IBM Application Discovery for IBM Z Batch Server V5.1.0

Configuration Guide

SC27-8973-06

Contents

Chapter 1. Accessibility Features for IBM Application Discovery for IBM Z

Accessibility features assist users who have a disability, such as restricted mobility or limited vision, to use information technology content successfully.

Overview

IBM® Application Discovery for IBM Z® includes the following major accessibility features:

- Keyboard-only operation
- Operations that use a screen reader

IBM Application Discovery for IBM Z uses the latest W3C Standard, WAI-ARIA 1.0 (www.w3.org/TR/wai-aria/), to ensure compliance with US Section 508 (www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards) and Web Content Accessibility Guidelines (WCAG) 2.0 (www.w3.org/TR/WCAG20/). To take advantage of accessibility features, use the latest release of your screen reader and the latest web browser that is supported by IBM Application Discovery for IBM Z.

The IBM Application Discovery for IBM Z online product documentation in IBM Knowledge Center is enabled for accessibility. The accessibility features of IBM Knowledge Center are described in the Accessibility section of the IBM Knowledge Center help (https://www.ibm.com/support/knowledgecenter/en/about/releasenotes.html).

Keyboard navigation

This product uses standard navigation keys.

Interface information

For alternative installation using Command Line Installation (CLI), refer to section Alternative Installation for ADDI Using CLI in *IBM AD Installation and Configuration Guide*.

The IBM Application Discovery for IBM Z user interfaces do not have content that flashes 2 - 55 times per second.

The IBM Application Discovery for IBM Z web user interface relies on cascading style sheets to render content properly and to provide a usable experience. The application provides an equivalent way for low-vision users to use system display settings, including high-contrast mode. You can control font size by using the device or web browser settings.

The IBM Application Discovery for IBM Z web user interface includes WAI-ARIA navigational landmarks that you can use to quickly navigate to functional areas in the application.

Related accessibility information

In addition to standard IBM help desk and support websites, IBM has a TTY telephone service for use by deaf or hard of hearing customers to access sales and support services:

TTY service
800-IBM-3383 (800-426-3383)
(within North America)

For more information about the commitment that IBM has to accessibility, see IBM Accessibility (www.ibm.com/able).

Chapter 2. Introduction

After **IBM Application Discovery for IBM Z (AD) Batch Server** is installed, a folder named `sample-conf` is created in the **IBM AD Batch Server** installation folder. This `sample-conf` folder contains the templates for all the configuration files needed to customize the functioning of **IBM AD Batch Server**. To configure **IBM AD Batch Server**, copy the templates from the `sample-conf` folder, and place them in the `conf` folder.

Once the setup is completed, before running **IBM AD Batch Server**, some preliminary configurations should be performed. You need to specify on which projects you want **IBM AD Batch Server** to run the reports, which reports to generate, where to store the generated reports etc. Also, you need to specify the parameters for **IBM AD Web Service**.

The configuration parameters are stored in `server.properties`, `project.properties`, and `webservice.properties` files. The following sections contain a detailed description of the parameters available in these three files as well as a description of other configuration files for specific components such as **Annotations**, **GraphDB**, **Search Index**, **Rule Based**, **Reports**, and **Web Service Metrics**.

Important: It is vital that you set the parameters for the **Search Index**, **Annotations** and **GraphDB** components. Failing to do so will make some of the analyses impossible to generate in **IBM AD Analyze**.

Terms and Conventions

For the purposes of writing this guide, the following terms and conventions have been used:

- **`Command`** names are printed as shown.
- "Terms and Conventions" on page 3 in this guide are indicated as shown. For page numbers, refer to the Table of Contents at the beginning of this guide.
- `File references` are printed as shown.
- **Button names** and options/functions within a dialog box are printed as shown.

Chapter 3. Configuration Guides

Server Properties

About this task

In `server.properties` file, enter the desired values for the properties that are detailed below.

Note: Make sure to restart **IBM AD Batch Server** after you modify the configuration file.

Procedure

1. Add the IP of the computer where the **IBM AD Configuration Server** host is installed.

```
## Coordination and Configuration Server host
ccs.server.host=127.0.0.1
```

2. Add the port number to be used by **IBM AD Configuration Server**; the default port is 2181, if no other value is specified the default value is used.

```
## Coordination and Configuration Server port
## default 2181
#ccs.server.port=2181
```

3. Add the ID of the Environment set in **IBM AD Configuration Server**.

```
## Coordination and Configuration Server environment
ccs.environment=fb9107c2-8770-43ef-990c-15b387f19698
```

4. By default the audit of **IBM AD Batch Server** components is enabled, by setting it to false the Audit stops.

```
##Audit server
# default true
#audit.enable=true
```

5. Add the IP used by **IBM AD Batch Server** to communicate with **IBM AD Configuration Server**

```
## client host
## default: one of the available ip-s
# client.host=
```

6. Add the IP of the computer where **IBM AD Batch Server** is installed.

```
##aplication server host
#application.server.host=
```

7. Add the port number where **IBM AD Batch Server** listens to admin requests; the default port is 8686, if no other value is specified the default value is used.

```
## port on which the server listens to admin requests
## default: 8686
#application.server.port=
```

8. Specify how much to wait before the server stops (milliseconds); the default value is 5000, if no other value is specified the default value is used.

```
## how much to wait for the server to stop (milliseconds)
## default: 5000
#stop.timeout=
```

