Version 2 Release 2

IBM i2 Enterprise Insight Analysis Deployment examples

SC27-5091-00

Note

Before using this information and the product it supports, read the information in “Notices” on page 35.

This edition applies to version 2, release 2, modification 1 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 Installing and Deploying section of the IBM Knowledge Center, and ensure that you select the correct version.

Contents

Contacting IBM Support

IBM Support provides assistance with product defects, answers FAQs, and helps users to resolve problems with the product.

About this task

After trying to find your answer or solution by using other self-help options such as technotes, you can contact IBM Support. Before contacting IBM Support, your company or organization must have an active IBM software subscription and support contract, and you must be authorized to submit problems to IBM. For information about the types of available support, see the Support portfolio topic in the *Software Support Handbook*.

Procedure

To contact IBM Support about a problem:

1. Define the problem, gather background information, and determine the severity of the problem. For more information, see the Getting IBM Support topic in the *Software Support Handbook*.
2. Gather diagnostic information.
3. Submit the problem to IBM Support in one of the following ways:
   - Online through the IBM Support Portal at Support Portal. You can open, update, and view all of your service requests from the Service Request portlet on the Service Request page.
   - By phone. For the phone number to call in your region, see the Directory of worldwide contacts web page at https://www.ibm.com/planetwide/

Results

If the problem that you submit is for a software defect or for missing or inaccurate documentation, IBM Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the IBM Support website daily, so that other users who experience the same problem can benefit from the same resolution.

Using your deployment toolkit

The deployment toolkit contains scripts and components that you need to deploy and maintain i2 Analyze. Configuring components within the toolkit allows you to consistently deploy to the same locations with the same settings.

IBM i2 Analyze can be deployed both as a part of IBM i2 Enterprise Insight Analysis and as a standalone product. To support both these mechanisms, there are two types of i2 Analyze deployment toolkit available. The core difference between the two types of deployment toolkit, is the inclusion of additional components to help connect i2 Analyze to Cognos for deployments that require the i2 Analyze Onyx services.

Directories

The deployment toolkit contains files in several directories. On most occasions, you need to interact with only three of the directories:

Directory	Contents
`examples`	The `examples` directory contains the configurations that you use to create example deployments.
`configuration`	The `configuration` directory contains files that you must update with information specific to your deployment. **Note:** When the deployment toolkit is first installed, this directory does not exist, rather it is expected that you select one of the example configuration files to create this directory.
`scripts`	The `scripts` directory contains the `setup` script that you use to create, deploy, and upgrade i2 Analyze.

The `configuration` directory contains files that you must update with information specific to your deployment. When you have provided this information, you can use the `setup` script to administer your i2 Analyze deployment.

The `setup` script

The `setup` script completes specific actions that are called tasks. For a list of available tasks and other information, on the i2 Analyze server, open a command prompt and navigate to `toolkit\scripts`, then run one of the following commands:

`setup -h`

The `-h` argument displays the usage, common tasks, and examples of use for the `setup` script.

`setup -a`

The `-a` argument displays the same content as when you use `-h`, and a list of additional tasks.

`setup -dh`

The `-dh` argument displays the help information for data access points.

Note: On Linux, whenever you run the `setup` script from the `toolkit/scripts` directory, you must prefix the script with `./`. For example, `./setup -t deploy`.

Deployment toolkit tasks

The following toolkit options and tasks are available to use with the `setup` script in the deployment toolkit.

Usage: setup [-h] [-dh] [-a] [-s SERVER] [-w WAR | -id IDENTIFIERS | -hn HOST | --all] [--force] -t TASK [--stacktrace]

Task name	Task Description
`-h, --help`	Shows this help message, and exits
`-a, --additional`	Shows additional arguments and tasks, and exits
`-dh, --dahelp`	Shows Data Access specific tasks, and exits
`-t, --task <task>`	Specifies the task to perform
`-s, --server <server>`	Specifies the server profile to manage
`-w, --war <war>`	Specifies the war name
`-co, --collection <collection>`	Specifies the Solr collection ID
`-id, --id <ids>`	Specifies the ids of the components to manage
`-hn, --hostname <hostname>`	Restrict the task to operate only on components with the specified hostname attribute
`-l, --locale <locale>`	Specify the language code of the schema files to use in the example deployment
`--all`	Apply the operation to all applicable components
`-sn, --schemaName <Schema>`	Specify the name of the schema file to use in the example deployment
`-e, --exampleData <Example Data>`	Specify the name of the directory containing the example data to ingest
`-f, --force`	Suppress the warning prompt that is associated with tasks that might result in data loss
`-st, --stacktrace`	Print the stacktrace if an error occurs

The following installation and deployment tasks are available:

Task name	Task Description
`installLiberty`	Installs Liberty from the Liberty binaries in the toolkit, to the directory specified in environment.properties.
`installZookeeper`	Installs ZooKeeper from the ZooKeeper binaries in the toolkit, to the directory specified in environment.properties.
`installSolr`	Installs Solr from the Solr binaries in the toolkit, to the directory specified in environment.properties.

Task name	Task Description
`deployExample`	Deploys i2 Analyze with default files and settings.
`ingestExampleData`	Ingests entity and link record examples into the Information Store.
`deploy`	Creates the databases, creates the application profile, and deploys i2 Analyze.
`deployLiberty`	Deploys the i2 Analyze application into Liberty.
`start`	Starts all i2 Analyze services on the current server.
`stop`	Stops all i2 Analyze services on the current server.
`restart`	Restarts all i2 Analyze services on the current server.
`configSummary`	Summarizes the configuration of the toolkit.

Examples of use:

- `setup -t deployExample`
- `setup -t ingestExampleData`
- `setup -t deploy`
- `setup -t start`
- `setup -t configSummary`

The following upgrade tasks are available:

Task name	Task Description
`upgrade`	Upgrades the configuration files, and upgrades i2 Analyze.
`upgradeConfiguration`	Upgrades the configuration files for an Information Store deployment.
`upgradeZookeeper`	Upgrades Solr and ZooKeeper, and upgrades ZooKeeper configuration files to the version required by the toolkit.
`upgradeSolr`	Upgrades Solr and Solr configuration files to the version required by the toolkit.
`upgradeDatabases`	Upgrades the Information Store database and clears the search index if required.
`upgradeSolrCollections`	Upgrades ZooKeeper and Solr, and creates a new collection if required.
`upgradeLiberty`	Upgrades Liberty to the version required by the toolkit.

Examples of use:

- `setup -t upgrade`
- `setup -t upgradeConfiguration`
- `setup -t upgradeSolr -hn "example.solr.hostname"`

The "upgradeZookeeper", "upgradeSolr", "upgradeDatabases", and "upgradeSolrCollections" tasks support an optional -hn argument that restricts their effect to a single host.

