SE V3.02	SE V3.5	V4.0
Minimum system requirements:			
Operating System	• z/OS V1R1, or • OS/390 V2R7 or higher	• z/OS V1R1, or • OS/390 V2R8 or higher	• z/OS V1R1, or • OS/390 V2R8 or higher

Table 2. Summary of SE V3.02, SE V3.5, and V4.0 J2EE server characteristics, for migration purposes (continued)

Characteristic	SE V3.02	SE V3.5	V4.0
System Configuration	OS/390 HTTP Server	OS/390 HTTP Server	• OS/390 HTTP Server • Sysplex (monoplex minimum) • Workload management in goal mode • RRS • System logger • LDAP • DB2 for OS/390 V7.1
Software Development Kit (SDK)	Sun or IBM JDK 1.1.8	IBM Java 2 Standard Edition (J2SE) V1.3 for OS/390	IBM Java 2 Standard Edition (J2SE) V1.3 for OS/390
Process/ Execution Model	Provides a Go Web Server (GWAPI) Plug-in Routine. See "Process/execution model differences" on page 28 for a detailed description of differences.	Provides a Go Web Server (GWAPI) Plug-in Routine. See "Process/execution model differences" on page 28 for a detailed description of differences.	The J2EE Server contains a Web container.
WebSphere Administration Database	No database is required. Server configuration is provided in a configuration file. Server operations are performed via HTTP server facilities.	No database is required. Server configuration is provided in a configuration file. Server operations are performed via HTTP server facilities.	Administration Database is required to be resident and accessed within DB2 V7.1. An Administration application is provided for configuring and managing J2EE and system servers. HTTP servers that are configured to route Web requests to J2EE Servers are managed using existing HTTP server facilities.
Application Assembly and Deployment	The notion of a Web Application is supported. See "Application assembly and deployment differences" on page 31.	The notion of a Web Application is supported. See "Application assembly and deployment differences" on page 31.	WebSphere for z/OS accept enterprise applications in the form of an Enterprise Archive (.ear) file.
WebSphere HTTP Session State Database repository	Database must exist in DB2 for OS/390 V5 (with PTFs) or V6 (with PTFs). See "WebSphere HTTP session state database repository" on page 33.	Database must exist in DB2 for OS/390 V5 (with PTFs) or V6 (with PTFs). See "WebSphere HTTP session state database repository" on page 33.	Database must exist in DB2 for OS/390 V7.1.

Table 2. Summary of SE V3.02, SE V3.5, and V4.0 J2EE server characteristics, for migration purposes (continued)

Characteristic	SE V3.02	SE V3.5	V4.0
Security Mechanism	SAF-based, LocalOS. See "Security mechanism" on page 35.	SAF-based, LocalOS. See "Security mechanism" on page 35.	SAF-based, LocalOS.
Common Connector Framework (CCF) support	Compliant with the IBM Common Connector Framework V1.1. Minimal qualities of service and runtime integration are provided. See "Common Connector Framework support" on page 38.	Compliant with the IBM Common Connector Framework V1.1. Minimal qualities of service and runtime integration are provided. See "Common Connector Framework support" on page 38.	Compliant with the IBM Common Connector Framework V1.1. Minimal qualities of service and runtime integration are provided.
Access to CICS	CICS Transaction Gateway (CTG) product (5648-B43) provides a CCF based connector that allows access to CommArea based CICS Transaction programs. See "Accessing CICS" on page 40.	CICS Transaction Gateway (CTG) product V4.0 provides a CCF based connector that allows access to CommArea based CICS Transaction programs. See "Accessing CICS" on page 40.	CICS Transaction Gateway (CTG) product V4.0 provides a CCF based connector that allows access to CommArea based CICS Transaction programs.
Access to IMS	IMS Connect (5655-E51) provides a CCF based connector that allows access to IMS Transaction Programs. See "Accessing IMS" on page 42 .	See "Accessing IMS" on page 42 for additional information.	
DB2/ESA Access via JDBC V2.0 Standard Extension DataSource APIs	Database must exist in a DB2 subsystem at either a V5 level (with PTFs) or a V6 level (with PTFs). See "Accessing DB2 for OS/390 through JDBC V2.0 Standard Extension DataSource APIs" on page 44 for more specific information.	Database must exist in a DB2 subsystem at either a V5 level (with PTFs) or a V6 level (with PTFs). See "Accessing DB2 for OS/390 through JDBC V2.0 Standard Extension DataSource APIs" on page 44 for more specific information.	DB2 V7.1.

Part 2. Migrating from WebSphere V3.02SE or V3.5SE to WebSphere V4.0.1

This part of the book is intended for those who are migrating to WebSphere V4.0.1 from either WebSphere V3.02SE or V3.5SE and contains the above chapters.

Chapter 3. Specification and functional differences between WebSphere V3.02SE, V3.5SE, and V4.0.1

The following table shows releases V3.02SE, V3.5SE, and V4.0.1 of the WebSphere for z/OS family and the specification and functional differences between them.

Table 3. Specification and functional differences between WebSphere V3.02SE, V3.5SE, and V4.0.1

Specifications	WebSphere V3.02SE	WebSphere V3.5SE	WebSphere V4.0.1
Java related	• Requires SDK 1.1.8 • Supports specification levels: – Servlet 2.1 – JSP 1.0 – JDBC 1.1	• Requires SDK 1.3 • Supports specification levels: – Servlet 2.1/2.2 – JSP .091/1.0/1.1 – JDBC 1.2	• Requires SDK 1.3 • Supports J2EE 1.2 levels: – Servlet 2.2 – JSP 1.1 – JDBC 2.0 – EJB 1.1 – JNDI 1.2 – JTA 1.0 – RMI/IIOP 1.0 – JMS 1.1 – JavaMail 1.1 – JAP 1.0 – Client container
Web Services	N/A	N/A	Introduces Web Services equivalent to those provided by: • WebSphere AE V4.0 – HTTP to Stateless Session beans • Requires SDK 1.3 PTF available 10/2001 • Supports specification levels: – SOAP 1.1 – Apache SOAP V2.2

- **WebSphere V3.02SE** provides a JAVA runtime for executing Web applications consisting of servlets/JSPs.
- **WebSphere V3.5SE** adds support for Servlet/JSP specification levels required by the J2EE 1.2 specifications, allowing customers to take advantage of the latest levels and to start early migration of applications in preparation for movement to J2EE 1.2 servers.
- **WebSphere V4.0.1** introduces a J2EE server that provides support for enterprise applications consisting of Web applications and enterprise java beans (EJBs). Support for Servlets, JSPs, and EJBs is compliant to the J2EE 1.2 architecture and provides all the benefits of this architecture. Tooling supports the J2EE 1.2 required packaging scheme. The J2EE server, unlike prior JAVA runtimes provided by WAS SE, provides container managed services for items such as transactions and security, simplifying the application programmers job. In addition the J2EE server utilizes the z/OS infrastructure to provide QOS

expected on the 390 platform. WebSphere V4.0.1 finishes delivery of all functional items required for certification. This includes delivery of JMS, JAVAMail, and client container support. In addition, WebSphere 4.0.1 provides the initial delivery of Web Services.

Chapter 4. Preparing to install the WebSphere for z/OS V4.0.1 runtime

This chapter addresses the various preparation steps needed to migrate to WebSphere Application Server V4.0.1 for z/OS and OS/390 from either WebSphere V3.02SE or V3.5SE, including:

- "Choosing mountpoints for WebSphere V4.0.1"
- "Determining system requirements for WebSphere V4.0.1" on page 18

Choosing mountpoints for WebSphere V4.0.1

WebSphere for z/OS V4.0.1 uses two different hierarchical filesystem (HFS) directories to store executable code and sample files:

1. a Java (SDK) directory, usually `/usr/lpp/java`
2. a WebSphere directory, usually `/usr/lpp/WebSphere`

These directories and their subdirectories are usually contained in HFS data sets which are separate from the z/OS or OS/390 root filesystem HFS data set. In the following instructions, it is assumed that all three HFS data sets are separate, as recommended during WebSphere Application Server product installation.

When two different versions of WebSphere Application Server are used concurrently on a z/OS or OS/390 system, it is necessary to have HFS data sets from both versions mounted. Either the old or new HFS data sets must be mounted at version-specific mountpoints during the interval that the two WebSphere Application Servers are being run side by side. For ease of migration, we recommend, when necessary, that the older WebSphere version be moved to allow for installation of the new WebSphere code at the customary mountpoints. Using the customary mountpoints for WebSphere for z/OS V4.0.1 will make migration to future WebSphere releases easier.

Creating version specific mountpoints to move WebSphere V3.02SE for coexistence with V4.0.1

Both the Java and WebSphere levels in WebSphere V3.02SE are incompatible with those on WebSphere V4.0.1. Before installing WebSphere for z/OS V4.0.1 on your target system, perform the following steps:

1. Create version specific mountpoints such as `/usr/lpp/java118` and `/usr/lpp/WebSphere302` on your target system. This can be done using either the OMVS shell or the TSO MKDIR command. Note that the HFS data set containing the `/usr/lpp/` hierarchy must be mounted read/write.
2. Shut down all Web servers using WebSphere V3.02SE.
3. Unmount the WebSphere V3.02SE HFS data sets, using the TSO UNMOUNT command:

   ```
   UNMOUNT FILESYS('old.java.hfs.data.set')
   UNMOUNT FILESYS('old.was.hfs.data.set')
   ```
4. Remount the WebSphere V3.02SE HFS data sets at the new version-specific mountpoints using the TSO MOUNT command:

   ```
   MOUNT FILESYS('old.java.hfs.data.set') TYPE(HFS) MOUNTPOINT('/usr/lpp/java118')
   MOUNT FILESYS('old.was.hfs.data.set')  TYPE(HFS) MOUNTPOINT('/usr/lpp/WebSphere302')
   ```

5. Each Web server running WebSphere V3.02SE will have a Web server configuration file, a Web server environment variables file, and a WebSphere Application Server file; the usual names for these files are `httpd.conf`, `httpd.envvars` and `was.conf`. Find the copies of these files used by each WebSphere V3.02SE enabled Web server on your system, and use an editor of your choice to change all occurrences of the old Java and WebSphere mountpoints to the new version-specific Java and WebSphere mountpoints. (Remember to save unmodified copies of these files in case you need to return to using the old Java and WebSphere mountpoints.)

6. Restart all Web servers that use WebSphere V3.02SE. Test your existing WebSphere applications to make sure they work correctly. If problems arise, review your Web configuration files.

You are now ready to install WebSphere for z/OS V4.0.1.

Creating version specific mountpoints to move WebSphere V3.5SE for coexistence with V4.0.1

The WebSphere level in WebSphere V3.5SE is incompatible with that contained in WebSphere V4.0.1; the Java levels are compatible. Therefore, only the WebSphere HFS data set needs to be moved to a version-specific mountpoint. Before installing WebSphere V4.0.1 on your target system, perform the following steps:

1. Create a version specific mountpoint such as `/usr/lpp/WebSphere35` on your target system. This can be done using either the OMVS shell or the TSO MKDIR command. Note that the HFS data set containing the `/usr/lpp/` hierarchy must be mounted read/write.

2. Shut down all Web servers using WebSphere V3.5SE.

3. Unmount the WebSphere V3.5SE HFS data sets, using the TSO UNMOUNT command:

   `UNMOUNT FILESYS('old.was.hfs.data.set')`

4. Remount the WebSphere V3.5SE HFS data sets at the new version-specific mountpoint using the TSO MOUNT command:

   `MOUNT FILESYS('old.was.hfs.data.set')  TYPE(HFS) MOUNTPOINT('/usr/lpp/WebSphere35')`

5. Each Web server running WebSphere V3.5SE will have a Web server configuration file, a Web server environment variables file, and a WebSphere Application Server file; the usual names for these files are `httpd.conf`, `httpd.envvars`, and `was.conf`. Find the copies of these files used by each WebSphere V3.5SE enabled Web server on your system, and use an editor of your choice to change all occurrences of the old WebSphere mountpoint to the new version-specific mountpoint. (Remember to save unmodified copies of these files in case you need to return to using the old WebSphere mountpoint.)

6. Restart all Web servers using WebSphere V3.5SE Application Server 3.5. Test your existing WebSphere applications to make sure they work correctly. If problems arise, review your Web configuration files.

You are now ready to install WebSphere for z/OS V4.0.1.

Determining system requirements for WebSphere V4.0.1

See the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834 for determining the system requirements for the following:

- z/OS or OS/390 hardware requirements

- z/OS or OS/390 software requirements for WebSphere V4.0.1
- Workstation requirements
- Software requirements for developing WebSphere for z/OS applications
- Requirements for J2EE components
- Recommendation for developing EJB components
- Recommendation for developing Web components
- Requirements for CORBA (MOFW) components

Chapter 5. Installing the WebSphere Application Server V4.0.1 for z/OS and OS/390 runtime

Now you are ready to install the WebSphere Application Server V4.0.1 for z/OS and OS/390 runtime on your target system, using the product program directory or ServerPac instructions, and customizing it according to *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. When customization is complete, you will be able to run servlets, JSPs and Enterprise Java Beans on your WebSphere Application Server V4.0.1 runtime system. This chapter addresses:

- "Installing the WebSphere V4.0.1 runtime"
- "Related and useful information" on page 22

and directs you to related information that will help in your install and customization of the WebSphere V4.0.1 runtime.

Installing the WebSphere V4.0.1 runtime

To migrate from either WebSphere V3.02SE or V3.5SE to WebSphere V4.0.1 involves multiple changes in support of the J2EE servers introduced by WebSphere V4.0.1. You must setup the supporting infrastructures (DB2 7.1, RRS, WLM, LDAP, etc.), set-up the system server structure (daemon, naming, system management, etc.) and utilize the new administration application to define and execute the IVP. These changes require you to do a cold start and initialize the WebSphere 4.0.1 runtime as if it was your first installation.

Note: Release changes after WebSphere for z/OS V4.0 do not require the cold start method.

Running the customization dialog: The customization dialog is intended for the system programmer or administrator responsible for installing and customizing WebSphere for z/OS. For more information see "Running the customization dialog" in the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*. The dialog covers a portion of WebSphere for z/OS customization. Specifically, it creates tailored jobs to:

- Copy the generated jobs into your system libraries
- Create the system management HFS structure and the initial environment file
- Create and customize the LDAP server
- Set up WebSphere for z/OS security controls (RACF)
- Define the WebSphere for z/OS runtime configuration (systems management server, naming server, interface repository server, daemon server)
- Run the installation verification programs (IVPs)

Installation and customization topics to review: See the section "Installing and customizing your first runtime" in the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization* which contains the steps needed to go to V4.0 from V3.02SE and V3.5SE.

Related and useful information

The following references will provide help in installing the WebSphere V4.0.1 runtime and can be found at the following Web site:

`http://www.ibm.com/software/webservers/appserv/`

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Program Directory*, GI10-0680, describes the elements of and the installation instructions for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: License Information*, LA22-7855, describes the license information for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834, describes the planning, installation, and customization tasks and guidelines for WebSphere for z/OS.

