IBM Systems - iSeries

e-business and Web serving
WebSphere Application Server - Express Version 5.1
Migration

Version 5 Release 4

IBM Systems - iSeries

e-business and Web serving
WebSphere Application Server - Express Version 5.1
Migration

Version 5 Release 4

Note

Before using this information and the product it supports, be sure to read the information in "Notices," on page 23.

Third Edition (February 2006)

This edition applies to version 5.1 of WebSphere Application Server - Express (5722-E51) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

Contents

Migration

Migration tasks and tools allow you to transfer your existing configuration settings and existing applications to the new version of the product.

Product migration functions are provided by the migration tools in IBM WebSphere Application Server - Express. This table shows the supported migration paths:

Existing version	New version
WebSphere Application Server Version 3.5.6 (or higher) Standard Edition	WebSphere Application Server - Express Version 5.1
WebSphere Application Server - Express Version 5.0	WebSphere Application Server - Express Version 5.1

See the Migration

section in the WebSphere Application Server and Network Deployment Version 5.1 documentation for information on migrating to those releases.

This documentation provides detailed step-by-step instructions to successfully migrate your older WebSphere Application Server releases to WebSphere Application Server - Express Version 5.1. The following topics take you through the migration process. It is recommended that you read each topic in the order listed, and that you read through the migration documentation at least once before you start the migration process.

For more information on the typical migration scenario, see **"Overview" on page 2**.

"Step 1: Plan your migration" on page 7

"Step 1: Plan your migration" on page 7
This topic describes the steps you should complete before starting the migration process. It also includes information on how to evaluate your current environment to ensure you meet the requirements for migration.

"Step 2: Migrate your applications" on page 11

"Step 2: Migrate your applications" on page 11
This topic describes how to migrate your applications to WebSphere Application Server - Express. This involves modifying your applications to use the specifications supported by WebSphere Application Server - Express.

"Step 3: Migrate your WebSphere Application Server instances" on page 14

"Step 3: Migrate your WebSphere Application Server instances" on page 14
This topic provides step-by-step instructions to migrate an older WebSphere Application Server instance to a new WebSphere Application Server - Express Version 5.1 instance.

"Step 4: Complete the migration" on page 19

"Step 4: Complete the migration" on page 19
This topic describes the manual steps that you might need to perform to complete the migration process. These are steps that are not performed automatically by the WebSphere Application Server - Express migration tools.

Overview

Migration consists of stages that include scoping, skills migration, code migration, and runtime migration. Refer to these articles for reference on these stages and other general information on the migration process:

- Part 1: Designing Software for Change

- Part 2: Stages of Migration

- Part 3: Migration assessment

- Migrating Applications from WebSphere Application Server Standard Edition to WebSphere Application Server-Express V5

- WebSphere Application Server - Express V5.0 for iSeries

Note: These articles are not iSeries specific and may contain information not pertinent to iSeries.

WebSphere Application Server - Express migration leverages the existing environment and applications and changes them to be compatible with the WebSphere Application Server - Express environment. Existing application components and configuration settings are applied to the WebSphere Application Server - Express environment during the migration process.

Migration involves modifying your applications so that they run in WebSphere Application Server - Express and then migrating those applications and your environment to WebSphere Application Server - Express. The latter step can be performed by using the migration tools shipped with the product.

Migration Tools

Product migration functions are provided by the the WebSphere Application Server migration tools. These tools perform migration from Version 3.5.x and from WebSphere Application Server - Express Version 5.0.x to WebSphere Application Server - Express Version 5.1. The migration tools are comprised of the following commands:

- **WASPreUpgrade** saves Version 3.5 or WebSphere Application Server - Express Version 5.0.x configuration data and applications from a previous version to a backup directory. For more information, see "The WASPreUpgrade script" on page 3.

- **WASPostUpgrade** restores Version 3.5 or WebSphere Application Server - Express Version 5.0.x configuration data and applications into WebSphere Application Server - Express Version 5.1. This command uses the output from the WASPreUpgrade command. For more information, see "The WASPostUpgrade script" on page 5.

The following tools are also available to assist you in migrating your applications:

- **MigrateWC** converts JSP Specification .91 or 1.0 application components to JSP Specification 1.1 components. It also converts Servlet 2.1 specification components to Servlet 2.2 specification components. The converted components are written to a new file. This tool is not shipped with WebSphere Application Server. For more information and to download this tool, see the WebSphere Developer Domain

. Search in the **Downloads** section for MigrateWC.

- **CACT** analyzes compiled servlet and enterprise bean Java class files and provides information on any API that is deprecated or not supported in WebSphere Application Server Version 4, Version 5, or Version 5.1. This tool is not shipped with WebSphere Application Server. For more information and to download this tool, see the WebSphere Developer Domain

. Search in the WSDD **Library** section for CACT.

- **WebSphere Development Studio Client (WDSC)** is part of the WebSphere Studio family of application development tools. This tool can be used to import your application source files, set the desired J2EE version in the product, for example to J2EE Version 1.3, and rebuild the source files. Use of changed, not supported, and deprecated APIs are reported through compilation errors. After it is built, the application can be tested with the version of WebSphere Application Server that is included with the tool.

The WASPreUpgrade script

The **WASPreUpgrade** script is a migration tool for migrating the configuration and applications of previous WebSphere Application Server and WebSphere Application Server - Express versions to a WebSphere Application Server - Express V5.1 application server node. The script is located in the /QIBM/ProdData/WebASE51/ASE/bin directory.

Authority

To run this script, your user profile must have *ALLOBJ authority.

Syntax

The syntax of the WASPreUpgrade script is:

```
WASPreUpgrade backupDirectory currentWASDirectory administrationNodeName  -instance instance
  [ -nameServiceHost host_name [ -nameServicePort port_number ] ]
  [ -traceString trace_spec [ -traceFile file_name ] ]
```

Parameters

The parameters of the WASPreUpgrade script are:

- *backupDirectory*
  This is a required parameter and must be the first parameter that you specify. The value *backupDirectory* specifies the name of the directory where the script stores the saved configuration and files. This is also the directory from which the WASPostUpgrade tool reads the configuration and files.