The following administration tasks are available:

Task name	Task Description
replayFromTimestamp	Starts Liberty in a mode that replays all events since the time specified by the 'datetime.to.replay.from' property in environment-advanced.properties.
configureHttpServer	Sets up the reverse proxy configuration for IBM HTTP Server.
migrateFileRegistry	Migrate a file based user registry from a full WebSphere Application Server profile. The was.home.dir and wlp.home.dir properties must be set in environment.properties.
enableLibertyAdminCenter	Enable the Liberty Admin Center.
disableLibertyAdminCenter	Disable the Liberty Admin Center.
generateDefaults	Configures the environment with default property values.
ensureBasicUserRegistry	Configures the application for basic user registry authentication.
ensureExampleUserRegistry	Populates the user registry with an example user and user groups that map to the example security schema.
ensureDefaultUserProfileProvider	Configures the application with the default user profile provider.
addInformationStore	Generates a fragment for the Information Store, and updates topology.xml.
addI2Connect	Updates topology.xml for i2 Connect.
updateInformationStoreSchema	Updates the Information Store database to conform to the schema file referenced in the i2 Analyze configuration.
updateInformationStoreSecuritySchema	Updates i2 Analyze and the Information Store to use the security schema file referenced in the i2 Analyze configuration.
createDatabaseStorage	Creates the database storage*
createDatabases	Creates the database storage and tables*
modifyInformationStoreDatabase	Runs the informationStoreModifications.sql script on the Information Store database.
dropTables	Drops all of the tables from the database but leaves the database intact*
dropDatabases	Drops the entire database and de-allocates storage*

Task name	Task Description
`emptyInformationStore`	Empties the Information Store of data, apart from metadata.
`addInformationStoreIngestionSource`	Adds or replaces information about an ingestion source to the Information Store.
`createInformationStoreStagingTable`	Creates an Information Store staging table for a specific entity or link type.
`ingestInformationStoreRecords`	Ingests records into the Information Store.
`deleteProvenance`	Deletes (entity/link) provenance from the Information Store.
`previewDeleteProvenance`	Previews deleting (entity/link) provenance from the Information Store.
`syncInformationStoreCorrelation`	Synchronizes data in the Information Store after a correlation operation failed during ingestion.
`enableMergedPropertyValues`	Creates the database views used to define the property values of merged i2 Analyze records.
`disableMergedPropertyValues`	Removes the database views used to define the property values of merged i2 Analyze records.
`createEtlToolkit`	Creates a DataStage ETL toolkit that contains the files DataStage requires to run pipeline jobs.
`clearData`	Clears the search index and all the data in the database.
`clearSearchIndex`	Clears the search index.
`clearLTPAkeys`	Clears the LTPA keys.

* The exact behavior of these tasks may change depending on the chosen database engine.

Example of use:

- `setup -t replayFromTimestamp`
- `setup -t configureHttpServer`
- `setup -t clearData -co "solr.collection.id"`

The "clearData" and "clearSearchIndex" tasks support an optional -co argument that restricts their effect to a single Solr collection.

Tasks for DB2 only:

Task name	Task Description
`catalogRemoteDB2Nodes`	Adds a remote database server entry to the DB2 node directory for each remote DB2 database that is defined in topology.xml
`uncatalogRemoteDB2Nodes`	Removes the remote database server entry in the DB2 node directory for each remote DB2 database that is defined in topology.xml.

Task name	Task Description
`recatalogRemoteDB2Nodes`	Removes, then re-adds the remote database entry in the DB2 node directory for each remote DB2 database that is defined in topology.xml.
`listDB2NodeDirectory`	Lists the contents of the DB2 node directory.
`catalogDB2Databases`	Adds an entry to the system database directory for each DB2 database that is defined in topology.xml. If the database is remote from the i2 Analyze server, the database is cataloged at the node specified for that database in topology.xml.
`uncatalogDB2Databases`	Removes the entry in the system database directory for each DB2 database that is defined in topology.xml.
`recatalogDB2Databases`	Removes, then re-adds the entry in the system database directory for each DB2 database that is defined in topology.xml.
`listDB2SystemDatabaseDirectory`	Lists the contents of the local DB2 system database directory.

In addition to the start, stop and restart tasks, the following tasks are available:

Task name	Task Description
`startLiberty`	Starts Liberty.
`stopLiberty`	Stops Liberty.
`restartLiberty`	Restarts Liberty.
`startSolrAndZk`	Starts the Solr nodes and ZooKeeper hosts.
`stopSolrAndZk`	Stops the Solr nodes and ZooKeeper hosts.
`restartSolrAndZk`	Restarts the Solr nodes and ZooKeeper hosts.
`startSolrNodes`	Starts the Solr nodes.
`stopSolrNodes`	Stops the Solr nodes.
`restartSolrNodes`	Restarts the Solr nodes.
`startZkHosts`	Starts ZooKeeper hosts.
`stopZkHosts`	Stops ZooKeeper hosts.
`restartZkHosts`	Restarts ZooKeeper hosts.

Examples of use:

- `setup -t startSolrAndZk`
- `setup -t stopZkHosts -id "1,3"`
- `setup -t restartSolrNodes -id node1`
- `setup -t startSolrNodes -hn "example.solr.hostname"`

The "SolrNodes" and "ZkHosts" tasks support an optional -id argument.

The comma-separated list of identifiers that you specify restricts the task to the nodes and hosts with matching identifiers in the topology.

The "SolrNodes" and "ZkHosts" tasks support an optional -hn argument that restricts their effect to a single host.

The following Solr and ZooKeeper tasks are available:

Task name	Task Description
`createSolrNodes`	Creates the Solr nodes that are defined in topology.xml. If the nodes already exist, their configuration is updated.
`createZkHosts`	Creates the ZooKeeper hosts that are defined in topology.xml. If the hosts already exist, their configuration is updated.
`getZkStatus`	Reports the status of the ZooKeeper hosts that are defined in topology.xml
`createAndUploadSolrConfig`	Creates and uploads the Solr configuration to the ZooKeeper hosts.
`createSolrCollections`	Creates the Solr collections that are defined in topology.xml.
`deleteSolrCollections`	Deletes the Solr collections that are defined in topology.xml.

Example of use:

- `setup -t createSolrNodes -hn "example.solr.hostname"`
- `setup -t createSolrCollections -co "solr.collection.id"`

The "createAndUploadSolrConfig", "createSolrCollections", and "deleteSolrCollections" tasks support an optional -co argument that restricts their effect to a single Solr collection.

These tasks support an optional -hn argument that restricts their effect to a single host.

Data access options:

Task name	Task Description
`-x, --schema <schema>`	Specifies the full path including the file name of the schema file used to generate the mapping
`-o, --mappingJAR <mappingJAR>`	Specifies the full path including the file name to the mapping JAR file to be created
`-z, --xsdPath <xsdPath>`	Specifies the full path to a directory where the generated XSD files will be stored
`-dn, --datasourceName <datasourceName>`	Specifies the name of the data source to be created or modified
`-sc, --singleCardFormat`	Generates files that validate using a single card format instead of the standard multiple card format
`-s, --server <server>`	Specifies the server profile to manage
`-c, --configFile <configFile>`	Specifies the full path including the file name of the Connector Creator configuration file

Task name	Task Description
`-lp, --logProperties <logProperties>`	Specifies the full path including the file name of the logging properties file

The following tasks are available:

Task name	Task Description
`generateMappingJar`	Creates mapping classes and XSD files for the specified schema
`addDaodDataSource`	Generates a fragment for data access on-demand, and updates topology.xml
`addDelpsDataSource`	Generates a fragment for data load ELP stage, and updates topology.xml
`addConnectorCreator`	Generates a fragment for Connector Creator, and updates topology.xml
`createI2AnalyzeSchemaFromIBase`	Creates an i2 Analyze schema that reflects the entity, link, and property types in an existing iBase database
`replaceIBaseSchemaFromI2Analyze`	Replaces the schema of a new iBase database with one that is consistent with an existing i2 Analyze deployment
`addIBaseConnector`	Generates a fragment for an iBase connector, and updates topology.xml
`addIBaseDatasource`	Adds or updates an iBase data source in the deployment configuration
`uploadCcConfig`	Uploads a configuration file to Connector Creator
`downloadCcConfig`	Downloads the configuration file from Connector Creator
`updateConnectorsConfiguration`	Updates the i2 Analyze server with the connection details of the connectors defined in topology.xml.