Chapter 6. Migration considerations for new and changed function in WebSphere V4.0.1

The following WebSphere V4.0.1 functions will contain the following:
- Description
- Summary of the WebSphere for z/OS tasks or interfaces that might be affected
- Coexistence considerations, if any, that are associated with the item
- Migration procedures, if any, that are associated with the item
- References to other publications that contain additional detailed information

The following new or changed function topics include:
- "Operating system and database requirements" on page 24
- "Process/execution model differences" on page 28
- "Application assembly and deployment differences" on page 31
- "WebSphere HTTP session state database repository" on page 33
- "Security mechanism" on page 35
- "Common Connector Framework support" on page 38
- "Accessing CICS" on page 40
- "Accessing IMS" on page 42
- "Accessing DB2 for OS/390 through JDBC V2.0 Standard Extension DataSource APIs" on page 44
- "JRas support" on page 46
- "J2EE services for Web applications" on page 48
- "RunAs" on page 83

Operating system and database requirements

Description

This section describes new operating system and database requirements that affect your migration.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	None
Application development	None
Auditing	None
Customization	See "Migration tasks" on page 26.
General user	None
Operations	New operational procedures for running servers. See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Operations and Administration*, SA22-7835.
Interfaces	None

Dependencies

For a complete list of WebSphere V4.0.1 requirements, see "Determining WebSphere for z/OS system requirements" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

Coexistence considerations

The following are compatibility or coexistence issues introduced by the J2EE run time:

- A Standard Edition V3.02 or V3.5 system can coexist on the same system or sysplex with WebSphere V4.0.1, provided they have different mount points (you cannot use the default mount point for both products). See "Choosing mountpoints for WebSphere V4.0.1" on page 17 for more information on choosing mount points. You may want to create a separate test system or LPAR to provide isolation for test purposes.

- DB2 for OS/390 V7.1 is required at run time. Customers may wish to stage the migration of DB2 7.1 into application environments not related to WebSphere. DB2 provides several ways to support coexistence between DB2 7.1 and earlier DB2 versions. Consider:
  - DB2 for OS/390 V7.1 can coexist with an earlier DB2 on same image with unique test data
  - DB2 for OS/390 V7.1 can do a distributed call to an earlier DB2 to access test data
  - DB2 for OS/390 V7.1 can do datasharing with an earlier DB2 to access test data. Note that only two levels of DB2 for OS/390 can be in the same datasharing group. If datasharing, you must install DB2 for OS/390 compatibility APARs.

  Recommendation: Keep data sharing between multiple releases of DB2 for OS/390 to a limited timeframe.

Figure 1 shows possible DB2 for OS/390 configurations for migration to DB2 for OS/390 V7.1.
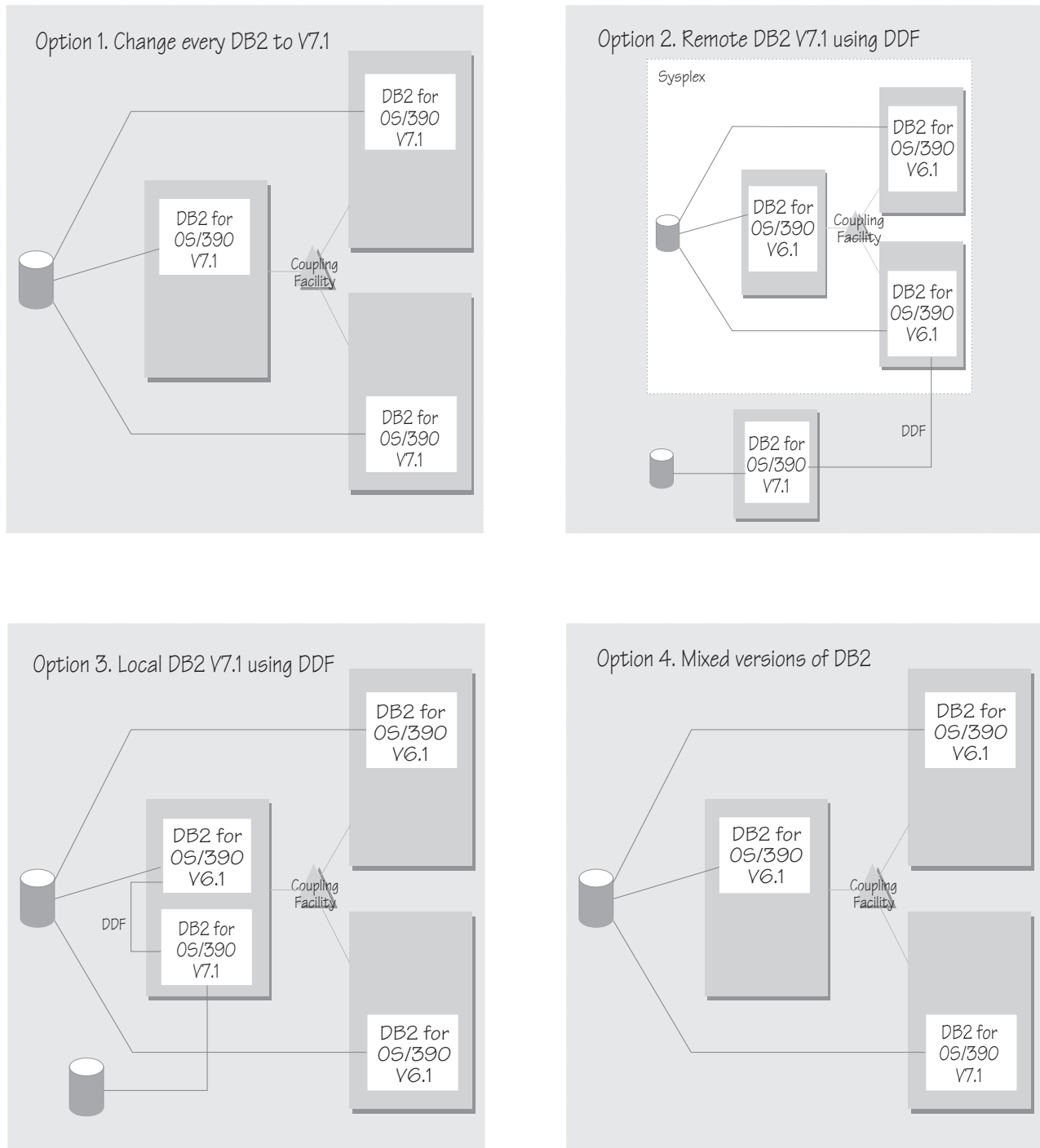
Figure 1. Possible configurations for migration to DB2 for OS/390 V7.1

- If you want to interoperate with a Standard Edition V3.5 system, you must install a compatibility PTF on the SDK for V3.5. See the latest PTF information in the PSP bucket.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Task	Condition	Reference Information
Upgrade your hardware, if necessary. There are significant performance advantages for those applications doing floating point arithmetic if the machine has binary floating point hardware, such as S/390 Parallel Enterprise Server-Generation 5 and later systems.	Highly recommended	
Upgrade your operating system. WebSphere V4.0.1 requires OS/390 V2R8 or later or z/OS. WebSphere sysplex configurations should use the shared HFS function introduced in OS/390 V2R9 to minimize complexity. New installations should consider OS/390 V2R10 as the base operating system level.	Required	*z/OS Planning for Installation*, GA22-7504
Install PTFs. See the PSP bucket for required PTFs for: • Workload management • RACF • LDAP • XML parser • SSL/security (optional) • RRS • OS/390. Review PTFs that may be required for mixed releases of the operating system running in the sysplex.	Required	Documentation accompanying the PTFs.
Migrate to DB2 for OS/390 V7.1	Required	*DB2 Release Planning Guide*, SC26-9943
Choose different mountpoints for coexistence of two different releases of WebSphere. See "Choosing mountpoints for WebSphere V4.0.1" on page 17 for more information.	Optional	
Install and customize WebSphere V4.0.1. There are operating system requirements: • Sysplex (minimum: monoplex) • Workload management in goal mode • RRS and Logger • LDAP • FTP server	Required	"Preparing the base OS/390 or z/OS environment" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834 and "Installing and customizing your first runtime" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834

Task	Condition	Reference Information
There are some prereqs for JMS and Java Mail: • Define J2EE mail session resources if you're an administrator who supports the JavaMail/JAVA Beans Activation Framework. • Run the bbomcfg script. • Declare JMS Administered objects as J2EE resource references in the deployment descriptor. • Ensure that MQSeries Version 5 Release 2 and support pack MA88 are installed	Optional	See "JAVA Mail/JAVA Beans Activation Framework" on page 78 and "Java Message Service (JMS)" on page 80.

For more information

For more detailed information about this support, refer to the following WebSphere for z/OS publications:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834
- *z/OS Planning for Installation*, GA22-7504
- *DB2 Release Planning Guide*, SC26-9943
- *z/OS MVS Planning: Workload Management*, SA22-7602
- *z/OS MVS Setting Up a Sysplex*, SA22-7625z/OS MVS Programming: Resource Recovery*, SA22-7616
- *z/OS Communications Server: IP Configuration Guide*, SC31-8775

Process/execution model differences

Description

This section compares the process/execution model for WebSphere for z/OS with the process/execution models for versions 3.02 and 3.5 of the Application Server.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	• New ServerInit, Service, and ServerTerm directives must be added to the HTTP server's httpd.conf file. • Web containers need to be configured within a J2EE server.
Application development	Application installation process has changed.
Auditing	None
Customization	None
General user	None
Operations	Method for defining a JVM has changed.
Interfaces	None

Dependencies

For a complete list of Standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes the differences in the SE V3.02, SE V3.5 and V4.0.1 process/execution model.

Table 4. Process/execution model comparison

SE V3.02	SE V3.5	V4.0.1
Uses a Go Web Server (GWAPI) plug-in routine to initialize a JDK 1.1.8 Virtual Machine within the HTTP server address space.	Uses a Go Web Server (GWAPI) plug-in routine to initialize an SDK 1.3 Virtual Machine within the HTTP server address space.	Uses a Go Web Server (GWAPI) plug-in routine to initialize an SDK 1.3 Virtual Machine within the HTTP server address space.

Table 4. Process/execution model comparison (continued)

SE V3.02	SE V3.5	V4.0.1
Web Components are executed in an Application Server instance that executes within the address space of an IBM HTTP Server. The Application Server is initialized via a GWAPI plug-in routine that is provided with the V3.02 product. The Application Server gets its configuration from a was.conf configuration file that is provided as input to the plug-in.	Same as for V3.02.	Web Applications execute inside of a Web Container that resides in the Server region of a J2EE Server. Web Containers are configured within J2EE Servers using properties contained in the webcontainer.conf. One or more HTTP servers containing the WebSphere V4.0.1 Plug-In Routine is required to be configured within the same Sysplex as the J2EE Servers containing the installed Web applications. The HTTP server handles traffic from Web clients. **Note:** Web applications previously run in a V3.5 Application Server can temporarily be run in the V4.0.1 plug-in. For more information on the V4.0.1 plug-in, see "Migrating from V3.5 SE" on page 105.
The GWAPI plug-in routine is configured to the HTTP server address space by adding ServerInit, Service, and ServerTerm directives within the HTTP server's httpd.conf file.	Same as for V3.02.	Same as for V3.02
Only one GWAPI plug-in routine can be configured to a single Web server address space. To run multiple levels of the plug-in routine on a system simultaneously, the plug-in routines must be configured within separate HTTP server address spaces.	Same as for V3.02	Same as for V3.02
Does not support Web containers.	Does not support Web containers.	Web containers are configured within J2EE servers using properties contained in a webcontainer.conf configuration file that is provided to the J2EE server as an environment variable. **Note:** The V4.0.1 Systems Management facilities is used to install Web applications into the Web container. The V4.0.1 Systems Management facilities is used to install Web applications into the Web container.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Application assembly and deployment differences

Description

This section compares the application assembly and deployment process for WebSphere for z/OS with the application assembly and deployment process for versions 3.02 and 3.5 of the Application Server.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	Application deployment process has changed.
Application development	Application assembly and deployment process has changed.
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

For a complete list of Standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes the differences in Standard Editions V3.02, SE V3.5 and WebSphere V4.0.1 application assembly and deployment process

Table 5. Application assembly and deployment comparison

SE V3.02	SE V3.5	V4.0.1
Physical properties, such as the document root in the HFS where HTML and JSPs are stored, the classpath for locating servlet, and Java bean implementations, as well as the application's address (rootURI specification within its host), used during application deployment are specified in the **deployedwebapp** properties in the was.conf configuration file.	Same as for V3.02.	Enterprise applications (which can include Web applications) must exist as an Enterprise Archive (.ear) file in order to be deployed into a Web container. The .ear file is provided as input to the System Management Application provided with V4.0.1. The Administration application is able to do full deployment of the application, including resource resolution and installation of the physical files. The application level deployment descriptor for each .war file within the application is assigned a "context root". Context roots are equivalent to the root.URI specification that is provided on deployedwebapp was.conf file properties in previous versions of the Application Server.

Table 5. Application assembly and deployment comparison (continued)

SE V3.02	SE V3.5	V4.0.1
Web application definition properties, such as servlet definitions, and initialization properties, are either specified using **webapp** properties in the was.conf file, or in a separate webapp.xml file that exists within the Application Server classpath. (The separate document structure allows the developer to supply application assumptions to the administrator in a formalized manner.)	Same as for V3.02.	Web application definition properties, such as servlet definitions, and initialization properties, are contained in a separate web.xml file located in the WEB-INF directory.
Virtual host definitions and binding to the deployed Web applications that are to be served through them are specified in **host** properties in the was.conf file.	Same as for V3.02 with one addition; it also provides utilities that create the necessary deployment information (.webapp files, or deployedwebapp properties for the was.conf file) from an industry standard Web Application Archive (.war file). This allows applications that have been developed and packaged in a .war file format, using application development tools, to be deployed within the Application Server. The full set of functions contained in the deployment descriptors for Web applications are not supported.	Web applications are able to be imported and assembled into an Enterprise application using the Application Assembly tool that is provided with the product. Once an application is installed into a J2EE Server, it can be exposed through a Virtual Host that is defined within the webcontainer.conf file.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

WebSphere HTTP session state database repository

Description

This section compares the WebSphere HTTP Session State database repositories used in Standard Editions V3.02, V3.5 and WebSphere V4.0.1.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	DB2 is set up as part of the J2EE server installation process instead of as part of the WebSphere for OS/390 and z/OS installation process.
Application development	None
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

For a complete list of Standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes the differences in how DB2 for OS/390 databases for storing session data are set up in Standard Editions V3.02, V3.5 and WebSphere V4.0.1 environments.

Table 6. Setup differences for WebSphere HTTP Session State database repositories

SE V3.02	SE V3.5	V4.0.1
If you are using persistent HTTP Session State, a DB2 for OS/390 database, at a V5.0 or V6.0 level, must be defined as described in *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835.	If using persistent HTTP Session State, a DB2 for OS/390 database, at a V5.0, V6.0 or V7.0 level, must be defined as described in *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835.	If using persistent HTTP Session State a DB2, a DB2 for OS/390 database, at a V7.1 level, must be defined as described in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.
A Session State database at the DB2 V5 or V6 level can be concurrently shared between V3.02 and V3.5.	A Session State database at the DB2 V5 or V6 level can be concurrently shared between V3.02 and V3.5; a Session State database at the DB2 V7.1 level can be concurrently shared between V3.5 and V4.0.1.	A Session State database at the DB2 V7.1 level can be concurrently shared between V3.5 and V4.0.1.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806

- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Security mechanism

Description

This section compares the security mechanism in Standard Editions V3.02, V3.5, and WebSphere V4.0.1.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	Changes need to be made to the HTTP server's httpd.conf file
Application development	Security authentication parameters need to be specified within the Web application's deployment descriptors
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

For a complete list of Standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes the differences in how security is handled in an SE V3.02, SE V3.5 and V4.0.1 environment.