9. Specify how frequently database version changes are tested (seconds); the default value is 120, if no other value is specified the default value is used.

```
## how frequently database version changes are tested (seconds)
## default: 120
#pool.interval=
```

10. Specify the installation directory or the place where **IBM AD Batch Server** is installed. All paths are either absolute or relative to the installation directory. Relative paths start with the first directory under `${install.dir}`.

```
## ${install.dir} place where server is installed
## all paths are either absolute or relative to ${install.dir}
## relative paths start with the first directory under ${install.dir}
```

a) Enter the path to the directory where configuration files are stored; the default path is `${install.dir}/conf`; if no other path is specified the default path is used.

```
## path to the directory where configuration files are placed
## default: ${install.dir}/conf
conf.dir={install.dir}/conf
```

Important:

- Under Windows, make sure that all backslash characters in all paths are doubled as in ****. Example:

```
conf.dir=C:\\Program\Files\\IBM\Application\Discovery\Batch\Server/conf
```

- Under Linux, remember that all paths use **/** as separators and spaces are escaped by using ****. Example:

```
conf.dir=/home/username/IBM\Application\Discovery\Batch\Server/conf
```

b) Enter the path to the directory where data files are stored; the default path is `${install.dir}/data`; if no other path is specified the default path is used.

```
## path to the directory where data files are placed
## default: ${install.dir}/data
data.dir={install.dir}/data
```

Important:

- Under Windows, make sure that all backslash characters in all paths are doubled as in ****. Example:

```
data.dir=C:\\Program\Files\\IBM\Application\Discovery\Batch\Server/data
```

- Under Linux, remember that all paths use **/** as separators and spaces are escaped by using ****. Example:

```
data.dir=/home/username/IBM\Application\Discovery\Batch\Server/data
```

c) Enter the path to the directory where temporary data is stored; the default path is `${data.dir}/tmp`; if no other path is specified the default path is used. Example:

```
## path to the directory where temporary data is placed
## default: ${data.dir}/tmp
tmp.dir={data.dir}/tmp
```

Important:

- Under Windows, make sure that all backslash characters in all paths are doubled as in ****. Example:

```
tmp.dir=C:\\Program\Files\\IBM\Application\Discovery\Batch\Server/tmp
```

- Under Linux, remember that all paths use **/** as separators and spaces are escaped by using ****. Example:

```
tmp.dir=/home/username/IBM\Application\Discovery\Batch\Server/tmp
```

Project Properties

The `project.properties` file contains a set of global settings, followed by the specific setting(s) for each type of component.

For each available component, there can be one or more configuration files. In `project.properties` file we need to enter the paths to these configurations files for each component. Additionally, in these files you can configure project specific settings. For example, you can configure **IBM AD Batch Server** to run only **Rule Based** reports for a specific project. Or you can configure **IBM AD Batch Server** to save the reports for one project in one folder and for other project in another folder. For more optional settings, see the project specific settings in the corresponding section for each component.

Global Settings

The global settings specify the projects on which the **IBM AD Batch Server** will operate and which components will run on the specified projects.

The `project.properties` file is where you can set the following global properties.

- Comma separated list of project names that will be the only ones considered for this service.

  Important: If no value is set for this parameter no report will be generated. ***** means all projects.

```
projects.whitelist=<Project>
```

- Comma separated list of project names that will be excluded for this service (lower priority than whitelist). Leave empty or remove property if not needed.

```
projects.blacklist=<Project>
```

- Comma separated list of component names that will be the only ones considered for this service. Ex.: **index** must be added as a component.

```
components=index,gdbImport,annUpdate
```

  Optional components can be considered for this service.

Table 1. Optional Components

Component	Description
ruleBased	The **Rule Based** component generates reports for the resources specified in the configuration files according to the rules and parameters defined in the corresponding configuration files. **Note:** If the **Rule Based** component is used, make sure that the `ruleBased.properties` file is configured. For more information, go to “ruleBased.properties File” on page 14.
reports	The **Reports** component is used to generate the complexity reports. For more information, go to IBM AD Analyze User Guide, Complexity Reports chapter.

Table 1. Optional Components (continued)	
Component	**Description**
cobolPP	The **cobolPP**, **jclPP**, and **pl1PP** components generate the expanded sources for Cobol, JCL, and PL/I. For more information, go to IBM AD Analyze User Guide, View Expanded Source chapter.
jclPP	
pl1PP	
wsmetrics	The **wsmetrics** component is needed only if IBM Application Delivery Intelligence for IBM Z (ADI) is used on the system. **Note:** Additionally, in order for the **wsmetrics** component to be executed, make sure that the **gdbImport** component is included in the components list of this service.

Important: It is vital that you set the parameters for the **Search Index**, **Annotations** and **GraphDB** components. Failing to do so will make some of the analysis impossible to generate in **IBM AD Analyze**.

Annotations

About this task

IBM AD Batch Server reads the configurations present under Annotations Database configurations from **IBM AD Configuration Server**. The configurations were set in `server.properties` in previous versions, and according to those settings it creates, if necessary, the repository where all user-defined annotations for all projects are stored.

Note: To make sure that Annotations Database was created in your database repository server instance check the following steps.