Examples of use:

- `setup -t generateMappingJar -x MySchema -o MyMappingJAR [-sc]`
- `setup -t addDaodDataSource -dn MyDataSource`
- `setup -t addDelpsDataSource -dn MyDataSource`
- `setup -t addConnectorCreator`
- `setup -t uploadCcConfig -c "C:/ccConfigIn.xml"`
- `setup -t downloadCcConfig -c "C:/ccConfigOut.xml"`

Deployment process

Before you start a deployment of i2 Analyze, it is important to understand the process that is used. Use the following information to ensure that you choose the correct deployment method for your system.

The components that you need to deploy differ depending on the deployment pattern that you are using, and the features that you would like to take advantage of. To determine the pattern that best suits your needs, see the Deployment patterns overview.

Deployment sequence

Use the deployment toolkit to complete the actions for the deployment. By building up the deployment incrementally, you can tailor your system to the individual requirements of your organization, before you deploy into a production environment.

Create an example deployment

The following list is an outline of the steps that you follow to deploy an example deployment, from preparing the environment to connecting the first user:

1. Install the prerequisite software.
2. Create the configuration directory.
3. Provide the user names and passwords that the deployment process requires.
4. Create the example deployment.
5. Connect to i2 Analyze as a user, and test that the application can be accessed.

An example deployment demonstrates whether your prerequisites are set up correctly, and provides an opportunity to demonstrate the system without the need for further configuration.

Configure i2 Analyze

You can use the deployment toolkit from the example deployment as a starting point for creating a custom configuration.

After you have an example deployment, customize the configuration files to match your requirements:

1. Configure the environment.
2. Create an i2 Analyze security schema for your environment.
3. Select or create an i2 Analyze schema and charting scheme for your data.
4. Set the i2 Analyze application global properties.
5. Configure the application server security for your environment.
6. Connect to i2 Analyze as a user, and test that the application can be accessed.

At this stage, i2 Analyze can be redeployed after every change, so that you can test the deployment. If you want to make changes to the structure of your data, or the permissions that are used to access that data, you must clear the data and the search indexes before you redeploy.

Migrate i2 Analyze

When you have completed customizing the deployment, you can migrate i2 Analyze to a different system by completing the following sequence:

1. Install the prerequisite software.
2. Copy the deployment toolkit from the original system to the new system.
3. Change the host names and any other specific settings for the new environment.
4. Run the script to deploy i2 Analyze.
5. Configure the application server security for your environment.
6. Connect to i2 Analyze as a user, and test that the application can be accessed.

Selecting the type of example deployment

In the IBM i2 Analyze deployment toolkit, several example configurations are available. To reduce the amount of configuration that you need to complete, select the example that is closest to the requirements for your final deployment.

The i2 Analyze deployment toolkit includes the following example configurations:

information-store-opal

The information-store-opal example contains configuration settings for a deployment that includes:

- An Information Store that is accessed by using the i2 Analyze Opal services

daod-opal

The daod-opal example contains configuration settings for a deployment that includes:

- i2 Analyze with i2 Connect, which can provide access to other data stores through connectors that you create

information-store-daod-opal

The information-store-daod-opal example contains configuration settings for a deployment that includes:

- An Information Store that is accessed by using the i2 Analyze Opal services
- i2 Analyze with i2 Connect, which can provide access to other data stores through connectors that you create

analysis-repository

The analysis-repository example contains configuration settings for a deployment that includes:

- An Analysis Repository that is accessed by using the i2 Analyze Onyx services

By default these configurations deploy with the law enforcement schema in US English, but you can specify different schemas or locale codes as parameters.

The IBM i2 Enterprise Insight Analysis deployment toolkit also includes the following example configurations:

ar-information-store-opal

The ar-information-store-opal example contains configuration settings for a deployment that includes:

- An Information Store that is accessed by using the i2 Analyze Opal services
- An Analysis Repository that is accessed by using the i2 Analyze Onyx services

information-store-onyx

The information-store-onyx example contains configuration settings for a deployment that includes:

- An Information Store that is accessed by using the i2 Analyze Onyx services
- An Analysis Repository that is accessed by using the i2 Analyze Onyx services

This example configuration includes Cognos reports that are designed to use the signals intelligence schema. To use the i2 Analyze Onyx services with a different schema, you must modify the reports.

By default the Enterprise Insight Analysis configurations deploy with the signals intelligence schema in US English, but you can specify different schemas or locale codes as parameters.

It is possible to change the type of your deployment after you select and deploy your example by adding the appropriate data stores and configuration files. For information about modifying the topology of your deployment after you select an example, see Adding additional data sources.

Deploying your example

When you have decided on the type of starting deployment that you require, you can configure and deploy your example. Follow the instructions that are relevant to your example.

Deploying i2 Analyze with the Information Store

An installation of i2 Analyze includes example settings for deploying the server with an Information Store. After you deploy the Information Store, you can access the data that it contains by using Analyst's Notebook Premium.

Before you begin

Install IBM i2 Analyze and any software prerequisites. For more information, see Installing IBM i2 Enterprise Insight Analysis.

About this task

The following procedure describes how to create an example deployment of i2 Analyze with the Information Store. The i2 Analyze toolkit contains an example configuration for the deployment. The `deployExample` task generates the default values for the mandatory settings, provides the JDBC driver, and deploys the platform.

The example deployment demonstrates a working i2 Analyze system with an example user so that you can log in.

In the example deployment, i2 Analyze runs with the example security schema and matching Liberty security groups and users. The example user has the following credentials:

- The user name is `Jenny`
- The password is `Jenny`

The example deployment uses the `law-enforcement-schema.xml` schema as the i2 Analyze schema with the associated `law-enforcement-schema-charting-schemes.xml` as the charting scheme.

Procedure

1. Create the configuration directory:

a) Navigate to the `\toolkit\examples\configurations\information-store-opal` directory.

   This directory contains the preconfigured files that you require to deploy a system that uses the Information Store to store data. The data can be accessed by using Analyst's Notebook Premium as a rich desktop client.

b) Copy the `configuration` directory to the root of the toolkit.
   For example, `C:\IBM\i2analyze\toolkit\configuration`.

2. Specify the credentials to use for the deployment.

   For more information about credentials, see .

   a) Using a text editor, open the `toolkit\configuration\environment\credentials.properties` file.

   b) Enter the user name and password to use with the database.

   c) Enter the user name and password to use with the Solr index.

   d) Enter the password to use to encrypt LTPA tokens.

   e) Save and close the `credentials.properties` file.

3. Run the setup script to create the example deployment.

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

   b) To deploy the example, run the following command:

```
setup -t deployExample
```

   c) To start the WebSphere Application Server Liberty profile server, run the following command:

```
setup -t start
```

   d) Start, or restart, the HTTP server that hosts the reverse proxy.

4. Optional: To populate your Information Store with the provided example data, run the following command:

```
setup -t ingestExampleData
```

What to do next

When you start i2 Analyze, the URI that users must specify in Analyst's Notebook Premium is displayed in the console. For example, `This application is configured for access on http://`*`host_name`*`/opal`.

For the Opal experience, deploy Analyst's Notebook Premium and connect to your deployment. For more information, see Installing IBM i2 Analyst's Notebook Premium and Connecting IBM i2 Analyst's Notebook Premium to IBM i2 Analyze.

After you test your deployment, configure the deployment to match your requirements. For more information, see Configuring IBM i2 Analyze.

Deploying i2 Analyze with the Information Store and i2 Connect

An installation of i2 Analyze includes example settings for deploying the server with an Information Store and support for i2 Connect. The Information Store is accessed by using the i2 Analyze Opal

services, and i2 Connect enables analysts to search for and retrieve data from an example external data source.