For example, if you are migrating a configuration from WebSphere Application Server Version 3.5, you could specify the /home/was35_instancename/backup directory. If the directory does not exist, the WASPreUpgrade script creates it.

- ***currentWASDirectory***
  This is a required parameter and must be the second parameter that you specify. The value *currentWASDirectory* specifies the name of the instance root directory for instance that you want to migrate. For Version 3.5.x this is /QIBM/UserData/WebASStd/*instance*, where *instance* is the name of the instance you want to migrate. For WebSphere Application Server - Express Version 5.0.x this is /QIBM/UserData/WebASE/ASE5/*instance*, where *instance* is the name of the instance you want to migrate.
- ***administrationNodeName***
  This parameter is required and must be the third parameter that you specify. The value *administrationNodeName* specifies the name of the node that contains the administrative server for the previous version of the product. The value of this argument is case-sensitive and must match the node name given in the topology tree on the **Topology** tab of the administrative console for the previous version. The WASPreUpgrade tool uses this parameter to call the XMLConfig tool.
- **-instance**
  This is a required parameter. The value *instance* specifies the name of the WebSphere Application Server - Express V5.1 instance to which you are migrating the configuration.
- **-nameServiceHost**
  This is a required parameter if you are migrating from Version 3.5.x. The value *host_name* specifies the TCP/IP host name of the iSeries server. If you do not specify this parameter, the default host name is localhost.
- **-nameServicePort**
  This is a required parameter if you are migrating from Version 3.5.x. The value *port_number* specifies specifies the bootstrap port for the administrative server for the instance from which you are migrating. If you do not specify this parameter, the default value is 900.
- **-traceString**
  This is an optional parameter. The value *trace_spec* specifies the trace information that you want to collect. To gather all trace information, specify `"*=all=enabled"` (with quotation marks). The default is to not gather trace information. If you specify this parameter, you must also specify the -traceFile parameter.
- **-traceFile**
  This is an optional parameter. The value *trace_file* specifies the name of the output file for trace information. If you specify the -traceString parameter but do not specify the -traceFile parameter, the script does not generate a trace file.

Logging

The WASPreUpgrade tool displays status to the screen while it is running. It also saves a more extensive set of logging information in the WASPreUpgrade.log file. This file is located in the *backupDirectory* directory, where *backupDirectory* is the value specified for the backupDirectory parameter.

Security

The WASPreUpgrade script uses the XMLConfig script to export configuration data from WebSphere Application Server Version 3.5 Standard Edition. If WebSphere security is enabled, ensure that the properties are configured properly in the sas.client.props property file for the Version 3.5 instance before running the WASPreUpgrade script or the script fails when it attempts to invoke the XMLConfig script.

Examples

Migrate from WebSphere Application Server Version 3.5.x Standard Edition

This example illustrates how to migrate from WebSphere Application Server Version 3.5.x Standard Edition. The backup directory is named /home/was35_instancename/WebASAdv, and the instance root of the previous version is /QIBM/UserData/WebASAdv/default. The administrative server is contained in myNode.

```
> WASPreUpgrade /home/was35_instancename/WebASAdv /QIBM/UserData/WebASAdv/default MYISERIES
```

The WASPostUpgrade script

The **WASPostUpgrade** command is a migration tool for migrating the configuration and applications of previous versions to a WebSphere Application Server - Express application server node. The script is located in the /QIBM/ProdData/WebASE51/ASE/bin directory.

The WASPostUpgrade tool installs all migrated applications into the /QIBM/UserData/WebASE51/ASE/*instance*/installedApps directory, where *instance* is the name of the instance to which you are migrating. The tool includes applications from the installedApps directory and other directories from the previous version.

Authority

To run this script, your user profile must have *ALLOBJ authority.

Syntax

The syntax of the WASPostUpgrade script is:

```
WASPostUpgrade backupDirectory  -instance instance
 [ -cellName cell_name ] [ -nodeName node_name ]
 [ -webModuleAdditionalClasspath classpath ]
 [ -documentRootLimit number ]
 [ -substitute"key1=value1[;key2=value2;[...]]" ]
 [ -portBlock port_starting_number ] [ -backupConfig < true | false > ]
 [ -replacePorts < true | false > ]  [ [ -traceString trace_spec
 [ -traceFile file_name ] ]
```

Parameters

The parameters of the WASPostUpgrade script are:

- ***backupDirectory***
  This is a required parameter. The value *backupDirectory* specifies the name of the directory in which the WASPreUpgrade tool stores the saved configuration and files, and from which the WASPostUpgrade tool reads the configuration and files.
- **-instance**
  This is a required parameter. The value *instance* specifies the name of the WebSphere Application Server - Express instance to which the script migrates your configuration.
- **-cellName**
  This is an optional parameter. The value *cell_name* specifies the cell name that the script updates. If you do not specify this parameter, the script reads the configuration to determine the cell name. If only one cell name is found, the script updates that cell. If no cell name is found, or if more than one cell name is found, the script returns an error.
- **-nodeName**
  This is an optional parameter. The value *node_name* specifies the node name that the script updates. If you do not specify this parameter, the script reads the configuration to determine the node name. If only one node name is found, the script updates that cell. If no node name is found, or if more than one node name is found, the script returns an error.
- **-webModuleAdditionalClasspath**
  This is an optional parameter. The value *classpath* specifies the path or the path and file names of specific directories or files that you do not want copied into the Web archive (WAR) file that the script

processes. Instead, the script adds the paths and files to the Web Module extension (ibm-web-ext.xmi) additionalClassPath attribute. If you want to specify multiple values for this parameter, separate them with a colon (:).

- **-documentRootLimit**
  This is an optional parameter. The value *number* specifies the number of files that the script copies from the document-root field of your Web application. The default value is 300.
- **-substitute**
  This is an optional argument. This parameter specifies values for security variables to substitute. (for example, `-substitute "NODE_NAME=admin_node;APP_SERVER=default_server"`).