Table 7. Security mechanism comparison

SE V3.02	SE V3.5	V4.0.1
SAF-based, LocalOS.	Same as for V3.02.	Same as for V3.02.
User Registry: Users are defined in operating system SAF repository.	**User Registry:** Same as for V3.02.	**User Registry:** Same as for V3.02.
Challenge Mechanism for Authentication: HTTP Server protection directives can be set up within the httpd.conf file to make use of the HTTP Basic Authentication function, which require users to provide userid and password for authentication. Client Certificate is provided over HTTPS SSL Connection. The Client Certificate must resolve to a userid within the SAF User Registry.	**Challenge Mechanism for Authentication:** Same as for V3.02.	**Challenge Mechanism for Authentication:** Same as for V3.02 with the following additions: • Userid and Password may be obtained via Form-based Login as prescribed by the Servlet V2.2 Specification. • The challenge mechanism for components of a Web Application may be configured using either information in the .webapp file that is part of the deployed Web application, or using HTTP server protection directives within the httpd.conf file.

Table 7. Security mechanism comparison (continued)

SE V3.02	SE V3.5	V4.0.1
URL Access Checks: URL access checking can be performed using the authenticated identity. These checks can be configured using HTTP Server protection directives within the httpd.conf file.	**URL Access Checks:** Same as for V3.02.	**URL Access Checks:** Same as for V3.02.
Operating System Execution Identity: The system identity in which the request will execute is determined by the protection directives within the HTTP server. The identity resulting from the HTTP server authentication is then used as the Operating System execution identity.	**Operating System Execution Identity:** Same as for V3.02.	**Operating System Execution Identity:** All requests inside of the V4.0.1 Web container execute with a system identity equal to that of the application server region.
J2EE Execution Identity: N/A	**J2EE Execution Identity:** N/A	**J2EE Execution Identity:** Maintains information about the requestor for use by Web components at runtime. APIs on the input request object allow servlets to retrieve information about the subject of the request, such as information from an X509 certificate or userid. J2EE Services, such as JDBC and Java 2 Connectors, can obtain the proper information about a requestor at runtime for use in its service level security checking.
Access Control Checks: SAF Checks are performed against resources in the SOMDOBJS facility class. Properties in the was.conf. configuration file specify which resources to check.	**WebSphere Access Control Checks:** Same as for V3.02.	**WebSphere Access Control Checks:** Access Control checks to Web Components based on requestor access to roles associated with the WebComponent. SAF checks are performed against profiles in the EJBROLE class.
Single Sign-On Capability: No support provided.	**Single Sign-On Capability:** No support provided.	**Single Sign-On Capability:** Single Sign-on to a Web Application is supported as described in the Servlet Specification V2.2.
Recommendations and Usage: Authentication must be performed by the HTTP server, using parameters specified in either HTTP server protection directives and/or was.conf file properties.	**Recommendations and Usage:** Same as for V3.02	**Recommendations and Usage:** Use the deployment descriptors packaged with Web applications as the basis for authentication and authority checking. Security processing configured within the HTTP server is performed prior to entering the J2EE server environment. Existing protection directives can initially be left in the HTTP server's httpd.conf file. However, you should eventually consider removing them to prevent redundant authentication processing.

For more information

For more detailed information about this support, see:

- "Setting up security" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834
- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Common Connector Framework support

Description

This section describes Common Connector Framework support.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	Should transition to the use of Sun Microsystems Corporation's J2EE Connector Architecture as it becomes available. See the following link for more information: `http://www.ibm.com/software/webservers/appserv/download_v4z.html`
Application development	None
Auditing	None
Customization	See "Migration tasks".
General user	None
Operations	None
Interfaces	See "Migration tasks".

Dependencies

See "Migration tasks".

Coexistence considerations

See "Migration tasks".

Migration tasks

Table 8. Common Connector Framework comparison

SE V3.02	SE V3.5	V4.0.1
Minimal qualities of service and runtime integration are provided using IBM Common Connector Framework (CCF) V1.1. CCF support is not enabled to be user transaction aware. Connectors are configured to the runtime via installing their implementation files within the Application Server classpath. Client programs gain access to connectors via static access to the CCF Connection Factory that is provided by the runtime.	Same as for V3.02.	CCF Connector support is provided for Web components at the same level as it was provided for versions 3.02 and 3.5. This support is intended as a migration aid for existing Standard Edition customers. It is recommended that customers transition to the use of Javasoft compliant Sun Microsystems Corporation's J2EE Connector Architecture as it becomes available.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Accessing CICS

Description

This section compares how to access CICS in Standard Editions V3.02, V3.5 and WebSphere V4.0.1.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	None
Application development	See Table 9.
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

For a complete list of Standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes how to access CICS in an SE V3.02, SE V3.5 and V4.0.1 environment.

Table 9. Accessing CICS comparison

SE V3.02	SE V3.5	V4.0.1
CICS Transaction Gateway (CTG) product (5648-B43) Version 3.1.x provides a CCF based connector that allows access to CommArea based CICS Transaction programs. This connector is not transaction aware. It uses the System Identity (ACEE) on the execution thread for access control checking (see "Security mechanism" on page 35).	Same as for V4.0.1.	CICS Transaction Gateway (CTG) product V4.0 provides a CCF based connector that allows access to CommArea based CICS Transaction programs. This connector is not transaction aware when used within the J2EE Server runtime. It uses the System Identity (ACEE) on the execution thread for access control checking (see "Security mechanism" on page 35).

Table 9. Accessing CICS comparison (continued)

SE V3.02	SE V3.5	V4.0.1
Recommendations and Usage: Client authentication and access control checking should be applied to the Web component that is being accessed. Requests should be executed with a system identity equal to that of the HTTP server. Therefore, HTTP protection directives should be configured to allow requests to execute as %%SERVER, thereby enabling system resources, such as existing CICS, IMS, DB2, and files, to only allow access control from Application Server instances. With the most current service level installed, the HTTP server no longer needs to be configured with a Unix System Services ID of UID=0. Therefore, the HTTP Server can be configured with a UID that only provides user level access rights.	Same as for V3.02.	**Recommendations and Usage:** Client authentication and access control checking should apply to the Web component being accessed via the HTTP client, enabling system resources, such as existing CICS, IMS, DB2, and files, to only allow access control from WebSphere for OS/390 and z/OS instances. (All Web components within the WebSphere V4.0.1 J2EE runtime execute with a system identity equal to that of the J2EE Server.) J2EE servers can be configured with minimal access rights and privileges.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
- `http://www.ibm.com/software/webservers/appserv/`
- The IBM Redbook "Revealed! Architecting Web Access to CICS", located at:

  `http://www.redbooks.ibm.com/abstracts/sg245466.html`

Accessing IMS

Description

This section compares how to access IMS in Standard Editions V3.02, V3.5 and WebSphere V4.0.1.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	None
Application development	See "Coexistence considerations".
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

For a complete list of Standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes how to access IMS in Standard Editions V3.02, and V3.5, and a WebSphere for z/OS V4.0.1 environment.

Table 10. Accessing IMS comparison

SE V3.02	SE V3.5	V4.0.1
IMS Connect (5655-E51) provides a CCF based connector that allows access to IMS Transaction Programs. This connector is not user transaction aware. It makes use of the System Identity (ACEE) on the execution thread for performing access control checks against the IMS transaction (see "Security mechanism" on page 35).	Same as for V3.02.	This connector makes use of the System Identity (ACEE) on the execution thread for performing access control checks against the IMS transaction. All requests within the V4.0.1 Web container execute with a System Identity equal to that of the server (see "Security mechanism" on page 35).

Table 10. Accessing IMS comparison (continued)

SE V3.02	SE V3.5	V4.0.1
Recommendations and Usage: Client authentication and access control checking should be applied to the Web component that is being accessed. Requests should be executed with a system identity equal to that of the HTTP server. Therefore, HTTP protection directives should be configured to allow requests to execute as %%SERVER, thereby enabling system resources, such as existing CICS, IMS, DB2, and files, to only allow access control from Application Server instances. With the most current service level installed, the HTTP server no longer needs to be configured with a Unix System Services ID of UID=0. Therefore, the HTTP Server can be configured with a UID that only provides user level access rights.	Same as for V3.02.	**Recommendations and Usage:** Client authentication and access control checking should apply to the Web component being accessed via the HTTP client, enabling system resources, such as existing CICS, IMS, DB2, and files, to only allow access control from WebSphere for OS/390 and z/OS instances. (All Web components within the WebSphere V4.0.1 J2EE runtime execute with a system identity equal to that of the J2EE Server.) J2EE servers can be configured with minimal access rights and privileges.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Accessing DB2 for OS/390 through JDBC V2.0 Standard Extension DataSource APIs

Description

This section compares how to access DB2 for OS/390 through JDBC V2.0 Standard Extension DataSource APIs in standard Editions V3.02, and V3.5 and WebSphere V4.0.1.

What this change affects

This support might affect the following areas of WebSphere for z/OS processing.

Area	Considerations
Administration	DB2 tables may need to be modified.
Application development	Web applications needing to use JDBC must include a deployment descriptor indicating that JDBC is an external resource that needs to be accessed. The com.ibm.ejs.ns.jndi.CNInitialContext Factory class can not be used for V4.0.1 Web applications.
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

For a complete list of standard Editions V3.02, V3.5, and WebSphere 4.0.1 J2EE server characteristics, for migration purposes, see Table 2 on page 10.

Coexistence considerations

The following table summarizes how to access DB2/ESA via JDBC V2.0 Standard Extension DataSource APIs in an SE V3.02, SE V3.5 and V4.0.1 environment.

Table 11. Accessing DB2 for OS/390 through JDBC comparison

SE V3.02	SE V3.5	V4.0.1
Connection pools can be configured using connection pool properties in the was.conf file. Pool configuration settings includes min and max connections, idle connection timeout, and the name of the database driver.	Same as for V3.02.	WebSphere Application Server V4.0.1 for z/OS and OS/390 requires datasources to be configured using the Systems Administration Utility. Datasources for connection pools are configured using the Systems Administration Utility. Web applications needing to use JDBC must include a deployment descriptor indicating that JDBC is an external resource that needs to be accessed.

Table 11. Accessing DB2 for OS/390 through JDBC comparison (continued)

SE V3.02	SE V3.5	V4.0.1
A JNDI name can be specified for a datasource object. This name can be used to obtain and release JDBC connections. The com.ibm.ejs.ns.jndi. CNInitialContextFactory class provides an initial context factory for gaining access to the JNDI name space. The datasource implementation returned from the namespace contains implementation for the following methods: • getConnection (userid,Password) • getConnection()	Same as for V3.02	As part of application deployment, the System Management Application resolves references and establishes name spaces which enable Web components using J2EE programming techniques to locate such datasources at runtime. The datasource implementation returned from the name space contains implementation for the following methods: • getConnection (userid,Password) • getConnection()
When getConnection is performed with no input parameters, the resultant JDBC handle is established with a primary authorization identity equal to the identity under which the current thread is executing. JDBC Connections can not be used in conjunction with user transactions.	Same as for V3.02	Same as for V3.02
Recommendations and Usage: Client authentication and access control checking should be applied to the Web component that is being accessed. Requests should be executed with a system identity equal to that of the HTTP server. Therefore, HTTP protection directives should be configured to allow requests to execute as %%SERVER, thereby enabling system resources, such as existing CICS, IMS, DB2, and files, to only allow access control from Application Server instances. With the most current service level installed, the HTTP server no longer needs to be configured with a Unix System Services ID of UID=0. Therefore, the HTTP Server can be configured with a UID that only provides user level access rights.	Same as for V3.02.	**Recommendations and Usage:** Client authentication and access control checking should apply to the Web component being accessed via the HTTP client, enabling system resources, such as existing CICS, IMS, DB2, and files, to only allow access control from WebSphere for OS/390 and z/OS instances. (All Web components within the WebSphere V4.0.1 J2EE runtime execute with a system identity equal to that of the J2EE Server.) J2EE servers can be configured with minimal access rights and privileges.

For more information

For more detailed information about this support, see:

- *WebSphere Application Server for OS/390 V3.02 Standard Edition Planning, Installing, and Using*, GC34-4806
- *WebSphere Application Server for OS/390 V3.5 Standard Edition Planning, Installing, and Using*, GC34-4835
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

JRas support

Description

The JRas support has been changed as follows:
- New interfaces allow Java applications to obtain message or trace loggers.
- A customer-supplied trace settings file, instead of runtime environment variables, now enables or disables the collection of trace data.
- Message collection is always enabled.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	To enable the collection of trace data for Java applications: • Provide a trace settings file, and • Modify the application server runtime environment variables to point to that settings file.
Application development	For new Java applications, use the new JRas interfaces for obtaining message and trace loggers. Although the previous interfaces are deprecated, you do not have to change any of the Java applications that currently use them. For additional details, see the topic about logging messages and trace data for Java applications in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.
Auditing	None
Customization	None
General user	None
Operations	Messages or trace data for Java applications might appear in either the error log, the CTRACE data set, or both. Also, because message collection is always enabled, this support might increase message traffic on the master console.
Interfaces	None

Dependencies

There are no additional hardware, software, or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 12. Migration tasks

Task	Condition	Procedure reference
Recode Java applications to use the new JRas interfaces.	Optional	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
Prepare the runtime environment for logging Java application messages and trace requests, which includes: • Creating a trace settings properties file • Updating the JVM properties file for the application server Required if existing applications use JRas support.	Required	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
Update the environment variables for the application server, to remove the obsolete JRas variables.	Optional	

For more information

For more detailed information about this support, refer to the following publications:

• *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

• *WebSphere Application Server V4.0.1 for z/OS and OS/390: Messages and Diagnosis*, GA22-7837

• The JRas topic in the Information Center on your WebSphere workstation

J2EE services for Web applications

Description

J2EE services for Web applications consists of the following:

- **Java Message Service (JMS):** which provides a framework for developing and supporting Java software components that communicate by creating, sending, and receiving messages. This method of communication, known as messaging, allows components to interact asynchronously and reliably, without knowing more about their communication partners than message formats and destinations. For more information on JMS, see "Java Message Service (JMS)" on page 80.
- **JAVA Mail/JAVA Beans Activation Framework:** which provides a framework for developing and supporting Java applications that send, store, and receive mail. According to the Sun Microsystems J2EE specification, a JavaMail configuration consists of the following:
  - The JavaMail API implementation, which provides general facilities for reading and sending E-mail.
  - The JavaBeans Activation Framework (JAF), another Java API that handles mail in forms that are more elaborate than plain text (in other words, MIMEs, URL pages, file attachments, and so on).
  - Service providers, which implement protocols for mail transport and storage. In other words, these service providers allow applications to send mail through mail servers and to access stored mail.