Before you begin

Install IBM i2 Analyze and any software prerequisites. For more information, see Installing IBM i2 Enterprise Insight Analysis.

Before you create the example deployment, you must download and install Node.js to host the example connector. Download Node.js for your operating system from: https://nodejs.org/en/download/. You can install Node.js with the default settings.

About this task

The following procedure describes how to create an example deployment of i2 Analyze with the Information Store and i2 Connect. The i2 Analyze toolkit contains an example configuration for the deployment, and an example connector. The `deployExample` task generates the default values for the mandatory settings, provides the JDBC driver, and deploys the platform.

The example deployment demonstrates a working i2 Analyze system with an example user so that you can log in.

In the example deployment, i2 Analyze runs with the example security schema and matching Liberty security groups and users. The example user has the following credentials:

- The user name is `Jenny`
- The password is `Jenny`

The example deployment uses the `law-enforcement-schema.xml` schema as the i2 Analyze schema with the associated `law-enforcement-schema-charting-schemes.xml` as the charting scheme.

Procedure

1. Create the configuration directory:
   a) Navigate to the `\toolkit\examples\configurations\information-store-daod-opal` directory.
      This directory contains the preconfigured files that you require to deploy a system that uses i2 Connect to connect to an external data source and uses the Information Store to store data.
   b) Copy the configuration directory to the root of the toolkit.
      For example, `C:\IBM\i2analyze\toolkit\configuration`.
2. Specify the credentials to use for the deployment.
   For more information about credentials, see .
   a) Using a text editor, open the `toolkit\configuration\environment\credentials.properties` file.
   b) Enter the user name and password to use with the database.
   c) Enter the user name and password to use with the Solr index.
   d) Enter the password to use to encrypt LTPA tokens.
   e) Save and close the `credentials.properties` file.
3. Run the setup script to create the example deployment.
   a) Open a command prompt and navigate to the `toolkit\scripts` directory.

b) To deploy the example, run the following command:

```
setup -t deployExample
```

4. Start the example connector.

   Note: The example connector uses port number 3700. Ensure that no other processes are using this port number before you start the connector.

   a) In a command prompt, navigate to the `toolkit\examples\connectors\example-connector` directory.

   b) To install the dependencies that are required for the example connector, run the following command:

   ```
   npm install
   ```

   Note: You must be connected to the internet to install the dependencies.

   c) To start the Node.js server, run the following command:

   ```
   npm start
   ```

5. Start i2 Analyze.

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

   b) To start i2 Analyze, run the following command:

   ```
   setup -t start
   ```

6. Start, or restart, the HTTP server that hosts the reverse proxy.

What to do next

When you start i2 Analyze, the URI that users must specify in Analyst's Notebook Premium is displayed in the console. For example, `This application is configured for access on http://`*`host_name`*`/opal`.

Install Analyst's Notebook Premium and connect to your deployment. For more information, see Installing IBM i2 Analyst's Notebook Premium and Connecting IBM i2 Analyst's Notebook Premium to IBM i2 Analyze.

Production deployments of i2 Analyze use client-authenticated SSL communication between i2 Analyze and any connectors. This example does not use it, Analyst's Notebook Premium displayed a warning to that effect. For more information about configuring client authenticated SSL, see Client authenticated Secure Sockets Layer with IBM i2 Connect.

You can create your own connectors to use with the deployment of i2 Analyze. For more information about creating your own connectors, see IBM i2 Analyze and i2 Connect.

After you test your deployment, configure the deployment to match your requirements. For more information, see Configuring IBM i2 Analyze.

Deploying i2 Analyze with i2 Connect

An installation of i2 Analyze includes example settings for deploying the server with support for i2 Connect. i2 Connect enables analysts to search for and retrieve data from external data sources by

using the Opal quick search functions, and then analyze the results on a chart in Analyst's Notebook Premium.

Before you begin

Install IBM i2 Analyze and any software prerequisites. For more information, see Installing IBM i2 Enterprise Insight Analysis.

Before you create the example deployment of i2 Analyze with i2 Connect, you must download and install Node.js to host the example connector. Download Node.js for your operating system from: https://nodejs.org/en/download/. You can install Node.js with the default settings.

About this task

To use i2 Connect, you must obtain or create a custom connector to the external data source that you want to search. The i2 Analyze toolkit contains an example configuration for the deployment, and an example connector.

The example deployment demonstrates a working i2 Analyze system that can query and retrieve data from an external data source by using i2 Connect and an example connector. You can log in by using an example user. In the example deployment, i2 Analyze runs with the example security schema and matching Liberty security groups and users. The example user has the following credentials:

- The user name is Jenny
- The password is Jenny

Procedure

1. Create the configuration directory:
   a) Navigate to the `\toolkit\examples\configurations\daod-opal` directory.
      This directory contains the preconfigured files that you require to deploy a system that uses i2 Connect to connect to an external data source.
   b) Copy the `configuration` directory to the root of the toolkit.
      For example, `C:\IBM\i2analyze\toolkit\configuration`.
2. Specify the credentials to use for the deployment.
   For more information about credentials, see .
   a) Using a text editor, open the `toolkit\configuration\environment\credentials.properties` file.
   b) Enter the user name and password to use with the Solr index.
   c) Enter the password to use to encrypt LTPA tokens.
   d) Save and close the `credentials.properties` file.
3. Run the setup script to create the example deployment.
   a) Open a command prompt and navigate to the `toolkit\scripts` directory.
   b) To deploy the example, run the following command:

```
setup -t deployExample
```

4. Start the example connector.

   Note: The example connector uses port number 3700. Ensure that no other processes are using this port number before you start the connector.

   a) In a command prompt, navigate to the `toolkit\examples\connectors\example-connector` directory.

b) To install the dependencies that are required for the example connector, run the following command:

```
npm install
```

Note: You must be connected to the internet to install the dependencies.

c) To start the Node.js server, run the following command:

```
npm start
```

5. Start i2 Analyze.
   a) Open a command prompt and navigate to the `toolkit\scripts` directory.
   b) To start i2 Analyze, run the following command:

```
setup -t start
```

6. Start, or restart, the HTTP server that hosts the reverse proxy.

What to do next

When you start i2 Analyze, the URI that users must specify in Analyst's Notebook Premium is displayed in the console. For example, `This application is configured for access on http://`*`host_name`*`/opaldaod`.

Install Analyst's Notebook Premium and connect to your deployment. For more information, see Installing IBM i2 Analyst's Notebook Premium and Connecting IBM i2 Analyst's Notebook Premium to IBM i2 Analyze.

Production deployments of i2 Analyze use client-authenticated SSL communication between i2 Analyze and any connectors. This example does not use it, Analyst's Notebook Premium displayed a warning to that effect. For more information about configuring client authenticated SSL, see Client authenticated Secure Sockets Layer with IBM i2 Connect.

You can create your own connectors to use with the deployment of i2 Analyze. For more information about creating your own connectors, see IBM i2 Analyze and i2 Connect.

Deploying i2 Analyze with the Analysis Repository

An installation of i2 Analyze includes example settings for deploying the server with an Analysis Repository data store. After you deploy the Analysis Repository, you can connect to it by using the Intelligence Portal and Analyst's Notebook Premium.

Before you begin

Install IBM i2 Analyze and any software prerequisites. For more information, see Installing IBM i2 Enterprise Insight Analysis.

About this task

The following procedure describes how to create an example deployment of i2 Analyze with the Analysis Repository that uses DB2 as the database management system. The i2 Analyze toolkit contains an example configuration for the deployment. The `deployExample` task generates the default values for the mandatory settings, provides the JDBC driver, and deploys the platform.