  In the input XML data file, each key appears as key for substitution. This argument replaces all occurrences of $NODE_NAME$ and APP_SERVER with admin_node and default_server, respectively, in the input XML file.

  If the substitution string contains semicolons, use $semiColon$ to separate it from the ";" delimiter.
- **-portBlock**
  This is an optional parameter. The value *port_starting_number* specifies the first of a block of 10-15 consecutive port numbers that zare not in use on the iSeries server where the migration is being performed. It is recommended that you always specify the -portBlock parameter if you do not want your instance's ports to conflict with the default instance's ports.
- **-backupConfig**
  This is an optional parameter. Valid values for this parameter are true and false. The default value is true. If true is specified, then the existing Version 5.1 configuration is saved before any changes are made by the WASPortUpgrade tool.
- **-replacePorts**
  This is an optional parameter. Valid values for this parameter are true and false. The default value is false. If true is specified, all virtual host alias port settings are replaced during migration. If migrating from Version 5.0.x, virtual host and transport settings in the Version 5.1 instance are replaced by the settings from the Version 5.0.x instance.
- **-traceString**
  This is an optional parameter. The value *trace_spec* specifies the trace information that you want to collect. To gather all trace information, specify `"*=all=enabled"` (with quotation marks). The default is to not gather trace information. If you specify this parameter, you must also specify the -traceFile parameter.
- **-traceFile**
  This is an optional parameter. The value *trace_file* specifies the name of the output file for trace information. If you specify the -traceString parameter but do not specify the -traceFile parameter, the script does not generate a trace file.

Logging

The WASPostUpgrade tool displays status to the screen while running. It also saves a more extensive set of logging information in the WASPostUpgrade.log file. This file is located in the /QIBM/UserData/WebASE51/ASE/*instance*/logs directory, where *instance* is the name of the instance that you are migrating to.

Examples

Migrate from WebSphere Application Server Version 3.5.x Standard Edition

This example illustrates how to migrate from WebSphere Application Server Version 3.5.x, Advanced Edition. The WASPreUpgrade script created the backup directory named /home/was35_instancename/WAS35Std. The WASPostUpgrade script reads the configuration from this directory to migrate to WebSphere Application Server - Express.

```
WASPostUpgrade /home/was35_instancename/WAS35Std
```

Step 1: Plan your migration

Complete these steps before you migrate to WebSphere Application Server - Express.

1. Before you can migrate, you must install WebSphere Application Server - Express. For more information on installing WebSphere Application Server - Express, see WebSphere Application Server - Express installation in the *Installation* topic.
2. In addition to migrating your instances and applications, you also need to familiarize yourself with the tools and features of WebSphere Application Server - Express. The Administration topic describes administrative tasks and the tools that are provided to perform those tasks.
3. Evaluate the deprecated Version 5.1 items. For more information, see "Deprecated Version 5.1 items."
4. Verify that you have the minimum prerequisites required for migration. For more information, see "Migration prerequisites" on page 9.
5. Evaluate the changes to API specification levels to determine what needs to be changed, if anything, in the applications that are to be migrated. Use these topics to plan your application migration requirements. For more information, see "API and Specifications for version 3.5.x" on page 10.

Deprecated Version 5.1 items

This document summarizes deprecated features in Version 5.1. A deprecated class or interface is supported for at least two full product releases, or three full years (whichever is longer) before it is removed from the product.

Use the following information to calculate the earliest release when the item might be removed, according to the deprecation policy.

Installation and migration tools

The currently packaged version of JDOM in WebSphere Application Server will not be packaged in future releases of WebSphere Application Server. JDOM is a Java representation of an XML document that provides an API for efficient reading, manipulating and writing documentation.

Recommended migration action: Go to JDOM

, obtain the latest copy of JDOM, and bundle it inside your application.

Note: Customers running WSADIE V4.1 applications need to migrate them to WSADIE V5.0.

Servers and clustering features

No deprecated features.

Application programming model and container support features

- Bean Scripting Framework (BSF), which is a JSP execution and debugging function, is being deprecated in WebSphere Application Server Version 5.1.

  Recommended migration action: If you use the JavaScript, Tcl, and Python languages, the functionality needs to be rearchitected. If you use BSF scripting in your own custom applications, they are unaffected. Custom scripts that are written for the WebSphere Application Server administrative console are also unaffected.

  BSF functionality continues to exist in WebSphere Application Server Version 5.1 but will be fully deprecated in a future version of the application server, according to the deprecation policy. If you debug JSPs, you might have to restart the application server during JavaScript debugging sessions.
- Data access programming interfaces in com.ibm.websphere.rsadapter.

Relational resource adapter interface: com.ibm.websphere.rsadapter

Methods have been deprecated in these types:

```
com.ibm.websphere.rsadapter.OracleDataStoreHelper
  public void doSpecialBLobWork(ResultSet rset, InputStream[] data,
    String[] blobColumnNames)
  public String assembleSqlString(String[] blobColumnNames,
    StringBuffer whereClause, String[] varValues, String tableName)
```

Recommended migration action: These relational resource adapter deprecated methods do not impact the application.

Note: You do not need to implement these deprecated methods in their subclasses if you have the subclass of OracleDataStoreHelper class. These deprecated methods are not called by the WebSphere Application Server run-time environment.

- Web container API modifications:

  Note: There are no declared deprecations. A Java API has changed between versions 1.3 and 1.4.

  – Old method signature:

  ```
  public String getStackTrace();
     // returns a String representation of the exception stack
  ```

  – New method signature (J2SE(TM) 1.4 in WebSphere Application Server Version 5.1):

  ```
  public StackTraceElement[] getStackTrace();
      // returns an array of stack trace elements
  ```

  – WebSphere Application Server Version 5.1 replacement method (a replacement method that supports the old functionality is provided):

  ```
  public String getStackTraceAsString();
      // returns a String representation of the Exception Stack
  ```

  See Migrate Web applications for more information.