    The WebSphere for z/OS JavaMail package supports the use of the JavaMail API by all types of application components: Servlets, JavaServer Pages (JSPs), Enterprise JavaBeans, and application clients. This package contains:
    - The JavaMail API implementation
    - The JAF API
    - Two service providers: An SMTP service provider and an IMAP service provide

    For more information on JMS, see "JAVA Mail/JAVA Beans Activation Framework" on page 78 and "Java Message Service (JMS)" on page 80.

For more information

For more detailed information about this support, refer to the following publications:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

RunAs

Description

EJB RunAs is the mechanism defined by J2EE for determining the identity under which an EJB method will run. This support does not directly implement any of the J2EE specification levels, but is a composite of these levels, along with enhanced function. The composite part is that there are three RunAs settings defined, caller, server, and role. The enhanced function is that, RunAs is set at the method level, rather than the J2EE suggested bean level, and there is an option to force the operating system identity to match that of the RunAs setting. The RunAs settings are controlled using the AAT tool, and the synchronization of the operating system identity is controlled using a combination of the SMGUI, and the AAT tool. For more information on RunAs, see "RunAs" on page 83.

For more information

For more detailed information about this support, refer to the following publications:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Chapter 7. Migrating existing WebSphere V3.02SE and V3.5SE applications to WebSphere V4.0.1

Once you have migrated the operating system and subsystems to required levels and the WebSphere run time, you must migrate your applications from Standard Editions V3.02 or V3.5.

The following sections contain information about migrating applications to the WebSphere for z/OS J2EE server environment. Because WebSphere for z/OS V4.0.1 requires compliance with the latest J2EE programming and packaging specifications, and requires the use of specific development, assembly, and installation tools, consider reading the following introductory material before using the instructions to migrate application components:

- "Overview of application tools" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for a brief introduction to tools and processes for developing, assembling, and installing J2EE applications in the WebSphere for z/OS J2EE server.

The following sections contain information on:

1. "Background on migration" for a brief introduction to migration concepts, tasks, and recommendations.
2. "Migration scenarios for applications running on WebSphere Application Server for z/OS or OS/390 Standard Edition" on page 53

Background on migration

Generally speaking, migration encompasses actions that you complete to ensure that existing applications continue to function correctly on a new version or release of an IBM product.

Table 13 is a checklist of migration actions that might be either required or optional, depending on the type of application you want to migrate and its current runtime environment. The checklist also indicates the role of the most appropriate person to complete each action or set of actions, based on the Sun Microsystems J2EE specification. Many actions are the same as steps for developing, assembling, deploying, and installing a new application, as described in "Creating, assembling and deploying J2EE server applications" found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836. Familiarity with those procedures will help you complete the migration process more efficiently.

Table 13. Checklist of roles and potential migration actions

✔	Potential migration actions:	Condition / Comments
	For system programmers or administrators:	
✔	Understand how product, system, database, or configuration changes might affect your applications.	**Required** for any migration path. See "Operating system and database requirements" on page 24 for more information.

Table 13. Checklist of roles and potential migration actions (continued)

✔	Potential migration actions:	Condition / Comments
✔	Preconfigure J2EE resources, such as DB2	**Optional**, depending on requirements of application to be installed (required for connection pooling)
For application component providers:		
✔	Understand how tooling changes might affect your applications	**Required** because WebSphere for z/OS requires specific tools. See "Application assembly and deployment differences" on page 31 for more information.
✔	Understand how interface changes might affect your applications	**Required** because WebSphere for z/OS requires compliance with particular levels of programming specifications. Servlets and JSPs must be updated to comply with the Servlet 2.2, JSP 1.1, and SDK 1.3 specifications. Applications using CTG 3.x must change to use CTG 4.0.
✔	Change component design or rewrite source code because of interface changes	**Required**, depending on the type of application component and its compliance with required levels of programming specifications
✔	Regenerate code to pick up interface enhancements, or enhancements to or maintenance for development tools	**Optional**, modify to use JRAS function. Modify to use JMS/JAVAMail capability provided in WAS 4.0.1
For application assemblers or deployers:		
✔	Repackage applications and regenerate metadata because of enhancements to or maintenance for application assembly and deployment tools	**Required** for all applications to be installed in a WebSphere for z/OS J2EE server.
For application installers:		
✔	Re-install any reassembled applications to replace existing or add new EAR files.	**Required** for all applications to run in a WebSphere for z/OS J2EE server
✔	Modify conversation elements because of enhancements to or maintenance for the WebSphere for z/OS Administration application	**Required** to define and activate a WebSphere for z/OS J2EE server and the J2EE resources associated with it, and to install J2EE applications

Table 13. Checklist of roles and potential migration actions (continued)

✔	Potential migration actions:	Condition / Comments

✔	Modify z/OS or OS/390 constructs because of product, system, or configuration changes. Such changes include moving or reconstructing application databases, changing WLM service goals, and so on.	**Required** for:
		• Applications that require role-based security
		  – **Required:** Changes to z/OS Security Server definitions
		  – **Optional:** Redeploy of application deployment with rolename matching z/OS Security Server definitions
		• Web applications that formerly ran in the WebSphere for z/OS Standard Edition environment

Migration scenarios for applications running on WebSphere Application Server for z/OS or OS/390 Standard Edition

Depending on the version of the Standard Edition product you are currently using, you might have to upgrade elements of your z/OS or OS/390 system as well as perform migration actions for your Web applications.

For additional information about the Standard Edition products and WebSphere for z/OS, see:

- "Migrating from V3.5 SE" on page 105 for information about migrating from Standard Edition V3.02 or V3.5, or if you want to run existing Web applications in a Standard Edition environment at the same time you run new Web applications in a WebSphere for z/OS J2EE server.
- *WebSphere Application Server for OS/390 Application Server Planning, Installing and Using, Version 3.5*, GC34–4835, to determine how to:
  - Install and configure Standard Edition V3.5.
  - Migrate Web applications from previous versions of Standard Edition to the WebSphere for z/OS Plug-in.

Chapter 8. Removing WebSphere V3.02SE or WebSphere V3.5SE

When all applications have been successfully migrated to WebSphereV4.0.1, you can delete your WebSphere Application Server Version 3.02SE or V3.5SE data sets, and remove any version-specific mountpoints you created.

Part 3. Migrating from WebSphere V4.0 to WebSphere V4.0.1

This part of the book is intended for those who are migrating to WebSphere V4.0.1 from WebSphere V4.0 and contains the above chapters:

Chapter 9. Differences between WebSphere for z/OS V4.0 and WebSphere for z/OS V4.0.1

Following are the differences between WebSphere V4.0 and WebSphere V4.0.1:

Table 14. Differences between WebSphere V4.0 and WebSphere V4.0.1

WebSphere V4.0	WebSphere V4.0.1

Introduces J2EE server capable of running enterprise applications consisting of Web and enterprise java bean components:
• Requires SDK 1.3
• Supports J2EE 1.2 levels:
  – Servlet 2.2
  – JSP1.1
  – JDBC 2.0
  – EJB 1.1
  – JNDI 1.2
  – JTA 1.0
  – RMI/IIOP 1.0
	Completes J2EE certified server capable of running enterprise applications consisting of Web and enterprise java bean (EJB) components:
	• Requires SDK 1.3 and PTF PQ52841.
	• Supports J2EE 1.2 levels:
	  – Servlet 2.2
	  – JSP1.1
	  – JDBC 2.0
	  – EJB 1.1
	  – JNDI 1.2
	  – JTA 1.0
	  – RMI/IIOP 1.0
	  – JMS 1.1
	  – JavaMail 1.1
	  – JAF 1.0
	  – Client container

	Introduces Web Services equivalent to those provided by:
	• WebSphere AE V4.0
	  – HTTP to Stateless Session beans
	• Supports specification levels:
	  – SOAP 1.1
	  – Apache SOAP V2.2

- **WebSphere V4.0** introduced a J2EE server that provided support for enterprise applications consisting of Web applications and enterprise java beans (EJBs). Support for Servlets, JSPs, and EJBs is compliant to the J2EE 1.2 architecture and provides all the benefits of this architecture. Tooling supports the J2EE 1.2 required packaging scheme. The J2EE server, unlike prior JAVA runtimes provided by WAS SE, provides container managed services for items such as transactions and security, simplifying the application programmers job. In addition the J2EE server utilizes the z/OS infrastructure to provide Quality of Service (QOS) expected on the 390 platform.

- **WebSphere V4.0.1** finishes delivery of all functional items required for certification. This includes delivery of JMS, JAVAMail, and client container support. In addition, WebSphere 4.0.1 provides the initial delivery of Web services.

Chapter 10. Overall migration tasks to go from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1

This chapter contains the following overall migration task sections:

- "Gathering the appropriate information"
- "Migrating your runtime from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1" on page 62b
- "Exploiting new function in V4.0.1" on page 63
- "Upgrading applications that are already installed in a WebSphere for z/OS J2EE server" on page 63

Gathering the appropriate information

Gather the supporting installation documentation for WebSphere V4.0.1. Following are a list of books and other information that are in the WebSphere for z/OS library. They can be found at the following Web site:

`http://www.ibm.com/software/webservers/appserv/`

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Program Directory*, GI10-0680, describes the elements of and the installation instructions for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: License Information*, LA22-7855, describes the license information for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834, describes the planning, installation, and customization tasks and guidelines for WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Messages and Diagnosis*, GA22-7837, provides diagnosis information and describes messages and codes associated with WebSphere for z/OS.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Operations and Administration*, SA22-7835, describes system operations and administration tasks.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836, describes how to develop, assemble, and install J2EE applications in a WebSphere for z/OS J2EE server. It also includes information about migrating applications from previous releases of WebSphere Application Server for OS/390, or from other WebSphere family platforms.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling CORBA Applications*, SA22-7848, describes how to develop, assemble, and deploy CORBA applications in a WebSphere for z/OS (MOFW) server.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: System Management User Interface*, SA22-7838, describes the system administration and operations tasks as provided in the Systems Management User Interface.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: System Management Scripting API*, SA22-7839, describes the functionality of the WebSphere for z/OS Systems Management Scripting API product.

Migrating your runtime from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1

Following are some basic steps in preparing to upgrade from WebSphere for z/OS V4.0 to WebSphere for z/OS V4.0.1:

- **Creating the proper HFS structure for upgrades:** You can install new functional levels of WebSphere for z/OS without disrupting service to your clients provided you have the proper HFS structure in a sysplex and you use what we call a rolling upgrade. See "Overview of creating the proper HFS structure for upgrades" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834 for more information.

- **Backing up your WebSphere for z/OS V4.0 system:** If you want to back up all persistent data for your WebSphere for z/OS system, you must back up:
  - The system management database
  - The LDAP database tables containing the naming space and the interface repository
  - Files in the HFS WebSphere for z/OS runtime information
  - WebSphere for z/OS PROCLIBs
  - WebSphere for z/OS LOADLIBs

  For more backup information, see "Guidelines for backup of the WebSphere for z/OS system" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

- **Applying the minimum service levels for your WebSphere for z/OS V4.0 system in order to migrate to WebSphere for z/OS V4.0.1:** There are two paths for migrating from V4.0 to V4.0.1 which have the following prereqs:
  1. If you are on a monoplex, or you are on a sysplex and can afford an outage, you need the following WebSphere 4.0 APARS/PTFs:
     - APAR PQ48857 (PTF UQ90027)
     - APAR PQ49276 (PTF UQ55643)
  2. If you are on a sysplex and want to do a rolling warmstart, you need the following WebSphere 4.0 APARs/PTFs:
     - APAR PQ48857 (PTF UQ90027)
     - APAR PQ49276 (PTF UQ55643)
     - APAR PQ53552
     - IBM Developer Kit for OS/390, Java2 Technology Edition indicated: APAR PQ52841 (PTF UQ99325)

  Note: The above service information is current with this edition. See the latest PTF information in the PSP bucket.

- **Using the customization dialog or samples in customizing WebSphere for z/OS V4.0.1:** There are various configuration choices you may have to make based on whether you used the customization dialog in configuring V4.0. Following are some decision paths that may help you in your installation and configuration to V4.0.1.

Table 15. Customization dialog decisions.

If you are upgrading from V4.0 to V4.0.1 and . . .	Then follow. . .	Notes...
used the customization dialog in V4.0	the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834	The customization dialog allows you to load the V4.0 variables you used.
did not use the customization dialog but want the advantages the customization dialog brings	the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834	You may need to modify the jobs created by the customization dialog according to your specific environment. For example, you may have updated a steplib that did not appear in the V4.0 samples.
did not use the customization dialog and prefer to use the sample jobs instead	See the section called "Migrating from WebSphere for z/OS V4.0 to V4.0.1 without the customization dialog" in the appendix of *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.	

Exploiting new function in V4.0.1

No application changes are required if the warmstart process is utilized when migrating to WebSphere for z/OS V4.0.1 from WebSphere for z/OS V4.0. Existing application and server definitions will be preserved across the migration installation, and existing applications will run as is. However, some changes are required, in the run time and in the applications, to exploit new function introduced in WebSphere for z/OS V4.0.1. These changes are documented in the line item information provided in the following sections. For each function that you plan to exploit, carefully review the line item information for:

- Additional prerequisite products
- Additional configuration tasks
- Interface changes

VisualAge for Java V4.0 and Webstudio V4.0 are recommended for new application development. Also, some functions introduced in WebSphere for z/OS V4.0.1 require the use of the AAT level available on the WebSphere for z/OS download Web-site at General Availability, located at:

```
http://www.ibm.com/software/webservers/appserv/
```

Upgrading applications that are already installed in a WebSphere for z/OS J2EE server

The following notes apply to applications that you have previously installed in a WebSphere for z/OS J2EE server:

Notes:

1. You may install multiple versions of the same application in the same J2EE server, as long as each application version is uniquely named, separately packaged and installed through the WebSphere for z/OS Application Assembly tool. If each version of the application requires a different database schema, you may define a separate J2EE resource connection for each application.

2. If you make changes to an already installed application, you may:
   a. Create a new conversation in the SMEUI tool
   b. Replace the existing application EAR file
   c. Validate, commit, and activate the conversation with the SMEUI tool
   d. Issue a z/OS or OS/390 workload manager (VARY REFRESH) command to drain current work and start new server regions, for example:

   ```
   v wlm,applenv=(servername,refresh)
   ```

   Use this process as a replacement for the Standard Edition servlet reloading function. The servlet reloading function that existed in previous versions of the Application Server is no longer supported. WLM commands are now used to refresh servlets without causing an interruption of service.

Chapter 11. New function in WebSphere for z/OS V4.0.1

This chapter discusses new functions delivered in WebSphere V4.0.1.

The new WebSphere V4.0.1 functions will contain the following:
- Description
- Summary of the WebSphere for z/OS tasks or interfaces that might be affected
- Coexistence considerations, if any, that are associated with the item
- Migration procedures, if any, that are associated with the item
- References to other publications that contain additional detailed information

WebSphere V4.0.1 Release Summary

The following list summarizes the WebSphere updates that were introduced with WebSphere Application Server V4.0.1 for z/OS and OS/390. For more information, refer to the detailed section for each item.