The example deployment demonstrates a working i2 Analyze system with an example user so that you can log in.

In the example deployment, i2 Analyze runs with the example security schema and matching Liberty security groups and users. The example user has the following credentials:

- The user name is `Jenny`
- The password is `Jenny`

The example deployment uses the `law-enforcement-schema.xml` schema as the i2 Analyze schema with the associated `law-enforcement-schema-charting-schemes.xml` as the charting scheme.

Procedure

1. Create the configuration directory:
   a) Navigate to the `\toolkit\examples\configurations\analysis-repository` directory.

      This directory contains the preconfigured files that you require to deploy a system that uses the Analysis Repository to store data. The data can be accessed using both the Intelligence Portal as a thin web client and Analyst's Notebook Premium as a rich desktop client.

   b) Copy the `configuration` directory to the root of the toolkit.
   For example: `C:\IBM\i2analyze\toolkit\configuration`
2. Specify the credentials to use for the deployment.

   For more information about credentials, see .
   a) Using a text editor, open the `toolkit\configuration\environment\credentials.properties` file.
   b) Enter the user name and password to use with the database.
   c) Enter the password to use to encrypt LTPA tokens.
   d) Save and close the `credentials.properties` file.
3. Run the setup script to create the example deployment.
   a) Open a command prompt and navigate to the `toolkit\scripts` directory.
   b) To deploy the example, run the following command:

      ```
      setup -t deployExample
      ```

   c) To start i2 Analyze, run the following command:

      ```
      setup -t start
      ```

   d) Start, or restart, the HTTP server that hosts the reverse proxy.

What to do next

Deploy any other components that are required for the deployment pattern. After you deploy all the components that you require, test your deployment by connecting a client to your deployment. For more information, see “Connecting clients” on page 33.

After you test your deployment, configure the deployment to match your requirements. For more information, see Configuring IBM i2 Analyze.

Deploying both the Analysis Repository and the Information Store

Create an example deployment with the Analysis Repository, and Information Store with Opal services. The example uses an example i2 Analyze schema that is designed for signals intelligence data.

Before you begin

Install IBM i2 Enterprise Insight Analysis, and any software prerequisites. For more information, see Installing IBM i2 Enterprise Insight Analysis.

About this task

Create an example deployment of i2 Analyze with the Analysis Repository and the Information Store. In this example, the Information Store is used with the Opal services. The `deployExample` task generates the default values, provides the JDBC driver, and deploys the platform.

Example data is provided, which you can ingest into the Information Store.

An example deployment demonstrates a working i2 Analyze system. An example user is created so that you can log in.

In an example deployment, i2 Analyze runs with the example security schema and matching WebSphere Application Server Liberty profile security groups and users. The example user has the following credentials:

- The user name is `Jenny`
- The password is `Jenny`

These example deployments use the `sigint-schema.xml` schema as the i2 Analyze schema with the associated `sigint-schema-charting-schemes.xml`.

Setting up the Opal and Onyx configuration

If you would like to set up an Information Store that uses the i2 Analyze Opal services along with the Analysis Repository, use the `ar-information-store-opal` configuration example.

Procedure

1. Create the configuration directory:
   a) Navigate to the `toolkit\examples\configurations\ar-information-store-opal` directory.

      This directory contains the preconfigured files that you require to deploy the example system.
   b) Copy the `configuration` directory to the root of the toolkit.
      For example, `C:\IBM\i2EIA\toolkit\configuration`.
2. Specify the credentials to use for the deployment.

   For more information about credentials, see Modifying the credentials.
   a) Using a text editor, open the `toolkit\configuration\environment\credentials.properties` file.
   b) Enter the user name and passwords to use with the databases.
   c) Enter the user name and password to use with the Solr index.
   d) Enter the password to use to encrypt LTPA tokens.
   e) Save and close the `credentials.properties` file.
3. Run the setup script to create the example deployment.

a) On the i2 Analyze server, open a command prompt and navigate to the `toolkit\scripts` directory.

b) To deploy the example, run the following command:

```
setup -t deployExample -sn sigint-schema.xml
```

c) To start the applications, run the following commands:

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

d) Start, or restart, the HTTP server that hosts the reverse proxy.

4. Populate your Information Store with the provided example data:

```
setup -s opal-server -t ingestExampleData -e signal-intelligence-data-set-1
```

What to do next

Connect a client to test the deployment of Enterprise Insight Analysis. For more information, see Connecting clients.

Setting up the Onyx only configuration

If you would like to set up an Information Store that uses the i2 Analyze Onyx services along with the Analysis Repository, use the `information-store-onyx` configuration example. This example cannot be deployed until the configuration is extended to use the i2 Analyze Onyx services.

About this task

Note: It is useful to make backups of any files that you edit as part of this procedure in case you need to refer to the original file. This method helps you to track your changes if you need to consider revisions or corrections to solve a problem.

Procedure

1. Create the configuration directory:

   a) Navigate to the `toolkit\examples\configurations\information-store-onyx` directory.

   This directory contains the preconfigured files that you require to deploy the example system.

   b) Copy the `configuration` directory to the root of the toolkit.
   For example, `C:\IBM\i2EIA\toolkit\configuration`.

2. Open a command line and navigate to the `toolkit\scripts` directory. Using an account with administrator permissions, run the following command to set default values in the configuration:

```
setup -t generateDefaults
```

3. In the `toolkit\configuration\environment` directory, verify that the default values in `http-server.properties` are correct for your environment.
4. In the `toolkit\configuration\environment\onyx-server` directory, verify that the default values in `environment.properties` are correct for your environment.
5. Ensure that the `db2jcc4.jar` file is in the `toolkit\configuration\environment\common\jdbc-drivers` directory.

6. Using a text editor, open the `toolkit\configuration\environment\credentials.properties` file:
   - Populate the values of the `db.write1.user-name` and `db.write1.password` properties with the user account credentials that you want the i2 Analyze deployment scripts to use to authenticate with the Analysis Repository database.
   - Populate the values of the `db.infostore.user-name` and `db.infostore.password` properties with the user account credentials that you want the i2 Analyze deployment scripts to use to authenticate with the Information Store database.
   - Populate value of the `ltpakeys.password` property, with the value that is used to encrypt the LTPA keys file.
7. Using a text editor, open the `toolkit\configuration\environment\topology.xml` file and enter the DB2 edition and version values in the `<database>` element.
8. Copy the `toolkit\configuration\examples\security-schema\example-dynamic-security-schema.xml` file to the `toolkit\configuration\fragments\common\WEB-INF\classes` directory.

Enabling the Information Store to use the Onyx services

The IBM i2 Enterprise Insight Analysis installation includes configuration example files. You can use these example files to setup an example deployment that contains the Information Store with the Onyx services.

About this task

The following tasks guide you through the process of deploying and configuring the key components of a deployment that uses the Onyx services to connect to the Information Store:

- i2 Analyze
- IBM Cognos Analytics, or IBM Cognos Business Intelligence

Before you begin, you must install the prerequisite software. For more information, see:

- Installing IBM i2 Enterprise Insight Analysis
- Installing IBM Cognos

Use the following steps to deploy all the components of the example deployment on a single server. The example files include an example schema, and compatible Cognos reports.