- Web services gateway customization API

  Recommended migration action: No action is required at this time. However, use of Java API for XML-based Remote Procedure Call (JAX-RPC) handlers is recommended over Web services gateway-specific interfaces, such as filters, where possible. The Web services gateway API will be replaced in a future release.

Application services features

- Data access binaries — Common Connector Framework:

  The following .jar files are deprecated in Version 5.1:

  – ccf.jar
  – ccf2.jar
  – recjava.jar
  – eablib.jar

  Recommended migration action: The J2EE Connector Architecture solution should be used instead of the Common Connector Framework.

Security features

- Security programming interfaces:
  – The API is being deprecated for the following interface:

  ```
  com.ibm.websphere.security.auth.WSPrincipal.getCredential()
  ```

  Recommended migration action: Obtain the Subject that contains the WSCredential using one of the following methods:

  - The RunAs Subject is the Subject used for outbound requests.
  - The Caller subject is the Subject that represents the authenticated caller for the current request.
  - The methods to use to get the runAs and caller subjects are as follows:

com.ibm.websphere.security.auth.WSSubject.getRunAsSubject()

com.ibm.websphere.security.auth.WSSubject.getCallerSubject()

– The following interface is being deprecated:

com.ibm.websphere.security.auth.WSSecurityContext

Recommended migration action: Use JAAS for any authentication-related functionality.

– The following exception is being deprecated:

com.ibm.websphere.security.auth.WSSecurityContextException

Recommended migration action: Use JAAS for any authentication-related functionality.

– The following class is being deprecated:

com.ibm.websphere.security.auth.WSSecurityContextResult

Recommended migration action: Use JAAS for any authentication related functionality.

Environment features

No deprecated features.

System administration features

No deprecated features.

Performance features

No deprecated features.

Problem determination features

No deprecated features.

Migration prerequisites

Before you migrate your old version of WebSphere Application Server, verify that you meet these requirements:

- **Minimum version of WebSphere Application Server Version 3.5.x**
  If you are migrating from Version 3.5.x, you must be at Version 3.5.6 or higher.

  To determine the current level of WebSphere Application Server installed on your system, perform these steps:

  1. Enter this command on the CL command line:

     WRKLNK '/QIBM/ProdData/WebASAdv/properties/com/ibm/websphere/product.xml'

  2. Specify option 5 (Display) next to the product.xml file to view the contents. The number within the <version> tags show the current version you have installed.

  If you do not meet the minimum version, obtain the latest group PTF. See WebSphere Application Server PTFs for iSeries

  

  for information on the correct group PTF for your i5/OS release level and WebSphere Application Server - Express Version 5.0.x product.
- **Minimum version of WebSphere Application Server - Express Version 5.0.x**
  If you are migrating from WebSphere Application Server - Express, you must be at Version 5.0 or higher.
- **WebSphere Application Server - Express must be installed.**
  Follow the instructions Installation topic for more information. WebSphere Application Server - Express must be installed on the same iSeries server as the product to be migrated.

- ***ALLOBJ authority is required.**
  When you call the WasPreUpgrade and WasPostUpgrade migration tools, your user profile must have *ALLOBJ authority.

API and Specifications for version 3.5.x

If your existing applications currently support different specification levels than are supported by this version of the product, it is likely you must update at least some aspects of the applications to comply with the new specifications.

In many cases, IBM provides additional features and customization options that extend the specification level even further. If your existing applications use IBM extensions from earlier product versions, it might be necessary for you to perform mandatory or optional migration to use the same kinds of extensions in WebSphere Application Server - Express.

From Version 3.5.x to WebSphere Application Server - Express, main migration areas concern IBM entensions and migrating the Java specifications to Java 2.

The following table summarizes potential migration areas due to changes in supported specifications:

Specification	Support in Version 3.5	Support in WebSphere Application Server - Express	Must migrate from Version 3.5?	Details
JDBC	JDBC 1.0	JDBC 2.0	Yes	Many applications can run unchanged in WebSphere Application Server - Express although some changes may be required or recommended.
JavaServer Pages	JSP .91	JSP 1.2	Yes	JSP 1.0 and 1.1 APIs are a pure subset of JSP 1.2. For more information, see Migrate Web applications.
	JSP 1.0		No	
	JSP 1.1		No	
Servlets	Servlet 2.1	Servlet 2.3	Yes	Many Servlet 2.1 applications can run unchanged in WebSphere Application Server - Express although changes might be required or recommended. For more information, see Migrate Web applications.
	Servlet 2.2		No	Servlet 2.2 APIs are a pure subset of Servlet 2.3. For more information, see Migrate Web applications.

The following table summarizes potential migration areas due to changes in supported APIs:

API	Must migrate from Version 3.5?	Details
Sessions	Yes	Many applications can run unchanged in WebSphere Application Server - Express, although changes may be required or recommended. For more information, see "Migrate HTTP sessions" on page 12.

The following table summarizes potential migration areas due to changes in supported tools:

Tool	Must migrate from Version 3.5?	Details
XML Configuration Tool	Yes	Use JMX support provided by WSAdmin. For more information, see The wsadmin administrative tool in the *Administration* topic.
WebSphere Control Program	Yes	Use JMX support provided by WSAdmin. For more information, see The wsadmin administrative tool in the *Administration* topic.

Step 2: Migrate your applications

As technology advances, particularly in the area of Java components, new WebSphere Application Server product versions advance to support and extend the most recent open specification levels. If your existing applications currently support different specification levels than are supported by WebSphere Application Server - Express, it is likely you need to update at least a few aspects of the applications to comply with the new specifications.

Note: Evaluate the "Deprecated Version 5.1 items" on page 7 before you migrate your applications.

See these topics for instructions on how to migrate your applications:

"Application assembly in WebSphere Application Server - Express" on page 12
This topic describes the changes in application assembly in WebSphere Application Server - Express.

