Version 2 Release 1

IBM i2 Enterprise Insight Analysis
Maintaining a deployment

Note

Before using this information and the product it supports, read the information in "Notices" on page 13.

This edition applies to version 2, release 2, modification 0 of IBM® i2® Enterprise Insight Analysis (product number 5725-G23) and to all subsequent releases and modifications until otherwise indicated in new editions. Ensure that you are reading the appropriate document for the version of the product that you are using. To find a specific version of this document, access the Configuring section of the IBM Knowledge Center, and ensure that you select the correct version.

Contents

Maintaining an Enterprise Insight Analysis deployment

After you take a deployment of i2 Enterprise Insight Analysis out of development and put it into production, there are maintenance tasks to perform. For example, you might need to store new types of data, change the security settings, or examine the health of the deployment.

Modifying the schema and the security schema

Ideally, after the development process, your schema and security schema are as close as possible to their final form before you go live with a deployment of i2 Analyze. However, things do change over the life of a deployment, and some modifications that do not compromise the integrity of the data in the system are possible.

Modifying the i2 Analyze schema

i2 Analyze supports a limited set of changes to the schema of a production deployment. The changes that you can make, and the procedure for making them, vary slightly depending on which i2 data stores and services are in the deployment.

About this task

After you put a deployment of i2 Analyze into production, you can generally make additive changes to the schema, but not destructive ones. For example, you can add new item types, and add new property types to existing item types, but you cannot remove types from the schema. Removing item types or property types after production can result in stores with data that is not valid according to the schema. i2 Analyze has controls in place that prevent this situation from occurring.

Procedure

The following steps describe how to make small changes to the schema in a production deployment of i2 Analyze.

1. If the deployment contains only the Analysis Repository, then you can use Schema Designer to modify the schema. You do not need to follow any of the other steps in this procedure.

   Instead, see the *IBM i2 Analysis Repository Tools* documentation.

The Information Store restricts the changes that you can make to the schema slightly more than the Analysis Repository does. As a result, if your deployment contains both data stores, you must address the Information Store first.

Furthermore, the functionality that enables the deployment toolkit to manage the schema across the Information Store and the Analysis Repository is in the Opal services. Although you cannot use the Information Store with both the Onyx and Opal services, the Opal services must be present to update the Information Store schema.

2. If your i2 Analyze deployment contains the Information Store with only the Onyx services, then you must add the Opal services before you continue:
   a) On the i2 Analyze server, open a command prompt and navigate to the `toolkit\scripts` directory.
   b) Run the `addInformationStore` task to add the Opal server to the deployment:

```
setup -t addInformationStore
```

You are now ready to modify the schema.

3. Locate the XML file that contains the schema for the i2 Analyze deployment, and load it into Schema Designer.
4. Make your changes to the schema and the associated charting schemes, and then save the file.

   Note: In this mode, Schema Designer does not validate whether your changes are compatible with the deployed schema. Validation takes place when you apply the changes to your deployment.
5. At the command prompt, run the `deploy` task to update the Information Store with the modified schema:

   ```
   setup -t deploy -s opal-server
   ```

   The command recognizes that you modified the schema, determines whether the changes are valid for a running Information Store, and then applies them to the store. If the changes are not valid, the command displays messages to explain the problems.

   Note: If you customized the Information Store creation process by specifying `createdatabase="false"` in the topology file and running the scripts yourself, this command works in the same way. Execution stops so that you can customize the changes to the Information Store. After you apply the changes, you can run the task again to complete the process.
6. If your deployment of i2 Analyze includes the Analysis Repository, run the `deploy` task again to update it with the modified schema as well:

   ```
   setup -t deploy -s onyx-server
   ```

   If the command to update the Information Store ran successfully, this command will succeed too. The set of valid changes for the Information Store is a subset of the valid changes for the Analysis Repository. For full information, see *Permitted schema changes*.
7. Run the `start` task to restart the services that you were using before you modified the schema. That is, run one or both of the following commands to restart the Onyx and Opal servers respectively:

   ```
   setup -t start -s onyx-server
   setup -t start -s opal-server
   ```

   Note: If you added the Opal server only to update the Information Store with the modified schema, do not start the opal server.

What to do next

Because you can make only additive changes to a schema that you modify through this procedure, it is not mandatory to change other parts of your deployment. However, to take full advantage of your additions, consider the following complementary changes.