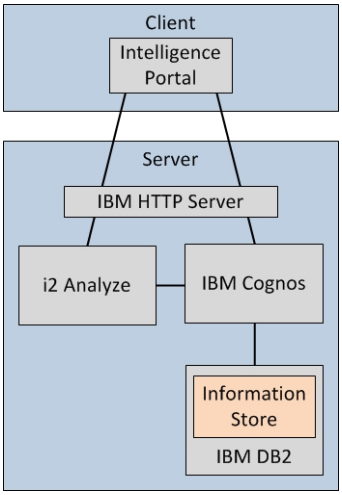


Figure 1: An example Onyx pattern deployment that uses a single server.

After you test your deployment, configure the deployment to match your requirements. To configure a deployment, see Configuring IBM i2 Enterprise Insight Analysis.

Creating the Content Manager database
When you install IBM Cognos Analytics 11.0.10 with the Custom option and all components selected, Cognos requires an IBM DB2® Content Manager database. Before you configure Cognos, you must create and configure this database.

About this task

If you are using your own instance of IBM DB2® for your content store, you can generate a DDL to create a DB2 database suitable for the content store. For more information, see Generating a script file to create a database for a DB2 content store. When the DDL is generated, run the script file to create a database in DB2. The script is created in the *install_location*\configuration\schemas\content\db2 directory.

Configuring Cognos
Enterprise Insight Analysis uses Cognos reports to query and retrieve search results from the Information Store. To deploy Cognos with Enterprise Insight Analysis, you must complete specific configuration steps that differ from a standard deployment of Cognos.

Before you begin

Important: Before you configure Cognos, ensure that you have access to an authentication provider, such as a Lightweight Directory Access Protocol (LDAP) server. The authentication provider must be able to provide an account to Enterprise Insight Analysis to access the Cognos SDK. You must know the details of your authentication provider server and at least one account so that you can specify those details when you configure Cognos.

About this task

Copy the required IBM DB2 drivers and the Enterprise Insight Analysis Morphlet into the IBM Cognos installation directories.

Test the content store connection. Then, complete the following procedure to configure the Cognos environment properties for Enterprise Insight Analysis.

Configure Cognos with the settings for your authentication provider. For more information, see Configuring IBM Cognos Components to Use an Authentication Provider. The information that you need to connect to your authentication provider server is input at step 5 of the procedure.

You can test the completed configuration or you can test some components individually by right-clicking the component in the Explorer panel and selecting **Test**. If the test fails, select **Details** for more information. Reconfigure the affected properties and then test again. Do not start the service until all tests are error-free.

Procedure

1. Copy the `db2jcc4.jar` and `db2jcc_license_cu.jar` drivers from the `java` directory of the DB2 installation to the Cognos installation.

   The default DB2 installation path depends on the operating system.

   - On Windows operating systems, the default DB2 installation path is `C:\Program Files\IBM\SQLLIB`.
   - On Linux operating systems, the default DB2 installation path is `/opt/ibm/db2/`*`version`* (where *version* is the DB2 version number).

   The destination directory for the driver files depends on the version of Cognos that you are using.

   - For Cognos 10.2.2, copy the drivers into the `webapps\p2pd\WEB-INF\lib` directory.
   - For Cognos 11.0.10, copy the drivers into the `drivers` directory and into the `webapps\p2pd\WEB-INF\lib` directory.

2. Deploy the Enterprise Insight Analysis Morphlet by copying the `i2.xts` file from the `cognos` directory in your EIA installation folder directory in the Enterprise Insight Analysis Integrated Components to the `templates\ps\portal` directory of the Cognos installation.
3. Start IBM Cognos Configuration.
   - On Linux operating systems, open a command line, navigate to the `bin64` directory of the Cognos installation, and enter the following command:

     ```
     ./cogconfig.sh
     ```

   - On Windows operating systems, click **Start** > **IBM Cognos Configuration**.
4. To test the content store connection, select the Content Store node and in the toolbar click **Test**.
5. Modify the Cognos environment configuration. In the Explorer pane, click **Environment**.
   a) Set the **Report server execution mode** property to **64-bit**.

      In a 64-bit Cognos installation, both 32-bit and 64-bit versions of the report server component are provided. The 32-bit version is enabled by default.

   b) Modify each URI property so that the host name section of the URI does not contain *localhost*, but the network host name of your server. Each URI must use a network host name.

      To ensure the validity of cookies, use a fully qualified domain name that is shared between the i2 Analyze server instance and Cognos. For example, *`host_name.my.domain.com`*.

   c) For Cognos 11.0.10, in the Gateway URI field, specify the URI to the gateway.

      Enter the fully qualified URL for the Cognos service. The URL must end in `/cgi-bin/cognos.cgi`. For example, `http://`*`cognosserver.mydomain.com`*`/analytics/cgi-bin/cognos.cgi` or `http://`*`cognosserver.mydomain.com`*`/ibmcognos/cgi-bin/cognos.cgi`.

   d) In the toolbar, with the Environment node selected, click **Test**.

Cognos Configuration tests the environment group properties.

6. Enterprise Insight Analysis requires authenticated access to the Cognos SDK. Complete the following steps to configure Cognos to connect to the external authentication service that provides the account that you want Enterprise Insight Analysis to use to access the Cognos SDK:
   a) In IBM Cognos Configuration, in the Explorer pane, click **Security**.
   b) Right-click **Authentication** and select **New resource** > **Namespace**.
   c) Enter a name for your authentication namespace, for the example, use EIAA. Then, select the appropriate namespace type from the list, and click **OK**.
   d) In the Explorer pane, under **Authentication**, click the new authentication provider resource that you created. For the **Namespace ID** property, specify a unique identifier for the namespace.
   For the example, use EIAA.
   e) Populate or modify any other values to match your authentication namespace.
   f) In the toolbar, with the EIAA node selected, click **Test**.
   For example, if the security you specified is LDAP, enter a user name and password that you know is contained in the LDAP repository.
   Cognos Configuration tests the namespace configuration.
7. If you are using Cognos Analytics, specify the subset of URLs for which cookies are valid.
   a) In the toolbar, click **Actions** > **Edit Global Configuration**.
   b) In the **General** tab, set the value of **Path** to the base URL for your deployment. If you installed Cognos in the default directory, the value is /analytics.
8. If you are using Cognos Analytics, set the IBM Cognos Application Firewall validation to false.
   a) Click **IBM Cognos Application Firewall**.
   b) Set **Enable CAF validation** to false.
9. Test and start the Cognos services:
   a) In the toolbar, click **Actions** > **Test**.
   Cognos Configuration tests the connections to content store and other resources, and the namespace configuration.
   b) Start the Cognos service, click **Actions** > **Start**.

   For more information about testing and starting the Cognos services, see Start the Application services components.

Configuring the IBM HTTP Server for Cognos
Modify the IBM HTTP Server configuration to provide access to Cognos directories.

About this task

First, create a Cognos configuration file to contain the information you need to provide the HTTP Server with access to Cognos directories. Then, update the HTTP Server configuration by incorporating the Cognos configuration. Examples are provided for the information that you need to provide to populate the Cognos configuration.

Procedure

1. In the `conf` directory of the IBM HTTP Server installation, use a text editor to create a Cognos configuration file, for example `cognos.conf`.
   The default path for the `conf` directory is `C:\IBM\HTTPServer\conf`.

2. Use the text editor to populate the Cognos configuration file with the information that the HTTP Server requires to access Cognos.

   For examples of how to provide information about the Cognos configuration, see "IBM HTTP Server configurations for Cognos" on page 27.
3. In the `conf` directory of the IBM HTTP Server installation, open the `httpd.conf` file.

   The default path for the `httpd.conf` file is `C:\IBM\HTTPServer`.
4. To modify the HTTP Server configuration for your version of Cognos, include the `cognos.conf` that you populated in step 2.

   Add the name of the Cognos configuration file to the end of the `http.conf` file as shown in the following example.