Table 16. Summary of WebSphere updates for WebSphere Application Server V4.0.1 for z/OS and OS/390.

For Information About:	Refer to :
Accessibility	66
Buffer limits	67
Client container	68
Customization panels	71
Distributed exceptions	74
JAVA Mail/JAVA Beans Activation Framework	78
Java Message Service (JMS)	80
Java Naming and Directory Interface (JNDI) caching	82
RunAs	83
SMF recording: Support of EJB container	85
Warm start	87
Web: Application Assembly tool (AAT) WebSphere Application and XDD support	89
Web Security	91
Web services for V4.0.1 (SOAP)	93

Accessibility

Description

Successful access to information and use of information technology by people who have disabilities is known as "accessibility".

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	The changes will be effective after the reinstall of the SMEUI.
Application development	None
Auditing	None
Customization	None
General user	Short cut changes were made (for example; save is now on Ctrl+s and no longer on F3).
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 17. Accessibility migration tasks

Task	Condition	Procedure reference
Reinstall the System Management User Interface (SMEUI) by doing a warmstart to WebSphere V4.0.1.	Required	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834

For more information

For more detailed information about accessibility, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834
- See also, the online help in the SMEUI.

Buffer limits

Description

The Server has been enhanced to reject requests greater than 10MB (local and remote). The 10MB limit is a GIOP message size limit, so the effective limit for application parameters will be slightly less than 10MB. Precisely how much less is dependent on the size of service contexts which are also passed with the request.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	The 10M GIOP message limit should be taken into account when developing applications that flow large quantities of data. If your application flows data in pieces below this limit, this new functionality will not affect you. Check your applications and make appropriate changes if necessary.
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 18. Buffer limits migration tasks

Task	Condition	Procedure reference
When developing applications that flow large quantities of data the 10M GIOP message limit should be taken into account. Check your applications and make appropriate changes if necessary. If your application flows data in pieces below this limit, this new functionality will not affect you.	Optional	See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for more information on buffer limits.

For more information

For more detailed information about buffer limits, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Client container

Description

The client container is a Windows-based implementation of the J2EE Application Client Container, which provides an execution environment for Java application client programs. Applications are launched, within their own JVM, from an ear file containing a fully-deployed application client jar. The application client program then executes within the container, using resources and services provided by the container. JNDI services allow the application client program to lookup EJB services executing in the WebSphere for z/OS EJB container, and other J2EE application client container API's such as JDBC and RMI provide access to additional local and distributed services.

Application clients are first tier client programs that execute in their own Java virtual machines. Application clients follow the model for Java technology-based applications - they are invoked at their main method and run until the virtual machine is terminated. Like other J2EE application components, application clients depend on a container to provide system services.

An application client container includes several services such as security, transaction, naming, application programming interfaces, and packaging and deployment. A minimal implementation must include security and packaging and deployment, and the supported J2EE APIs must include JDBC, RMI-IIOP, JNDI, and client-side EJB.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	None
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

The WebSphere for z/OS Application Client Container, as installed on Windows NT/2000, provides J2EE interoperability with the WebSphere for z/OS Application Server running on z/OS. Installation of this product will replace the default JVM for the workstation - the user is notified during the installation.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to

situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 19. Client container migration tasks

Task	Condition	Procedure reference
Install the Application Client container on a Windows NT or 2000 platform: 1. Use FTP binary to download the self-extracting zip file containing the installation from the HFS where WebSphere for z/OS is installed to the Windows workstation where you want to install the Application Client container. The file on the z/OS HFS is named `/usr/lpp/WebSphere/bin/J2EEClient_NT.zip` 2. Run the self-extracting zip file on your workstation to extract all of the files to a new work directory. 3. From the work directory, run the extracted setup.exe program. The defaults will generally be acceptable, and will result in the creation of a new directory named `C:\WebSphere\AppClient`, which is the installation root directory. All of the tools are supplied as batch files, and are located in the bin subdirectory under the installation root directory. 4. You will need to create a text file named jndi.properties, and place it in the \properties subdirectory within the installation root directory (e.g., `c:\WebSphere\AppClient\properties\jndi.properties`). This file overrides the property in `com/ibm/websphere/naming/jndiprovider.properties` (loaded from ns.jar in the app client lib) and should contain the following line: com.ibm.websphere.naming.namespaceroot=bootstraphostroot	Required	See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for details.
Deploy a J2EE Application containing an Application Client jar file. This requires the use of AAT for z/OS with Application Client Support (packaged separately), as well as the Application Client Resource Configuration Tool (packaged with the Application Client Container). 1. Use AAT with Application Client support to build an EAR file with an Application Client jar file. 2. Take the EAR file and run it through Application Client Resource Configuration Tool (`WebSphere\AppClient\bin\clientConfig.bat`) to define/add client resources.	Optional	See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for details.

Table 19. Client container migration tasks (continued)

Task	Condition	Procedure reference

Run an Application Client. This requires the use of the Client Container Launcher (packaged with the Application Client Container). This tool can be invoked using the following command line:
 	
`c:> C:\WebSphere\AppClient\bin\launchClient earFileName args`
 	
where *earFileName* is the fully qualified name of the deployed ear file from the preceding task and *args* are any additional Java arguments, e.g. special client container and/or application arguments, etc.. If the tool is invoked without any arguments, it will display a screen of help text, including optional client container arguments.
 	
If you need to modify the address of the WebSphere for z/OS Application Server to which your application client container connects, you can modify the following property in the setupCmdLine.bat file (found in the \bin subdirectory of the installation root directory):
 	
`set COMPUTERNAME=yourWebSphereAppServerOrbHost`
 	
where *yourWebSphereAppServerOrbHost* is a valid ip host address.
	Optional	See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for details.

For more information

For more detailed information about client container, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
- See the readme.html file included in the root installation directory of the Application Client Container. This page includes links to several useful online resources concerning the Application Client Container.

Customization panels

Description

The customization dialog is intended for the system programmer or administrator responsible for installing and customizing WebSphere for z/OS. The customization dialog was added in WebSphere for z/OS through a Small Programming Enhancement (SPE) PTF, and has been extended in WebSphere for z/OS Version 4.0.1 to provide support for migration from previous releases of WebSphere for z/OS Version 4. The dialog covers a portion of WebSphere for z/OS customization. Specifically, it creates tailored jobs to:

- Copy the generated jobs into your system libraries.
- Create the system management HFS structure and the initial environment file.
- Create and customize the LDAP server
- Set up WebSphere for z/OS security controls (RACF)
- Define the WebSphere for z/OS runtime configuration (systems management server, naming server, interface repository server, daemon server)
- Run the installation verification programs (IVPs)

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	• If you used the customization dialog in configuring V4.0 then go to the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. **Note:** The customization dialog allows you to load the V4.0 variables you used.
	• If you did not use the customization dialog but want the advantages the customization dialog brings, see the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. **Note:** You may need to modify the jobs created by the customization dialog according to your specific environment. For example, you may have updated a STEPLIB that did not appear in the V4.0 samples.
	• If you did not use the customization dialog and prefer to use the sample jobs instead, then see the section called "Migrating from WebSphere for z/OS V4.0 to V4.0.1 without the customization dialog" in the appendix of *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.
Application development	None
Auditing	None

Area	Considerations
Customization	• If you used the customization dialog in configuring V4.0 then go to the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. **Note:** The customization dialog allows you to load the V4.0 variables you used. • If you did not use the customization dialog but want the advantages the customization dialog brings the sections the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. **Note:** You may need to modify the jobs created by the customization dialog according to your specific environment. For example, you may have updated a STEPLIB that did not appear in the V4.0 samples. • If you did not use the customization dialog and prefer to use the sample jobs instead then see the section called "Migrating from WebSphere for z/OS V4.0 to V4.0.1 without the customization dialog" in the appendix of *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.
General user	None
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 20. Customization panels migration tasks

Task	Condition	Procedure reference
Run the customization dialog if you installed/configured your V4.0 system with the customization dialog.	Recommended	See the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.
Run the customization dialog if you did not use the customization dialog but want the advantages the customization dialog brings.	Recommended	See the sections named "Warm start" and "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.
Run the sample jobs included in V4.0.1 if you did not use the customization dialog and prefer to use the sample jobs instead.	Optional	See the section called "Migrating from WebSphere for z/OS V4.0 to V4.0.1 without the customization dialog" in the appendix of *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834

For more information

For more detailed information about this support, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834

Distributed exceptions

Description

Distributed Exceptions are enhanced Java exceptions which allow for exception chaining which eases debugging. Distributed Exceptions are part of the extended WebSphere programming model, which goes beyond the J2EE programming model. As such, they can be used by WebSphere applications. DistributedException classes are integrated into the WebSphere for z/OS runtime but not used by the runtime.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	Distributed Exceptions have been shipped with WebSphere AE 3.5. On WebSphere for z/OS these classes ship with Version 4.0.1. Applications can use these classes now, cross-familiy, without running into a portability problem when deploying code using these classes on WebSphere for z/OS.
Auditing	None
Customization	None
General user	Application programmers are now able to use Distributed Exceptions, cross-familiy, as part of the extended WebSphere Programming Model .
Operations	None

Area	Considerations

Interfaces	The following are the descriptions of the new interfaces , classes, exceptions are copied from the AE Documentation, located at:
	
	`http://www.ibm.com/software/webservers/appserv/doc/v40/ae/apidocs/index.html`
	
	• **Interface DistributedExceptionEnabled**
	
	`com.ibm.websphere.exception`
	`Interface DistributedExceptionEnabled`
	
	All Known Implementing Classes:
	
	` DistributedEJBCreateException,`
	` DistributedEJBRemoveException,`
	` DistributedException`
	
	` public interface DistributedExceptionEnabled`
	
	Enables an exception to be treated as a distributed exception. This interface should be used by an exception that is not a subclass of DistributedException.
	
	In addition to implementing the required methods, the implementing class should create an attribute, a DistributedExceptionInfo object, in each constructor, after it has done it's other work. This object will do most of the work for the methods. For all of the examples in the Javadoc of the methods, it is assumed that the name of this attribute is `distributedExceptionInfo`.
	
	Typically, the implementor of this interface will have multiple constructors. See `com.ibm.websphere.DistributedException` for examples.
	
	See Also: `com.ibm.websphere.DistributedException,`
	`com.ibm.websphere.DistributedExceptionInfo`
	
	• **Class DistributedExceptionInfo**
	
	`com.ibm.websphere.exception  Class DistributedExceptionInfo  java.lang.Object`
	
	`|`
	`+--com.ibm.websphere.exception.DistributedExceptionInfo`
	
	`public class DistributedExceptionInfo`
	`extends java.lang.Object`
	`implements java.io.Serializable`
	
	Does the work for exception classes that implement the DistributedExceptionEnabled interface. See Also: Serialized Form
	
	• **Class DistributedEJBCreateException**
	
	`com.ibm.websphere.exception`
	`Class DistributedEJBCreateException`
	`java.lang.Object`
	`│    +--java.lang.Throwable`
	`│         +--java.lang.Exception`
	`│              +--javax.ejb.CreateException`
	`│                   +--com.ibm.websphere.exception.DistributedEJBCreateException`
	`│`
	
	`public class DistributedEJBCreateException`
	`extends javax.ejb.CreateException`
	`implements DistributedExceptionEnabled`
	`A subclass of javax.ejb.CreateException that provides exception functions`
	`desirable in a distributed environment. This includes the following:`
	`Support to allow exceptions to be chained, in the situation where multiple`
	`exceptions are thrown during a series of method calls.`
	`Saving stack trace information so that printStackTrace() will provide the stack trace of all`
	`chained exceptions.`
	`Methods to retrieve specific exceptions in the chain.`
	`Support for localized messages.   See Also:  Serialized Form`

Area	Considerations

- **Class DistributedEJBRemoveException**

```
Class DistributedEJBRemoveException
com.ibm.websphere.exception
Class DistributedEJBRemoveException
java.lang.Object
 |
 +--java.lang.Throwable
      |
      +--java.lang.Exception
      |
 +--javax.ejb.RemoveException
|
             +--com.ibm.websphere.exception.DistributedEJBRemoveException
public class DistributedEJBRemoveException
extends javax.ejb.RemoveException
implements DistributedExceptionEnabled
```

A subclass of javax.ejb.RemoveException that provides exception functions desirable in a distributed environment. This includes the following:

Support to allow exceptions to be chained, in the situation where multiple exceptions are thrown during a series of method calls.

Saving stack trace information so that printStackTrace() will provide the stack trace of all chained exceptions.

Methods to retrieve specific exceptions in the chain.

Support for localized messages.
See Also: Serialized Form

- **Class ExceptionInstantiationException**

```
 ExceptionInstantiationException

com.ibm.websphere.exception
Class ExceptionInstantiationException

java.lang.Object
 |
 +--java.lang.Throwable
      |
      +--java.lang.Exception
          |
          +--com.ibm.websphere.exception.DistributedException
             |
             +--com.ibm.websphere.exception.ExceptionInstantiationException

public class ExceptionInstantiationException
extends DistributedException
```

Exception - ExceptionInstantiationException This indicates that an exception was thrown when trying to instantiate a previous exception in a chain of exceptions. The specific exception can by retrieved using getPreviousException().   See Also: Serialized Form

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 21. Distributed exceptions migration tasks

Task	Condition	Procedure reference
Use Distributed Exceptions, cross-familiy, as part of the extended WebSphere Programming Model.	Optional	See the AE Documentation, located at: `http://www.ibm.com/software/webservers/appserv /doc/v40/ae/apidocs/index.html`

For more information

For more detailed information about distributed exceptions, refer to the following:

- See the AE Documentation, located at:

  `http://www.ibm.com/software/webservers/appserv/doc/v40/ae/apidocs/index.html`

JAVA Mail/JAVA Beans Activation Framework

Description

JavaMail provides a framework for developing and supporting Java applications that send, store, and receive mail. According to the Sun Microsystems J2EE specification, a JavaMail configuration consists of the following:

- The JavaMail API implementation, which provides general facilities for reading and sending E-mail.
- The JavaBeans Activation Framework (JAF), another Java API that handles mail in forms that are more elaborate than plain text (in other words, MIMEs, URL pages, file attachments, and so on).
- Service providers, which implement protocols for mail transport and storage. In other words, these service providers allow applications to send mail through mail servers and to access stored mail.

The WebSphere for z/OS JavaMail package supports the use of the JavaMail API by all types of application components: Servlets, JavaServer Pages (JSPs), Enterprise JavaBeans, and application clients. This package contains:

– The JavaMail API implementation
– The JAF API
– Two service providers: An SMTP service provider and an IMAP service provide

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	Administrators who support this function would potentially need to define J2EE mail session resources. See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for details.
Application development	Developers who write applications using mail services should reference the Sun Specifications for JavaMail. Deployers of applications using JavaMail should review data in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.
Auditing	None
Customization	See "Using the JavaMail API in J2EE application components" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for details.
General user	None
Operations	None
Interfaces	The Java API's supported are as defined in the Sun specification for JavaMail and JAF.

