WebSphere® Process Server

Version 6.0

Securing your applications and their environment

Note
Before using this information, be sure to read the general information in "Notices" on page 13.

September 29 2005

This edition applies to version 6, release 0, of WebSphere Process Server (product number 5724-L01) and to all subsequent releases and modifications until otherwise indicated in new editions.

To send us your comments about this document, email doc-comments@us.ibm.com. We look forward to hearing from you.

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.

Contents

Securing applications and their environment

This section describes how to plan, administer and test the security of your applications and their environment in IBM WebSphere Process Server, Version 6.0.

These documents are supplemental to the core security information found in the WebSphere Application Server Information Center. You should refer to the WebSphere Application Server documentation for a more detailed discussion of securing your environment and applications.

WebSphere Process Server documentation PDFs

Security overview

The security of your data and processes is critical. WebSphere Process Server security is based on the WebSphere Application Server version 6.0 security. Please refer to the WebSphere Application Server Information Center for detailed information.

There are two aspects to securing an application. First is authentication, a user or a process that invokes an application must be authenticated. The second is access control – does the authenticated user have permission to perform the operation?

The remainder of this section details the security considerations at various stages of operation of the WebSphere Process Server.

Security considerations during installation

Complete these tasks to implement security before, during, and after installing WebSphere Process Server.

These tasks should be undertaken at installation time.

1. Secure your environment before installation.

   The commands required to install WebSphere Process Server with proper security will vary with operating system. For detailed information see the topic **Securing your environment before installation** in the WebSphere Application Server Information Center.
2. Prepare the operating system for installation of WebSphere Process Server.

   This step includes how to prepare the different operating systems for installation of WebSphere Process Server. For detailed information see the topic **Preparing the operating system for product installation** in the WebSphere Application Server Information Center.
3. Secure your environment after installation.

   This task provides information on how to protect password information after you install WebSphere Process Server. For detailed information see the topic **Securing your environment after installation** in the WebSphere Application Server Information Center.

When you have completed the installation security can be administered from the administrative console. For detailed information see the topic on **Administering security** in the WebSphere Application Server Information Center.

Creating basic authentication information at install time

When creating a new WebSphere Process Server profile you are prompted for a user name and password at various stages. Enter a valid user name and password each time you have the opportunity to do so. The service component architecture (SCA), BPEL and Common Event Infrastructure panels offer user name and password entry fields. The user names and passwords you supply are required when you enable security.

If you do not provide this information during installation, you must use the administrative console to enter a valid user name and password for each alias.

Creating WebSphere Process Server profiles with security

You can take steps to secure your environment when you create your WebSphere Process Server profile. Alternatively you can provide the same information on the administrative console after you create the profile.

When you create a WebSphere Process Server profile using the Profile Wizard graphical user interface, there are three panels where you can enter user names and passwords for security purposes. The three components of WebSphere Process Server which permit you to input these user names and passwords are: the service component architecture (SCA), BPEL and the Common Event Infrastructure.

These user names and passwords are used to create authentication aliases. The user names and passwords are required when you enable security. If you do not enter the user names and passwords when you create the profile, you can provide the same information using the administrative console, after you have created the profile.

More information on managing authentication aliases is provided in subsequent topics.

Setting up WebSphere Process Server security

Control the security of your WebSphere Process Server from the administrative console.

The steps required to set up security in WebSphere Process Server depend on the type of environment in which you are working. Instructions are provided for a standalone installation, with differences detailed for a Network Deployment installation.

Also discussed in this topic is setting the Lightweight Directory Access Protocol (LDAP) as your user registry.

Setting up security for a standalone WebSphere Process Server

Set up the security of a standalone installation of WebSphere Process Server by taking the following steps.

1. Start WebSphere Process Server.
2. Launch the administrative console.
3. Enable global security.

   Expand **Security**, click **Global security**, select the **Enable global security** check box.