- If your deployment uses the Opal services and you want users to see the new types in quick search filters, edit the configuration file that controls them. See *Creating and configuring facets*.
- If your deployment uses the Onyx services and you want users to be able to analyze data that has your new types through Cognos, edit the reports and mapping files. See *Configuring Cognos*.
- To enable the Information Store to ingest data for the new item types and property types, modify your ingestion artifacts. See *Ingesting data into an Information Store*.

Permitted i2 Analyze schema changes

After you create a schema and use it in a deployment, i2 Analyze restricts subsequent changes to that schema to ensure that data in the system remains accessible. In general, you can add content to a published i2 Analyze schema, but you cannot take content away.

To be specific, i2 Analyze prevents changes to a schema that might invalidate data that is already stored. For example, if you remove an item type from the schema, then a store might contain data for which i2 Analyze no longer has a definition. However, you can make additive changes to the schema. You can also disable or hide item or property types in some parts of the system, if you are certain that they are no longer required.

Permitted changes for all i2 data stores

You can make the following changes to the i2 Analyze schema of a live deployment, no matter which i2 data stores the deployment includes.

- Add an item type
- Change the display name of an item type
- Change the icon of an item type
- Add a property type
- Change the display name of a property type
- Change the display order of a property type
- Increase the value length of a property type
- Make a property type non-mandatory
- Add a grade type
- Add a labeling scheme

Permitted changes that do not affect the Information Store

You can make the following changes to the i2 Analyze schema of a live deployment. However, they do not affect the Information Store, which means that users still see items with hidden types when they run a quick search, for example.

- Add a property group type
- Add a link constraint
- Add a link strength
- Disable an item type
- Hide an item type

The disable and hide functions change how item and property types behave in the deployment without making any destructive changes. Use these features with caution, because they can affect the behavior of visual query and import operations. They can also affect the work of Analysis Repository users by making data unavailable.

Permitted changes when a deployment contains only the Analysis Repository

If your deployment of i2 Analyze does not include the Information Store, then you can additionally make the following changes to the schema.

- Decrease the value length of a property type
- Change the data type of a property type to a string

If you attempt these changes on a deployment that includes the Information Store, the redeployment command fails.

Prevented changes for all i2 data stores

i2 Analyze prevents all of the following schema changes from taking place against a live deployment.

- Change the schema identifier
- Remove an item type
- Remove an entity type from the permitted list for a link type
- Remove a property group type
- Remove a property type
- Make a property type mandatory
- Remove a default property value
- Remove a property value from a selected-from or suggested-from list
- Remove a grade type
- Remove a link strength

To protect your data when you redeploy with a modified schema, i2 Analyze carries out validation checks to ensure that the changes you made do not result in data loss.

Modifying the security schema

After you put your deployment of i2 Analyze into production, there are restrictions on the changes that you can make to the security schema. You are relatively free to create security groups and permissions, but you can only add security dimension values to the security dimensions that were present when you went live.

About this task

Important: To change the schema in the way that this documentation describes, your deployment must use the supplied `WebSphereDynamicAccessRoleBasedPrincipalProvider` security implementation.

Modifying security dimensions

There are few changes that you can make to the security dimensions of an i2 Analyze deployment that do not also require you to clear the data from the system. You can add dimension values to security dimensions, and you can make cosmetic changes to dimensions and values.

Before you begin

Before you modify a security schema, you must understand its structure, including the element definitions and their attributes.

It is good practice to make a copy of the deployed security schema file as a backup. By default, the file is in the following directory: `toolkit\configuration\fragments\common\WEB-INF\classes\`. The name of the security schema is specified in the `DynamicSecuritySchemaResource` property of the `ApolloServerSettingsMandatory.properties` file in the same directory.

About this task

The following table shows the changes that you can make to the security dimensions in a deployed security schema without clearing data from the system. It also states whether a reindex is required if the change takes place:

Change	XML elements or attributes	Allowed	Reindex required
Add a security dimension	`<Dimension>`	No	N/A
Modify an existing security dimension	`DisplayName`, `Description`	Yes	No
Remove an existing security dimension	`<Dimension>`	No	N/A
Add a dimension value to a security dimension	`<DimensionValue>`	Yes	Yes
Reorder the dimension values in an ordered dimension	`<DimensionValue>`	Yes	Yes
Modify an existing dimension value	`DisplayName`, `Description`	Yes	No
Remove an existing dimension value from a security dimension	`<DimensionValue>`	No	N/A

To add a security dimension value to a security dimension, add a `<DimensionValue>` element as a child of an existing `<Dimension>` element.