   ```
   ######End#IBM#IAP#Generated#Configuration#RewriteRules######
   include conf/cognos.conf
   ```
5. Optional: To add images to reports in Cognos® Report Studio, configure Web Distributed Authoring and Versioning (WebDAV) on your web server. No images are included in the example files.

   For information about configuring WebDAV, see Configuring WebDAV to view and browse images.
6. Restart IBM HTTP Server.

IBM HTTP Server configurations for Cognos
The Cognos configuration values are used by IBM HTTP Server to enable access to Cognos directories. Populate the `cognos.conf` file with the Cognos configuration information according the version of Cognos and the operating system.

IBM HTTP Server configuration for Cognos Analytics 11.0.10
For example, use the following configuration to define aliases for the `bi`, `cgi-bin`, and `webcontent` directories:

```
LoadModule headers_module modules/mod_headers.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so

<Location /analytics/bi>
    RequestHeader set X-BI-PATH /analytics/bi/v1
    Header always unset X-Frame-Options
    ProxyPass http://host_name.my.domain.com:9300/bi
    ProxyPassReverse http://host_name.my.domain.com:9300/bi
    ProxyPassReverseCookieDomain "." "my.domain.com"
</Location>

ScriptAlias /analytics/cgi-bin "installation_path/cognos/analytics/cgi-bin"

<Directory "installation_path/cognos/analytics/cgi-bin">
    Header always unset X-Frame-Options
    Header always append X-Frame-Options SAMEORIGIN
    Header always unset X-Frame-Options "ALLOW-FROM http://
host_name.my.domain.com/"
    AllowOverride None
    Options None
    Require all granted
</Directory>

Alias /analytics "installation_path/cognos/analytics/webcontent"
<Directory "installation_path/cognos/analytics/webcontent">
    AllowOverride None
    Options None
    Require all granted
</Directory>

<Location /analytics/cgi-bin/mod2_2_cognos.dll>
    SetHandler cognos-handler
    Require all granted
</Location>
```

Where *installation_path* is the Cognos installation location. Ensure that you use forward slashes (/) in all file paths, for both Linux and Windows paths.

Where *host_name* is the fully qualified domain name of the i2 Analyze server.

Note: `analytics` is the default value for the **Gateway URI** and **Controller URI for gateway** properties in IBM Cognos Configuration.

Note: For a Linux deployment, in the previous configuration change `mod2_2_cognos.dll` to `mod2_4_cognos.so`. You must also add the following configuration to the `cognos.conf` file:

```
LoadModule expires_module modules/mod_expires.so
LoadModule filter_module modules/mod_filter.so


LoadModule cognos_module "/opt/ibm/cognos/analytics/cgi-bin/mod2_4_cognos.so"

<IfModule mod_expires.c>
    <FilesMatch "\.(jpe?g|png|gif|js|css|json|html|woff2?|template)$">
        ExpiresActive On
        ExpiresDefault "access plus 1 day"
    </FilesMatch>
</IfModule>

 <IfModule mod2_4_cognos.c>
    CGIBinDir "installation_path/cognos/analytics/cgi-bin"
 </IfModule>

 <Directory installation_path/cognos/analytics>
    <IfModule mod_deflate>
        AddOutputFilterByType DEFLATE text/html application/json
          text/css application/javascript
    </IfModule>
    Options Indexes MultiViews
    AllowOverride None
    Require all granted
</Directory>
```

IBM HTTP Server configuration for Cognos 10.2.2
For example, use the following configuration to define aliases for the `bi`, `cgi-bin`, `webcontent/documentation`, `webcontent`, `webcontent/samples`, and `v5dataserver/XQE/Logs` directories:

```
ScriptAlias /ibmcognos/cgi-bin "installation_path/c10_64/cgi-bin"
<Directory "installation_path/c10_64/cgi-bin">
    Header always unset X-Frame-Options
    Header always set X-Frame-Options "ALLOW-FROM http://host_name/"
    Options FollowSymLinks
    AllowOverride FileInfo

    Require all granted
</Directory>
Alias /ibmcognos/help "installation_path/c10_64/webcontent/documentation"
<Directory "installation_path/c10_64/webcontent/documentation">
    Options Indexes FollowSymLinks MultiViews IncludesNoExec
    AddOutputFilter Includes html
    AllowOverride FileInfo
    Require all granted
</Directory>
Alias /ibmcognos "installation_path/c10_64/webcontent"
<Directory "installation_path/c10_64/webcontent">
    Options Indexes FollowSymLinks MultiViews IncludesNoExec
    AddOutputFilter Includes html
    AllowOverride FileInfo
    Require all granted
</Directory>
Alias /samples "installation_path/c10_64/webcontent/samples"
<Directory "installation_path/c10_64/webcontent/samples/">
    Options Indexes FollowSymLinks MultiViews IncludesNoExec
    AddOutputFilter Includes html
    AllowOverride FileInfo
    Require all granted
</Directory>
Alias /XQElogs "installation_path/c10_64/v5dataserver/XQE/Logs"
<Directory "installation_path/c10_64/v5dataserver/XQE/Logs">
    Options Indexes FollowSymLinks MultiViews IncludesNoExec
    AddOutputFilter Includes html
    AllowOverride None
    Require all granted
</Directory>
```

Where *installation_path* is the Cognos installation location. Ensure that you use forward slashes (/) in all file paths, for both Linux and Windows paths.

Where *host_name* is the fully qualified domain name of the i2 Analyze server.

Note: `ibmcognos` is the default value for the **Gateway URI** and **Controller URI for gateway** properties in IBM Cognos Configuration.

Deploying the Cognos reports and visualizations

Using IBM Cognos Administration, you need to deploy the sample reports and visualizations. There are named reports for each type of search request, and reports for the more complex Expand and Find Path analytics.

Procedure

1. Go to the EIA installation directory for Cognos, for example `i2EIA\cognos`.
2. Copy the `EIAA_Samples.zip` and `Visualizations.zip` files to the `deployment` directory of the Cognos installation, for example `C:\Program Files\IBM\cognos`. Do not extract the contents of the files.
3. In IBM Cognos Configuration, verify that the Cognos services are running.
4. In a web browser, start IBM Cognos Administration.

   Use the following URL format to start IBM Cognos Administration: `http://<hostname>:9300/bi/v1/disp?b_action=cogadmin`
5. On the **Configuration** tab, click **Content Administration**. Then, on the toolbar, click the **New Import** icon.
   a) On the **Select a deployment archive** page, select **EIAA_Samples** and click **Next**.
   b) On the **Specify a name and description** page, ensure that the name of the package is set to `EIAA_Samples` and click **Next**.
   c) On the **Select the public folders and directory content** page, select the **eia** directory, and then click **Next**.
   d) On the **Specify the general options** page, keep the default settings and click **Next**.
   e) On the **Review the summary** page, check the settings and click **Next**.
   f) On the **Select an action** page, select **Save and run once** and click **Finish**.
   g) On the **Run with options** page, select **Now** and click **Run**.
   h) Click **OK** to import the reports.
6. Click the **New Import** icon and to import the visualizations that are included with the reports, repeat the process in step 5. On the **Select a deployment archive** page, select **Visualizations**. On the **Select the public folders and directory content** page, select the **Visualizations** directory. Ensure that all the visualizations are imported into Cognos.

   Note: The reports and visualizations are automatically imported into the locations that are required by Enterprise Insight Analysis. If you move your Cognos reports to a different directory or modify the directory structure, the reports might not run because the package mapping is embedded within the report definition.
7. Import the EIA graphics that are used in the visualizations to the Cognos installation.
   a) Locate the `webcontent.zip` file in the EIA installation directory for Cognos, for example `i2EIA\cognos`.
   b) Extract the contents of the `webcontent.zip` file to `webcontent` directory of the Cognos installation, retaining the directory structure of the compressed file.