Procedure

1. Go to **IBM AD Batch Server** installation folder and search for the `conf` folder.
2. Open the `server.properties` file that can be found in `conf` folder and verify that **IBM AD Configuration Server** host and environment are correctly filled in. For more information, see “Server Properties” on page 5.
3. Go to **IBM AD Configuration Server** > **Environment** > **Configurations** > **Annotations Database** and make sure that **Database name**, **Database schema**, and **Database server** settings are correctly filled in. For more information, see Configuring the Annotations Database.
4. Start IBM AD Batch Server.
5. Go to the database repository instance set-up under Annotations Database configurations from **IBM AD Configuration Server** and you should find the database having the name mentioned in Database name field.

   Note: By default, **IBM AD Batch Server** will preserve in Annotations Database only those annotations for projects which are present in the projects list and will delete all other references/annotations. If you want to keep all annotations for all projects including the projects which are deleted at some point, uncomment and enter false.

   ```
   ## if true, all references of projects that do not appear in projects list will be deleted from annotations database.
   (the default behavior)
   ```

   ```
   #annUpdate.manageDeletedProjects=true
   ```

   In order to still use EZ#Annotations database created with installations lower than 5.1.0.0, see Configuring the Annotations Database.

Application Settings

This setting allows the user to specify which components will run on a specific project.

Application Project Settings

This is the list of components that will execute for the project. **Empty** means no component will be executed; use ***** for all. Use a separate line for each project for which you want to define the components list.

```
#project.<projectName>.components=
```

Index Project Files

The **Index** component will index the resources of a project so that a search in resources can be performed in **IBM AD Analyze** client, using **Search in Files** analysis.

Note: In order for the **Search in Files** analysis to be used by all **Analyze Client** users, make sure the Index location is a shared folder and has the appropriate permissions. For more information on **Search in Files**, see the dedicated KnowledgeCenter section.

Index Global Settings

Output folder where indexes are stored; the project name will be added by default.

Tip: This setting does not override database entry!

```
index.indexFolder=\\\\network path\\<Folder>\\Index
```

Note: The **Analyze Client** must have read access to **network path**\\<Folder>\\Index.

Once the indexes have been created in the default location, if there is a need to change this default folder, the following parameter should be used with value **true**.

```
index.forceFolder=true
```

If a folder matches one of the patterns specified below, it will not be indexed.

```
index.exclude.folderPattern=pattern,(pattern)...
```

If a file matches one of the patterns specified below, it will not be indexed.

```
index.exclude.filePattern=pattern,(pattern)...
```

Index Project Settings

All the parameters in the global settings can also be configured individually for each project. The parameters set for each project override the global parameters.

The output folder where the index for this project is stored must be unique per project.

Note: This setting does not override database entry!

```
project.<projectName>.index.indexFolder=
```

This is how you can override the database folder and create the index in a new location:

```
project.<projectName>.index.forceFolder=(true|false) default false
```

This is how you can specify extra folders to be indexed (apart from project folder):

Note: Make sure that all backslash characters in all paths are doubled: ****.

```
project.<projectName>.index.extraSources=path, path
```

If a folder matches one of these patterns, it will not be indexed.

```
project.<projectName>.index.exclude.folderPattern=pattern(,pattern)...
```

If a file matches one of these patterns, it will not be indexed.

```
project.<projectName>.index.exclude.filePattern=pattern(,pattern)...
```

ADI Indexing

About this task

The **ADI Index** component will index the resources of a project so that the **Business Rules Discovery (BRD)** feature, from **IBM Application Delivery Intelligence (ADI)** application, can display code snippets.

Procedure

1. Go to <IBM ADDI Installation Folder>\IBM Application Discovery Batch Server \conf and in adi.properties file, enter the desired values for the properties detailed below.
   a) Specify the protocol, the host name and the port for the ADI server as a URL.

   ```
   ##URL for the ADI indexing service, ex: http://my.adi.host:8080/addi-es
   es.serverUrl=
   ```

   b) Specify the user name used for ADI server authentication.

   ```
   ##user name used in ADI indexing service
   es.userName=
   ```

   c) Specify the password used for ADI server authentication.

   ```
   ##password used in ADI indexing service
   es.password=
   ```

2. In project.properties file, located in the \conf folder, enter the **adidx** component name.

   ```
   ## list of components that will be executed
   ## example components:
   ruleBased,reports,index,gdbImport,annUpdate,cobolPP,jclPP,pl1PP,wsmetrics,adidx
   ## empty means none will execute!!, use * for all
   components=adidx
   ```

Cobol, JCL and PL/I Preprocessing

View expanded source functionality in **IBM AD Analyze** can only be available if a preprocessing operation has been performed on the analyzed resources. Preprocessing of **Cobol** resources implies expanding all include files and applying all **REPLACE** compiler directives. For **JCL**, pre-processing implies expanding all **JCL** procedures and **JCL** include files.

Preprocessing of **PL/I** resources implies expanding all *include* files, having the following supported syntax:

```
File inclusion: %INCLUDE File, %XINCLUDE File, ++INCLUDE File, -INC File;
```

```
Library member inclusion: %INCLUDE MEMBER, %INCLUDE LIB (MEMBER).
```

The **Cobol**, **JCL** and **PL/I** preprocessing parameters allow the user to customize the details of the operation.