4. Enforce Java 2 security.

   Ensure that the **Enforce Java 2 security** check box is selected.

5. Set the Lightweight Third Party Authentication (LTPA) as the authentication mechanism.

   Choose **Lightweight Third Party Authentication (LTPA)** from the **Active authentication mechanism** list. LTPA is the only authentication mechanism supported by WebSphere Process Server. For more details on configuring LTPA as the authentication mechanism see the **Configuring single signon** topic in the WebSphere Application Server Information Center.

6. Enter a password to be used for LTPA key storage.

   Expand **Authentication mechanisms** and select **LTPA**. In the **Password** field, enter a password and type the same password in the **Confirm password** field. This password is used for the LTPA key storage. Confirm your changes by clicking **Apply**.

7. Provide necessary parameters for the user registry.

User Registry	Action
Operating System	Under **User registries**, choose **Local OS**. On the **Local OS user registry** page provide a user name and password. **Note:** This user name is used as the identity of the server. The user is automatically added to the **Administrator** role.
Lightweight Directory Access Protocol (LDAP)	See "Configuring Lightweight Directory Access Protocol (LDAP) as the user registry" on page 4 for details.

8. Save your changes

   Click **OK**.

9. Restart the WebSphere Process Server.

Setting up security for a Network Deployment environment

Setting up security in a Network Deployment environment requires steps in addition to those required for a standalone version of WebSphere Process Server.

You must be running the administrative console on the machine which is acting as the deployment manager.

1. Enable global security.

   On the administrative console, expand **Security**, click **Global security**, select the **Enable global security** check box.

2. Enforce Java 2 security.

   Ensure that the **Enforce Java 2 security** check box is selected.

3. Set the Lightweight Third Party Authentication (LTPA) as the authentication mechanism.

   Choose **Lightweight Third Party Authentication (LTPA)** from the **Active authentication mechanism** list. LTPA is the only authentication mechanism which is supported in WebSphere Process Server.

4. Enter a password to be used for LTPA key storage.

Expand **Authentication mechanisms** and select **LTPA**. In the **Password** field, enter a password and type the same password in the **Confirm password** field. This password is used for the LTPA key storage. Confirm your changes by clicking **Apply**.

5. Provide necessary parameters for the user registry.

User Registry	Action
Operating System	Under **User registries**, choose **Local OS**. On the **Local OS user registry** page provide a user name and password. **Note:** This user name is used as the identity of the server. The user is automatically added to the **Administrator** role.
Lightweight Directory Access Protocol (LDAP)	See "Configuring Lightweight Directory Access Protocol (LDAP) as the user registry" for details.

6. Ensure that the security information is propagated to the nodes of the cell.

   Select the **Synchronize with Nodes** check box.

7. Save your changes

   Click **OK**.

8. Restart the deployment manager, the nodes and the WebSphere Process Servers.

Configuring Lightweight Directory Access Protocol (LDAP) as the user registry

By default the user registry is the local operating system registry. If you prefer, use an external Lightweight Directory Access Protocol (LDAP) as the user registry.

This task assumes that you have global security switched on.

1. Start WebSphere Process Server.
2. Launch the administrative console.
3. Bring up the LDAP User Registry configuration page.

   Expand **Security**, click **Global security**, click **LDAP** under the **User Registries** heading.

4. Set the user name and password used to run WebSphere Process Server for security purposes.

   In the **Server user ID** field enter the user name and in the **Server user password** enter the corresponding password. This ID is not the LDAP administrator user ID. The entry should exist in the LDAP.

5. Choose the type of LDAP you are using.

   From the **Type** list choose the specific LDAP that you wish to use as your user registry.

6. Enter the name of the machine where the LDAP resides.

   In the **Host** field enter the name of the server where the LDAP resides.

7. Enter the port number on which the LDAP listens.

   In the **Port** field enter the port number on which the LDAP server is listening.