      For example, extract to `C:\Program Files\IBM\cognos\analytics\webcontent`.

   Note: If you want to modify the example reports or create more, the model for IBM Cognos Framework Manager is provided in the EIA installation dierctory for Cognos, for example `i2EIA \cognos`. To work with the model, you must download and install IBM Cognos Framework Manager.

Deploying i2 Analyze
When you deploy a system that includes an Information Store with the Onyx services, you must complete specific configuration steps that differ from a standard i2 Analyze deployment.

Procedure

1. Using a text editor, open the `toolkit\configuration\fragments\cognos-connector\WEB-INF\classes\ApolloServerSettingsDaodMandatory.properties`. Populate the values for the following properties.

   Guidance and examples are provided in the comment sections for each of the properties in the `ApolloServerSettingsDaodMandatory.properties` file. You must use the property values that you modified in the Cognos® environment configuration when you completed the procedure in "Configuring Cognos" on page 23. In IBM Cognos Configuration, check values in the Environment pane, and by going to **Security** > **EIAA**.

   - `CognosURL`: Include the fully qualified hostname for the Cognos server. For example, `http://<hostname>/analytics/cgi-bin/cognos.cgi`.
   - `AuthenticationNamespace`: Enter the name that you entered under Authentication, For example, `EIAA`.
   - `CognosSDKURL`: Include the fully qualified hostname for the Cognos server. For example: `http://<hostname>:9300/bi/v1/disp`
   - `CognosSDKUserID` and `CognosSDKPassword`: For example, enter the user and password for the account that you used to test security in the procedure in "Configuring Cognos" on page 23.

   Note: The `CognosSDKPassword` value, which is stored in the `ApolloServerSettingsDaodMandatory.properties` file is not encoded after you deploy i2 Analyze. After you deploy i2 Analyze, remove the value from the file.
2. In a command line, navigate to the `toolkit\scripts` directory. Using an account with administrator permissions and with permissions to create and modify the database, run the following command:

   ```
   setup -t deployExample -s onyx-server
   ```
3. Restart the IBM HTTP Server that hosts the reverse proxy.

What to do next
Set up the WebSphere Application Server Liberty user registry, then configure the connection between Cognos and the Information Store database.

Setting up the user registry
To allow users to log in to i2 Analyze, they must be members of groups. An example user registry is provided, which matches the names of the groups that are defined in the example security schema.

Procedure

1. To set up security, open a command line and navigate to the `toolkit\scripts` directory. Using an account with administrator permissions, enter the following command to stop the application server:

   ```
   setup -t stop
   ```
2. Using an XML editor, open the basic registry file, `deploy\wlp\usr\shared\config\user.registry.xml`. Configure users and groups that define the user access levels to items in i2 Analyze.

For example, when you complete the procedure in "Deploying i2 Analyze" on page 31, i2 Analyze is deployed with the following sample user registry.

```
<basicRegistry id="basic" realm="WebRealm">
    <user name="Jenny" password="{xor}FToxMSY="/>
    <group name="Analyst">
        <member name="Jenny"/>
    </group>
    <group name="Clerk">
        <member name="Jenny"/>
    </group>
    <group name="Controlled">
        <member name="Jenny"/>
    </group>
    <group name="Unclassified">
        <member name="Jenny"/>
    </group>
    <group name="Security Controller">
        <member name="Jenny"/>
    </group>
    <group name="Administrator">
        <member name="Jenny"/>
    </group>
</basicRegistry>
```

In this example, the user name is *Jenny* and the password is *Jenny*.

For more information about setting up security, Configuring a basic user registry for Liberty.

3. In a command line, navigate to the `toolkit\scripts` directory. Using an account with administrator permissions, enter the following command to start i2 Analyze with the WebSphere® Application Server security settings:

```
setup -t start
```

4. Start, or restart, the IBM HTTP Server that hosts the reverse proxy.

Configuring the connection between Cognos and the Information Store
In IBM Cognos Administration, configure the connection to the Information Store database.

Procedure

Complete the following steps to configure the connection to the Information Store database.

1. In the web browser, enter the URL to open IBM Cognos Administration.
   For example, `http://<hostname>:9300/bi/v1/disp?b_action=cogadmin`
2. In IBM Cognos Administration, on the **Configuration** tab, click **Data Source Connections**.
3. On the toolbar, click the **New Data Source** icon.
4. Enter the name for the data source: `EIAA_InformationStore`. Then, click **Next**.
5. In the **Type** menu, select **IBM DB2**. Then, click **Next**.
6. Specify the **DB2 database name**, this value is the name of the Information Store database that is specified in the `<database>` element for the Information Store in the `topology.xml` file. For example, `ISTORE`.

7. Specify the **DB2 connect string**, for example `jdbc:db2://host_name:port`. Where *host_name* is the host of your Information Store database, and *port* is the port number that the database is listening on.
8. In the **Signon** pane, select **Password**. Then, specify the **User ID** and **Password** of a user that can connect to the Information Store database.
9. Click **Test the connection** to verify that Cognos can connect to the Information Store database. If the connection is successful, `Succeeded` is displayed in the **Status** column of the **Test the connection** pane. Then, click **Next**.
10. In the **Specify the IBM DB2 connection string** page, specify the **Server name**, **Port number**, and **Database name** with the values for your Information Store database. Then, click **Finish**.

What to do next

Connect a client to test the deployment of Enterprise Insight Analysis. For more information, see Connecting clients.

Configure your example deployment to match your requirements. For more information, see Configuring IBM i2 Analyze.

Connecting clients

When your example deployment has completed, you can connect to your data store using one of the supported clients.

Before you begin

- You must have at least one user set up within the application server that has permission to access items in i2 Analyze.
- You must have started i2 Analyze.

Procedure

To use Analyst's Notebook to connect, follow the instructions in Connecting IBM i2 Analyst's Notebook Premium to IBM i2 Analyze.

Opening the web client

If you have deployed the Information Store, you can open the web client from any client computer with access to the HTTP server that hosts the reverse proxy. Ensure that you can access the web client, and search for data.

Before you begin

The web client is licensed as a part of the IBM i2 Enterprise Insight Analysis Investigate Add On. Ensure that you have the correct agreements in place before accessing the web client.

Procedure

1. Open a web browser, and navigate to `http://host_name/opal` (where *host_name* is the fully qualified domain name or IP address of the HTTP server).
   The web client displays a login dialog.
2. Enter the name and password of a user who is registered in the application server.
3. Search and visualise data, to ensure that the application is running.

Opening the Intelligence Portal

If you have deployed the Analysis Repository, you can open the Intelligence Portal from any client computer with access to the HTTP server that hosts the reverse proxy. Ensure that you can access the Intelligence Portal, and create, browse, and search for data.

Procedure

1. Open a web browser, and navigate to `http://`*`host_name`*`/apollo` (where *host_name* is the fully qualified domain name or IP address of the HTTP server).
   i2 Analyze displays a login dialog.
2. Enter the name and password of a user who is registered in the application server.
3. Create, browse, and search for data, to ensure that the application is running.

   Note: When you create test items, ensure that the permissions are set up so that you have access to view them.

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.

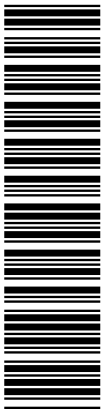
Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

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.

Part Number: 99F9999



SC27-5091-00