Cobol Preprocessor Global Settings

This parameter specifies the encoding used to read **Cobol** programs and copybooks when generating the preprocessed source. The default value is **UTF-8**, but any Java™ encoding is accepted. For example, for Japanese ideograms use **Shift_JIS**.

```
cobolPP.encoding=Shift_JIS
```

If the source code contains tabs instead of blank spaces, the preprocessor should know how many spaces to put instead of the tabs. The tab size default value is *4*, allowed values are *>= 1*.

```
#cobolPP.tabSize=4
```

Threads parameter is used to set the maximum number of threads per task to execute when preprocessing. The default value is *5*, allowed values are *>= 1*.

```
#cobolPP.threads=5
```

Emulate prefix parameter dictates whether or not to handle **PREFIX** and **-INC** directives when preprocessing, the default value is **true**.

```
#cobolPP.emulatePREFIX=true
```

Full path parameter dictates whether to specify in the inserted comment the full path of the expanded copy file in the preprocessed source, default value **true**. If set to **false**, only the name of the copy file and its content will be inserted in the preprocessed source.

```
#cobolPP.copy.fullPath=true
```

JCL Preprocessor Global Settings

This parameter specifies the encoding used to read files in order to generate preprocessed source. The default value is **UTF-8**, but any Java encoding is accepted. For example, for Japanese ideograms, use **Shift_JIS**.

```
jclPP.encoding=Shift_JIS
```

Threads parameter can be used to set the maximum number of threads per task to execute when preprocessing. The default value is *5*, but the allowed values are *>= 1*.

```
#jclPP.threads=5
```

PL/I Preprocessor Global Settings

This parameter specifies the encoding used to read files in order to generate preprocessed source. The default value is *UTF-8*, but any Java encoding is accepted. For example, in case of Japanese ideograms, use *Shift_JIS*.

```
pl1PP.encoding=Shift_JIS
```

Threads parameter can be used to set the maximum number of threads per task to execute when preprocessing. The default value is *5*, but the allowed values are *>= 1*.

```
pl1PP.threads=5
```

Important: If the source code contains characters or digits beyond the column 72 (in the inactive zone, for example *BG100 : PROCEDURE OPTIONS (MAIN);[ ]* ***00020002***) then the next additional setting must be made:

```
pl1PP.line_max_len=73
```

Rule Based Analysis Reports

The **Rule Based** component will generate reports for the resources specified in the configuration files according to the rules and parameters defined in the corresponding configuration files.

A configuration file is line based text organized as key/value pairs. General properties of a configuration file include:

- empty lines are ignored.
- lines that start with **#** are ignored (comments).
- keys do not contain spaces or equal sign **=**.
- a value starts at the first non-space character after **=**and finishes at the end of the line.
- values can expand on multiple lines by means of a single backslash character followed by newline.

Rule Based Global Settings

File specifying what information will be displayed on the first page of the generated reports.

```
ruleBased.configFile=ruleBasedConfig.properties
```

File containing the actual definition of the rules.

```
ruleBased.rulesFile=ruleBased.properties
```

Encoding of the file containing the actual definition of the rules, the default encoding is **UTF-8**, to modify, uncomment and enter desired encoding.

```
ruleBased.rulesFile.encoding=UTF-8
```

Option to generate an individual report for each source file where a rule failed, default value **true**.

```
ruleBased.reportsPerSource=true
```

Option to generate for each source file an additional file (named `controlTotal`) containing the number of rules that failed for that file. Valid only if the previous parameter, **reportsPerSource**, is set to **true**. The default value is **true**.

```
ruleBased.generateTotals=true
```

Output folder for rule based reports generated by source based rules. The default folder is the data folder, to set a different folder, uncomment and enter the path to that folder. The default format for the reports is `*.`PDF.

```
ruleBased.reportsFolder=
```

Output folder for reports generated in `*.`CSV format. The default folder is the data folder, to set a different one, uncomment and enter the path to that folder.

```
ruleBased.csvFolder=
```

Output folder for `controlTotals` files generated by source based rules. The `controlTotals` files are the files containing the number of rules that failed for a specific source file. The default folder is the data

folder, to set a different folder, uncomment and enter the path to that folder. This parameter is only valid only if `generateTotals` is set to **true**.

```
ruleBased.totalsFolder=
```

Default suffix for the `controlTotals` file. To set a different suffix, uncomment and enter the desired suffix.

```
ruleBased.totalsSuffix=.CDGSTDS.TOTALS
```

Prefix for the `controlTotals` file, by default the `controlTotals` do not carry a prefix, to use a prefix uncomment and enter the desired prefix.

```
ruleBased.totalsPrefix=
```

Rule Based Project Settings

All the parameters in the global settings can also be configured individually for each project. The parameters set for each project override the global parameters.

- Absolute path to rule configuration file used for Rule Based analysis component.

  ```
  project.<projectName>.ruleBased.rulesFile=projectName.rules.properties
  ```

- Option to generate an individual report for each source file where a rule failed. The default value is **true**.

  ```
  project.<projectName>.ruleBased.reportsPerSource=
  ```

- Option to generate for each source file an additional file (named controlTotal) containing the number of rules that failed for that file. Valid only if **reportsPerSource** is set to **true**.

  ```
  project.<projectName>.ruleBased.generateTotals=
  ```

- Output folder for rule based reports generated by source based rules. The default folder is the data folder. To set a different folder, uncomment and enter the path to that folder. The default format is `*.PDF`:

  ```
  project.<projectName>.ruleBased.reportsFolder=
  ```

- Output folder for `*.CSV` files generated by data based rules. The default folder is the data folder. To set a different folder, uncomment and enter the path to that folder:

  ```
  project.<projectName>.ruleBased.csvFolder=
  ```

- Output folder for controlTotals files generated by source based rules. The default folder is the data folder. To set a different folder, uncomment and enter the path to that folder. Valid only if **generateTotals** is set to **true**.

  ```
  project.<projectName>.ruleBased.totalsFolder=
  ```

ruleBasedConfig.properties File

All data set in this file will appear on the cover page of the reports.