Dependencies

To have a functional mail system on z/OS or OS/390, your installation also needs to have the appropriate mail servers and mail stores installed.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 22. JAVA Mail/JAVA Beans Activation Framework migration tasks

Task	Condition	Procedure reference
Define J2EE mail session resources if you're an administrator who supports the JAVA Mail/JAVA Beans Activation Framework.	Optional	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

For more information

For more detailed information about JavaMail/JAVA Beans Activation, refer to the following:

- Sun specifications for javamail and JAF out on the javasoft site, located at:

  `http://java.sun.com`

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Java Message Service (JMS)

Description

The Java Message Service (JMS) provides a framework for developing and supporting Java software components that communicate by creating, sending, and receiving messages. This method of communication, known as messaging, allows components to interact asynchronously and reliably, without knowing more about their communication partners than message formats and destinations.

The Java Message Service is a Java API for asynchronous messaging.

IBM has made an implementation of the Java Message Service (JMS) over MQSeries Version 5 Release 2 available as support pack MA88. The OS/390 version of JMS over MQ should behave like the workstation counterpart except it can exploit RRS for local and global transactions. This is a requirement for transactional operation within WebSphere for z/OS.

In addition to adding transactional support, the WebSphere for z/OS SMEUI and Application Assembly Tool have been enhanced to add support for the configuration and deployment of JMS administered objects and their declaration as resource references in J2EE enterprise archives.

Note: MQSeries does not currently provide a "message broker" on OS/390 or z/OS that can support the Publish/Subscribe domain. If applications would like to use the Pub/Sub messaging paradigm, an off-platform message broker such as MQSeries Integrator (or the MQSeries Support Pack MA0C) will need to be installed and configured. The JMS sample can be found in the WebSphere for z/OS samples directory. The default location for this file is:

```
/usr/lpp/WebSphere/samples
```

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	The only install requirement for JMS support is that bbomcfg needs to be run. The bbomcfg script will move the resource-ref templates into the appropriate places for the installation so that the EUI can find them. For more information on bbomcfg, see *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.
Application development	JMS Administered objects should be declared as J2EE resource references in the deployment descriptor. Support was added to the AAT and SMEUI to help with this. Utilize the latest AAT level that is available at: `http://www.ibm.com/software/webservers/appserv/download_v4z.html` See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for more information.
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	MQSeries Version 5 Release 2 available as support pack MA88 is a functional prereq.

Dependencies

MQSeries Version 5 Release 2 available as support pack MA88 is a functional prereq.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 23. Java Message Service (JMS) migration tasks

Task	Condition	Procedure reference
Run the bbomcfg script. The bbomcfg script will move the resource-ref templates into the appropriate places for the installation so that the EUI can find them. For more information on bbomcfg, see .	Required	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834
Declare JMS Administered objects as J2EE resource references in the deployment descriptor. Support was added to the AAT and SMEUI to help with this.	Optional	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
Ensure that MQSeries Version 5 Release 2 and support pack MA88 are installed. You also need APAR number (PQ52271).	Required	Are a functional prereq.

For more information

For more detailed information about this support, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
- MQSeries Family SupportPacs located at:

  `http://www.ibm.com/software/ts/mqseries/txppacs/txpm2.html`
- *MQSeries Using Java (SC34-5456)*

Java Naming and Directory Interface (JNDI) caching

Description

The Java Naming and Directory Interface (JNDI) supports caching global name space lookups. This capability already exists in WebSphere AE 3.5/4.0 and introduces proprietary features in the EJB programming model that must be matched by 390. Additionally, this capability is a much needed performance boost for EJB home lookups.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	JNDI caching is turned on by default in WebSphere for z/OS version 4.0.1. If caching is not a valid option for namespace lookups, you must turn caching off for those lookups.
Auditing	None
Customization	See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for details.
General user	None
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 24. Java Naming and Directory Interface (JNDI) migration tasks

Task	Condition	Procedure reference
You must turn caching off for lookups if caching is not a valid option for namespace lookups. JNDI caching is turned on by default in WebSphere for z/OS version 4.0.1.	Required	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

For more information

For more detailed information about JNDI, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

RunAs

Description

EJB RunAs is the mechanism defined by J2EE for determining the identity under which an EJB method will run. This support does not directly implement any of the J2EE specification levels, but is a composite of these levels, along with enhanced function. The composite part is that there are three RunAs settings defined, caller, server, and role. The enhanced function is that, RunAs is set at the method level, rather than the J2EE suggested bean level, and there is an option to force the operating system identity to match that of the RunAs setting. The RunAs settings are controlled using the latest AAT tool, and the synchronization of the operating system identity is controlled using a combination of the SMGUI, and the AAT tool.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	None
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	See the RunAs section found in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Dependencies

This support requires the installation of APAR PQ53621.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 25. RunAs migration tasks

Task	Condition	Procedure reference
None for migration	N/A	None

For more information

For more detailed information about this support, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

SMF recording: Support of EJB container

Description

The System Management Facilities (SMF) collects and records system and job related information on the OS/390 system. This information can be used to: bill users, report reliability, analyze the configuration, scheduling of work, profiling system resource use, and many others.

SMF Recording will produce additional SMF record subtypes (subtypes 5 and 6). They contain data which describe the specifics of EJB containers and the activities therein.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	To enable SMF recording, you must define the J2EE server to create SMF records, and perform other administration tasks; for further details, start with the SMF topic in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Operations and Administration*, SA22-7835.
Application development	There is no impact on WebSphere Application Server applications. There is an impact on SMF browsers, There are 2 new record subtypes, and 4 modified records (with new Version numbers).
Auditing	None
Customization	Definitions in SMEUI: Conversation->Sysplex->J2EE Server: New J2EE Server properties (Write Server Activity SMF Records, Write Container Activity SMF Records, Write Server Interval SMF Records, Write Container Interval SMF Records, SMF Interval Length)
General user	None
Operations	None
Interfaces	Definitions in SMEUI: Conversation->Sysplex->J2EE Server: New J2EE Server properties (Write Server Activity SMF Records, Write Container Activity SMF Records, Write Server Interval SMF Records, Write Container Interval SMF Records, SMF Interval Length)

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are 2 new record subtypes, and 4 modified records (with new Version numbers). The back level version of browsers will not be able to evaluate the new and modified records. The browsers will have to be migrated (either use the new browser shipped via the WebSphere Application Server Web site, located at:

`http://www.ibm.com/software/webservers/appserv/`

or modify the custom built browsers).

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 26. SMF recording: Support of EJB container migration tasks

Task	Condition	Procedure reference
Migrate to the new browser version.	Optional	Shipped via the WebSphere Application Server Web site, located at: `http://www.ibm.com/software/webservers/appserv/`

For more information

For more detailed information about this support, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Operations and Administration*, SA22-7835 which describes the new SMF Record types.
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Messages and Diagnosis*, GA22-7837

Warm start

Description

Warm start is a method to move from one functional level of WebSphere for z/OS to another (V4.0 to V4.0.1) that requires changes to persistent data (for example, the system management database). If performed in a sysplex with the proper HFS structure, the method does not disrupt WebSphere for z/OS service to clients.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	To be non-disruptive for your running applications, the warm start method requires that you have WebSphere for z/OS set up in a sysplex as explained in "Enabling WebSphere for z/OS on a sysplex" in the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. In particular, you need to have a shared HFS with two mount points, each holding a level of WebSphere for z/OS. The method allows you to bring down one clustered host instance at a time, then switch the HFS for that instance to the new level. Because other clustered host instances are operating when one is down, clients still get service from WebSphere for z/OS.
Application development	None
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	Capability and function levels are properties for servers and server instances visible on the Systems Management Operation Application EUI. The Operations Application flags servers ready for warm start with a green bullet. Warm start readiness is also indicated by the console message BBOU0579I for each server. Warm starts for server instances are initiated from the console via the additional start command parameter parms='-ORBCBI WARM'. Warm starts for application servers and server instances can also be initiated via the Operations application.

Dependencies

In order to have a non-disruptive rolling warmstart in a sysplex, you need to have a shared HFS with two mount points, each holding a level of WebSphere for z/OS. The method allows you to bring down one clustered host instance at a time, then switch the HFS for that instance to the new level. Because other clustered host instances are operating when one is down, clients still get service from WebSphere for z/OS.

Coexistence considerations

For WAS server side functional levels, the related and matching Administration and Operation application EUI functional levels have to be installed as described in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 27. Warm start migration tasks

Task	Condition	Procedure reference

• Backup your current system.
• If performing a rolling warm start, create the proper HFS structures
• Install the required service level.
  – If you are on a sysplex and want to do a rolling warmstart, you need the following WebSphere 4.0 APARs/PTFs:
    - APAR PQ48857 (PTF UQ90027)
    - APAR PQ49276 (PTF UQ55643)
    - APAR PQ53552
    - IBM Developer Kit for OS/390, Java 2 Technology Edition indicated: APAR PQ52841 (PTF UQ99325)
 	
Note: The above service information is current with this edition. See the latest PTF information in the PSP bucket. PSP Buckets are identified by UPGRADEs, which specify product levels, and SUBSETs, which specify the FMIDs for a product level. The **UPGRADE** and **SUBSET** values for WebSphere Application Server V4.0.1 for z/OS and OS/390 are:
  – **Upgrade:** WASAS401, **Subset:**WAS401, for the Application Server
  – **Upgrade:** JAVAOS390, **Subset:**HJVA130, for the IBM Developer Kit for OS/390 Java 2 Technology Edition
	Optional	See "Steps for performing a warm start from WebSphere for z/OS V4.0 to V4.0.1" in the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

Use the customization dialog to install V4.0.1.	Recommended	See "Running the customization dialog" in the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834.

For more information

For more detailed information about this support, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834

Web: Application Assembly tool (AAT) WebSphere Application and XDD support

Description

To assemble an application for installation on WebSphere for z/OS, you must use the WebSphere for z/OS Application Assembly tool, which:

- Generates code for z/OS or OS/390, including remote interfaces, home interfaces, ties and stubs, keys, handles, finder helpers, and code related to persistence.
- Converts the deployment descriptors for V1.0 Enterprise beans to match the V1.1 specification level. This capability enables WebSphere for z/OS to support V1.0 beans.

The AAT tool is compatible between V4.0 and V4.0.1 releases. XDD support is a new function for the new version, exploiting new runtime extensions via a GUI providing modifications to the runtime extensions.

The new AAT level is backward compatible. It can import EAR files that were exported by the previous version of AAT.

Download the latest copy from the WebSphere Application Server Web site (go to `http://www.ibm.com/software/webservers/appserv/` and click `Download` on the left frame).

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	None
Auditing	None
Customization	Use the new AAT. Directions on how to take advantage of these new features are described in the HELP within the AAT tool.
General user	If new function is to be utilized, then you would take advantage of these new functions during Assembly process of the new AAT. See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for more information.
Operations	See *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 and AAT's built-in HELP for more information.
Interfaces	Runtime interfaces do not change. Only new interfaces are in the new version of the AAT tool, which provide a WebSphere Application's Deployment Descriptor modification via GUI and provides eXtendedDeploymentDescriptor modification via GUI.

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

The previous version of AAT used the default values for the runtime extensions. There are no changes required for existing applications as they will be running with the default settings. In order to take advantage of the new XDD features, you

would have to redeploy only those applications that require new XDD features to be set to something other than the default after making changes to the XDD via the new version of AAT.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 28. Web: Application Assembly tool (AAT) WebSphere Application and XDD support migration tasks

Task	Condition	Procedure reference
Redeploy only those Applications that require new XDD features to be set to something other than the default. Do this after making changes to the XDD via the new version of AAT in order to take advantage of the new XDD features.	Optional	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

For more information

For more detailed information about this support, refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836
- AAT's built-in HELP

Web Security

Description

Similar to EJBs, access to Web Applications is controlled by setting up Roles. Application Assemblers are able to specify what Roles are allowed to access a particular URL by specifying security-constraints in the deployment descriptor of the Web application. (For complete information on specifying security constraints in Web applications see the JAVA Servlet Specification V2.2.)

The Application Assembly Tool is used to set up a correlation between the Roles specified in the deployment descriptor and the Roles recognized by RACF prior to installing the application in a Web container.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	Administrators who are responsible for installing Web applications into a Web container will have to reinstall any V4.0 Web applications for which security needs to be provided. See Chapters 4 and 11 in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for additional information.
Application development	Developers who write applications may want to add security role deployment descriptors, as defined in the JAVA Servlet Specification V2.2, to their applications.
Auditing	None
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 29. Web security

Task	Condition	Procedure reference
Re-install Web applications for which you want to include Web security.	Optional	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

For more information

For more detailed information about Web security, refer to the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

Web services for V4.0.1 (SOAP)

Description

WebSphere for z/OS uses SOAP (Simple Object Access Protocol) as its framework for supporting the creation and deployment of Web services implemented using Stateless Session beans.

For more information on "Using Web services" see *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.

What this change affects

This support might affect the following areas of processing:

Area	Considerations
Administration	None
Application development	See "Deploying an Enterprise application as a SOAP-accessible Web service" , "Creating a SOAP client", and "Using XML-SOAP for Remote Procedure Calls" in the *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.
Auditing	WebSphere for z/OS includes three options for providing security for SOAP services when using HTTP as the transport: 1. HTTP basic authentication 2. SSL (HTTPS) connections 3. SOAP signature See "Securing SOAP Services" in *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836 for more information.
Customization	None
General user	None
Operations	None
Interfaces	None

Dependencies

There are no software or functional dependencies associated with this support.

Coexistence considerations

There are no coexistence considerations associated with this support.

Migration tasks

Review the following high-level migration tasks to better understand the impacts to your environment. **Required** tasks apply to all installations enabling the function. **Optional** tasks apply to only specified operating environments or to situations where there is more than one way to set up or enable the function. For more details on the procedures associated with a task, see the reference listed.

Table 30. Web services for V4.0.1 (SOAP) migration tasks

Task	Condition	Procedure reference
There are no migration tasks in order to use this new function. See the reference for more information on "Deploying an Enterprise application as a SOAP-accessible Web service" , "Creating a SOAP client", "Using XML-SOAP for Remote Procedure Calls", and "Securing SOAP Services".	Required	*WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

For more information

For more detailed information about Web services (SOAP/UDDI support), refer to the following:

- *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836

- There are several SOAP clients available for most programming languages. The Web Services Toolkit, available at the following alphaWorks Web site includes a SOAP client:

    http://www.alphaworks.ibm.com/

Part 4. Migrating applications from WebSphere AE to WebSphere for z/OS V4.0.1

This part of the book is intended for those who are migrating to WebSphere V4.0.1 from WebSphere AE and contains the above sections.