8. Enter the **Base Distinguished Name**.

   Specifies the base distinguished name of the directory service, indicating the starting point for LDAP searches of the directory service.

For authorization purposes, this field is case sensitive. This specification implies that if a token is received (for example, from another cell or Domino) the base DN in the server must match the base DN from the other cell or Domino server exactly. If case sensitivity is not a consideration for authorization, enable the **Ignore case** field. This field is required for all Lightweight Directory Access Protocol (LDAP) directories except for the Domino Directory, where this field is optional.

9. Leave the remaining parameters with the default values and confirm your changes.

   Click **OK**.

Authentication

When security is turned on, clients must be authenticated.

If a client tries to access a secured application without being authenticated, an exception is thrown.

Web clients (for instance JSPs or servlets) can be setup for HTTP Basic authentication, so that when you reference the URL, the browser prompts for a user name and password.

Java clients should use JAAS for authentication.

WebServices clients can use Webservices/SOAP authentication.

Some of the components of the WebSphere® Process Server infrastructure have authentication aliases that are used to authenticate the runtime code for access to databases and the messaging engine. These BPEL and Common Event Infrastructure authentication aliases are outlined in subsequent topics. The WebSphere Process Server installer collects the user name and passwords to create these aliases.

Some runtime components have message driven beans (MDBs) that are configured with a runAs role. The WebSphere Process Server installer collects the user name and password for the runAs role.

Modifying authentication aliases

You might need to modify existing authentication aliases.

Modify authentication aliases from the administrative console.

1. Access the J2EE Connector Architecture (J2C) authentication data entries panel.

   From the administrative console, expand **Security**, click **Global security**. Under the Authentication heading, expand **JAAS Configuration**, and select **J2C Authentication data**.

   Note: This panel can also be used to add or delete authentication data.

2. Select the authentication alias that you want to modify.

   The J2EE Connector Architecture (J2C) authentication data entries panel contains a list of authentication aliases, the user ID associated with this alias and, optionally, a description of the alias. Click on the alias that you want to modify.

3. Change the properties of the alias.

On the configuration panel for the selected alias, you can modify the alias name, or the associated user ID and password. You can also modify the description of the authentication data entry.

4. Confirm your changes.

   Click **OK**, or **Apply**. Return to the J2EE Connector Architecture (J2C) authentication data entries panel and click **Save** to apply your changes to the master configuration.

   Note: For a Network Deployment installation, make sure that a file synchronized operation is performed to propagate the changes to other nodes.

BPEL authentication aliases

BPEL has the following authentication aliases. Modify these aliases using the administrative console.

Alias	Description	Information
BPEAuthDataAliasJMS_*node_server*	Used to authenticate with the messaging engine.	User name and password are entered on the BPEL configuration panel of the Profile Wizard.
BPEAuthDataAlias*DbType_node_server*	Used to authenticate with databases.	Configure the database using the BPEL-provided scripts.

runAs role	Description	Information
JMSAPIUser	Used by the BFM JMS API MDB in bpecontainer.ear.	Enter user name and password on the BPEL configuration panel of the Profile Wizard.
EscalationUser	Used by the task.ear MDB.	Enter user name and password on the BPEL configuration panel of the Profile Wizard.

Common Event Infrastructure authentication aliases

The Common Event Infrastructure has the following authentication aliases. Modify these aliases using the administrative console.

Alias	Description	Information
CommonEventInfrastructureJMSAuthAlias	Used to authenticate with the messaging engine.	Enter user name and password on the Common Event Infrastructure configuration panel of the Profile Wizard.
EventAuthAlias*DBType*	Used to authenticate with databases.	Enter user name and password on the Common Event Infrastructure configuration panel of the Profile Wizard.

Service component architecture authentication aliases

The service component architecture (SCA) has the following authentication aliases. Modify these aliases using the administrative console.

Alias	Description	Information
SCA_Auth_Alias	Used to authenticate with the messaging engine.	Enter user name and password on the SCA configuration panel of the Profile Wizard.