- Name of the owner of the report.

  ```
  owner=
  ```

- Name of the organization.

  ```
  organization=
  ```

- Role of report owner within the organization.

```
organizationRole=
```

- Image that will appear on cover page of the report.

```
coverImage=
```

- Project name that will appear on cover page; if not set here, the name of the project given as input will be used.

```
projectName=
```

ruleBased.properties File

This file specifies the actual rules on which the **Rule Based** reports are generated, different parameters for the rules as well as the source files on which the **Rule Based** analysis will run.

Here are some specific syntax elements of the `RuleBased.properties` file:

- <number>: is a positive integer.
- <identifier>: alphanumeric text with no spaces.
- <path>: absolute path to a file on disk.
- <class>: qualified class name for a rule.
- <text>: any string of characters on one line. Line continuation (single backslash at the end of line) can be used for multiple lines. A syntax comment is introduced hereby. It is not part of the file format.
- {key}: the value of the key as found in the file.

Rules can be parametrized, inputs being one of the optional parameters which can be set. Input values can be a list of data from the database based on a query. Rule characteristics:

- **id** - unique number to identify an input; it is used to create all subsequent keys for current input.
- **key** - unique key used for substitution in query rules.
- **name** – input name.
- **query** - query to obtain a list of values from the repository which will act as parameters.
- **firstColumnType** - declare the type for the first column of the query. The default value is numeric. Accepted, string.

Parameter values can be simple values (numbers or text) and are also optional. Parameter characteristics:

- **id** - unique number to identify a parameter; it is used to create all subsequent keys for current parameter.
- **key** - unique key used for substitution in query rules.
- **name** – parameter name.
- **pattern** - if parameter is a simple value, pattern can be used to validate the data the user entered.
- **length** - if parameter is a simple text value, length can be used to verify the max length of the data.
- **value.lessThan** - for numeric simple values it can be used for validation.
- **value.greaterThan** - for numeric simple values it can be used for validation.
- **prefix** - text to prefix the user chosen data (transforms the parameter as text).
- **suffix** - text to be appended to the user chosen data (transforms the parameter as text).

Rules can be organized in groups according to specific criteria. Rule group characteristics:

- **id** - uniquely identifies a group; it is used to create all subsequent keys for current group.
- **name** - name to display for the group.

- **parent** - a {group.id} value of another group that contains this group.
- **description** - additional information on the group displayed in the application.

Rules are the basis of the Rule Based analysis action. Rule characteristics include:

- **id** -unique number that identifies a rule; it is used to create all keys for current rule.
- **name** – the name that is visible for the user. This will appear in the report.
- **sourceBased** - classifies the rule. The report will be generated in `*.PDF` format if **sourceBased** is set to **true**. If **sourceBased** is set to **false**, the report will be generated in `*.CSV` format. The default value is **true**.
- **weight** – a number used to calculate the weight of a certain rule, appears in the report summary. The default value is **1**.
- **query** and **selectiveQuery** - paths to query files.
- **class** - qualified class name for a rule implementation.
- **groups** - list of group ids that will show the rule.
- **description** - long text that describes the rule to the end-user.
- **params** - list of parameter keys that are required by this rule.
- **inputs** - list of input keys that are required by this rule.

Reports

The **Report** component will generate the reports specified in the configuration file.

Report: Global Settings

Important: The global settings used by these reports is set in the `project.properties` file.

Absolute path to the file containing a description of the report headers: `#reports.configFile=${install.dir}.reportsConfig.properties`.

Comma separated list of report codes that will be generated.

```
## 15103=Halstead
## 15104=Cyclomatic
## 15105=Maintainability
## 15106=Heuristic
## 15108=Unused Routines Within Programs
#reports.whitelist=
```

Comma separated list of report codes that will NOT be generated (lower priority than whitelist).

```
#reports.blacklist=
```

Output folder for generated reports. The default folder is the data folder, to change, uncomment and enter absolute path to desired folder.

```
#reports.outputFolder=
```

Option to generate a unique report for all resources. The default value is **true**. To generate a separate report for each resource, enter **false**.

```
#reports.coalesceInputs=true
```

Report: Project Settings

Important: The global settings used by these reports is set in the `project.properties` file.

All the parameters in the global settings can also be configured individually for each project. The parameters set for each project override the global parameters.

Comma separated list of report codes that will be generated for the project.

```
## 15103=Halstead
## 15104=Cyclomatic
## 15105=Maintainability
## 15106=Heuristic
## 15108=Unused Routines Within Programs
#project.<projectName>.reports.whitelist=
```

Comma separated list of report codes that will NOT be generated (lower priority than whitelist)

```
#project.<projectName>.reports.blacklist=
```

Absolute path to custom report configuration file for the project.

```
#project.<projectName>.reports.configFile==${install.dir}.projectName.reportsconfig.properties
```

Option to generate a unique report for all resources in the specified project. The default value is **true**. To generate a separate report for each resource, enter **false**.

```
#project.<projectName>.report.coalesceInputs=<true|false>
```

reportsConfig.properties File

This file allows the user to determine what information will appear on the first page of the reports, whether to show the appendix of a report, as well as where to find the configuration files which allow the customization of different report types.