To modify the display name or description of a dimension or a dimension value, change the `DisplayName` or `Description` attributes an existing `<Dimension>` or `<DimensionValue>` element. You must not change the value of the `Id` attribute.

Procedure

1. Using an XML editor, open the security schema for the deployment.

   The security schema is in the `toolkit\configuration\fragments\common\WEB-INF\classes\` directory. The name of the security schema is specified in the `DynamicSecuritySchemaResource` property of the `ApolloServerSettingsMandatory.properties` file in the same directory.
2. Modify the security dimensions in the security schema according to your requirements.
3. Increment the version number that is stated in the `Version` attribute of the `<SecurityDimensions>` element in the security schema.
4. Check your updated schema to ensure that it remains possible for all users to get an access level that is not "none" for at least one value in every access dimension.
5. Save and close the file.
6. If you added a dimension value, clear the search index:

   a) Open a command prompt and navigate to the `toolkit\scripts` directory.

   b) Run the following command:

   ```
   setup -t clearSearchIndex
   ```
7. Redeploy i2 Analyze.

What to do next

When a user (or the system) creates an item in the Analysis Repository, i2 Analyze applies a default set of dimensions. During deployment, you can specify what these default dimensions are. For more information, see *Setting default dimension values*.

Modifying security permissions

It is possible to change the mapping between user groups and the security permissions that the security schema defines without reimporting or reindexing your data. You must take care to ensure that all i2 Analyze users retain the ability to access your deployment.

Before you begin

Before you modify a security schema, you must understand its structure, including the element definitions and their attributes.

It is good practice to make a copy of the deployed security schema file as a backup. By default, the file is in the following directory: `i2analyze\deploy\wlp\usr\servers\onyx-server\apps\onyx-services-ar.war\WEB-INF\classes`. The name of the security schema is specified in the `DynamicSecuritySchemaResource` property of the `ApolloServerSettingsMandatory.properties` file in the same directory.

About this task

The following table shows the changes that you can make to the security permissions in a deployed security schema without clearing data from the system:

Change	XML elements or attributes	Allowed	Reindex required
Add a security group	`<GroupPermissions>`	Yes	No
Modify an existing security group	`UserGroup`	Yes	No
Remove an existing security group	`<GroupPermissions>`	Yes	No
Add security dimensions to a security group	`<Permissions>`	Yes	No
Remove security dimension from a security group	`<Permissions>`	Yes	No
Add security permissions from a security dimension for a security group	`<Permission>`	Yes	No
Modify existing security level from a security dimension permission for a security group	`DimensionValue`, `Level`	Yes	No
Remove existing security permissions from a security dimension permissions element for a security group	`<Permission>`	Yes	No

If the requirements for security groups change, you can modify the `<GroupPermissions>` element and its children.

- To add a group, insert a complete `<GroupPermissions>` element. To use the new group, you must ensure that the user repository contains a group that matches the value of the `UserGroup` attribute.
- To modify the name that is associated with a group, change the value of the `UserGroup` attribute.
- To remove a group, remove the `<GroupPermissions>` element for that group.

If the requirements for the permissions of a security group change, you can add or remove `<Permissions>` elements, and add, modify, and remove child `<Permission>` elements.

- To change the dimensions that a group has permissions for, you can add or remove `<Permissions>` elements as follows:

  - To add a dimension that the group has permissions for, insert a `<Permissions>` element where the value of the `Id` attribute matches the value of the `Id` attribute of the dimension.
  - To remove a dimension that the group has permissions for, remove the `<Permissions>` element where the value of the `Id` attribute matches the value of the `Id` attribute of the dimension.
- To change the security permissions that a group has within a dimension, you can add, modify, and remove `<Permission>` elements as follows:
  - To add a permission to a group, insert a `<Permission>` element. The `DimensionValue` attribute must match a dimension value in the same dimension that is defined in the `Dimension` attribute of the parent `<Permissions>` element.
  - To modify the current permission that a group has in a dimension value, set the `Level` attribute to a different value.
  - To modify the dimension value that a permission is for, set the `DimensionValue` attribute to a different value.
  - To remove the current permission that a group has in dimension value, remove the `<Permission>` element in which the `DimensionValue` attribute matches that dimension value.