Access control

Access control refers to ensuring that an authenticated user has the permissions necessary to perform a specific operation.

When a user has authenticated themselves to the WebSphere Process Server, it is important for security that not every possible operation is available to the user. Allowing some users to perform certain tasks, while denying these tasks to other users is termed access control.

Access control can be arranged for components that you develop to make them secure. This is achieved using service component architecture qualifiers at development time. See the WebSphere Integration Developer Information Center for more information.

Some WebSphere Process Server components, packaged as ear files, secure their operation using J2EE role-based security. Details of these components are provided. BPEL and the Common Event Infrastructure are installed as part of the WebSphere Process Server. The role-based security associated with these components is outlined in detail in subsequent topics.

Developing secure components

Secure the components that you develop. Components implement interfaces that have methods. Use the service component architecture (SCA) qualifier SecurityPermission to secure an interface or method.

Develop your secured application in WebSphere Integration Developer. Export the application as an ear file for deployment in WebSphere Process Server.

Import a secured application into WebSphere Process Server with the following steps.

1. Install the application ear file.

   On the administrative console, expand **Applications** and click **Enterprise applications**. Click **Install** and fill in the details of the new application.
2. Assign security roles to the new application.

   Click **Map security roles to users/groups**. You have four choices of roles for the application.

Option	Description
Everyone	This is equivalent to no security.

Option	Description
All authenticated	Anyone who authenticates with a valid user name and password is a member of the role.
Mapped users	Individual users are listed as members of the role.
Mapped groups	Groups are the most convenient way to add the users, every member of the identified groups becomes a member of the role.

Use **Look up users** and **Look up groups** to list users and groups that can be mapped to the role.

In the sample SCDL below, access to the method **onwayinvoke** is restricted to users that are members of the **manager** role.

```
<?xml version="1.0" encoding="UTF-8"?>
<scdl:component xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:java="http://www.ibm.com/xmlns/prod/websphere/scdl/java/6.0.0"
xmlns:ns1="http://sample.recovery.security/Itarget"
xmlns:scdl="http://www.ibm.com/xmlns/prod/websphere/scdl/6.0.0"
xmlns:wsdl="http://www.ibm.com/xmlns/prod/websphere/scdl/wsdl/6.0.0"
displayName="secure" name="Component1">
  <interfaces>
    <interface xsi:type="wsdl:WSDLPortType" portType="ns1:Itarget">
      <method name="onewayinvoke">
       <scdl:interfaceQualifier xsi:type="scdl:SecurityPermission"
role="manager"/>
      </method>
      </interface>
  </interfaces>
  <references/>
  <implementation xsi:type="java:JavaImplementation"
class="sca.component.java.impl.Component1Impl1">
 </implementation>
</scdl:component>
```

Access control in BPEL applications

The following ear files are installed with access control as part of the BPEL installation. BPEL is installed as part of the WebSphere Process Server installation.

ear file	Roles	Default permission	Notes
bpecontainer.ear	BPESystemAdministrator	Group name entered during install.	Has access to all business processes and all operations.
bpecontainer.ear	BPESystemMonitor	All authenticated users.	Has access to read operations.
task.ear	TaskSystemAdministrator	Group name entered during install.	Has access to all human tasks.
task.ear	TaskSystemMonitor	All authenticated users.	Has access to read operations.
Bpcexplorer.ear	WebClientUser	All authenticated users.	Can access the BPCExplorer.

Access control in Common Event Infrastructure applications

The following ear file is installed with access control as part of the Common Event Infrastructure installation. The Common Event Infrastructure is installed as part of the WebSphere Process Server installation.

The EventServer.ear file is the only enterprise archive file installed as part of the Common Event Infrastructure installation.

Roles	Default permission
eventAdministrator	All authenticated users.
eventConsumer	All authenticated users.
eventUpdater	All authenticated users.
eventCreator	All authenticated users.
catalogAdministrator	All authenticated users.
catalogReader	All authenticated users.