The data set in this file will appear on the cover page of the reports.

- Name of the owner of the report.

  ```
  owner=
  ```

- Name of the organization.

  ```
  organization=
  ```

- Role of report owner within the organization.

  ```
  organizationRole=
  ```

- Image that will appear on cover page of the report.

  ```
  coverImage=
  ```

- Project name that will appear on cover page; if not set here, the name of the project given as input it will be used.

  ```
  projectName=
  ```

- Show appendix in reports; the default value is **false**.

  ```
  showAppendix=
  ```

- Absolute path to the file containing custom settings for reports (values for weights, colors for pie charts, etc.)

```
halstead.settingsFile=
heuristic.settingsFile=
cyclomatic.settingsFile=
```

```
maintainability.settingsFile=
unusedRoutines.settingsFile=
```

cyclomatic.properties File

This file contains settings which allow the customization of the **Cyclomatic** reports.

Note: This configuration file is located in the `conf/reports` folder.

The colors used in the pie charts present in the report:

```
#simpleColorPreference=0,255,0
#mediumColorPreference=250,196,69
#complexColorPreference=221,0,0
```

Values for thresholds used to separate simple/medium/complex groups:

```
#simpleThresholdPreference=10
#mediumThresholdPreference=50
```

halstead.properties File

This file contains settings which allow the customization of the **Halstead** reports.

Note: This configuration file is located in the `conf/reports` folder.

By default, all statements are considered operators. If you want to set a specific statement as operand, enter the statement id (from the **Statement** table in the database) then enter **false**.

```
##stmt id = check state
#if not present, falls to default. In default file, default value = check
#29=false
#28=false
```

heuristic.properties File

This file contains settings which allow the customization of the **Heuristic** reports.

Note: This configuration file is located in the `conf/reports` folder.

Colors used in pie charts present in the report:

```
#simpleColorPreference=0,255,0
#mediumColorPreference=250,196,69
#complexColorPreference=221,0,0
```

Values for thresholds used to define simple/medium/complex groups:

```
#simpleThresholdPreference=200
#mediumThresholdPreference=500
```

By default, all statements are assigned a weight of **1**, if you want to assign a different weight for a specific statement, enter the statement id (from the **Statement** table in the database) and then desired weight value.

```
##stmt id = weight value
#if not present, falls to default. In default file, default value = 1
#1815=100
#1814=1000
#109=2
#108=3
```

maintainability.properties File

This file contains settings which allow the customization of the **Maintainability** reports.

Note: This configuration file is located in the `conf/reports` folder.

Colors used in pie charts present in the report:

```
#complexColorPreference=0,255,0
#simpleColorPreference=221,0,0
#mediumColorPreference=250,196,69
```

Values for thresholds used to define simple/medium/complex groups:

```
#simpleThresholdPreference=65
#mediumThresholdPreference=85
```

By default, all statements are considered operators. If you want to set a specific statement as operand, enter the statement id (from the **Statement** table in the database) then enter **false**.

```
#stmt id=false
#29=false
#28=false
```

unusedRoutines.properties File

This file contains settings which allow the customization of the **Unused Routines** reports.

Note: This configuration file is located in the `conf/reports` folder.

By default, **Exit Procedures** will NOT be shown in the report, to include the **Exit Procedures** in the generated report, uncomment and enter **false**.

```
#filterExitProcedures=true
```

Web Service Metrics Component

Web Service Metrics component generates input data for **IBM AD Web Service**. The metrics values are packaged into **JSON** objects.

IBM AD Web Service

IBM AD Web Service collects the data provided by **Web Service Metrics** component and prepares it for delivery.

IBM AD Web Service parameters are stored in a `webservice.properties` file available in **IBM AD Batch Server** installation folder.

The following parameters can be set:

1. The IP of the computer where **IBM AD Web Service** is installed. The default value is **0.0.0.0** (all interfaces). Another valid value is the external IP of the computer.

   ```
   ## web service host
   #host=
   ```

2. The port no. to be used by the **IBM AD Web Service**; the default port is **8090**, if no other value is specified the default value will be used.

   ```
   ## web server port
   ## default: 8090
   #port=
   ```

3. The administrative port number to be used by **IBM AD Web Service**; the default administrative port is **8099**, if no other value is specified the default value will be used.

```
## web service administrative port
## default: 8099
#admin-port=
```

4. Absolute path to the **Web service** directory.

```
## web service directory
web-dir=webdata
```

5. Absolute path to the **JSON** files directory.

```
## json files directory
rest-dir=wsmetrics
```

6. Absolute path to file containing the Web service authentication data.

```
## authentication info for the web service
auth-file=auth.properties
```

7. Absolute path to file containing the SSL certificate.

```
##ssl keystore file
Keystore-file=keystore.jkl
```

8. Absolute path to the file containing log configuration data.

```
## log configuration file
log-conf-file=webservice.log4j.properties
```

IDCAMS Preprocessor

To display the dataset usage via **IDCAMS** utility in **Dataset Usage** in **Jobs**, a preprocessing operation must be performed on the analyzed resources. Running **IDCAMS** preprocessing implies saving data from all available **JCL** jobs that execute **IDCAMS** utility into the database repository project repository. **IDCAMS** preprocessing does not need to be added as a separate component, it runs on each project added in the <project.whitelist>.