Migrate Web applications
This topic describes how to determine what, if any, migration changes are required for your Web applications.

"Migrate HTTP sessions" on page 12
This topic describes how to determine what, if any, migration changes are required for your HTTP sessions.

Migrate from wscp to wsadmin
This topic describes how to determine what migration changes are required for your wscp commands.

Migrate applications to use Java keystores
This topic describes how to migrate your applications to use Java keystores.

Migrate Web services
This topic describes how to determine what, if any, migration changes are required for your Web services.

"Migrate applications that use the XML parser for Java"
This topic describes how to migrate applications that use the open source XML parser for Java.

Application assembly in WebSphere Application Server - Express

Version 3.5.x developers use the administrative console to create, edit, and view application configurations. WebSphere Application Server - Express developers use the WebSphere Development Studio Client for iSeries to package, edit, and view J2EE applications. For more information, see Step 4: Assemble your application in the *Application Development* topic.

Packaging J2EE applications includes:

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

Migrate HTTP sessions

If you have Version 3.5 applications running in Servlet 2.1 mode, some of the following WebSphere Application Server - Express differences might influence how you choose to track and manage sessions.

1. Session persistence is not supported in WebSphere Application Server - Express.
2. During application development, modify session-related APIs as needed. Some API changes are required in order to redeploy existing applications on WebSphere Application Server - Express. These include changes to the HttpSession API itself as well as issues associated with moving to support for the Servlet 2.3 specification. Certain Servlet 2.1 API methods have been deprecated in Servlet 2.3 API. These deprecated APIs still work in WebSphere Application Server - Express, but they may be removed in a future version of the API. Changes are summarized in the following list:
   - Replace instances of getValue() with getAttribute()
   - Replace instances of getValueNames() with getAttributeNames()
   - Replace instances of removeValue() with removeAttribute()
   - Replace instances of putValue() with setAttribute()
3. During application development, modify Web application behavior as needed. In accordance with the Servlet 2.3 specification, HttpSession objects must be scoped within a single Web application context; they may not be shared between contexts. This means that a session can no longer span Web applications. Objects added to a session by a servlet or JSP in one Web application cannot be accessed from another Web application. The same session ID may be shared (because the same cookie is in use), but each Web application has a unique session associated with the session ID. WebSphere Application Server - Express provides a feature that can be used to extend scope of a session to enterprise application.
4. Use administrative tools to configure Session Manager security settings as needed. Relative to session security, the default Session Manager setting for Integrate Security is now false. This is different from the default setting in some earlier releases.

Migrate applications that use the XML parser for Java

Recently, the Apache Software Foundation announced the creation of the xml.apache.org project for Open Source XML solutions. As part of that announcement, IBM announced that it was donating the XML4J, XML4C and LotusXSL technologies to the xml.apache.org project. The parsing technologies have been renamed Xerces, and the LotusXSL technology has been renamed Xalan.

IBM is shifting its XML parsing development resources to work on the Xerces parsers. The objective is to use the Xerces code base as the foundation for XML4J and XML4C. This version of XML4J is based on the Apache Xerces version 2.0 codebase.

What's the difference between XML4J and Xerces?

Xerces has been internally tested at IBM, in addition to the testing that is done by the xml.apache.org project. The main JAR files names changed from `xml4j.jar` to `xmlParserAPIs.jar` and `xercesImpl.jar`, and from `xml4jSamples.jar` to `xercesSamples.jar`.

What part of the API is public, and what is subject to change?

API status must be defined before this question can be answered:

- **Public** - This is the type of API that a typical client developer should code against. Attempts are made to fix severe bugs in this type of API. Much of this type of API (defined as public) has reached W3C Recommendation status or a similar status in xml-dev working group. The interface is not expected to change much.
- **Experimental** - These interfaces and classes reflect the latest W3C specifications and SAX specifications from the xml-dev working group. Since these specifications are not finalized, the interfaces are subject to change. As some of these experimental specifications work through the working groups to the highest level, such as W3C Recommendation, they will be upgraded to the "Public" category.
- **Internal** - These classes are considered to be internal to Xerces, even though they may be public and have public methods. They can be used by developers who have complex and specific needs, such as building their own XML Parser. However, note that the architecture is subject to change.

API Status	API contents (packages, interfaces, classes and methods)	Comments
Public	DOM Level 1 interfaces and DOM Level 2 (DOM2) interfaces The DOM2 interfaces have been implemented in the same interfaces as DOM1, but as new methods. SAX Level 1 interfaces and SAX Level 2 (SAX2) interfaces	DOM L1, DOM L2, SAX1 and SAX2 Interfaces are stable.
Experimental	DOM Level 3 (DOM3) and Core DOM Level 3 (DOM3) Abstract Schemas and Load and Save	DOM L3 is in working draft status. XML4J provides a subset of DOM L3 support.
Internal	All other packages are considered to be internal.	The internal Xerces architecture may change.

As an Apache Open Source project, the Xerces community is very interested in your questions and feedback regarding the whole API, not only that part designated above as public.

If you have specific questions, patches, or feedback regarding the Xerces API or code, see the Apache Web site

.

What APIs should I use for new development?

All new development should use the org.apache.xerces.parsers.* classes. The four compatibility parser classes have been replaced by these two classes:

```
org.apache.xerces.parsers.SAXParser
org.apache.xerces.parsers.DOMParser
```

The control over validation has been changed to be a feature of the parser, rather then requiring separate classes.

The future direction for the parser instantiation classes is a parser instantiation API that results from the W3C DOM Level 3 effort, which is just getting underway.

To make sure your code is as stable as possible, use the interfaces specified in the **Public** section of the preceding table (for example DOM1 and SAX1). The table is updated periodically to reflect new interfaces and classes that are given **Public** status (for example, DOM2 and SAX2).

Is IBM making any additional support guarantees with XML4J?

IBM is not making any additional support guarantees for XML4J. In particular, IBM is not certifying that XML4J is Y2K compliant. XML4J makes no internal date calculations.