Securing adapters

Two types of adapter are supported in WebSphere Process Server: WebSphere Business Integration Adapters and WebSphere Adapters. The security of both types of adapter is discussed..

Adapters are the mechanism by which applications communicate with Enterprise Information Systems (EISs). The information that is exchanged between an application and an EIS may be highly sensitive. It is important to ensure the security of this information transaction.

WebSphere Business Integration Adapters consist of a collection of software, application program interfaces (APIs) and tools that enable applications to exchange business data through an integration broker. WebSphere Business Integration Adapters rely on JMS messaging and JMS does not support security context propagation.

WebSphere Adapters enable managed, bidirectional connectivity between Enterprise Information Systems (EISs) and J2EE components supported by WebSphere Process Server.

For inbound communication from adapters into WebSphere Process Server, there is no authentication mechanism. For WebSphere Business Integration Adapters the reliance on JMS messaging precludes security context propagation. J2C also lacks inbound security support, so WebSphere Adapters also have no authentication mechanism for inbound communication.

The entry from an adapter to WebSphere Process Server always employs a service component architecture (SCA) export. The SCA export has to be wired to an SCA component, such as mediation, BPEL, SCA Java component or Selector.

The security solution is to define a `runAs` role on the component that is the target for the WebSphere Adapter export. This is done using the SCA qualifier **SecurityIdentity** during development (see the WebSphere Integration Developer Information Center for more information). When the component executes, it does so under the identity defined in the **runAs** role.

Security in Business Process Choreographer tasks and processes

There are a number of roles associated with human tasks and business processes. This topic describes the roles available.

Human tasks, by definition, require human intervention to complete them. Some business processes may also require human intervention. These human tasks and business processes are developed using WebSphere Integration Developer and are executed using Business Process Choreographer. When you develop the task or process, you must assign roles to users or groups involved in the human tasks and business processes. See the WebSphere Integration Developer Information Center for more information about assigning the roles or querying the roles associated with specific roles.

The human task manager uses the roles to determine the users' capabilities on a production system.

Roles associated with tasks and processes

Important: These roles are unique to tasks and processes that are running in the Business Process Choreographer business container and human task container.

WebSphere Process Server supports the following roles for tasks and processes:

Administrator
Users who belong to this role can monitor, end, or delete tasks and processes and also display information about tasks and processes.

Reader
Users who belong to this role can only display tasks and processes.

Starter
Users who belong to this role can start or display tasks and processes.

Tasks also have these additional roles:

Owner
Users who belong to this role can save, cancel, complete or display tasks that they have already claimed.

Potential owner
Users who belong to this role can claim and display tasks.

Network Deployment security considerations

In an Network Deployment of WebSphere Process Server there are additional security considerations.

The most important consideration is the user registry, since the user must have access to all machines in the network.

1. If the operating system is the user registry:
   a. The user registry must be a domain or network user registry.
   b. All usernames should be valid domain or network users.
2. A Windows domain user registry works with a Network Deployment.
3. In a UNIX or mixed UNIX and Windows environment, it is preferable to use the Lightweight Directory Access Protocol (LDAP) as the user registry.

Tutorial: Writing a Jacl script that lists security roles

This tutorial addresses how to write and execute a simple Jacl script that can access and manage a JMX MBean. This particular script is concerned with calling roles when global security is enables. Using this script, you will be able to print out the role name for each role in a relationship.

Objective of this tutorial

After completing this tutorial, you will be able to:

- Write a Jacl script that calls a JMX MBean requesting a list of all relationships.

For more information about writing scripts, refer to ″Using scripting (wsadmin)″ in the WebSphere Application Server Network Deployment, version 6.0 information center.

Time required to complete this tutorial

This tutorial requires approximately 15-30 minutes to complete.