Chapter 12. Migrating applications from WebSphere AE to WebSphere for z/OS V4.0.1

This part of the book is intended for those who may desire to move applications, that are currently running on WebSphere Application Server Advanced Edition (WebSphere AE), over to WebSphere for z/OS V4.0.1. You may have a desire to develop and test an application on the distributed platform in preparation for moving it into production on z/OS. You might also wish to move a production application running on distributed over to z/OS in order to take advantage of z/OS qualities of service. In either situation, you will need to understand any changes that would need to be made to move the application. For new applications slated to eventually move to z/OS, it would be helpful to understand how to develop the application in such a way as to make the movement to z/OS as simple as possible. The high level tasks associated with porting an application from AE to V4.0.1 can be viewed as:

1. Upgrade application development software on your workstation, if necessary.
2. Import your application component into an appropriate development tool. Then:
   - Rewrite code to eliminate functions that are deprecated because of new SDK levels or component specification levels.
   - For entity beans, review requirements for container-managed or bean-managed persistence.
   - If your Web components use J2EE resources, such as DB2, review the code and make any appropriate changes.
3. Regenerate component code and create a Web archive (WAR) file for the Web application components, or JAR files for Enterprise beans. For more information on creating a WAR file, go to The Java Servlet Specification, V2.2 located at:

   `http://www.javasoft.com`
4. Export the resulting WAR or JAR files from the application development tool.
5. Use the WebSphere for z/OS Application Assembly tool to import the component files and package the components into an Enterprise archive (EAR) file.
6. Use the WebSphere for z/OS Administration application to install the EAR file.
7. Review JNDI lookup names for your datasources and the Enterprise bean homes used in your components, and make sure the lookup names match the names under which the required components are registered on z/OS. If you are moving EJB 1.0 from AE 3.5, you must change your application to use java::comp

Some of the items that should be considered before porting your application are:

- Understanding database differences between WebSphere AE and WebSphere for z/OS and OS/390
- Understanding differences in specification levels and programming interfaces between WebSphere AE and WebSphere for z/OS and OS/390
- Managing connectivity to IMS and CICS applications between WebSphere AE and WebSphere for z/OS and OS/390
- Moving WebSphere AE platform applications with the Application and Assembly Tool (AAT)

The following sections will start to highlight some changes that may be required in moving applications from WebSphere AE to WebSphere for z/OS. It will not cover the infrastructure required to install and operate WebSphere for z/OS. For this

information, refer to *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834. It should be also noted that there is no support in WebSphere Application Server 4.0.1 family to move configuration data between different platforms. Servers must be redefined on the platform that applications are being moved to, using the Administration Application that runs there. This part is organized into the following sections:

Understanding database differences between WebSphere AE and WebSphere for z/OS and OS/390

WebSphere for z/OS and OS/390 utilizes the DB2/JDBC capability provided by DB2 7.1. If new applications are being developed on a distributed platform with 390 as the production target, you should consider using DB2 Connect to minimize the changes that are required. Existing applications being moved from the distributed platform will be utilizing a different resource manager (Oracle, DB2 UDB, etc.).

Moving the application will involve understanding the differences between DB2 7.1 and the resource manager being used by the application running on a distributed platform.

Test experience moving DB2 Universal Database (UDB) applications to DB2 390

Test experience to date has involved moving applications utilizing DB2 Universal Database (UDB) to the 390 platform. The following list of database differences encountered by the test team might be helpful, although it is not intended to be a complete list.

- DB2 390 is limited to 18 characters for table/column names.
- Primary key lengths on DB2 7.1 for S/390 are restricted to less then 256 columns.
- Although DB2 390 does not support 8 byte integers, the JDBC 390 driver will automatically convert 8 byte integer requests to Decimal (19,0), allowing the porting of JDBC application code without change. This does require a change in the associated CREATE TABLE statement to use Decimal(19,0).
- DDL must be modified for use on DB2 390. DB2 390 requires the addition of a **CREATE UNIQUE INDEX** on the DB2 table specifying the primary key. In addition, most customers will choose not to use defaults related to storage groups and VSAM datasets, and will need to make VCAT changes, etc. See the appropriate DB2 documentation for more information.
- LOB support on DB2 requires the addition of a rowid column.
- There are differences in the SQL features provided by DB2 UDB and DB2 390, but most applications can port between the two products without difficulty.

For a high level comparison of DB2 function on 390 and distributed platforms, see "DB2 Universal Databases for iSeries", located at:

`http://www.as400.ibm.com/developer/db2/db2common.html`

Understanding differences in specification levels and programming interfaces between WebSphere AE and WebSphere for z/OS and OS/390

The following sections describe the various differences in specification levels and programming interfaces between WebSphere AE and WebSphere for z/OS and OS/390:

- WebSphere differences within the J2EE 1.2 specification level
- WebSphere differences outside the J2EE 1.2 specification level
- Moving applications from an AE 3.5x level to any WebSphere 4.0.1 product

WebSphere differences within the J2EE 1.2 specification level

All members of the WebSphere Application Server 4.0 family support the J2EE 1.2 specification level. Applications written to this level running on the distributed platform should move easily to the z/OS platform. However, it should be noted that WebSphere Application Server V4.0.1 for z/OS and OS/390, consistent with the J2EE 1.2 specification, does not support the spawning of threads in the application server. Spawning of threads should be avoided when developing applications. Existing applications that do this need to be changed. It should also be noted that, consistent with the J2EE 1.2 specification, WebSphere Application Server V4.0.1 for z/OS and OS/390 does not support explicit JNDI look-up. Applications must use java:comp.

WebSphere differences outside the J2EE 1.2 specification level

There are several differences between the distributed platform and WebSphere Application Server V4.0.1 for z/OS and OS/390 outside of the realm of the J2EE specifications that should be noted:

- WebSphere AE 4.0 uses a proprietary authentication protocol, LightWeight Third Party Authentication (LTPA) for security interoperability. WebSphere Application Server V4.0.1 for z/OS and OS/390 supports a variety of industry standard authentication protocols, including SSL Digital Certificates and Kerberos. The common secure interoperability that can be used from the AE platform to z/OS and OS/390, is the use of basic authentication over a Secured Sockets Layer (SSL) pipe. Customers moving applications from AE to z/OS and OS/390 may need to make changes in this area. For more information on this topic see:
  - WebSphere Application Server Infocenter migration documentation at:
    `http://www.ibm.com/software/webservers/appserv/doc/v40/ae/infocenter`
  - *WebSphere Application Server V4.0.1 for z/OS and OS/390: Installation and Customization*, GA22-7834
- WebSphere Application Server V4.0.1 for z/OS and OS/390 JMS support is based on MQSeries functionality on the 390. At this time, Message brokers are not supported on z/OS and OS/390, but are supported on the distributed platform. Customers interested in this functionality can use a combined AE and z/OS and OS/390 topology with the message broker on the distributed platform. A sample has been provided with WebSphere Application Server V4.0.1 for z/OS and OS/390 that can be used in setting up this topology. When message brokers are supported on z/OS and OS/390, this combined topology will no longer be required.
- WebSphere Application Server V4.0.1 for z/OS and OS/390 has a GIOP message size limit of 10 MB. Applications moving from WebSphere AE may require change.

Moving applications from an AE 3.5x level to any WebSphere 4.0.x product

Additional changes are required to move an application from an AE 3.5.x level to any WebSphere 4.0.x product (AE or z/OS and OS/390) and are contained in the list below. It may be useful to review the migration documentation provided on infocenter for the WebSphere AE 4.0 product at:

http://www.ibm.com/software/webservers/appserv/doc/v40/ae/infocenter

- If starting development of a new application that is targeted to move to z/OS and OS/390, you can minimize the changes required if you're using WebSphere AE 4.0 as the development platform and you're using tooling that supports the J2EE specification levels/packaging.
- The J2EE 1.2 specification requires Servlets/JSP be developed to the Servlet 2.2/JSP 1.1 specification level. Changes may be required in existing applications. The differences in specification levels are documented at:

  http://java.sun.com

- The J2EE 1.2 specification also requires that Web applications be packaged as a WAR file. Since some levels of WebSphere AE 3.5.x did not support the use of a WAR file, and none of the levels mandated its use, it is likely that most Web applications will need to be repackaged.
- The J2EE 1.2 specification requires servers to support Enterprise JavaBeans (EJBs) written to an EJB 1.1 specification level, while WebSphere AE 3.5 supports EJB 1.0 specification levels. For migration purposes, the WebSphere 4.0 family will support deployment of EJB 1.0 beans into the server, automatically adding the necessary metadata. It should be noted that the JNDI direct reference lookup that was supported in AE 3.5, is not supported by the J2EE 1.2 specification, making a change necessary when moving existing EJB 1.0 beans to z/OS and OS/390.
- In WebSphere AE 3.5, APIs providing transactional support was provided by a set of WebSphere provided classes. Applications that took advantage of this support must change to utilize the JTA provided functionality. See the infocenter for WebSphere AE 4.0 at:

  http://www.ibm.com/software/webservers/appserv/doc/v40/ae/infocenter

  for more details.

- If the AE 3.5 application being migrated included Java clients that must now run against the application on WebSphere Application Server V4.0.1 for z/OS and OS/390, you may need to install the WebSphere for z/OS Java Technology Client available as a download at:

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

  However, it should be mentioned that secured interoperability with WebSphere Application Server V4.0.1 for z/OS and OS/390 is not supported at the WebSphere AE 3.5.x product level.

  Note: WebSphere ships an XML parser for its own use. This is not part of the J2EE programming model and applications should not be built dependent on the XML parser level shipped with the server. When developing applications, the XML parser required by the application should be packaged with the application ensuring that the application can be ported to different servers.

Managing connectivity to IMS and CICS applications between WebSphere AE and WebSphere for z/OS and OS/390

In addition to JMS and JDBC support, WebSphere Application Server V4.0.1 for z/OS and OS/390 provides the capability to connect to IMS and CICS applications.

Consistent with the WebSphere 4.0 family, WebSphere Application Server V4.0.1 for z/OS and OS/390 supports Common Connection Framework (CCF) access out of Servlets for migration purposes. It should be noted that WebSphere Application Server V4.0.1 for z/OS and OS/390 requires the use of CICS Transaction Gateway (CTG) 4.0, while WebSphere AE 3.5.x supported earlier levels. An application change may be required.

WebSphere Application Server V4.0.1 for z/OS and OS/390 also provides "beta" level connectors supporting the Common Client Interfaces defined by the J2EE Connector Architecture. They utilize z/OS and OS/390 function to provide container managed functions (i.e. 2 phase commit, etc.). These connectors are available as a download at:

```
http://www.ibm.com/software/webservers/appserv/
```

At this time, these are the only J2EE Connector Architecture connectors supported by WebSphere Application Server V4.0.1 for z/OS and OS/390. Applications being moved from the WebSphere AE platform that connect to CICS or IMS need to change to use these connectors.

Moving WebSphere AE platform applications with the Application Assembly Tool (AAT) to WebSphere for z/OS and OS/390

WebSphere Application Server V4.0.1 for z/OS and OS/390 provides an Application Assembly Tool (AAT) available as a download at:

```
http://www.ibm.com/software/webservers/appserv/
```

This tool must be used for applications being moved from the WebSphere AE platform. The AAT tool has an option to invoke the EJBDeploy tool as part of its processing. Ensure this option is enabled when moving applications that were generated on VisualAge for Java (VAJ) 3.5.x.

If using VAJ as the development tool, it should be noted that VAJ provides utility classes (above the J2EE architecture) to support functions such as the test client. For these items to function on WebSphere Application Server V4.0.1 for z/OS and OS/390, these utility classes must be packaged with the application EAR.

Part 5. Appendixes

Appendix A. Migration Considerations

Migrating from V3.5 SE

WebSphere for z/OS V4.0.1 includes a plug-in similar to the WebSphere Application Server V3.5 plug-in. If you have Web applications that you were previously running on WebSphere Application Server Standard Edition V3.5 that are not J2EE compliant, you can install them in the WebSphere for z/OS plug-in, and then migrate them over time, while creating new applications in a WAR file format that can be installed into the V4.0.1 Web container.

If you prefer to continue using WebSphere Application Server Standard Edition V3.5, it is possible for V3.5 to co-exist on the same z/OS or OS/390 system with WebSphere for z/OS V4.0.1 as long as the V3.5 HFS is mounted on a different mount point than the V4.0.1 HFS. The ability to have both the V3.5 and V4.0.1 Application Server on the same system also enables you to migrate existing V3.5 Web applications to your Web container over time, while creating new applications in a WAR file format that can be installed into the V4.0.1 Web container.

Regardless of which migration strategy you select, both sets of applications (the ones installed in the plug-in and the ones installed in the Web container) can be accessed, using HTTP protocol, from a browser. This capability enables you to:

- Continue to run existing V3.5 Web applications while becoming familiar with V4.0.1
- Develop new Web applications at the Java Servlet Specification Version 2.2 level, package them as WAR files, and install them in a Web container on the J2EE server.
- Slowly migrate existing V3.5 Web applications to a Web container. (These applications must be packaged into WAR files before you can use the Application Assembly Tool to install them into the Web container.)

  Note: Servlets can no longer be served by using their class name. Class names must be mapped to a servlet in a WAR file.

- Continue to run Web applications that do not comply with the Java Servlet Specification Version 2.2, that or require JavaServer Pages (JSPs) written at a 0.91 or 1.0 specification level, using either the V3.5 Application Server or the V4.0.1 plug-in.

To use the WebSphere for z/OS plug-in, you must:

- Specify the fully qualified name of the WebSphere for z/OS plug-in's was.conf file as the second parameter on the ServerInit directive in the HTTP server's httpd.conf configuration file that indicates the entry point to the WebSphere for z/OS plug-in's initialization routine.
- Copy the contents of your V3.5 was.conf file, except for the appserver.version property, to the WebSphere for z/OS plug-in's was.conf file. Be sure to include all of the **webapp** and **deployedwebapp** properties defining these applications.

If the HTTP Server detects a value in this second position of the ServerInit directive when it receives a request from a browser, it:

1. Searches the WebSphere for z/OS plug-in's was.conf file for a **deployedwebapp** property for the requested application. If a match is found, processing will be handled by the V3.5 Application Server.
2. If a matching **deployedwebapp** property is not found, the HTTP server communicates with WebSphere for z/OS to determine which J2EE server contains the Web and EJB containers for the requested application. The HTTP server then sends the request to the appropriate J2EE server for processing.

If a second parameter is not specified on the ServerInit directive, all requests will be sent directly to a J2EE server for processing.

When you are ready to migrate your Web applications to a Web container, you must:
1. Ensure that all of the servlets and JSPs contained in your Web applications conform to the Javasoft Servlet Specification V2.2 and the JavaServer Pages 1.1 specification level.
2. For each application, package all of the Web components into a WAR file, using standard Java Archive tools.
3. Using the Application Assembly Tool for z/OS and OS/390, convert each WAR file to an EAR file. Should you then wish to enable the EJBs and/or the Web Applications in the EAR as Web Services run the SoapEAREnabler tool. For information on how to use these tools, see *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.
4. Using the WebSphere for z/OS System Management User Interface, install the EAR file into a Web container on the V4.0.1 J2EE server. (For information on how to use this interface, see *WebSphere Application Server Version 4.0.1 for z/OS and OS/390: System Management User Interface*.)

Migrating from V3.02 SE

You have two options for migrating from V3.02SE:
1. You can first migrate to V3.5 and then to V4.0.1.

   Migrating to V3.5 and then to V4.0.1 enables you to continue using Web applications written to the V2.1 Javasoft Servlet Specification and JavaServer Pages written to the 0.91 and 1.0 specification levels, provided you configure your V3.5 Application Server to run in compatibility mode. (See the *WebSphere Application Server for OS/390 Planning, Installing and Using, Version 3.5*, GC34–4835 for more information about how to migrate from V3.02 and how to set up your V3.5 Application Server to run in compatibility mode.)