Global Settings

This parameter specifies the encoding used to read **JCL** jobs and **CTRL** files when parsing the sources for all available projects. The default value is **UTF-8**, but any Java encoding is accepted.

You can setup this encoding in the `project.properties` file.

For example, in case of Japanese ideograms use **Shift_JIS**:

```
ams.encoding=Shift_JIS
```

Project Settings

This parameter specifies the encoding used to read **JCL** jobs and **CTRL** files when parsing the sources for a specific project. The default value is **UTF-8**, but any Java encoding is accepted.

You can setup this encoding in the `project.properties` file, using the following entry:

```
project.<projectName>.ams.encoding=UTF-8
```

Appendix 1 - IBM AD REST API

The base path for the **Application Discovery REST API** is `/ws`. Also for all sample requests, add `/ws` after **{host}:{port}**:

```
curl--userusername:passwordhttps://{host}:{port}/ws/projects
```

GET /projects

Description: Get all projects.

Sample request:

```
curl --user username:password https://{host}:{port}/projects
```

Sample response:

```
{
    "projects": [
        {
            "name": "zMobile",
            "type": "mainframe",
            "uuid": "zMobile",
            "defined": 1511983789
        },
        {
            "name": "SAMPLE-PROJECT",
            "type": "mainframe",
            "uuid": "SAMPLE-PROJECT",
            "defined": 1511983787
        },
        {
            "name": "JKEBank",
            "type": "mainframe",
            "uuid": "JKEBank",
            "defined": 1511983790
        }
    ]
}
```

Where:

- **projects**: representing an array containing all the project entities. The project entity structure:
  - **name**: representing the name of AD project.
  - **type**: representing the type of the AD project (only *mainframe* at the moment).
  - **uuid**: representing the UUID of the AD project (the AD project name at this moment, it will be replaced with UUID variant 4).
  - **defined**: representing the UNIX timestamp (in seconds) when the project was added to the collection.

GET /projects/{project-uuid}/datasets

Where *{project-uuid}* is a unique identifier for a project.

Description: Get all datasets available in an AD project.

Sample request:

```
curl --user username:password https://{host}:{port}/projects/SAMPLE-PROJECT/datasets
```

Sample response:

```
{
    "meta": {
        "project": {
```

```
            "name": "SAMPLE-PROJECT",
            "uuid": "SAMPLE-PROJECT",
        },
        "snapshot": 1502354654
    },
    "data": [
        {
            "uid": "01da56d83ec1c04be901ebe162957e3d3eaf6e49a85a0f31497f535b1f567b32",
            "name": "CB12V51.GENAPP.KSDSCUST.REPORT",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "(,,,4)",
                "datasetType": "GDG"
            }
        },
        {
            "uid": "09c2ef33f622e62da876e1d73a7e9e8f3aa8236034973a3f232acbf4da8509ee",
            "name": "DFH520.CICS.SDFJAUTH",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "DATASET"
            }
        },
        {
            "uid": "0c296329019f68889434bc14103de4475c3a680c64ee91f9a8f331c736b30257",
            "name": "DFH530.CNTL.CICS.DFHTEMP",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "PDS"
            }
        }
    ]
}
```

Where:

- **meta.project.name**: representing the name of AD project.
- **meta.project.uuid**: representing the UUID of the AD project.
- **meta.snapshot**: representing the UNIX timestamp (in seconds) at the time the data was collected.
- **data**: representing an array containing all the CICS® regions artifacts.

The dataset artifact:

- **uid**: representing the string unique identifier.
- **name**: representing the dataset name.
- **type**: representing the DATASET (the artifact type).
- **properties.volumeSerial**: representing the **Volume** parameter from the DD definition.
- **properties.datasetType**: representing the dataset type (**PDS**, **GDG** or **DATASET**)

GET /projects/{project-uuid}/cics

Where *{project-uuid}* is a unique identifier for a project.

Description: Get all CICS regions available in an AD project.

Sample request:

```
curl --user username:password https://{host}:{port}/projects/SAMPLE-PROJECT/cics
```

Sample response:

```
{
    "meta": {
        "name": "SAMPLE-PROJECT",
        "uuid": "SAMPLE-PROJECT",
        "snapshot": 1513610223
    },
    "data": [
        {
            "uid": "56fa655b88f42e49246da037a2dabee69027befa07b31e1774464eb3c93bba0e",
```

```
            "name": "CICSTS53",
            "type": "REGION"
        },
        {
            "uid": "90e806231100cb8108d0b9a639cc5fd2cd92689c4b40d051329534a881fc6ea1",
            "name": "CICSTS52",
            "type": "REGION"
        }
    ]
}
```

Where:

- **meta.project.name**: representing the name of AD project.
- **meta.project.uuid**: representing the UUID of the AD project.
- **meta.snapshot**: representing the UNIX timestamp (in seconds) when the data was collected.
- **data**: representing an array containing all the CICS regions artifacts.

The CICS region artifact:

- **uid**: representing a string unique identifier
- **name**: representing the region name.
- **type**: representing the *REGION* (the artifact type).

GET /projects/{project-uuid}/cics/{region-uid}/datasets

Where:

- **{project-uuid}**: is a unique identifier for a project.
- **{region-uid}**: is a unique identifier of the a cics region artifact.

Description: Get all datasets used by a cics application.