Procedure

1. Using an XML editor, open the security schema for the deployment.

   The security schema is in the `toolkit\configuration\fragments\common\WEB-INF\classes\` directory. The name of the security schema is specified in the `DynamicSecuritySchemaResource` property of the `ApolloServerSettingsMandatory.properties` file in the same directory.
2. Modify the security permissions in the security schema according to your requirements.
3. Increment the version number that is stated in the `Version` attribute of the `<SecurityDimensions>` element in the security schema.
4. Check your updated schema to ensure that it remains possible for all users to get an access level that is not "none" for at least one value in every access dimension.
5. Save and close the file.
6. Redeploy i2 Analyze.

What to do next

When a user (or the system) creates an item in the Analysis Repository, i2 Analyze applies a default set of dimensions. During deployment, you can specify what these default dimensions are. For more information, see *Setting default dimension values*.

Administering user access

i2 Analyze provides features that enable you to change the behavior of the platform as users experience it. You can change which security groups each user is a member of, and you can configure the behavior of logout functionality in the Intelligence Portal.

Configuring i2 Analyze users in WebSphere Application Server Liberty

To give security permissions in i2 Analyze to a new user, or to change the security groups that a user is a member of, you can modify the WebSphere Application Server Liberty user registry.

About this task

WebSphere Application Server Liberty supports several approaches to user security. This procedure assumes that a basic user registry is in effect, which means that the server stores user configuration information in the `user.registry.xml` file.

Important: If your i2 Analyze deployment uses SPNEGO single sign-on, as described in *Configuring SPNEGO single sign-on*, then the users and groups are managed in Microsoft Active Directory. If your deployment uses any other security approach, follow the instructions for managing users in that approach.

Procedure

1. Open a command prompt and run the following command to stop the application server:

   ```
   setup -t stop
   ```

2. Use the WebSphere® Application Server Liberty `securityUtility` command to encode the password for any new users.

   a) Navigate to the `bin` directory of the WebSphere Application Server Liberty deployment that the deployment toolkit configured.

   b) Run the following command:

   ```
   securityUtility encode password
   ```

   The command displays the encoded password. Record it, including the `{xor}` prefix.

   Note: For more information about the security utility, see the WebSphere Application Server Liberty documentation.

3. In an XML editor, open the `user.registry.xml` file, which you can find in the `IBM\i2analyze\deploy\wlp\usr\shared\config` directory of your WebSphere Application Server Liberty installation.

4. Add any new users to the user registry by inserting new `<user>` elements as children of the `<basicRegistry>` element.

   Use the encoded password, including the `{xor}` prefix.

   For example:

   ```
   <user name="Jenny" password="{xor}FToxMSY="/>
   ```

5. To add a user to a group, so that they receive the security permissions that are assigned to that group, add a `<member>` element as a child of the appropriate `<group>` element.

   The `name` attribute of the `<member>` element must match the `name` attribute of the `<user>` element for that user in the file.

For example:

```
<user name="Jenny" password="{xor}FToxMSY="/>

<group name="Clerks">
    <member name="Jenny"/>
    <member name="Clerk1"/>
</group>
```

6. To remove a user from a group, so that they no longer receive the security permissions that are assigned to that group, delete the `<member>` element for the user from the relevant `<group>` element.

   Note: Every user in the registry must be a member of groups such that they receive a dimension value and a level from each access security dimension in the security schema. The same consideration does not apply to grant security dimensions.
7. Save and close the `user.registry.xml` file.
8. To start the application server, at the command prompt, navigate to `toolkit\scripts`. Then, run the following command:

   ```
   setup -t start
   ```
9. Start, or restart, the IBM HTTP Server that hosts the reverse proxy.

Configuring user logout and idle logout

By default, users remain logged in to i2 Analyze through the Intelligence Portal for as long as their session remains open. You can enable users to log out of the Intelligence Portal manually, or arrange for the system to log them out after a configurable idle period.

About this task

The settings that you must change to enable user logout or idle logout are in the `ApolloClientSettings.xml` file that can find in the `toolkit\congfiguration\fragments\onyx-services-ar` directory.

Procedure

1. Use an XML editor to open the `ApolloClientSettings.xml` file.
2. To enable users to log out, change the contents of the `<EnableLogout>` element to `true`.
   With this setting, users can log out of the Intelligence Portal by clicking **Log out**.
3. To enable idle logout, change the contents of the `<IdleLogoutMinutes>` element to an integer that is greater than zero.
   With this setting, users who are idle for the specified number of minutes are automatically logged out.