Prerequisites

This tutorial uses a script that is included with the JMX Security sample. This sample demonstrates the MBean function of printing out a list of role relationships.

Note: To use this script, you must select the option to install code samples during the installation of WebSphere Process Server.

The location of the sample Jacl script is in <wbi_root>/samples/JMXSample/scripts. The name of the script is: `RelServicesAdmin.jacl`.

To run the script, enter:

```
wsadmin -f ../samples/JMXSample/scripts/RelServicesAdmin.jacl
 -server servername -node nodename
```

This script will call up to 10 relationships in your environment and up to 10 roles for each relationship will be printed on the console.

Go to “Exercise: Writing a Jacl script” to learn how to write this `.jacl` script.

Exercise: Writing a Jacl script

The basic concepts in this script can be used to communicate with any MBean in the system. All that is required is the name and type of the MBean and the methods and attributes available on the MBean. The getAttribute and setAttribute commands are used for attributes. The invoke command is used for methods. Follow these steps to create a `.Jacl` script that manages the JMX Security MBean.

Note: The code in each step is prefaced with a statement explaining what the code does.

1. Determine the **nodename**

   The first part of the script shown below determines the nodename. If the `nodeName` is not specified correctly, the correct syntax is printed and the script exits.

```
# read and validate arguments

    if {{$argc == 1 } && { [lindex $argv $i] == "-nodeName" } {
            set nodeName [lindex $argv $i]
```

2. Identify the **MBean**

   An MBean is identified by a type and a name.

   Note: The name and type are hard coded in this case since you know the specific MBean you want to use.

   The second part of the script identifies the MBean.

```
# these two variables, mbeanName and mbeanType are used
  to uniquely identify the mbean.
       # for this sample, the mbean that access relationship
         services will be used.

       set mbeanName"RelService"
       set mbeanType"WBIRelServices"
```

3. Locate and set the **reference** to the MBean.

   You use the code shown here to set the reference for the MBean.

```
# locate the mbean and set a reference to in in "relSvcsMBean" variable

       set relSvcsMBean [$AdminControl queryNames
           name=$mbeanName,node=$nodeName,type=$mbeanType,*]
```

4. Call the **relationship** using the getAttribute command.

   The documentation of this specific MBean defines an attribute named `allRelationshipNames`. Ask the MBean for that attribute using the getAttribute command. The attribute value will be a list that you step through in the next step that invokes the command.

```
# request the list of relationships from the mbean

       set relationships
        [$AdminControl getAttribute $relSvcsMBean allRelationshipNames]
```

5. Invoke the **command** for each relationship name, you print the name, and then go back to the MBean for additional information.

   For each relationship name, you print the name, and then go back to the MBean for additional information. In this example the MBean defines a method named getAllRoleNames with a single parameter for the specific relationship name. You use the invoke command to call this method, passing the current relationship name. For each role in the relationship, a role name is printed.

```
# loop through the list of role names and print name

         foreach roleName $roles {
           puts "    Role: $roleName"
         }
       }
    } else {
         # arguments were not correct, print correct syntax
            puts "Usage: wsadmin -f RelServicesAdmin.jacl -nodeName nodeName"
    }
```

You have now written a script to call relationships.

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
577 Airport Blvd., Suite 800
Burlingame, CA 94010
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 document 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 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, a nd represent goals and objectives only.

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.

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

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

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

i5/OS
IBM
the IBM logo
AIX
AIX 5L
CICS
CrossWorlds
DB2
DB2 Universal Database
Domino
HelpNow
IMS
Informix
iSeries
Lotus
Lotus Notes
MQIntegrator
MQSeries
MVS
Notes
OS/390
OS/400
Passport Advantage
pSeries
Redbooks
SupportPac
WebSphere
z/OS

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. 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.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

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

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

This product includes software developed by the Eclipse Project (http://www.eclipse.org/).

Your product name with version, e.g., WebSphere Process Server, Version 6.0

Printed in USA