Migrating applications to use the XML4J 4.2.2 parser and the XSLT4J 2.5.4 transformer

JAXP defines a pluggability mechanism for a SAX and DOM parser via javax.xml.parsers APIs. Transformers are pluggable via javax.xml.transform APIs.

WebSphere Application Server Version 5.1 includes an XML4J 4.2.2 parser and an XSLT4J 2.5.4 transformer. To use a different implementation of JAXP in an application, package the parser and transformer in the application and change the class loader delegation to PARENT_LAST on the application or Web module.

It is recommended that applications do not use the parser or transformer implementation API directly, but that the applications use the JAXP API.

You can change an application to remove its dependency on the API in a previous version of the parser or the transformer from an earlier version of WebSphere Application Server. Package the JAR files in the application and set the classloader delegation mode to PARENT_LAST.

You must only recompile a Version 4.0.x XML application to migrate it to the Version 5.1 level.

Step 3: Migrate your WebSphere Application Server instances

After you have migrated your applications, you need to migrate your instance configurations.

See “Migration prerequisites” on page 9 to determine the currently installed product level of WebSphere Application Server Version 3.5.

“Migrate from a WebSphere Application Server Version 3.5.6 (or later) instance” on page 15
This topic contains detailed information and step-by-step instructions for migrating a WebSphere Application Server Version 3.5.6 (or later) instance to a new WebSphere Application Server - Express instance.

“Migrate from a WebSphere Application Server - Express Version 5.0.x instance” on page 17
This topic contains detailed information and step-by-step instructions for migrating a WebSphere Application Server - Express Version 5 instance to a new WebSphere Application Server - Express Version 5.1 instance.

Migrate from a WebSphere Application Server Version 3.5.6 (or later) instance

Tools for migrating administrative configurations are provided for Versions 3.5 and later. This support enables Version 3.5.6 Standard Edition (or later) to be upgraded to WebSphere Application Server - Express.

If multiple application servers exist in your Version 3.5.x environment, you have these options:

- The migration tools merge applications from multiple application servers of a Version 3.5.x instance into a single EAR file containing multiple Web modules, one Web module for each application server to be migrated. The EAR file is deployed as a single application within the WebSphere Application Server - Express instance.
- If you do not want your applications merged into a single EAR file and run in the same application server, you need to separate them into different Version 3.5.x instances prior to running the migration tools. Each instance should have a single application server containing the desired Web application or applications. You then migrate each Version 3.5.x instance to a corresponding WebSphere Application Server - Express instance.
- Use the WebSphere Development Studio Client environment to build an EAR file for each 3.5.x application you have and deploy each EAR file into a single Express instance. This option does not use the migration tools shipped with WebSphere Application Server - Express.

A summary of the product migration process is as follows.

- Create a WebSphere Application Server - Express instance (page 15).
- Start the Version 3.5 instance that is being migrated (page 15).
- Save the Version 3.5 configuration (page 15).
- Restore the Version 3.5 configuration into a WebSphere Application Server - Express instance (page 16).
- Start the WebSphere Application Server - Express instance (page 17).

Create a WebSphere Application Server - Express instance

See Create a new application server in the *Administration* topic for details on how to create a WebSphere Application Server - Express instance.

Start the Version 3.5 instance that is being migrated

Perform the following steps to start the WebSphere Application Server Version 3.5 instance:

1. Enter the following command from the CL command line:

   ```
   STRQSH
   ```

   This starts the Qshell environment so that you can run WebSphere Application Server scripts.
2. Run the strwasinst script with the following parameters:

   ```
   /QIBM/ProdData/WebASAdv/bin/strwasinst -instance 3.5.xInstanceName
   ```

   where *3.5.xInstanceName* is the name of the Version 3.5.x instance that is being migrated. You must wait for the administrative server to start successfully before continuing.

Save the Version 3.5 configuration

If you have servlets in the Version 3.5.x **default_app** Web Application, the WasPreUpgrade migration tool does not migrate them. If you wish to have these servlets migrated, you must change the **default_app** name before calling WasPreUpgrade. See Issues concerning the migration of JSPs and Servlets in the *Application Development* topic for more information.

Perform the following steps to save the Version 3.5 configuration:

1. Enter the following command from the CL command line:

```
STRQSH
```

This starts the Qshell environment so that you can run WebSphere Application Server scripts.

2. Run the WebSphere Application Server - Express WasPreUpgrade script as shown below:

```
/QIBM/ProdData/WebASE51/ASE/bin/WASPreUpgrade
/backup/myBackupDirectory
/QIBM/UserData/WebAsAdv/3.5.xInstanceName
adminNodeName
-nameServiceHost adminNodeName
-nameServicePort port_number
```

where:

- */backup/myBackupDirectory* (required parameter) is the fully qualified path to the integrated file system directory where the WasPreUpgrade migration tool stores the saved configuration and files. The directory is created if it does not already exist. Additionally,the tool writes a log file called WasPreUpgrade.log that chronicles the steps taken by WasPreUpgrade.
- /QIBM/UserData/WebAsAdv/*3.5.xInstanceName* (required parameter) is the fully qualified path of the Version 3.5.x administrative instance being migrated.
- *adminNodeName* (required parameter) is the name of the administration node for the Version 3.5.x instance. Generally, this is the iSeries host name from **CFGTCP** option 12. The WASPreupgrade tool invokes the Version 3.5 XMLConfig tool using this parameter.
- The -nameServiceHost and -nameServicePort parameters are also passed to XMLConfig. They are needed to override the default host name and port number used by XMLConfig and are required parameters when the Version 3.5.x instance being migrated is not the default instance. The value for -nameServiceHost is the TCP/IP host name of the iSeries server. The value for the -nameServicePort is the bootstrap port for the Version 3.5.x administrative server. The default bootstrap port is 900.

For a full explanation of the WasPreUpgrade migration tool and parameters, see "The WASPreUpgrade script" on page 3.