   Once you have your V3.5 Application Server running, you can follow the instructions in the previous section, "Migrating from V3.5 SE" on page 105, and add V4.0.1 to the same z/OS or OS/390 system.

   The **appserver.compliance.mode** was.conf file property, that was added for V3.5, enables you to continue to use most of your V3.02 applications, while simultaneously supporting the Java Servlet API 2.2 specification. To ensure compatibility, this new property enables you to indicate to the Application Server whether the Web applications you are running comply with the Java Servlet API 2.1 or 2.2 specification. If the components in a Web application comply with the Java Servlet API 2.1 specification, you specify **false** to indicate to the Application Server that you want it to run in compatibility mode; if they comply with the Java Servlet API 2.2 specification, you specify **true** for this property. **false** is the default value for this property.

The following table describes how the **appserver.compliance.mode** property setting affects Servlet API classes and methods, and various application functions. In this table, "Compatibility Mode" indicates that the applications comply with the Java Servlet API 2.1 specification, and "Compliance Mode" indicates that the applications comply with the Java Servlet API 2.2 specification. Make sure you understand all of the application processing implications noted in this table before changing the setting of this property to **true**.

Method or function	Compatibility mode	Compliance mode
Error-page tags in the .webapp file	The >error-page< contains a string that is the relative path to the Web application's default error page.	The >error-page< contains the following tags: • >location< • >exception-type< • >error-code< These tags are only available in a .webapp file. Since there is no corresponding property in the was.conf file, this function is only available when a .webapp file is used to define a Web application.
getCharacter Encoding() method	If the client request did not send any character encoding data, the default encoding of the server JVM is returned.	If the client request did not send any character encoding data, **null** is returned.
Default content type on *response buffer reset*	On *response buffer reset*, the content type of the request is reset to **text/html**.	On *response buffer reset*, the content type is cleared and not set to a default value.
getMimeType() method	If the file extension does not map to a valid mime type, the mime type **www/unknown** is returned.	If the file extension does not map to a valid mime type, **null** is returned.
HTTP Session scoping	Values placed in the HTTP Session object have a **global scope**, across all Web applications.	Values placed in the HTTP Session object have a **scope limited to the Web application** that created the value.

Request mapping behavior	• Exact mapping is not supported.
	• Wildcard mapping is an implied wildcard. That is, /Servlet really means /Servlet*.
	• Any URL pattern specified without /* on the end is assumed to be a wildcard rule, and /* is added in the Servlet runtime environment.
	• Any URL pattern provided with /* on the end is accepted and used as is.
		• The servlet specification pattern mapping logic is followed, including support for exact matches.
		• To specify the URL, the Servlet 2.2 specification allows the following syntax:
		  a. A string beginning with / and ending with /* specifies a wildcard match.
		  b. A string beginning with *. specifies an extension mapping.
		  c. All other strings are used as exact matches.
		• The Servlet 2.2 specification indicates how requests for resources are mapped to the appropriate resources. Mapping occurs in the following order:
		  a. exact match
		  b. longest wildcard match
		  c. matching extension
		  d. default servlet (defined by / URL)

2. You can migrate directly to V4.0.1. To migrate directly to V4.0.1, you must:
   a. Ensure that all of the servlets and JSPs contained in your Web applications conform to the Javasoft Servlet Specification V2.2 and the JavaServer Pages 1.1 specification level.
   b. For each application, package all of the Web components into a WAR files, using standard Java Archive tools.
   c. Using the Application Assembly Tool for z/OS and OS/390, convert each WAR file to an EAR file. Should you then wish to enable the EJBs and/or the Web Applications in the EAR as Web Services run the SoapEAREnabler tool. For information on how to use these tools, see *WebSphere Application Server V4.0.1 for z/OS and OS/390: Assembling J2EE Applications*, SA22-7836.
   d. Using the WebSphere for z/OS System Management User Interface, install the EAR file into a Web container on the V4.0.1 J2EE server. (For information on how to use this interface, see *WebSphere Application Server Version 4.0.1 for z/OS and OS/390: System Management User Interface*.)

Migrating from JDK 1.1x to SDK 1.3

Regardless of whether you migrate directly to V4.0.1 or migrate to V3.5 first, both the V4.0.1 and V3.5 Application Server run-time environments are built on SDK 1.3. You should be able to run most programs that ran under JDK 1.1x with little or no modification. However, the following list summarizes some minor potential incompatibilities that may require your applications to be modified:

1. There are now two Timer classes:
   - java.util.Timer (new)
   - javax.swing.Timer (existed in V1.1x)

   If an application has the following two import statements:
   ```
   import java.util.*;
   import javax.swing.*;
   ```

   and refers to javax.swing.Timer by its unqualified name, the following import statement must be added in order for the ambiguous reference to class Timer to be correctly resolved:
   ```
   import javax.swing.Timer;
   ```

2. The implementation of method **java.lang.Double.hashcode** has been changed to conform to the API specification. This change should not affect the behavior of existing applications because hashcode returns a truncated integer value.

3. A new Permission class, **java.sql.SQLPermission**, has been added in version 1.3. WebSphere Application Server V3.5 on MultiPlatforms supports this new class; WebSphere Application Server for OS/390 V3.5 does not.

4. The internal implementation of the Java Sound APIs (in class **com.sun.media.sound.SimpleInputDevice**) now checks **javax.sound.sampled.AudioPermission**. This new check means that, under 1.3, applets must now be given the appropriate AudioPermission to access audio system resources.

5. JInternalFrames are no longer visible by default. Developers must set the visibility of each JInternalFrame to true in order to have it show up on the screen.

6. The **TableColumn.getHeaderRenderer** method returns null by default. Therefore, you must use the new **JTableHeader.getDefaultRenderer** method instead to get the default header renderer.

7. The JTable's default text editor now gives setValueAt objects of the appropriate type, instead of always specifying strings. For example, if setValueAt is invoked for an Integer cell, then the value is specified as an Integer instead of a String. If you implemented a table model, you might have to change its **setValueAt** method to take the new data type into account. If you implemented a class that is used as a data type for cells, make sure that your class has a constructor that takes a single String argument.

8. It is no longer possible for sufficiently trusted code to modify final fields by first calling **Field.setAccessible(true)** and then calling **Field.set()**. An IllegalArgumentException will be thrown when an attempt is made to modify a final field. The JNI Set<Field> routines can be used to set non-static final fields.

9. The specification and behavior of the constructors **BasicPermission(String name)** and **BasicPermission(String name, String actions)** in class **java.security.BasicPermission** have been modified. When the name parameter is null, the constructors now throw a NullPointerException. When name is an empty string, the constructors now throw an IllegalArgumentException. This change of behavior is inherited by subclasses of **BasicPermission**. The change also affects the behavior of **java.lang.System.getProperty()** and **java.lang.System.setProperty()** whose implementations construct an instance of **PropertyPermission**, a subclass of **BasicPermission**. Because of this change, a call to **getProperty** or **setProperty** with an empty property name (that is, **getProperty("")** or **setProperty("", value)**) will result in an IllegalArgumentException. When using JDK instead of SDK, such a call would return quietly with no exception.

10. The behavior of **java.net.URL** has changed for cases where a URL instance is constructed from a String. A final slash ('/') is not automatically added to a URL when the URL is constructed without one. For example, the following code:

```
URL url = new URL("http://www.xxx.yyy");
System.out.println(url.toString());
```

now results in the following output:

```
http://www.xxx.yyy
```

11. The javac complier has a new implementation with the following implications:
    - Inherited members of an enclosing class are now accessible.
    - A local variable or catch clause parameter can be hidden when it is declared within the scope of a like-named method parameter, local variable, or catch clause parameter.
    - It is now illegal for a package to contain a class or interface type and a subpackage with the same name. A package, class, or interface is presumed to exist if there is a corresponding directory, source file, or class file accessible on the classpath or the sourcepath, regardless of its content.
    - A qualified name in a constant expression must be of the form TypeName.identifier.
    - Member classes of interfaces are inherited by implementing classes

12. **java.io.ObjectInputStream** has been optimized to buffer incoming data. This change should improve performance. This change causes ObjectInputStream to more frequently call the multi-byte read(byte[], int, int) method of the underlying stream. If underlying stream classes incorrectly implement this method, serialization failures may occur.

13. A public input method engine SPI as been included so that a client side adapter can be developed and distributed as a separate product and installed

into any implementation of the Java 2 platform. Environments that are currently set up to allow text entry using a server-based input method should updated to use a different solution, such as host input methods.

For the most current Java for OS/390 documentation, go to URL:

`http://www.ibm.com/s390/java/`

Setting runtime properties

In V3.5 of the Application Server, runtime settings, such as the location of the JVM properties file, the level of logging that is to be performed, and the location of the working directory, were set in the was.conf file. In V4.0.1, the runtime settings established for the J2EE server configuration apply to the containers within that server. Therefore, properties, such as the **appserver.jvmpropertiesfile** and **appserver.loglevel** properties, do not exist in the webcontainer.conf file.

Setting Session properties

You can continue to use most of the session settings you had in effect in V3.x of the Application Server. The following session properties can be copied from your V3.x was.conf file and added to the V4.0.1 Web container configuration file, webcontainer.conf:

- session.enable
- session.urlrewriting.enable
- session.cookies.enable
- session.protocolswitchrewriting.enable
- session.cookie.name
- session.cookie.comment
- session.cookie.maxage
- session.cookie.path
- session.cookie.secure
- session.tablesize
- session.invalidationtime
- session.tableoverflowenable
- session.dbenable
- session.dbtablename
- session.domain

Accessing services

In V3.5 of the Application Server, access to services such as JDBC and JNDI, was established through settings in the was.conf file. In V4.0.1, access to these tools is provided by the J2EE server. Therefore, properties, such as the **jdbcconnpool** properties, do not exist in the webcontainer.conf file, which contains the Web container's configuration settings. (See *WebSphere Application Server Version 4.0.1 for z/OS and OS/390: Assembling J2EE Applications* for more information about the webcontainer.conf file.)

Note: WebSphere for z/OS does not require a was.conf file unless you are using Web applications that are installed in the WebSphere for z/OS plug-in instead of in the Web container. Even if you continue to use a was.conf file, the property settings within this file only affect the plug-in and the Web

applications that are installed in the plug-in. They do not affect the Web container configuration settings or the Web applications that are installed in the Web container.

Appendix B. Notices

This information was developed for products and services offered in the U.S.A.
IBM may not offer the products, services, or features discussed in this document in
other countries. Consult your local IBM representative for information on the
products and services currently available in your area. Any reference to an IBM
product, program, or service is not intended to state or imply that only that IBM
product, program, or service may be used. Any functionally equivalent product,
program, or service that does not infringe any IBM intellectual property right may
be used instead. However, it is the user's responsibility to evaluate and verify the
operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter
described 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
North Castle Drive
Armonk, NY 10504-1785
USA

For license inquiries regarding double-byte (DBCS) information, contact the IBM
Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

**The following paragraph does not apply to the United Kingdom or any other
country where such provisions are inconsistent with local law:**
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS
PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS
FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or
implied warranties in certain transactions, therefore, this statement may not apply
to you.

This information could include technical inaccuracies or typographical errors.
Changes are periodically made to the information herein; these changes will be
incorporated in new editions of the publication. IBM may make improvements
and/or changes in the product(s) and/or the program(s) described in this
publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for
convenience only and do not in any manner serve as an endorsement of those Web
sites. The materials at those Web sites are not part of the materials for this IBM
product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it
believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

Examples in this book

The examples in this book are samples only, created by IBM Corporation. These examples are not part of any standard or IBM product and are provided to you solely for the purpose of assisting you in the development of your applications. The examples are provided "as is." IBM makes no warranties express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose, regarding the function or performance of these examples. IBM shall not be liable for any damages arising out of your use of the examples, even if they have been advised of the possibility of such damages.

These examples can be freely distributed, copied, altered, and incorporated into other software, provided that it bears the above disclaimer intact.

Programming Interface information

This publication documents information that is NOT intended to be used as Programming Interfaces of WebSphere for z/OS.

Trademarks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

CICS	RAMAC
DB2	RMF
IBM	SecureWay
IMS	S/390
IMS/ESA	VTAM
MVS	WebSphere
OS/390	z/OS
RACF	

The term CORBA used throughout this book refers to Common Object Request Broker Architecture standards promulgated by the Object Management Group, Inc..

Lotus, Notes, Domino, and Lotus Go Webserver, are trademarks of the Lotus Development Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, ActiveX, Visual Basic, Visual C++, Windows, Windows NT, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Glossary

For more information on terms used in this book, refer to one of the following sources:

- Sun Microsystems Glossary of Java Technology-Related Terms, located on the Internet at:

  `http://java.sun.com/docs/glossary.html`

- *IBM Glossary of Computing Terms*, located on the Internet at:

  `http://www.ibm.com/ibm/terminology/`

- The Sun Web site, located on the Internet at:

  `http://www.sun.com/`

Index

A

AAT 89
Accessibility 66
accessing CICS 40
accessing DB2 for OS/390 through
 JDBC 44
accessing IMS 42
administration
 considerations 5
application assembly and
 deployment 31
Application Assembly tool 89
application development
 considerations 5
Application Server V3.02, migrating
 from 106
Application Server V3.5SE, migrating
 from 105
auditing considerations 5

B

buffer limits 67

C

Client container 68
coexistence, definition 3
coexistence with V3.5SE 105
Common Connector Framework 38
customization
 general considerations 5
Customization panels 71

D

developing a migration strategy 4
Distributed exceptions 74

G

general user considerations 6

H

HTTP session state database 33

I

interface considerations 6

J

J2EE services for Web applications 48
JAVA Mail/JAVA Beans Activation
 Framework 78
Java Message Service (JMS) 80

JDBC, accessing 110
JDNI, accessing 110
JNDI 82
JRas support 46

M

migrating from V3.02 106
migrating from V3.5SE 105
migration
 overview 3
 roadmap 7
 strategy 4
 terminology 3
migration considerations 105

O

operating system and database 24
operational considerations 6
overview, migration 3

P

planning for migration 4
process/execution model 28
processing considerations 5

R

release overview 23, 97
reloading servlets 64
roadmap, migration 7
RunAs 49, 83
runtime properties 110

S

security mechanism 35
servlet reloading 64
SMF recording: Support of EJB
 container 85
strategy, migration 4
supported migration paths 7
system tools, accessing 110

T

task considerations 5

W

Warm start 87
Web: AAT WebSphere Application and
 XDD support 89
Web Security 91
Web services for V4.0.1 (SOAP) 93

WebSphere updates
 Common Connector Framework 38
 J2EE services for Web applications 48
 JRas support 46
 operating system and database 24
 RunAs 49
WebSphere V4.0.1 new function
 Accessibility 66
 buffer limits 67
 Client container 68
 Customization panels 71
 Distributed exceptions 74
 JAVA Mail/JAVA Beans Activation
  Framework 78
 Java Message Service (JMS) 80
 JNDI 82
 RunAs 83
 SMF recording: Support of EJB
  container 85
 Warm start 87
 Web: AAT WebSphere Application and
  XDD support 89
 Web security 91
 Web services for V4.0.1 (SOAP) 93

119

IBM ®

Program Number: 5655–F31

Printed in the United States of America