Sample request:

```
curl --user username:password https://{host}:
{port}/projects/90e806231100cb8108d0b9a639cc5fd2cd92689c4b40d051329534a881fc6ea1/datasets
```

```
{
    "meta": {
        "project": {
            "name": "SAMPLE-PROJECT",
            "uuid": "SAMPLE-PROJECT",
        },
        "snapshot": 1502354654
    },
    "data": [
        {
            "uid": "01da56d83ec1c04be901ebe162957e3d3eaf6e49a85a0f31497f535b1f567b32",
            "name": "CB12V51.GENAPP.KSDSCUST.REPORT",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "DATASET"
            }
        },
        {
            "uid": "09c2ef33f622e62da876e1d73a7e9e8f3aa8236034973a3f232acbf4da8509ee",
            "name": "DFH520.CICS.SDFJAUTH",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "DATASET"
            }
        },
        {
            "uid": "0c296329019f68889434bc14103de4475c3a680c64ee91f9a8f331c736b30257",
            "name": "DFH530.CNTL.CICS.DFHTEMP",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "DATASET"
```

```
            }
        }
    ]
}
```

Where:

- **meta.project.name**: representing the name of AD project.
- **meta.project.uuid**: representing the UUID of the AD project.
- **meta.snapshot**: representing the UNIX timestamp (in seconds).

GET /projects/{project-uuid}/jobs

Description: Get all jobs available in an AD project.

Sample request:

```
curl --user username:password https://{host}:{port}/projects/SAMPLE-PROJECT/jobs
```

Sample response:

```
{
    "meta": {
        "project": {
            "name": "SAMPLE-PROJECT",
            "uuid": "SAMPLE-PROJECT",
        },
        "snapshot": 1502354654
    },
    "data": [
        {
            "uid": "0b934ea4cfb7f446c7d926e711c45e12383bbfbea8d030dd34a154c7e3974076",
            "name": "CICSTS53",
            "type": "JCL"
        },
        {
            "uid": "5763b2fdc5a06f4af67e65f994585bdbe2eb00a7fc9aa72c58195d4906eda17f",
            "name": "CRTAIXS",
            "type": "JCL"
        },
        {
            "uid": "6b6f6023b61db6ee87798a6ecb6272aec0cf44b10d609e9ffe3807c7378dd786",
            "name": "RUNCOBOL",
            "type": "JCL"
        }
    ]
}
```

Where:

- **meta.project.name**: representing the name of AD project.
- **meta.project.uuid**: representing the UUID of the AD project.
- **meta.snapshot**: representing the UNIX timestamp (in seconds) when the data was collected.
- **data**: representing an array containing all the job artifacts.

The job artifact:

- **uid**: representing a string unique identifier.
- **name**: representing the job name.
- **type**: representing the JCL (the artifact type) at the time the data was collected.
- **data**: representing an array containing all the datasets artifacts in the CICS region identified by **{region-uid}**.

GET /projects/{project-uuid}/jobs/{job-uid}/datasets

Description: Get all datasets defined in DD cards for a given Job.

Sample request:

```
curl --user username:password https://{host}:{port}/projects/SAMPLE-
PROJECT/jobs/0b934ea4cfb7f446c7d926e711c45e12383bbfbea8d030dd34a154c7e3974076/datasets
```

Sample response:


```
{
    "meta": {
        "project": {
            "name": "SAMPLE-PROJECT",
            "uuid": "SAMPLE-PROJECT",
        },
        "snapshot": 1502354654
    },
    "data": [
        {
            "uid": "01da56d83ec1c04be901ebe162957e3d3eaf6e49a85a0f31497f535b1f567b32",
            "name": "CB12V51.GENAPP.KSDSCUST.REPORT",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "(,,,4)",
                "datasetType": "GDG"
            }
        },
        {
            "uid": "09c2ef33f622e62da876e1d73a7e9e8f3aa8236034973a3f232acbf4da8509ee",
            "name": "DFH520.CICS.SDFJAUTH",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "DATASET"
            }
        },
        {
            "uid": "0c296329019f68889434bc14103de4475c3a680c64ee91f9a8f331c736b30257",
            "name": "DFH530.CNTL.CICS.DFHTEMP",
            "type": "DATASET",
            "properties": {
                "volumeSerial": "",
                "datasetType": "PDS"
            }
        }
    ]
}
```


Where:

- **meta.project.name**: representing the name of AD project.
- **meta.project.uuid**: representing the UUID of the AD project.
- **meta.snapshot**: representing the UNIX timestamp (in seconds) when the data was collected.
- **data**: representing an array containing all the datasets artifacts defined in DD cards for the job identified by **{job-uid}**.

Documentation Notices for IBM Application Discovery for IBM Z

This edition applies to version 5.1.0 of IBM Application Discovery for IBM Z with the corresponding fix packs.

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 might 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, MD-NC119
Armonk, NY 10504-1785
US
```

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

```
Intellectual Property Licensing
Legal and Intellectual Property
Law
IBM Japan Ltd.
19-21, Nihonbashi-
Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan
```

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 jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information might 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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide 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 Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
```

```
Armonk, NY 10504-1785
US
```

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.

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.

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

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary. This information is for planning purposes only.

The information herein is subject to change before the products described become available.

This information contains examples of data and reports that are 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 actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE: This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You can 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. Therefore, IBM 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.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows: © (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

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 Copyright and trademark information.

SC27-8973-06