Restore the Version 3.5 configuration into a WebSphere Application Server - Express instance

Perform the following steps to restore the Version 3.5 configuration into a WebSphere Application Server - Express instance:

1. Enter the following command from the CL command line:

```
STRQSH
```

This starts the Qshell environment so that you can run WebSphere Application Server scripts.

2. Run the WasPostUpgrade script with the following parameters:

```
/QIBM/ProdData/WebASE51/ASE/bin/WASPostUpgrade
backupDirectoryName
[-instance 5_instance_name]
[-portBlock port_starting_number]
```

The first argument is required. Supported arguments include:

- **backupDirectory**
  Required name of the directory in which the WASPreUpgrade tool stores the saved configuration and files, and from which the WASPostUpgrade tool reads the configuration and files. The WASPreUpgrade tool creates this directory if it does not already exist.
- **-portBlock**
  This is an optional parameter. The value portblock specifies the first number of a block of port numbers that your instance uses. Specify the first port in a group of unused ports on your iSeries server. You can use the Work with TCP/IP Network Status (NETSTAT *CNN) command to display a list of port numbers that are currently being used. This parameter is case sensitive.

  Note: Although this is an optional parameter, it is recommended that you always specify the parameter (or the specific port parameters, described below) if you do not want your instance's ports to conflict with the default instance's ports.

For the -portBlock parameter, the script checks instances of WebSphere Application Server - Express. The script is not able to detect port usage by other applications, including previous versions of WebSphere Application Server.

For a full explanation of the WasPostUpgrade migration tool and parameters, see "The WASPostUpgrade script" on page 5.

Start the WebSphere Application Server - Express instance

Perform the following steps to start the WebSphere Application Server - Express instance:

1. Enter the following command from the CL command line to start the QASE51 subsystem if it is not already started:

   ```
   STRSBS QASE51/QASE51
   ```

2. Enter the following command from the CL command line:

   ```
   STRQSH
   ```

   This starts the Qshell environment so that you can run WebSphere Application Server scripts.

3. Run the startServer script with the following parameters:

   ```
   /QIBM/ProdData/WebASE51/ASE/bin/startServer
   -instance 50InstanceName 50ApplicationServerName
   ```

   where *50InstanceName* is the name of the WebSphere Application Server - Express instance created in an earlier step, and *50ApplicationServerName* is the name of the WebSphere Application Server - Express application server created in an earlier step.

Migrate from a WebSphere Application Server - Express Version 5.0.x instance

Tools for migrating administrative configurations are provided for WebSphere Application Server - Express Version 5.0.x to be upgraded to WebSphere Application Server - Express Version 5.1.

Before performing the instructions on this page, do the steps in "Step 1: Plan your migration" on page 7.

A summary of the product migration process is as follows:

1. Create a Version 5.1 instance to receive the WebSphere Application Server - Express Version 5.0.x configuration (page 17).
2. Save the WebSphere Application Server - Express Version 5.0.x configuration (page 17).
3. Restore the WebSphere Application Server - Express Version 5.0.x configuration into a Version 5.1 instance (page 18).
4. Start the Version 5.1 instance that receives the WebSphere Application Server - Express Version 5.0.x (page 18).

Create a Version 5.1 instance to receive the WebSphere Application Server - Express Version 5.0.x configuration

See Create a new application server in the *Administration* topic for details on how to create a WebSphere Application Server - Express instance. The name of your WebSphere Application Server - Express Version 5.1 instance must be identical to the Version 5 instance that is to be migrated.

Save the WebSphere Application Server - Express Version 5.0.x configuration

Perform the following steps to save the WebSphere Application Server - Express Version 5.0.x:

1. Enter the following command from the CL command line:

   ```
   STRQSH
   ```

   This starts the Qshell environment so that you can run WebSphere Application Server scripts.

2. Run the WasPreUpgrade script with the following parameters:

   Express Version 5.0.x:

   ```
   /QIBM/ProdData/WebASE51/ASE/bin/WASPreUpgrade
   myBackupDirectory
   /QIBM/UserData/WebASE/ASE5/5.xExpressInstanceName
   -instance 5.xExpressInstanceName
   ```

   where:

   - *myBackupDirectory*(required parameter) is the fully qualified path to the integrated file system directory where the WasPreUpgrade migration tool stores the saved configuration and files. The directory is created if it does not already exist. It is also the directory where the WasPreUpgrade migration tool writes a log file called WasPreUpgrade.log that chronicles the steps taken by WasPreUpgrade.
   - *5.xExpressInstanceName* (required parameter) is the name of the WebSphere Application Server - Express Version 5.1 instance to which you are migrating the configuration. This parameter must be identical to the Version 5.0.x instance that is to be migrated.

   For a full explanation of the WasPreUpgrade migration tool and parameters, see "The WASPreUpgrade script" on page 3.

Restore the WebSphere Application Server - Express Version 5.0.x configuration into a Version 5.1 instance

Perform the following steps to restore the WebSphere Application Server - Express Version 5.0.x configuration into a Version 5.1 instance:

1. Enter the following command from the CL command line:

   ```
   STRQSH
   ```

   This starts the Qshell environment so that you can run WebSphere Application Server scripts.

2. Run the WasPostUpgrade script with the following parameters:

   ```
   /QIBM/ProdData/WebASE51/ASE/bin/WASPostUpgrade
   myBackupDirectory
   -instance 51_instance_name
    [-portBlock port_starting_number]
   ```

   Supported arguments include:

   - **backupDirectoryName**
     where *myBackupDirectory* (required parameter) is the fully qualified path to the integrated file system directory where the the WasPreUpgrade migration tool had previously been used to save the WebSphere Application Server - Express Version 5.0.x instance configuration.
   - **-instance**
     This is a required parameter. The value *51_instance_name* specifies the name of the WebSphere Application Server - Express Version 5.1 instance to which the script migrates your configuration. This parameter must be identical to the Version 5 instance that is to be migrated.
   - **-portBlock**
     This is an optional parameter. The value *port_starting_number* specifies the first of a block of 10-15 consecutive port numbers that are not in use on the iSeries server where the migration is being performed. It is recommended that you always specify the -portBlock parameter if you do not want your instance's ports to conflict with the default instance's ports.