A warning can be displayed in the Intelligence Portal to inform a user before they are automatically logged out.

4. To define the time in minutes that the user is warned before they are logged out, change the contents of the `<AlertBeforeIdleLogoutMinutes>` element. Set the value to an integer that is greater than zero but less than the value of `<IdleLogoutMinutes>`.
5. Save and close the file.

What to do next
You must redeploy i2 Analyze for the changes to take effect.

Troubleshooting the deployment process

i2 Analyze provides mechanisms that assist with deployment before, while, and after you do it. You can validate the deployment toolkit before you begin, read status messages during the process, and read the log files afterward.

Deployment toolkit validation messages

The i2 Analyze deployment toolkit validates some of the configuration files that are required for a deployment. By reading the validation messages, you can be certain that required properties have values, that property values have the correct format, and that required files are in the correct locations.

Validation takes place when you run the `setup -t deploy` command. Validation can complete in the following ways:

Success

On success, the validation process displays a brief configuration summary, and the main deployment process continues. For example:

```
============== Configuration Summary (brief) ==============
 + configuration/fragments/common/WEB-INF/classes/ApolloServerSettings
Mandatory.properties:
   - ChartingSchemesResource='law-enforcement-schema-charting-schemes.xml'
   - SchemaResource='law-enforcement-schema.xml'
   - DynamicSecuritySchemaResource='security-schema.xml'
============================================================
```

Warning

When there is a warning, the validation process displays a brief configuration summary and a `WARNINGS` section. The `WARNINGS` section identifies settings that might not be configured correctly, but the deployment process continues. For example:

```
====================== WARNINGS (1) ======================
 + configuration/fragments/common/WEB-INF/classes/ApolloServerSettings
Mandatory.properties:
   - The <SchemaResource> property has not been set.
============================================================
```

Here, the schema is not set, and so the default law enforcement schema is used.

Error

If an error occurs, the validation process displays a longer configuration summary, and an `ERRORS` section. The `ERRORS` section identifies missing values that must be present. The deployment process stops, and you must correct the errors before you attempt to deploy again. For example:

```
====================== ERRORS (1) ======================
 + configuration/environment/opal-server/environment.properties:
   - "db.database.location.dir.db2" has not been set
============================================================
```

Here, the DB2 directory is not set, so the database cannot be configured.

Deployment progress messages

During the deployment process, i2 Analyze displays detailed messages that provide information about the state and configuration of the system.

The output from the deployment process describes each task that the `setup` command performs during deployment. If a task runs successfully, then only its name appears. For example:

```
:buildApplication
```

.

There are two reasons why a task might not run, but deployment can still proceed.

`UP-TO-DATE`

The task was performed earlier, or its output is already present. For example:

```
:installJDBCDrivers UP-TO-DATE
```

.

`SKIPPED`

The task is not required for this deployment. For example:

```
:importLTPAKey SKIPPED
```

.

If an error occurs, deployment stops in a controlled manner. i2 Analyze displays a stack trace that contains the name of the task that failed, and information about the location of the error. For example:

```
:createDatabasesIfNecessary FAILED

FAILURE: Build failed with an exception.

* Where:
Script 'C:\IBM\i2analyze\toolkit\scripts\gradle\database.gradle' line: 173

* What went wrong:
Execution failed for task ':createDatabasesIfNecessary'.
```

The messages are displayed on screen and sent to the log files that are stored in the `toolkit \configuration\logs` directory.

Deployment log files

i2 Analyze produces logging information about deployment tasks and the transactions that take place when the system is operational. After deployment, you can check these logs to help diagnose potential issues.

The following types of logged information are available for you to review:

Deployment logging

Each time that you run the `setup` command, a log file is created in the deployment toolkit. The messages in these logs describe which tasks were called, whether the tasks completed successfully, and the details of any issues that occurred.

These log files contain the same information as the console output. You can find them in the `toolkit\configuration\logs` directory.

WebSphere Application Server Liberty profile logging

In addition to the information in the i2 Analyze logs, extra information that relates to the application server is also logged in `C:\IBM\i2analyze\deploy\wlp\usr\servers\<server>\logs`.

Solr logging

Information that relates to Solr is logged in `C:\IBM\i2analyze\deploy\solr\server\logs`.

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.

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 United Kingdom Limited Hursley House Hursley Park Winchester, Hants, SO21 2JN UK

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, and 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. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

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

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

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 and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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