     For a full explanation of the WasPostUpgrade migration tool and parameters, see "The WASPostUpgrade script" on page 5.

Start the Version 5.1 instance that receives the WebSphere Application Server - Express Version 5.0.x

Perform the following steps to start the Version 5.1 instance that receives the WebSphere Application Server - Express Version 5.0.x:

1. Start the QASE51 subsystem if it is not already started. Enter this command from the CL command line:
   ```
   STRSBS QASE51/QASE51
   ```
2. Enter the following command from the CL command line:
   ```
   STRQSH
   ```
   This starts the Qshell environment so that you can run WebSphere Application Server scripts.
3. Run the startServer script with the following parameters:
   ```
   /QIBM/ProdData/WebASE51/ASE/bin/startServer -instance 51InstanceName
   ```
   where *51InstanceName* is the name of the Version 5.1 instance created in an earlier step.

Step 4: Complete the migration

This topic describes the manual steps that you might need to perform to complete the migration process. These are steps that are not performed automatically by the WebSphere Application Server - Express migration tools.

Enable the administrative console

The administrative console is disabled by default in Express instances, and must be enabled before it can be used. Even if you migrate from a 5.0.x instance where the administrative console is enabled, the console is not enabled automatically in the 5.1.x instance. For instructions on enabling the administrative console, see Start the WebSphere administrative console.

Note: If migrating a secured 5.0.x Express instance, the migrated 5.1.x instance will also have security enabled. Secure instances cannot be managed through the HTTP Server Administration interface, so to enable the administrative console in a secured instance, follow the instructions for starting the console without using the HTTP Server Administration interface.

Change the HTTP transport after migration
If the -portBlock parameter was not specified in the WASPostUpgrade script, you need to use the WebSphere Application Server - Express administrative console to adjust the ports manually after migration has completed."

1. Use the WebSphere Application Server - Express administrative console to change the HTTP transport port:
   a. Navigate to **Servers —> Application Servers —> *my_server*** where *my_server* is the name of the application server that was migrated.
   b. Click **Web Container**.
   c. Click **HTTP transports**.
   d. Click on a host.
   e. Adjust the **Port** property by entering a port not currently in use on your iSeries server. Record both the old and new port numbers; they are needed for step 2.
   f. Click **OK**.
   g. Repeat for each host.
2. Adjust your Virtual hosts to correspond to your HTTP transport.
   a. Click **Environment —> Virtual Hosts**.
   b. Select your virtual host.
   c. Click **Host Aliases**.
   d. Click **New**.
   e. Select one of the ports that were changed in step 1.
   f. Enter the port that this was changed to in step 1.

g. Click **OK**.
h. Repeat for each port that was changed in step 1.

3. Click **Save.**
4. Regenerate your plugin configuration.
   a. Navigate to **Environment —> Update Web Server Plugin**.
   b. Click **OK**.

Migrate the web server plug-in
If the WebSphere Application Server Version 3.5.x plug-in for your web server uses Open Servlet Engine (OSE) transport, you must switch to HTTP transport when migrating to WebSphere Application Server - Express.

The following instructions are specific to the Web server being supported and assume that you can successfully migrate existing Web applications:

Plug-in migration has been tested with the following Web server products:

- IBM HTTP Server (original) for iSeries (V4R5)
- IBM HTTP Server (powered by Apache) for iSeries (V5R1)
- Lotus Domino HTTP Server

Use the following steps to migrate the plug-in configuration:

1. **Configure an HTTP server instance**
   There are two options to chose from:
   - Create a new HTTP server instance to be used by the WebSphere Application Server - Express instance. This method allows both Websphere Version 3.5.x and WebSphere Application Server - Express instances to continue operating correctly.
   - Update the HTTP server instance configuration for the Websphere Version 3.5.x instance that is being migrated. This method changes the HTTP instance configuration to work with the WebSphere Application Server - Express instance and makes the Websphere Version 3.5.x instance no longer usable.

   For more information, see Start the HTTP Server Administration interface.
2. **Configure the virtual host for the WebSphere Application Server - Express instance.**
   This step ensures that both the host and HTTP transport port number exist in the virtual host list. For more information, see Manage virtual hosts for your application server in the *Administration* topic.
3. **Regenerate the Web server plugin configuration**
   This step needs to be done after any configuration changes have been made.

Change the ConnectionIOTimeOut properties for the Web container

Peformance changes to the WebSphere HTTP plug-ins may result in **InterruptedIOTimeout** exceptions while reading large requests, such as receiving file uploads to a servlet. Increase the **ConnectionIOTimeout** value in the Web Container of your application as follows:

1. In the topology tree, expand **Servers** and click **Application Servers**.
2. On the **Application Servers** page, click the name of the server that you want to modify.
3. On the server's detail page, click **Web Container**.
4. On the **Web Container** page, click **HTTP Transports**.
5. On the **HTTP Transports** page, click the transport that you want to modify.
6. Click **Custom properties**.
7. Click **New** and add a property named **ConnectionIOTimeout**. Set the property to the maximum time a servlet or JSP waits for a client to transmit request data.

8. After you add a virtual host alias, you must restart the application server. See Start and test your application server in the *Administration* topic for more information.

Appendix. 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 grant 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
U.S.A.
```

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 106-0032, 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
Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

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, IBM License Agreement for Machine Code, 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 measurements 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.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

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

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming Interface Information

This WebSphere Application Server - Express publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM i5/OS.

Trademarks

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

AIX
AIX 5L
e(logo)server
eServer
i5/OS
IBM
IBM (logo)
iSeries
pSeries
WebSphere
xSeries
zSeries

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the p <?Pub Caret?>ublications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED ″AS-IS″ AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Printed in USA